

Oracle Fusion Cloud HCM

HCM Data Loader



Oracle Fusion Cloud HCM
HCM Data Loader

G34704-03

Copyright © 2011,2024, Oracle and/or its affiliates.

Author: Oracle Human Resources Information Development Team

Contents

Get Help

i

1	Introduction to HCM Data Loader	1
	Overview of HCM Data Loader	1
	HCM Data Loader and Implementation Scenarios	2
	How Data Is Loaded Using HCM Data Loader	3
	How You Enable Access to HCM Data Loader	4
	How You Configure HCM Data Loader Business Object Access	7
	Supported Key Types	10
	Source Keys	12
	User Keys	13
	Oracle Fusion GUIDs and Surrogate IDs	14
	Business-Object Structure	15
	FAQs for Introduction to HCM Data Loader	16
2	Data File Instructions and Delivery	19
	HCM Data Loader Data and .zip Files	19
	File Line Instructions and File Discriminators	20
	The METADATA Instruction	22
	The MERGE Instruction	23
	The DELETE Instruction	24
	The SET Instruction	25
	Generate and Use Template Files	29
	FAQs for Data File Instructions and Delivery	31
3	Preparing to Use HCM Data Loader	33
	Sources of Business-Object Information	33
	HCM Data Loader Configuration Parameters	35
	How You Audit the Configure HCM Data Loader Parameters	42
	How You Define Referenced Objects	43
	How You Review Lists of Values	44
	Define the Source-System Owner	45

How You Test the HCM Data Loader Process Flow and Connections	45
How You Audit Objects Loaded in Bulk	46
FAQs for Preparing to Use HCM Data Loader	48
4 Data Preparation	49
Guidelines for Preparing the Source Data	49
General Data Considerations for HCM Data Loader	50
How You Load Images, Attachments, and Large Strings	51
How You Supply Source Keys	52
How You Update Source Keys	53
How You Supply User Keys	54
Examples of Supplying Oracle Fusion Surrogate IDs	55
Examples of Supplying Oracle Fusion GUIDs	55
How You Manage Reserved Characters	56
How You Load and Maintain Translated Objects	57
How You Include Reference Values in Data Files	59
Data Deletion Using HCM Data Loader	60
5 Loading Date-Effective Data	63
Overview of Loading Date-Effective Objects	63
How You Load Objects with Multiple Changes per Day	64
Options for Updating Date-Effective Objects with Future-Dated Records	66
Example of Retaining Attribute Values in Future-Dated Records	68
Options for Carrying Forward Changed Attribute Values in Retain Mode	69
Example of Replacing Attribute Values in Future-Dated Records	71
Examples of Changing First Effective Start Date and Last Effective End Date	72
6 Loading Flexfield Data	75
Guidelines for Loading Flexfield Data	75
How You Load Data for Descriptive Flexfields	76
Example of Loading Data for Descriptive Flexfields	78
How You Load Data for Extensible Flexfields	79
Example of Loading Data for Extensible Flexfields	80
7 Importing and Loading Data Using HCM Data Loader	83
How Data Is Imported and Loaded	83
Uploading and Securing Files on the Web Center Content Server	84

Import and Load Data	85
Example of using HCM Data Loader	87
How You Review and Correct Errors	90
Start or Resubmit a Load	92
Roll Back HCM Data Loader Data Set or Business Object	93
Stop HCM Data Loader Processing	94
Overview of Automating Data Loading	95
How to Review the Objects Processed in a Data Set	98
Maintaining Oracle Search Indexes	99
Manually Submitting the Bulk-Ingestion Process for Oracle Search Indexes	100
How to Enable Bulk Ingestion for Oracle Search Indexes	100
FAQs for Importing and Loading Data Using HCM Data Loader	100
FAQs for Bulk Ingestion for Oracle Search Indexes	103
8 Initiate File Processing on the Go	105
Using the Recent File Loads Page	105
9 Data Loading Using Spreadsheets	107
Spreadsheet Data Loaders	107
Set Up Desktop Integration for Excel	108
Troubleshoot Desktop Integration for Excel	109
10 Introduction to HCM Spreadsheet Data Loader	111
Guidelines for Using HCM Spreadsheet Data Loader	111
How Data Is Uploaded Using HCM Spreadsheet Data Loader	113
How You Test the HCM Spreadsheet Data Loader Process Flow and Connections	114
Spreadsheet Business Object Access	115
Manage Spreadsheet Business Object Access	117
How You Enable Access to HCM Spreadsheet Data Loader Tasks	119
How You Enable Administrator Access to HCM Spreadsheet Data Loader	121
How You Enable Audit of Role Configuration for a Spreadsheet Template	122
How You Audit Objects Loaded by HCM Spreadsheet Data Loader	125
FAQs for HCM Spreadsheet Data Loader	125
11 HCM Spreadsheet Data Loader Templates	127
HCM Spreadsheet Data Loader Templates	127
Define Categories for Spreadsheet Templates	129

Create and Edit Spreadsheet Templates	130
How You Associate a User Name with a Data Upload	133
Set Spreadsheet Template Parameters	134
Spreadsheet Template Design	137
Guidelines for Designing Spreadsheet Templates	139
How You Configure Default Values for Spreadsheet Attributes	140
Configure Lists of Values for Attribute Validation	141
Autogenerated Attribute Values in Spreadsheet Templates	143
Include Reconciliation and Reporting Information	144
Copy Spreadsheet Templates	145
Export and Import Spreadsheet Templates	146
How You Enable Business Users with Custom Roles to Load Spreadsheet Data	147
Add a Role to a Predefined Spreadsheet Template	147
How You Assign Access to a Spreadsheet Template	149
FAQs for HCM Spreadsheet Data Loader Templates	150

12 Importing and Loading Data Using HCM Spreadsheet Data Loader 153

Generate Spreadsheets	153
Import Data to a Spreadsheet	153
Manage Spreadsheet Data Sets	155
Upload Data Using HCM Spreadsheet Data Loader	156
Roll back Data using HCM Spreadsheet Data Loader	158
How to Set a Value to Null	158
Review Existing Data Sets	159
How You Delete Data Using HCM Spreadsheet Data Loader	160
FAQs for Importing and Loading Data Using HCM Spreadsheet Data Loader	160

13 Postload Processing 163


Processes to Run After Loading Data	163
How You Validate Loaded Data	166
How You Review Automatically Submitted Post Processes	167
Business Object Post Processes	169
Data Set Post Processes	173
How You Manage Business Object Post Processes	175
How You Manage Data Set Post Processes	176

14	Removing Person Information	177
	Remove Person Information	177
	Remove Person Payroll Information	185
	Initiating Remove Person Information using REST API	186
15	Purging Person Data from Test Environments	189
	Overview of the Purge Person Data in Test Environments Process	189
	Purge Person Data from Test Environments	190
	Tables Purged by the Purge Person Data in Test Environments Process	192
16	Maintaining Stage Tables	199
	Overview of Maintaining the Stage Tables	199
	Example of the Automatic Deletion and Backup of Stage Table Data	203
	Maintain the HCM Data Loader Stage Tables	204
	Maintain the HCM Spreadsheet Data Loader Stage Tables	208
17	Best Practices for HCM Data Loader and HCM Spreadsheet Data Loader	213
	HCM Data Loader Best Practices	213
18	Reporting and Diagnostics	215
	Overview of HCM Data Loader Extracts	215
	Setting Up HCM Data Loader Extracts	215
	Running HCM Data Loader Extracts	220
	Diagnostic Tests for HCM Data Loader	224
	Diagnostic Tests for HCM Spreadsheet Data Loader	226
	Overview of OTBI Reporting	228
19	Transferring Encrypted Data	231
	Set up Encryption for File Transfer	231
	Encrypt and Upload Files Automatically	233

Get Help

There are a number of ways to learn more about your product and interact with Oracle and other users.

Get Help in the Applications

Some application pages have help icons  to give you access to contextual help. If you don't see any help icons on your page, click your user image or name in the global header and select Show Help Icons. If the page has contextual help, help icons will appear.

Get Training

Increase your knowledge of Oracle Cloud by taking courses at [Oracle University](#).

Join Our Community

Use [Cloud Customer Connect](#) to get information from industry experts at Oracle and in the partner community. You can join forums to connect with other customers, post questions, suggest [ideas](#) for product enhancements, and watch events.

Share Your Feedback

We welcome your feedback about Oracle Applications user assistance. If you need clarification, find an error, or just want to tell us what you found helpful, we'd like to hear from you.

You can email your feedback to oracle_fusion_applications_help_ww_grp@oracle.com.

Thanks for helping us improve our user assistance!

1 Introduction to HCM Data Loader

Overview of HCM Data Loader

HCM Data Loader is a powerful tool for bulk-loading and maintaining data. It supports file-based upload of business objects, components, and component attributes using a flexible, delimited file.

You can use HCM Data Loader for data migration, ongoing maintenance of HCM data, and coexistence scenarios, where core HR data is uploaded regularly.

This topic introduces the main features and advantages of HCM Data Loader.

Tip: You can access Oracle By Example tutorials for HCM Data Loader in the Oracle Learning Library. Search for HCM Data Loader to find HCM Data Loader content.

Business-Object Support

You can load business objects for most Oracle Fusion Human Capital Management (HCM) products. These products include Oracle Fusion Absence Management, Compensation, Global Human Resources, Global Payroll, Performance Management, Profile Management, Talent Review and Succession Management, and Workforce Management. You can also update business objects using HCM Data Loader, regardless of how they were created.

You can review the complete list of supported business objects using the View Business Objects task, available in the Data Exchange work area.

Ease of Use

HCM Data Loader has a comprehensive user interface for loading data, monitoring progress, and reviewing any errors. It provides real-time information for all stages of its processing. HCM Data Loader's user interface also includes detailed information about the component hierarchies and attributes of supported business objects.

You can load data from either delimited data (.dat) files or spreadsheets. You can load most supported business objects using either method. You can generate business-object templates to use as the basis of your own .dat files. For a selected business-object component, the template includes every attribute, including those for configured flexfields. When using spreadsheets, you define spreadsheet templates to suit business needs and generate spreadsheets from those templates.

Performance

HCM Data Loader supports multithreaded processing, which enables you to upload complete data extracts without severe performance impacts. References among objects that are processed on separate threads are managed automatically. You can set HCM Data Loader configuration parameters to optimize processing for your environment.

You can also perform partial or incremental loads to update existing objects, thereby minimizing the related processing.

Supported Key Types

HCM Data Loader supports all of the following key types for most business objects:

- Oracle Fusion GUID
- Oracle Fusion surrogate ID
- Source key
- User key

As user keys and source keys are supported, knowledge of Oracle Fusion internal IDs isn't required.

Business-Object Features

HCM Data Loader supports most business-object features and requirements. For example, you can upload:

- Current and historical records for date-effective objects. You determine the amount of history to load.
- End-dated, terminated, or inactive records.
- Translated attributes in multiple languages. You specify the character set of the data file on the **File Character Set** configuration parameter for HCM Data Loader.
- Descriptive flexfields and extensible flexfields.
- Hierarchical tree data, such as organization and department trees.
- Attachments and pictures.
- Data from multiple sources. You can include source-system references in uploaded data.

Automation

You can initiate HCM Data Loader using a web-service call, which enables you to automate data upload.

Related Topics

- [Sources of Business-Object Information](#)

HCM Data Loader and Implementation Scenarios

You can implement Oracle HCM Cloud in either full or coexistence mode. In a coexistence implementation, you use Oracle Fusion Talent Management or Oracle Fusion Compensation, but continue to use your existing HR applications.

For both full and coexistence implementations, you implement Oracle HCM Cloud by performing the tasks that appear in your implementation project or for selected functional areas. This topic describes how HCM Data Loader supports these implementation types.

Full Implementations

If you're performing a full implementation of Oracle HCM Cloud, then you can use HCM Data Loader to bulk-load any existing HCM data at appropriate stages in the implementation. Typically, you load each type of data once only for this type of implementation. Following successful upload, you manage your data in Oracle HCM Cloud.

Coexistence Implementations

In a coexistence implementation, you use Oracle Fusion Talent Management or Oracle Fusion Compensation, but maintain your existing HR applications. For this type of implementation, you:

- Move talent data permanently to Oracle HCM Cloud, which becomes the application of record for talent data.
- Upload other types of data, such as person records, periodically to Oracle HCM Cloud. The source system remains the application of record for this data.

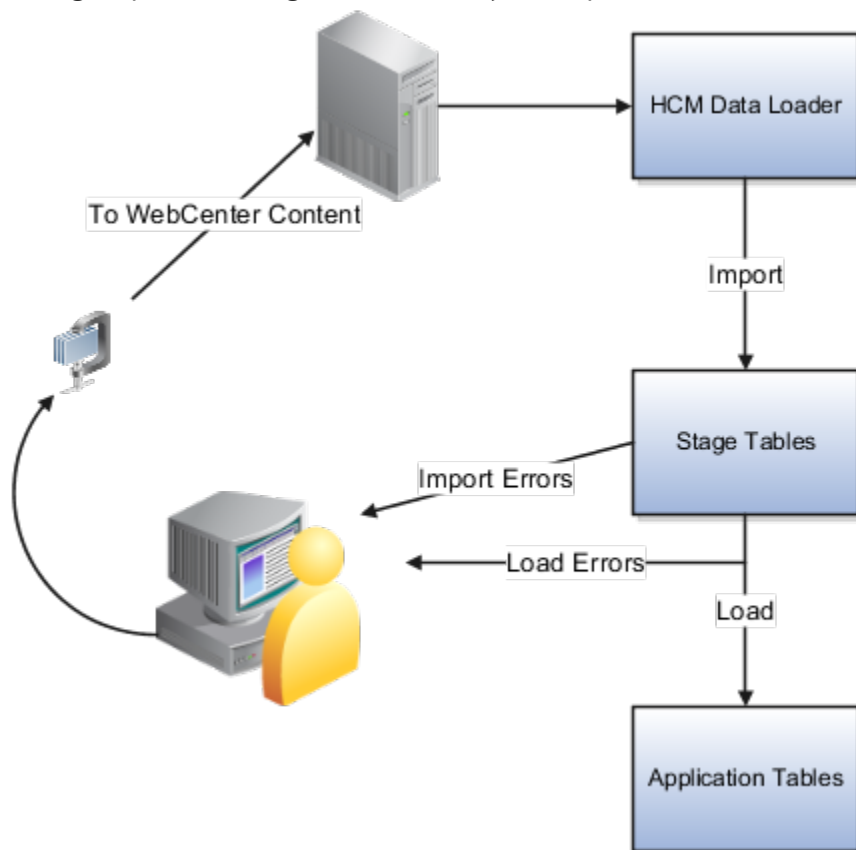
To implement an HCM coexistence scenario, for any source system, you can use HCM Data Loader for data upload. Follow the general instructions provided for HCM Data Loader. Oracle doesn't supply tools to extract data from your source systems. Work with your implementation partner or develop your own process for extracting the source data.

HCM Data Loader provides a Compensation Changes extract. You can extract assignment, bonus, and salary changes for a specified compensation run if you're using Oracle Fusion Compensation in a coexistence scenario.

How Data Is Loaded Using HCM Data Loader

This topic outlines the process of loading data in bulk from .dat files using HCM Data Loader. The data can be from any source.

This figure provides a high-level summary of the process.



The steps of this process are:

1. You place a .zip file containing your .dat files on the Oracle WebCenter Content server. You can upload your file directly to the server, or use the HCM Data Loader interface for local files.
2. You submit a request to HCM Data Loader to import and load the .zip file. For this step, you can use either the HCM Data Loader interface or the HcmCommonDataLoader web service.
3. HCM Data Loader decompresses the .zip file and imports individual data lines to its stage tables. In the stage tables, related data lines are grouped to form business objects.
Any errors that occur during the import phase are reported on the HCM Data Loader interface.
4. HCM Data Loader elevates the user before calling the relevant logical object interface method (delivered in product services) to load valid objects to the application tables.
Any errors that occur during the load phase are reported in the HCM Data Loader interface.
5. You review any errors from the import and load phases. You can perform this step either on the HCM Data Loader interface or using the HCM Data Loader Data Set Summary extract.
6. You correct errors from the import and load phases in your source data.
7. You load a new .zip file containing the corrected data to the WebCenter Content server.

You repeat this process from step 2 until all of the data is successfully loaded.

Note: By loading your data with the elevated user, HCM Data Loader bypasses the data security policies configured for the session user. However, you must supply data in the language configured for the FUSION_APPS_HCM_ESS_LOADER_APPID user, usually it is the base language of your environment.

Tip: You can also correct load errors interactively on the HCM Data Loader interface and resubmit any corrected data from there.

Related Topics

- [How Data Is Uploaded Using HCM Spreadsheet Data Loader](#)

How You Enable Access to HCM Data Loader

This topic explains how to give other roles access to these tasks. You must have the IT Security Manager job role or privileges.

Access to Import and Load Data Files

A single duty role provides access to:

- The Data Exchange work area
- The View Business Objects task
- The Recent File Loads task
- The Import and Load Data task
- The Delete Stage Table Data task and process.
- The HCM Data Loader processes for transferring, importing and loading data.

To enable this access, edit your custom role to add the following duty role:

Name	Code
HCM Data Load	ORA_HRC_HCM_DATA_LOAD_DUTY

You must also grant access to the Oracle WebCenter Content server folders that HDL uses for staging files to upload and generating template and error files. Grant your custom role these role hierarchies:

Role Hierarchy	Code
Upload data for Human Capital Management file based import	HCM_DATALOADER_IMPORT_RWD
Download data from Human Capital Management file based Export	HCM_DATALOADER_EXPORT_RWD

Access to the View Business Objects Task

On the View Business Objects task, you can review the supported business objects, their hierarchical shape and details of their supported attributes. If you're not using the HCM Data Load duty role, to enable this access, you edit the custom roles to add the following Function Security Privileges:

Function Security Privilege Name	Function Security Privilege Code
Review HCM Data Loader Business Objects	HRC_REVIEW_HDL_BUSINESS_OBJECTS
Generate HCM Data Loader Templates	HRC_GENERATE_HDL_TEMPLATES

In addition, if the Manage Data Exchange Work Area (`HRC_MANAGE_DATA_EXCHANGE_PRIV`) function security privilege isn't already granted to the role, then you must grant it.

Access to the Delete Stage Table Data Task

On the Delete Stage Table task, you initiate the deletion of data sets that are no longer required and create schedules for the regular maintenance of the staging tables. If you're not using the HCM Data Load duty role, to enable this access, you edit the custom roles to add the following Function Security Privilege:

Function Security Privilege Name	Function Security Privilege Code
Delete HCM Data Loader Stage Data	HRC_DELETE_HDL_STAGE_DATA

In addition, if the Manage Data Exchange Work Area (`HRC_MANAGE_DATA_EXCHANGE_PRIV`) function security privilege isn't already granted to the role, then you must grant it.

Note: This privilege also gives access to the Recent File Loads task. If you want to delete protected data sets before they expire, or maintain schedules to regularly purge protected data sets, you also need the following Function Security Privilege:

Function Security Privilege Name	Function Security Privilege Code
Delete HCM Data Loader Protected Data-Sets	HRC_HDL_DEL_PROTECTED_DS

Access to the Configure HCM Data Loader Task

On the Configure HCM Data Loader task, you configure the parameters that control HCM Data Loader and HCM Spreadsheet Data Loader import and load processes. This task is only available in **Setup and Maintenance**.

To enable access to this task, you edit the custom roles to add the following Function Security Privilege:

Function Security Privilege Name	Function Security Privilege Code
Manage Configuration of HCM Data Loader	HRC_MANAGE_CONFIGURATION_HCM_DATA_LOADER

Access to Bulk-Load Security Related Data

Enabling access to HCM Data Loader allows your users to bulk-load data into most of the objects that HCM Data Loader supports. To enable this capability, set the override value of the **Enable Restrict Access to Security Related Business Objects** parameter to **Yes** in the **Configure HCM Data Loader** task. An additional privilege is required for roles responsible for bulk-loading security related data using these objects:

- User
- Delegated Role
- Role Mapping
- Legislative Data Group Security Profile
- Organization Security Profile
- Country Security Profile
- Position Security Profile
- Document Type Security Profile
- Areas of Responsibility
- Exclusion Rule
- Job Requisition Security Profile

To enable access to bulk-load data with these objects, you edit the custom roles to add the following Function Security Privilege:

Function Security Privilege Name	Function Security Privilege Code
Load HCM Security Data	HRC_LOAD_HCM_SECURITY_DATA_PRIV

Note: Data sets imported prior to enabling this feature can continue to be processed without this privilege. Business object access is only evaluated when importing a data set.

How You Configure HCM Data Loader Business Object Access

You can configure product areas and/or business objects to be bulk loaded by a job role using HCM Data Loader to provide users access to the tools needed to fulfill their roles.

This topic explains the steps to provide access by roles or business objects. To provide access -

- By role: Select a single role and assign one or more business objects
- By business object: Select a single business object and assign one or more roles to it

Create Custom User Roles

Create custom user roles using the **HCM Data Load duty** role to grant HCM Data Loader access as detailed in the [How You Enable Access to HCM Data Loader](#) topic.

Manage HCM Data Loader Business Object Access By Role

- Access the **HCM Data Loader Business Object Access** task on the **HCM Data Loader** functional area under **Setup and Maintenance**.
- In the **Job and Abstract Roles**, enter the custom role in the **Role** query-by-example field .

- In the results section, select the created custom role. The **Assigned Business Objects** section title is automatically updated to include the role name.

HCM Data Loader Business Objects Access : By Role ▾

Job and Abstract Roles

Role	Code	Desc
Payroll Manager - Vision Operations	PAY_PAYROLL_MANAGER_VISION_OPERATIONS_DATA	Payro
Payroll Manager - Vision Services	PAY_PAYROLL_MANAGER_VISION_SERVICES_DATA	Payro
Payroll Administrator - Fusion Corp USA	PAY_PAYROLL_ADMINISTRATOR_FUSION_CORP_USA_DATA	Payro
Payroll Manager - Fusion Corp USA	PAY_PAYROLL_MANAGER_FUSION_CORP_USA_DATA	Payro
Payroll Administrator	ORA_PAY_PAYROLL_ADMINISTRATOR_JOB	Assis
Payroll Interface Coordinator	ORA_HRY_PAYROLL_COORDINATOR_JOB	Mana

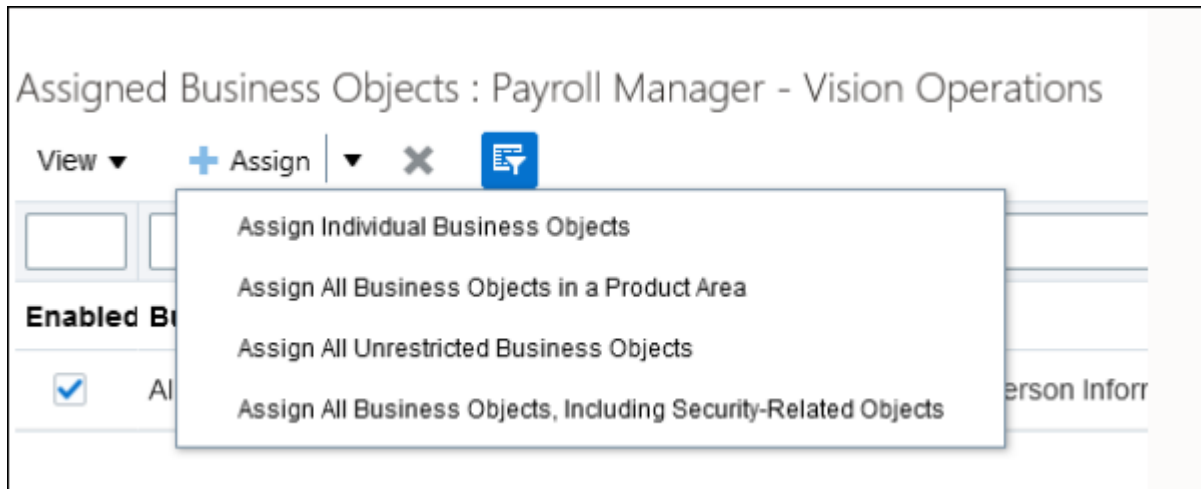
Assigned Business Objects : Payroll Manager - Vision Operations

View ▾

Assign ▾

Enabled Business Object	Product Area	Description
<input checked="" type="checkbox"/> All	Global Payroll - Person Information	All business

- Click **Assign** dropdown for the selected custom role under **Assigned Business Objects** section. The following role-business object mappings are displayed:
 - Assign Individual Business Objects
 - Assign All Business Objects in Product Area
 - Assign All Unrestricted Business Objects
 - Assign All Business Objects, Including Security-Related Objects



If you select **Assign Individual Business Objects**, then:

- Search and select the business objects in the **Search and Select Business Objects** dialog box.
- Click **Add** to add the selected business objects to the role. An entry appears in the **Assigned Business Objects** section for each of the selected business object.

If you select **Assign All Business Objects in a Product Area**, then:

- Select the product area in the **Select Product Area** dialog box.
- Click **Add**. A single entry appears for the product area in the **Assigned Business Objects** section.

Note: This role will have access to load data for all the business objects within the product area. Use the **View Business Objects** task to see which business objects are available in each product area.

If you select **Assign All Unrestricted Business Objects** then:

- A warning message appears to indicate that users with this role can bulk-load data with any business object that doesn't load security-related data.
- Click **Add** to close the warning and continue. A single entry appears for all unrestricted business objects in the **Assigned Business Objects** section.

If you select **Assign All Business Objects, Including Security-Related Objects** then:

- A warning message appears to indicate that users with this role will be able to use the security-related objects only if they have the **Load HCM Security Data** function security privilege

- Click **Add** to close the warning and continue. A single entry appears for all business objects for all product areas in the **Assigned Business Objects** section.

Note: When the **Restrict Access to Security Related Business Objects** is set to Yes in the **Configure HCM Data Loader** page, only users with the custom role with the **Load HCM Security Data** function security privilege will be able to bulk load data using security related business objects.

- Click **Save**.

Manage HCM Data Loader Business Object Access By Business Object

To provide access by business objects:

- On the **HCM Data Loader Business Object Access** page, select the **By Business Object** option in the page title.
- In the **Business Objects** section, enter the name of the business object in the **Business Object** query-by-example field and press **Enter**.
- In the results section, select the business object. The **Assigned Roles** section title is automatically updated to include the business object name.
- In the **Assigned Roles** section, click **Add** icon.
- In the **Search and Select Roles** dialog box, search and select one or more roles.
- Click **Add**. An entry appears in the **Assigned Roles** section for each of the selected roles.
- Click **Save**.

Enable HCM Data Loader Business Objects Access

To enable the capability of configuring the role based access of the business objects, set the override value of the **Enable Configuration of Role-Based Business Object Access** parameter to **Yes** in the **Configure HCM Data Loader** task under **Setup and Maintenance**.

Note: Data sets imported prior to enabling this feature can continue to be processed regardless of the user's business object access. Business object access is only evaluated when importing a data set.

Here's a tutorial to configure roles to access HCM Data Loader (HDL) functionality and business objects and to generate the PGP key for HDL file encryption [Configure Access to HCM Data Loader \(HDL\) and Understand HDL Security Options](#)

Related Topics

- [How You Enable Access to HCM Data Loader](#)
- [HCM Data Loader Configuration Parameters](#)

Supported Key Types

When planning to integrate data from one system to another, you must select key types to identify records uniquely. This topic introduces the key types that HCM Data Loader supports.

What Keys Identify

In HCM Data Loader, you use unique keys to identify:

- The record that you're creating or updating.
- The parent of the record that you're creating or updating.
The parent record may either be in the same data file or exist already in Oracle HCM Cloud.
- Any objects referenced by the record that you're creating or updating.

Supported Key Types

HCM Data Loader supports the following key types, which you can use for all types of object references. When more than one key value exists for an object, key references are resolved in this order:

1. Oracle Fusion Globally Unique Identifier (GUID)
2. Source key
3. Oracle Fusion surrogate ID
4. User key

If you supply multiple key values, then no cross-validation occurs. For example, if you supply both a GUID and a source key, then the GUID is used and the source key is ignored. If the source key identifies a different record from the GUID, then no error is raised.

Tip: Whenever possible, use source keys for both creating and updating business objects.

Key-Type Features

The following table summarizes the main features of the available key types. The **Create** and **Update** columns indicate whether you can use the key type when creating and updating objects.

Key Type	Create	Update	Held on Object	Type	Generated Automatically
GUID	No	Yes	No	Hexadecimal	Yes
Surrogate ID	No	Yes (see note)	Yes	Numeric	Yes
Source Key	Yes	Yes	No	Alphanumeric	Conditionally (see note)
User Key	Yes	Yes (see note)	Yes	Alphanumeric	No

Note: You can use surrogate IDs when updating objects, but the IDs may not be readily available to Oracle HCM Cloud users. Default source keys are generated only if you don't supply a source key when creating an object. You can't use user keys alone when updating some objects because their values are subject to change.

Keys that aren't held on the object exist in the Integration Key Map table.

Foreign-Object References

Oracle Fusion GUIDs and surrogate IDs are generated only when the associated record is successfully created in Oracle HCM Cloud. Source keys aren't recognized in Oracle HCM Cloud until the associated record exists in Oracle HCM Cloud. Therefore, before you attempt to load data that references foreign objects, you must ensure that those foreign objects exist in Oracle HCM Cloud. For a new implementation, you're recommended to load each business object separately and ensure that it loads successfully before importing the next. If you supply all business objects in the same .zip file, then HCM Data Loader loads them in order of dependency. In this case, a reference to a foreign object fails only if the referenced object failed to load.

Related Topics

- [Source Keys](#)
- [User Keys](#)
- [Oracle Fusion GUIDs and Surrogate IDs](#)

Source Keys

A source key is the ID that identifies a business object in its source system. Source keys are supported only for integration-enabled business objects. You can use source keys when creating or updating business objects.

You're recommended to use source keys for all implementations but particularly for ongoing integrations, where data is maintained in the source system.

Source-Key Structure

A source key has the two components shown in this table.

Component	Description
Source-system owner	Identifies where the data originated.
Source-system ID	ID used in the source system. The ID must be unique for the business-object component and source-system owner.

Having a source-system owner component to the key enables you to upload data from multiple source systems for the same business objects. For example, you may have person data in both US and UK databases to combine in one Oracle HCM Cloud system. The source-system ID doesn't have to be unique across both source systems. It must be unique only for the business-object component and source-system owner.

If you don't have an obvious source-system ID for a record, then you can generate or derive one. For example, you could derive a source-system ID for a person address by concatenating the person number with the address type.

Source keys aren't held against the created record. They exist in the Integration Key Map table.

Source Keys for Date-Effective Business Objects

If you're supplying date-effective history for an object, then you must supply the source system IDs for every date-effective record in the file. The values must be the same for each line of the date-effective history.

Default Source Keys

If you provide no source key when creating an object, then the source key is created automatically. The default source-system owner is **FUSION** and the default source-system ID is the surrogate ID. You can update the source key later using the Source Key business object. In the Source Key business object, you supply the new source key and a reference to the object that you're updating.

Source-System Owner Values

The source-system owner value is validated against the HRC_SOURCE_SYSTEM_OWNER lookup type. You must add your source-system names to this lookup type before loading data using source-system references. Use the **Manage Common Lookups** task in the Setup and Maintenance work area to update HRC_SOURCE_SYSTEM_OWNER.

Related Topics

- [Supported Key Types](#)
- [Oracle Fusion GUIDs and Surrogate IDs](#)
- [User Keys](#)

User Keys

Most Oracle HCM Cloud business objects, regardless of whether they're enabled for integration, have one or more attributes that make up a user key. The user key, which is also known as a natural key, is always visible on the user interface.

For example:

- The user key for an organization is the organization name.
- The user key for an area of responsibility is the responsibility name plus the person number.

When to Use User Keys

User keys are part of the business object definition. They're always required when you create a logical object, regardless of how you create it.

You can also use user keys when updating logical objects. However, the value of a user key can change, and some user-key attributes are translatable. For this reason, you can't update some business objects if you supply only user keys to reference them. Therefore, whenever possible, you're recommended to use source keys when updating objects.

User keys are recommended when:

- You're referencing or maintaining an object that wasn't created with a source key.
- The source-key value is unknown.

User Keys for Child Objects

When a business object is bound by another, the user key must include the user key for its parent. For example:

- Jobs are always part of a set. Therefore, job code alone doesn't uniquely identify a job. Instead, the set code must be part of the user key for a job.
- Job grades apply to a specific job. Therefore, the user key for a job grade must include both the user key for the grade and the user key for the parent job. Therefore, the user key for the job grade is made up of the grade code, job code, and set code.

Related Topics

- [Supported Key Types](#)
- [Oracle Fusion GUIDs and Surrogate IDs](#)
- [Source Keys](#)

Oracle Fusion GUIDs and Surrogate IDs

Oracle Fusion Globally Unique Identifiers (GUIDs) and surrogate IDs are two of the four types of keys that HCM Data Loader supports. This topic describes when you can use GUIDs and surrogate IDs to identify business objects uniquely.

Oracle Fusion GUIDs

When you create an integration-enabled business object in Oracle HCM Cloud, a GUID is generated for the object automatically. As the GUID is generated when you create the business object, it's suitable only for identifying business objects that already exist. You can't supply a GUID when creating a business object. The GUID, a hexadecimal value, isn't held on the created object but exists in the Integration Key Map table.

GUIDs are useful when you share changes with down-stream applications, such as third-party payroll providers. You extract the changes for the third-party payroll and provide the GUID as the unique key to the worker. When reporting any updates back to Oracle HCM Cloud, you can use GUIDs to identify the records to update.

Oracle Fusion Surrogate IDs

The surrogate ID, a numeric value, is the internal system identifier that's generated when you create a business object. All objects, including those that aren't integration-enabled, have surrogate IDs. The surrogate ID is held on the object. The ID is also used in the Integration Key Map table to provide a mapping for GUID and source-key values. As the surrogate ID isn't generated until the object is created, you can't use this value when creating objects. Access to surrogate IDs may also be limited. On-premises customers are more likely to have access to surrogate IDs than Oracle HCM Cloud customers.

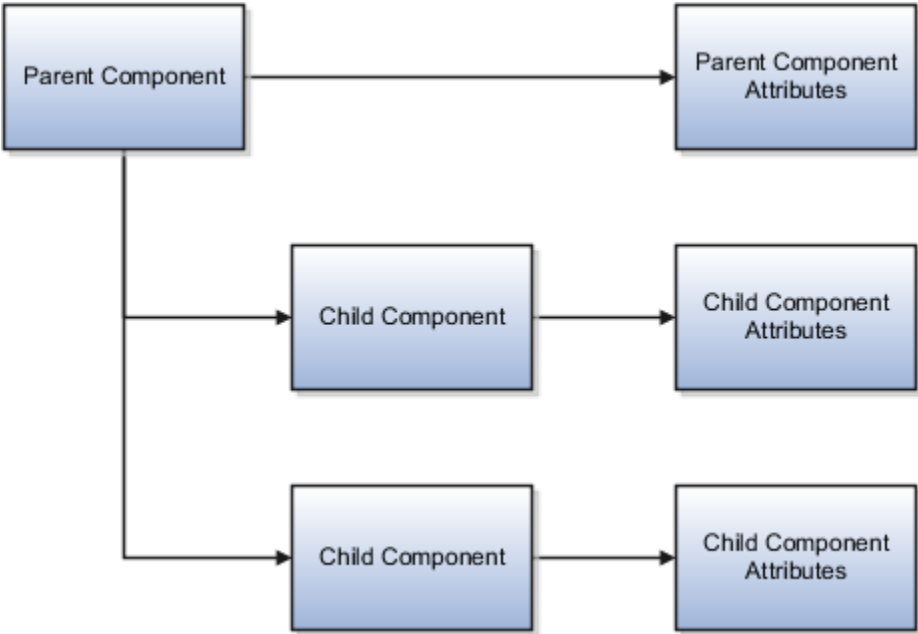
Related Topics

- [Supported Key Types](#)
- [Source Keys](#)
- [User Keys](#)

Business-Object Structure

Each Oracle HCM Cloud business object is a hierarchy of business-object components. At the top of the hierarchy is the parent component, and beneath it may be child and grandchild components. Each business-object component has multiple attributes.

This figure shows a generic business-object structure.



For example, the Rating Model business object includes Rating Model, Rating Level, and Rating Category components. The Rating Model component is the parent of the other two components. Each component has attributes such as **RatingName**, **RatingModelCode**, **RatingLevelCode**, and so on.

The most complex business object supported by HCM Data Loader is the Worker object, where five levels exist in the object hierarchy. These range from the Worker component at the top to Assignment Work Measure, Assignment Manager, Assignment Grade Steps, and Assignment Extra Information at the bottom. By contrast, the Person Type object has only the Person Type component.

Terminology

This table defines the terms used by HCM Data Loader to refer to business objects.

Term	Meaning
Object or business object	Refers to the complete object, which is the parent component and all child components. For example, Grade and Worker are business objects.
Component or business-object component	Refers to components of a business object. For example, Person Name and Work Relationship are components of the Worker object.

Term	Meaning
Logical object	Refers to a group of related components that form one occurrence of a business object. For example, the grade IC1 is a logical object.

Loading Logical Objects

When you deliver multiple components for the same business object together, HCM Data Loader groups them and loads the complete logical object. It doesn't process the components separately. If any component of the logical object fails validation, then the whole logical object is rejected. As HCM Data Loader loads only complete logical objects, you can be sure of exactly which data has been loaded. For example, when loading jobs, you know which jobs loaded successfully and which jobs failed to load.

Integration-Enabled Objects

HCM Data Loader can load any integration-enabled business object. An integration-enabled business object supports these four key types:

- Oracle Fusion GUID
- Oracle Fusion surrogate ID
- Source key
- User key

An integration-enabled object has an entry in the Integration Key Map table, where its GUIDs and source keys exist.

HCM Data Loader can also load a few objects, such as Department Tree, that aren't integration enabled. Typically, Oracle HCM Cloud doesn't own such objects and although you can't identify the object by a source key nor GUID, you can often reference foreign objects using integration keys, such as the parent and child departments while creating a department tree.

Tip: You can review business-object details on the View Business Objects page. The **Integration Keys Supported** field on the Component Details tab for a selected component indicates whether the component is integration enabled.

Here's a tutorial on using the View Business Objects task which is the data dictionary for all supported business objects:
[View Business Object task](#)

Related Topics

- [Sources of Business-Object Information](#)

FAQs for Introduction to HCM Data Loader

Why can't I access HCM Data Loader?

You may not have the correct permissions. Refer the topic:[How You Enable Access to HCM Data Loader](#) for details on providing access to HCM Data Loader.

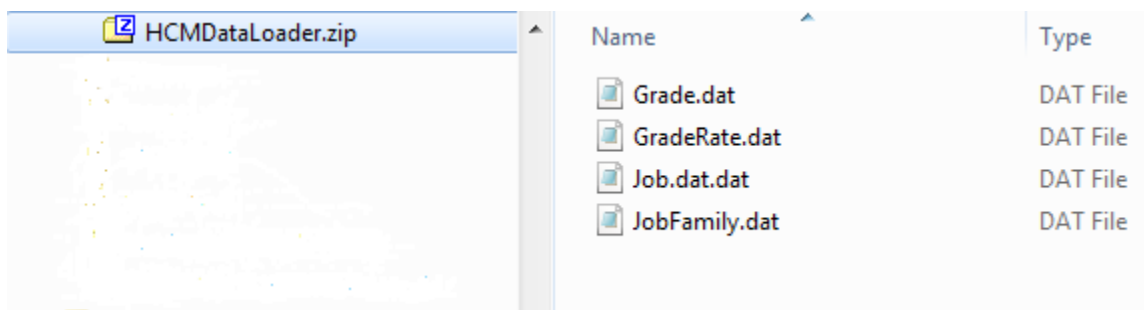
2 Data File Instructions and Delivery

HCM Data Loader Data and .zip Files

You use .zip files to load data with HCM Data Loader. Each .zip file can contain one or more data (.dat) file. Each file includes data for a single business object, such as Location, Grade, Worker, or Salary.

Your data file has the name of a business object that's specified by HCM Data Loader, such as Job.dat or Worker.dat. In any .zip file, you supply data files only for the business objects that you're loading.

This figure shows an example .zip file structure. The .zip file contains no folders for data (.dat) files.



You define the name of the .zip file, which must be in alphanumeric characters (a-z, A-Z, and 0-9). The .zip file can be encrypted as a whole, but HCM Data Loader doesn't support the encryption of individual data files.

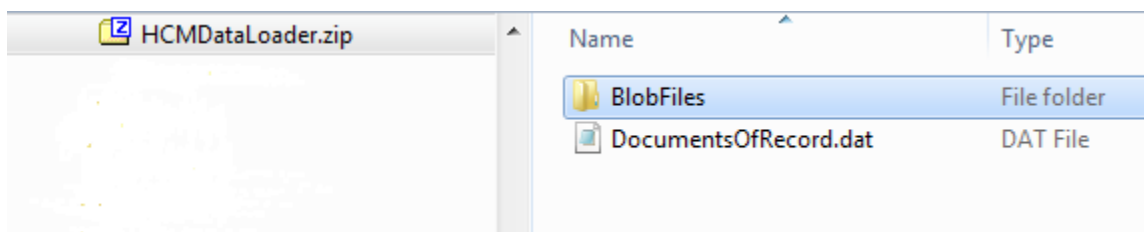
BlobFiles and ClobFiles Folders in the .zip File

The only two folders that can appear in the .zip file are:

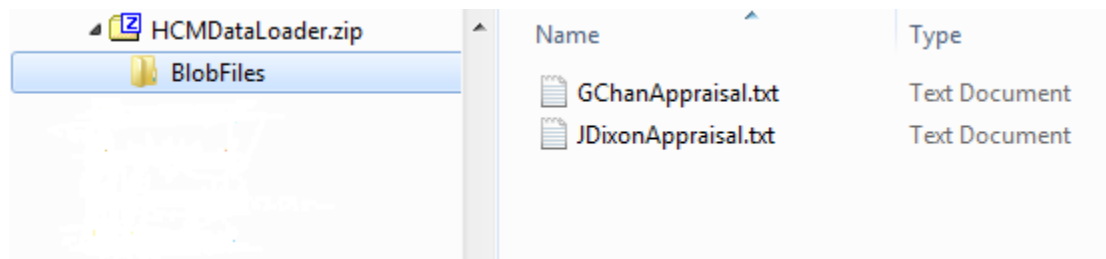
- BlobFiles, which is for binary large objects
- ClobFiles, which is for character large objects

You use these folders for files you want to load as attachments or into large objects. The file names of these attachments or images must be in alphanumeric characters (a-z, A-Z, and 0-9). The data type of the attribute that's used to load your attachment or large-object data determines which folder to use. For example, the **File** attribute in the Document Record object is used for loading attachment files and has a data type of BLOB. Therefore, you place files to be loaded as attachments to document records in the BlobFiles folder.

This figure shows a .zip file structure when attachments are included.



This figure shows the contents of the BlobFiles folder.



Data Files

Each HCM Data Loader data file is specific to a business object. You provide all components that you want to load for a business-object hierarchy in the same data file. For example, you provide Job, Job Valid Grade, Job Evaluation, Job Extra Information, and Job Legislative Extra Information components in Job.dat. This approach lets you to see all data for a business object and supports the validation of parent references on all child components. It also simplifies the construction of the .zip file, as separate directories aren't required and fewer data files have to be generated.

Tip: You can generate data-file templates for all supported objects on the View Business Objects page. Each template defines the complete structure of the selected object.

Third Party Data Files

You can use third party .zip files to load data with HCM Data Loader, such as pay slips that were generated by a payroll application. The data file must conform to the requirements for the business object you want to load, and the .zip file must have been compressed with `java.util.zip`. If not, then you must extract the files and compress again using `java.util.zip` for the load to work properly.

Here's a tutorial on creating and maintaining data with HCM Data Loader [Create and Maintain Data with HDL](#)

To create and upload a HDL worker.dat file for new hires, see: [Create and Load New Hires with HCM Data Loader \(HDL\)](#)

Related Topics

- [How You Load Images, Attachments, and Large Strings](#)
- [Generate and Use Template Files](#)

File Line Instructions and File Discriminators

Each line in a data file begins with an instruction, and most lines include a file discriminator. This topic introduces the available instructions and describes the file discriminators.

File Line Instructions

This table identifies the instructions that can appear in an HCM Data Loader data file.

Instruction	Line Type	Description
METADATA	Definition	Identifies the business-object component and the attributes of that component for which you're including values in the data file
MERGE	Data	Provides data to be merged in Oracle HCM Cloud
DELETE	Data	Identifies a business-object component to be purged from Oracle HCM Cloud
SET	Control	Overrides aspects of the default processing for a file
COMMENT	Comment	Adds a comment to the data file

File Discriminators

METADATA, MERGE, and DELETE lines must include a file discriminator value, which identifies the business-object component to which the instruction applies. For example, the available file discriminators for the Job business object are shown in this table.

Business-Object Component	File Discriminator
Job	Job
Job Valid Grade	JobValidGrade
Job Evaluation	JobEvaluation
Job Extra Information	JobExtraInfo
Job Legislative Extra Information	JobLegislative

The file discriminator appears immediately after the instruction in a file line. HCM Data Loader templates for a business object include the file discriminator for each of the object's components. This example shows instructions, the file discriminator, attribute names, and attribute values for the Job business object.

```
METADATA|Job|EffectiveStartDate|EffectiveEndDate|SetCode|JobCode|Name
MERGE|Job|1950/01/01|4712/12/31|COMMON|CFO|Chief Financial Officer
MERGE|Job|1950/01/01|2013/01/01|COMMON|PM|Manager Projects
MERGE|Job|2013/01/02|4712/12/31|COMMON|PM|Manager Projects
MERGE|Job|1950/01/01|4712/12/31|COMMON|VP_OF_HR|VP Human Resources
MERGE|Job|1950/01/01|4712/12/31|COMMON|PROG_MGR|Program Manager
```

In this example:

- The instruction, METADATA or MERGE, appears at the start of the line.
- The discriminator on all lines is Job.
- Attribute names appear in the METADATA line after the Job discriminator and separated by the vertical bar character (|).
- Attribute values appear in MERGE lines after the Job discriminator and separated by the vertical bar character (|).

Related Topics

- [The METADATA Instruction](#)
- [The MERGE Instruction](#)
- [The DELETE Instruction](#)
- [The SET Instruction](#)

The METADATA Instruction

The METADATA instruction in an HCM Data Loader data file identifies a business-object component and its attributes. The attributes are those for which you're including values in the data file.

Instruction Format

The METADATA line has this format:

```
METADATA|<DISCRIMINATOR>|<ATTRIBUTE 1>|<ATTRIBUTE 2>|<ATTRIBUTE n>
```

For example:

```
METADATA|Job|SetCode|JobCode|JobFamilyName|JobName|EffectiveStartDate|EffectiveEndDate
```

Restrictions

These restrictions apply to the METADATA instruction:

- The METADATA line must appear before the MERGE or DELETE data line to which it relates.
- You can include multiple METADATA lines in a single data file. However, each line must be for a different business-object component. For example, in a Job.dat file, you can include METADATA lines for the Job and Job Valid Grade components. However, you can't include two Job METADATA lines.

Validation

Each METADATA line must:

- Include a valid discriminator for the object that's identified in the data file name. For example, in a Job.dat file, the METADATA line must include one of these discriminators: Job, JobGrade, JobEvaluation, JobExtrainfo, and JobLegislative.

- Be unique in the file. That is, you can't provide multiple METADATA lines for the same discriminator in a single data file.
For example, having two METADATA lines for the JobGrade discriminator in the same data file is invalid.
- Include only valid attributes for the discriminator. Attribute names are case-sensitive, but they can appear in any order.
- Include the attributes for at least one of the supported key types. For example, if you're using source keys to identify a record uniquely, then you must include SourceSystemOwner and SourceSystemId attributes in the METADATA line.

HCM Data Loader business-object templates include METADATA lines with all available attributes for each component of a business object.

Related Topics

- [File Line Instructions and File Discriminators](#)

The MERGE Instruction

The MERGE instruction identifies a data line. You use the MERGE instruction whether creating or updating objects, and HCM Data Loader identifies the correct action.

You can supply either partial or complete objects using the MERGE instruction. The data supplied on the MERGE line is merged into existing Oracle HCM Cloud data. The order in which you specify MERGE lines isn't necessarily the order in which HCM Data Loader processes them.

Instruction Format

The MERGE line has this format:

```
MERGE|<DISCRIMINATOR>|<ATTRIBUTE 1>|<ATTRIBUTE 2>|<ATTRIBUTE n>
```

For example:

```
MERGE|Job|SE|Software Engineer|Common|2015/01/01|4712/12/31
```

Restrictions

These restrictions apply to the MERGE line:

- You must not supply multiple MERGE lines for the same component in a single data file, unless the object is date-effective. For example, you can't supply two MERGE lines, one to create a Person Ethnicity component and one to update it, in the same data file. As the component isn't date-effective, these instructions would correct the data rather than update it. In such cases, you can supply only current data. Alternatively, you can create the record in one data file and update it in a different data file.
Note: HCM Data Loader doesn't process individual file lines, but groups related lines. This grouping works for date-effective records because the file lines are processed in order of effective start date.
- You can include MERGE and DELETE instructions in the same file, but not for the same logical record. For example, you can't load multiple date-effective records for the same component, such as a single Job, using a

mix of MERGE and DELETE instructions. However, you can provide both MERGE and DELETE instructions for components of a single logical object. For example, you can provide:

- MERGE instructions for Worker and Person Name
- DELETE instructions for the Person Email and Person Phone components of the same Worker object

Validation

Each MERGE line must:

- Be preceded by the METADATA line for the same discriminator.
- Contain the same attributes as the relevant METADATA line and in the same order.
- Not include values for attributes that aren't in the METADATA line.
- Contain a unique identifier for itself, using any of the supported key types. In addition, each data line for a child component must include a unique reference to its parent. For example, GradeRateValue must include a unique reference to its associated GradeRate.

Related Topics

- [File Line Instructions and File Discriminators](#)

The DELETE Instruction

The DELETE instruction identifies a business-object component to purge from Oracle HCM Cloud. The component doesn't have to have been loaded using HCM Data Loader.

Instruction Format

The DELETE instruction has this format:

```
DELETE|<DISCRIMINATOR>|<ATTRIBUTE 1>|<ATTRIBUTE 2>|ATTRIBUTE n>
```

For example:

```
METADATA|JobFamily|EffectiveStartDate|EffectiveEndDate|JobFamilyName  
DELETE|JobFamily|2012/10/01|4712/12/31|Sales01
```

Restrictions

These restrictions apply to the DELETE instruction:

- You can delete a business object or a component of a business object. However, you can't delete individual date-effective records.
- You can't provide MERGE and DELETE instructions for the same record in a single data file. The order in which HCM Data Loader processes instructions is unpredictable.
- You can't recover deleted records. Therefore, try to correct data rather than delete and recreate it.
- Not all business-object components can be deleted. For information about deletion support, see the business-object details available from the View Business Objects page in the Data Exchange work area.

Related Topics

- [File Line Instructions and File Discriminators](#)

The SET Instruction

The SET instruction is optional in a .dat file and has no discriminator. You use SET instructions to alter default processing for the file in which they appear.

You can include any number of SET instructions in a .dat file, but they must appear before the first METADATA line. This topic describes when to use the SET instruction.

This table lists the SET instructions and their default values.

Instruction	Default Value
SET PURGE_FUTURE_CHANGES Y N	Y
SET DISABLE_POST_PROCESS_TASKS <process>	Not applicable
SET INVOKE_POST_PROCESS Y N	Y
SET FILE_DELIMITER <delimiter>	Vertical bar ()
SET FILE_ESCAPE <escape character>	Backslash (\)
SET FILE_NEWLINE <newline character>	n
SET ENABLE_AUDIT_DATA Y N	N
SET PURGE_AUDIT_DATA Y N	N
SET CALCULATE_FTE Y N	N
SET CREATE_DEFAULT_WORKING_HOUR_ PATTERN Y N	N
SET ENABLE_INCREMENTAL_LOAD_ EVENTS Y N	N

Instruction	Default Value
Note: Enabling events for your file may increase the time it takes to load.	
SET_EXTEND_DATA_SET_RETENTION Y N Note: Extending the retention period of a data set will lengthen the time it takes to load, as after processing the staging table data is moved from the primary to the secondary staging tables.	N
SET_EVALUATE_LIFE_EVENTS Y N	N
SET_RETAIN_SUCCESSION_PLAN_HISTORY Y N	Y
SET_ORA_ENABLE_OS_BULK_INGEST Y N	Y
SET_ORACLE_SEARCH_UPDATE_MODE ORA_BULK ORA_INITIAL	ORA_BULK

Tip: The default values shown here are predefined for the enterprise. You can override these values for the enterprise using the **Configure HCM Data Loader** task in the Setup and Maintenance work area.

SET Instruction Descriptions

This section provides an alphabetic list of all SET instructions with their descriptions.

SET CALCULATE_FTE

To request automatic calculation of full-time equivalent (FTE) values for all worker assignments in a Worker.dat file, include the **SET_CALCULATE_FTE** instruction. For example:

```
SET_CALCULATE_FTE Y
```

SET CREATE_DEFAULT_WORKING_HOUR_PATTERN

To request automatic creation of a default working-hour pattern for all worker assignments in a Worker.dat file, include the **SET_CREATE_DEFAULT_WORKING_HOUR_PATTERN** instruction. For example:

```
SET_CREATE_DEFAULT_WORKING_HOUR_PATTERN Y
```

SET DISABLE_POST_PROCESS_TASKS

By default, these processes run automatically after you load workers:

- Refresh Manager Hierarchy
- Update Person Search Keywords

To prevent **Refresh Manager Hierarchy** from running automatically, you can include this SET instruction in the Worker.dat file:

```
SET DISABLE_POST_PROCESS_TASKS RefreshManagerHierarchy
```

To prevent **Update Person Search Keywords** from running automatically, you can include this SET instruction in the Worker.dat file:

```
SET DISABLE_POST_PROCESS_TASKS UpdatePersonSearchKeywords
```

By default, the **FlattenPositionHierarchy** process runs automatically after you load HCM position hierarchies. To prevent **FlattenPositionHierarchy** from running automatically, include this SET instruction in the PositionHierarchy.dat file:

```
SET DISABLE_POST_PROCESS_TASKS FlattenPositionHierarchy
```

Tip: To prevent postload processes from running automatically, you can use the `SET INVOKE_POST_PROCESS` instruction instead.

SET ENABLE_AUDIT_DATA

Audit of objects loaded using HCM Data Loader is enabled or disabled for the enterprise using the **Enable Audit Data** configuration parameter. You can override the enterprise setting for individual .dat files by including this SET instruction. For example, if audit is enabled for the enterprise, then you can disable it for an individual .dat file.

SET FILE_DELIMITER

Use this instruction to change the default file delimiter, which is the vertical bar (|). For example, to change it to a comma, include this instruction:

```
SET FILE_DELIMITER ,
```

The new value can be up to 10 characters.

SET FILE_ESCAPE

Use this instruction to change the default file escape character, which is the backslash (\). For example, to change it to the equal sign (=), include this instruction:

```
SET FILE_ESCAPE =
```

The new value can be up to 10 characters.

SET EXTEND_DATA_SET_RETENTION

By default, all data sets are automatically and permanently deleted after a maximum of 30 days since they were last updated. If you are loading data which is to be validated, or may need to be rolled back, you can use this instruction to extend the retention period of the data set.

Supply this instruction only in business object files that support validate or roll back. If you are supplying multiple business object files that support validate or roll back, you need only supply this instruction once.

```
SET EXTEND_DATA_SET_RETENTION Y
```

Note: The time it takes to complete the processing of your file will be longer, as after loading the valid data into the application tables, the staging table data is moved from the primary to the secondary staging tables, so it doesn't negatively impact the processing time of subsequent files.

SET FILE_NEWLINE

Use this instruction to change the default file newline character, which is the character `n`. For example, to change it to the string `newline`, include this instruction:

```
SET FILE_NEW_LINE newline
```

The new value can be up to 10 characters.

SET INVOKE_POST_PROCESS

By default, these processes run automatically after you load workers:

- Refresh Manager Hierarchy
- Update Person Search Keywords

By default, the **FlattenPositionHierarchy** process runs automatically after you load HCM position hierarchies.

To prevent such processes from running automatically, you can include this SET instruction in the relevant .dat file:

```
SET INVOKE_POST_PROCESS N
```

SET PURGE_AUDIT_DATA

Use this instruction to purge the audit data for the objects in a .dat file. This feature is intended primarily to remove the audit records for personally identifiable information in worker records.

You can't include both of the following instructions in a single .dat file:

- SET ENABLE_AUDIT_DATA Y
- SET PURGE_AUDIT_DATA Y

If auditing of loaded objects is enabled for the enterprise, then you must disable it in any .dat file where you're purging audit data. That is, you include the following instructions:

- SET ENABLE_AUDIT_DATA N
- SET PURGE_AUDIT_DATA Y

SET PURGE_FUTURE_CHANGES

When you create business objects containing date-effective components, you include this instruction at the start of the file:

```
SET PURGE_FUTURE_CHANGES Y
```

This instruction sets the date-effective maintenance mode to Replace. In this mode, which is the default mode, the contents of the .dat file replace any existing data.

When you update business objects containing date-effective components, you include this instruction at the start of the file:

```
SET PURGE_FUTURE_CHANGES N
```

This instruction sets the date-effective maintenance mode to Retain. In this mode, any existing future-dated changes are retained.

SET ENABLE_INCREMENTAL_LOAD_EVENTS

To enable the firing of atom feeds and events that are triggered by the changes supplied in a file, include the SET ENABLE_INCREMENTAL_LOAD_EVENTS instruction. For example

```
SET ENABLE_INCREMENTAL_LOAD_EVENTS Y
```

Note: Events are only triggered if they are enabled in your environment. Enabling events for your file may increase the time it takes to load.

SET RETAIN_SUCCESSION_PLAN_HISTORY

Determines whether historical data is retained for changes to succession plans. To prevent the retention of history, include the SET RETAIN_SUCCESSION_PLAN_HISTORY instruction in your SuccessionPlan.dat files. For example:

```
SET RETAIN_SUCCESSION_PLAN_HISTORY N
```

SET ORA_ENABLE_OS_BULK_INGEST

To automatically submits the **Update Oracle Search with HCM Data Loader Changes** process to maintain the Oracle Search indexes with the changes made.

```
SET ORA_ENABLE_OS_BULK_INGEST Y
```

SET ORA_UPDATE_ORACLE_SEARCH

To enable bulk-ingestion include the following instruction at the top of your file:

```
SET ORA_UPDATE_ORACLE_SEARCH ORA_BULK
```

It enables the capture of Fusion surrogate IDs for Oracle Search bulk-ingestion and the automated submission of the **Update Oracle Search with HCM Data Loader Changes** process.

Related Topics

- [File Line Instructions and File Discriminators](#)
- [How You Manage Reserved Characters](#)
- [Overview of Loading Date-Effective Objects](#)
- [Enable Audit for Oracle HCM Cloud Business Objects](#)
- [HCM Data Loader Configuration Parameters](#)
- [How You Review Automatically Submitted Post Processes](#)

Generate and Use Template Files

HCM Data Loader provides a template file for each supported business-object hierarchy. The template contains:

- A COMMENT line, which identifies the business object, its version, and the file-creation date.

- A METADATA line for each component of the business-object hierarchy that you can load for the business object. The METADATA line includes every attribute of the component, including environment-specific configured flexfield attributes.

You can generate template files and use them as the basis of your own data files. The topic describes how to generate and use template files.

Generate a Single Template File

Follow these steps:

1. Open the **Data Exchange** work area and select **View Business Objects**.

By default, the View Business Objects page lists all business objects that HCM Data Loader supports. You can sort the list by product area, business object, or load order, for example. You can also retrieve a specific object by entering its name in the **Business Object** query-by-example field and pressing **Enter**.

2. If a template already exists for the object, then you can click the icon in the **File** column to download it. The **Last Refreshed** column shows when the object definition was last updated and the template was last generated. Alternatively, you can update the object definition before you download the template, as follows:
 - a. Select the object row in the business objects table.
 - b. Click **Refresh Object**.
 - c. In the Schedule Request dialog box, click **Submit**. Click **OK** to close the confirmation message.

When you click **Submit**, you start the process that updates the object definition and generates the latest template file.

Tip: To monitor progress, click **View Process Results** to open the View Process Results page.

3. On the View Business Objects page, click the **Refresh** icon.

Check the value in the **Last Refreshed** column to confirm that the template has been updated.

4. Click the icon in the **File** column to download the template.

Tip: You can download the complete object definition for offline use. To download the object definition, click the icon in the **Overview** column. The spreadsheet that you download is documentation for the object, not a template file.

Generate Template Files for All Business Objects

Follow these steps to generate template files for all business objects in a product area or all business objects:

1. Click the **Refresh All Objects** button on the View Business Objects page.
2. In the Schedule Request dialog box, you can either select a product area or leave **Product Area** set to **All**.
3. Click **Submit**.
4. When the generation process completes, you can download templates individually. Click the icon in the **File** column for an object on the View Business Objects page. Alternatively, follow these steps to download a single .zip file of the generated templates:
 - a. Click View Process Results.
 - b. On the View Process Results page, click the icon in the **File** column for your process.

Use Template Files

Template files identify the names of all available attributes of a business object, including those of environment-specific attributes, such as flexfield segments. Don't simply copy the METADATA lines from the template file and use them in your own data files. Instead, include in your data files:

- METADATA lines for the components that you want to load
- Attribute names for the values you want to supply

You can improve the speed and efficiency of the import and load processes by including only the attributes for which you're supplying data. Remove METADATA lines and attribute names that you don't plan to use.

HCM Data Loader validates every attribute name on every METADATA line.

FAQs for Data File Instructions and Delivery

How should I deliver data when creating objects?

Deliver one object per .zip file. For example, create one .zip file for jobs, one for workers, and so on. Correct any errors before loading the next object to avoid data-reference errors.

How should I deliver data for incremental updates?

Supply only changed and required attribute values. Deliver all business-object files in the same .zip file. HCM Data Loader processes them in the correct order and loads referenced data before the data that references it.

Can I include comments in a .dat file?

Yes. Use the COMMENT instruction to add a comment to a .dat file.

For example:

```
COMMENT Data for Business Object: Job Version: V2 Created on: 25-03-2017
```

You can include any number of COMMENT instructions and they can appear anywhere in the file. They have no effect on processing.

How do I trigger ATOM feeds and events for the changes uploaded by my file?

Atom feeds and events are triggered for changes supplied in the file by including the following SET instruction at the top of your file: SET ENABLE_INCREMENTAL_LOAD_EVENTS Y.

3 Preparing to Use HCM Data Loader

Sources of Business-Object Information

To load a business object or component successfully using HCM Data Loader, you must understand its structure, supported key systems, data formats, and required attributes. You must also understand any business rules that apply to the object.



This topic identifies where you can find this information.

Business-Object Details

The View Business Objects table lists all business object available for bulk-loading. From here you can:

- Download business object template files.
- Download the complete definition of an object for offline use.
- Identify which objects support being rolled back.
- Review the object hierarchy and supported attributes for an object.

You're recommended to refresh all objects after an update. The refresh ensures that any changes from the update appear on the View Business Objects page and in object templates. If no details exist for a selected object when you view it, then you're prompted to refresh the object.

You can refresh business objects using the Refresh Object icon and the Refresh All Objects button. When you refresh an object:

- Attribute and flexfield information for the selected business objects is updated.
- Up-to-date templates of METADATA lines are generated for the selected business objects.
- The Overview spreadsheet can be downloaded for the selected business object.
 - a. Click the icon in the Overview column to download the complete definition of an object for offline use. The default file name of the downloaded spreadsheet is GenericBusObjDetails.xlsx, but you can change the name when you save the file.

Note: Whenever you open a downloaded spreadsheet, you're invited to sign in again. If you sign in, then the spreadsheet is updated automatically with latest information as you select each tab. These updates overwrite any changes that you may have made to the spreadsheet. If you prefer to keep your changes, then don't sign in.

When you click the name of a business object on the View Business Objects page, you access information about the object. This information includes:

- The component hierarchy of the object.
- Component details for the currently selected component. These include:

- Its parent component, file discriminator, .dat file, and translation object
- Whether it supports integration keys
- Its supported actions, for example, whether you can delete the component
- Whether it's date effective and, if so, whether you can change the component's first start date and last end date
- Attribute details for each attribute of the currently selected component. These include:
 - Its name, type, and length
 - A translated label and description
 - Whether it's required always or for new objects only
 - Whether it's a key value or can be used as a key value and, if so, the key type
 - For attributes that are keys of type surrogate ID, parent surrogate ID, or foreign-object reference, whether integration keys are also supported
 - The names of any user-key attributes that you can use in place of surrogate IDs
 - Names of lookup types, for attributes that are lookup-validated
- Flexfield attribute details for each of the configured and deployed flexfield segments. These include:
 - The attribute name, flexfield segment label and description
 - The data type, length, and required status
 - The value set used to validate the flexfield segment
- **Note:**
 - Two attributes are supplied for flexfield segments, which are validated by a value set. The attribute that includes `_Display` in the name accepts the meaning, and the other attribute accepts the ID.
 - Some components restrict the flexfield contexts you can bulk-load using HCM Data Loader. Hence, you may not see a flexfield context you created.
- On the View Business Objects page you will now see a red alert icon against object hierarchies that are being withdrawn and a yellow warning icon against object hierarchies that include components that are being withdrawn. You can also see icons against components in the business object hierarchy that are always or conditionally required when creating new records.

If you can load flexfields for the component, then the Flexfield Attributes tab appears. For descriptive flexfields and extensible flexfields, you can see the attributes for a selected flexfield and context. For extensible flexfields, category also appears. When a flexfield context supports multirow entries, one or more flexfield attributes are identified as user keys to reference a single record uniquely. These attributes are identified in the **User Key** column on the Flexfield Attributes tab. This column doesn't appear if the flexfield doesn't support multirow entries.

You can refresh business objects using the **Refresh Object** icon and the **Refresh All Objects** button. When you refresh an object:

- Attribute and flexfield information for the selected business objects is updated.
- Up-to-date templates of METADATA lines are generated for the selected business objects.

Business Rules

Business rules for many objects are included in the *Integrating with Oracle HCM Cloud* guide. For objects that aren't described in the *Integrating with Oracle HCM Cloud* guide, see HCM Data Loader: Business Object Documentation (2020600.1) on My Oracle Support at <https://support.oracle.com>.

Example .dat files for many business objects are also available on Cloud Customer Connect. Select **Navigator > Others > Cloud Customer Connect**. After signing in, select the HCM Cloud Integrations forum on the Human Capital Management page.

Other documents on My Oracle Support provide information about business objects for specific localizations. Many are linked from the document All White Papers for Oracle Fusion HCM (1504483.1). For example:

- Oracle Fusion HCM: HCM Address Validation (2140848.1)
- Oracle Fusion HCM: Person Name Validation (2146270.1)
- Validating National Identifiers in Oracle Fusion Human Capital Management (2159758.1)

Here's a tutorial on using the View Business Objects task which is the data dictionary for all supported business objects:
[View Business Object task](#)

Related Topics

HCM Data Loader Configuration Parameters

The HCM Data Loader configuration parameters control the HCM Data Loader and HCM Spreadsheet Data Loader import and load processes. The default settings of the parameters suit most requirements.

However, you can override these settings to suit the configuration of your environment and the amount of data being loaded. Users can also override most enterprise settings for individual load processes.

In the Setup and Maintenance work area, use the following to review or override the predefined settings for the enterprise:

- Functional Area: HCM Data Loader
- Task: Configure HCM Data Loader

On the Configure HCM Data Loader page, select the HCM Data Loader tab. You can filter the parameters by these category values:

- Availability
- Diagnostics
- File Definition
- Performance
- Scheduling Default

Parameters that are specific to a business object appear in a separate section. This topic describes each of the configuration parameters within its category. Object-specific parameters are described following the category sections.

Note: HCM Spreadsheet Data Loader has its own versions of some configuration parameters, which you manage on the HCM Spreadsheet Data Loader tab. For these parameters, you can set different default values for HCM Data Loader and HCM Spreadsheet Data Loader. Users can override the HCM Spreadsheet Data Loader default values in spreadsheet templates.

Note: A More Information icon is introduced on the names of the parameters which can be overridden by a SET instruction. When you click the orange icon, the More Information pop up appears, and you will see the details of the SET instruction that overrides the data loading behavior of your file.

Note: You can audit changes to the Configure HCM Data Loader parameters. Refer: [How You Audit the Configure HCM Data Loader Parameters](#)

Availability Parameters

Enable Automatic Extended Data Set Retention

Determines whether data sets for business objects that support validate or rollback are automatically retained beyond the standard retention period. You can override the default behavior using a SET instruction. You control how long extended retention data sets are kept for using the Days to Preserve Extended Retention Data Sets parameter.

Default Value: No

Enable Audit Data

Specifies whether audit data is captured for business objects loaded using HCM Data Loader. If you set this parameter to **Yes**, then audit must also be enabled for Oracle Fusion Applications.

You can set this parameter separately for HCM Data Loader and HCM Spreadsheet Data Loader.

Default Value: No

Enable Bulk-Ingestion of Oracle Search Indexes

Enable changes uploaded by HDL and HSDL to be bulk-ingested into Oracle Search indexes. When disabled, incremental ingestion is used.

Default Value: Yes

Enable Configuration of Role-Based Business Object Access

Determines whether access to HCM Data Loader business objects is configurable by role. When enabled you must configure the business objects a custom role can load data with.

Default Value: No

Enable File Generation for Failed File Lines

Determines whether you can generate business object files for the files lines that failed to load.

Default Value: No

Enable HCM Data Loader Post Process Configuration

Determines whether configuration of the processes to be automatically submitted after a data load is enabled.

Default Value: Yes

Enable Load Events for Incremental Updates

Enable events to be triggered for the data supplied. Enabling load events will significantly increase the time it takes for your data to load. You can override this parameter with a set instruction followed by the override value. Place SET instructions before the first METADATA line in your business object file.

Default Value: No

Enable Load Group Size Calculation

Specifies whether the load-group size is to be calculated automatically. Set to **Yes** to enable the optimal load-group size to be calculated based on the number of objects to process and the available concurrent threads. When **Enable Load Group Size Calculation** is **Yes**, the **Load Group Size** parameter doesn't appear on the Schedule Request page. Also, the default value of the **Load Group Size** parameter is ignored.

Default Value: Yes

Enable UI Message Translation

Specifies whether messages on the Import and Load Data page are to appear in the language of the signed-in user. Set to **Yes** to enable translated messages. If this parameter is set to **No**, then messages appear in the language of the user who submitted the import and load request.

Default Value: No

Generate Business Object Mapping Files

Specifies one of three options for generating Business Object Mapping Files. The mapping file describes the HCM Data Loader shape, and you can import this into Oracle Integration Cloud to define transformation and mappings rules.

The default value of **No** means you haven't enabled the option to generate mapping files.

The **Custom only** value lets you generate mapping files for business object dat files you upload to HCM Data Loader. Only the attributes included in the uploaded file will be included in the mapping file. Under the Parameters section, select the Generate Objects Mapping File feature. Refresh the Business Objects task bar until the Download Mapping File link activates and then you can retrieve the map.

A parameter value of **Yes** lets you request a mapping file from the Import and Load Data interface, just like Custom only. In addition, on the View Business Objects page, you can Refresh an individual object and it will automatically generate a mapping file for you. Refresh the page until Last Refreshed indicates today's date. Download the mapping file using the link under Integration Mapping File.

Default Value: No

Purge Person Enabled Key

Key value supplied by Oracle Support to enable the **Purge Person Data in Test Environments** process.

You can't enable and run this process in production environments.

Default Value: None

Enable Load Events for Incremental Updates

Specifies whether incremental load events are enabled when bulk-loading data.

Default Value: No

Note: You can't override the default value for the enterprise. You can include a SET instruction in individual .dat files to enable incremental load events for the objects in the files.

Enable Mandatory Descriptive Flexfield Segment Validation

Determines whether an error is raised when loading descriptive flexfield data, a mandatory segment is without a value, and no default is configured.

Default Value: No

Restrict Access to Security Related Business Objects

Determines whether users with a role granted with the Load HCM Security Data function security privilege can bulk-load data using security related objects. When set to Yes, only users with a role that is granted the Load HCM Security Data function security privilege will be able to bulk-load data using security related objects, such as User and Delegated Role. When set to No, users that aren't granted the Load HCM Security Data function security privilege will encounter warnings when loading security-related data, but the uploads will continue.

Note: Once you enable this feature you can't disable it later. This functionality will be enabled by default in a future release.

For more information, refer [How You Enable Access to HCM Data Loader](#).

Diagnostics Parameters

Complex Error Stack Trace Occurrences by Thread

Maximum number of complex error message occurrences on a processing thread for which stack trace is recorded.

Default Value: 5

Data Error Stack Trace Occurrences by Thread

Maximum number of data error message occurrences on a processing thread for which stack trace is recorded.

Default Value: 2

File Definition Parameters

Data Set Name

For data loads initiated by web service, specifies whether the Oracle WebCenter Content document title or the .zip file name is used as the data set name.

Default Value: File name

File Character Set

Character set used for business object and attachment files.

Default Value: UTF-8

File Delimiter

A value of up to 10 characters used to separate attribute names on METADATA lines and attribute values on data lines.

Default Value: Vertical bar (|)

File Encryption

Default file encryption.

Default Value: None

File Escape Indicator

A value of up to 10 characters used to indicate that the next character is to be ignored on data lines.

Default Value: Backslash (\)

File New Line Indicator

A value of up to 10 characters used to indicate a new line in an attribute value. This value must be preceded by the file escape indicator.

Default Value: n

Flexfields Enabled

Controls whether flexfield data can be loaded.

Default Value: Yes

Purge Audit Data

Specifies whether audit data is to be purged automatically for objects successfully updated or deleted.

Default Value: No

Note: You can't override the default value for the enterprise. You can include a SET instruction in individual .dat files to purge audit data for the objects in the files.

Performance Parameters

Days to Preserve Standard Data Sets Before Archiving

Number of days since a standard data set was last updated before it can be automatically archived. Data sets are only archived if the number of stage table lines exceed the configured limit.

Default Value: 5

Days to Preserve Standard Data Sets Before Deleting

The number of days since a standard data set was last updated before it's automatically and permanently deleted from the stage tables.

Default Value: 30

Days to Preserve Extended Retention Data Sets

Number of days since an extended retention data set was last updated before it's automatically deleted from the stage tables. Only data sets that contain data which can be validated or rolled back can be retained for extended periods.

Default Value: 60

Load Group Size

Number of business objects processed as a single unit of work by a single thread.

Default Value: 100

Maximum Concurrent Threads for Import

Maximum number of threads to run concurrently when importing data to the stage tables.

You can set this parameter separately for HCM Data Loader and HCM Spreadsheet Data Loader.

Default Value: 8

Maximum Concurrent Threads for Load

Maximum number of threads to run concurrently when loading data from the stage tables to the application tables.

This parameter is used in conjunction with the **Load Group Size** parameter. Only the threads required to load your objects based on the **Load Group Size** parameter are created.

Default Value: 8

Maximum Data Lines Before Archiving

The number of data lines that can be held in the stage tables before data sets are automatically backed up.

Default Value: 10000000

Transfer Group Size

Number of file lines processed as a single unit of work by a concurrent thread.

Default Value: 100000

Scheduling Default Parameters

Date-Effective Update Mode

Specifies whether future-dated records should be purged when date-effective objects are loaded.

You can set this parameter separately for HCM Data Loader and HCM Spreadsheet Data Loader.

Default Value: Replace

Delete Source File

Specifies whether to delete source files from the Oracle WebCenter Content server once they have been processed.

Default Value: Yes

Disable Post Process Tasks

This parameter is read-only. You can disable individual postload processes by including a SET instruction in the relevant Worker.dat or Position.dat file.

Default Value: None

File Action

Default file-processing action.

Import and load means that valid objects are loaded automatically once import completes. **Import only** means that your data is imported to the stage tables, but load isn't initiated.

Default Value: Import and load

Initiate Business Object Post Processing

Specifies whether processes registered to run automatically after an object is loaded should run.

You can set this parameter separately for HCM Data Loader and HCM Spreadsheet Data Loader.

Default Value: Yes

Maximum Percentage of Import Errors

Percentage of file lines in error that can occur in a business object before the import process stops for the object.

Default Value: 100

Maximum Percentage of Load Errors

Percentage of business-object instances in error that can occur for a business object before the load process stops.

You can set this parameter separately for HCM Data Loader and HCM Spreadsheet Data Loader.

Default Value: 100

Business Object Parameters: Worker

You can set these parameters separately for HCM Data Loader and HCM Spreadsheet Data Loader.

Calculate Worker Full-Time Equivalent

Specifies whether the full-time equivalent (FTE) value should be calculated automatically for all worker assignments in a Worker.dat file.

Default Value: No

Create Worker Default Work Hour Pattern

Specifies whether a default working-hour pattern should be created automatically for all worker assignments in a Worker.dat file.

Default Value: No

Enable Worker Address Validation

Specifies whether a default address validation should be applied for all worker assignments in a Worker.dat file.

Default Value: Yes

Business Object Parameters: Location

Enable Location Address Validation

Specifies whether a default address validation should be applied for all locations in a Location.dat file.

Default Value: Yes

Business Object Parameters: Participant Enrollment

Evaluate Life Events

Determines whether life events should be evaluated when loading Participant Enrollment data.

Default Value: No

Note: You can override the value of this parameter using the **Configure HCM Data Loader** task in the Setup and Maintenance work area.

Business Object Parameters: Succession Plan

Retain Succession Plan History

Determines whether historical data is retained for changes to succession plans.

You can set this parameter separately for HCM Data Loader, HCM Spreadsheet Data Loader and HCM Spreadsheet Data Loader templates.

Default Value: Yes

Note: You can override the value of this parameter using the **Configure HCM Data Loader** task in the Setup and Maintenance work area.

Here's a tutorial to generate business object specific mapping files that can be imported into the Oracle Integration Cloud in order to transform data into the HCM Data Loader business object file shape: [Generate Oracle Integration Cloud \(OIC\) mapping files for HCM Data Loader \(HDL\)](#)

Related Topics

- [The SET Instruction](#)

How You Audit the Configure HCM Data Loader Parameters

Monitor changes to the parameters that control the behavior and availability of bulk-loading functionality by enabling an audit of the parameters available on Configure HCM Data Loader.

To enable audit for Configure HCM Data Loader:

1. Perform the **Manage Audit Policies** task in the **Setup and Maintenance** work area.
2. Configure the audit parameters for **Oracle Fusion Applications**.
3. Select **HCM Common Architecture** as the **Product**.
4. Check the checkbox for **Configure HCM Data Loader Parameters** and select the node in the hierarchy.
5. Add attribute **Parameter Value**.
6. Save your changes.

Changes made using the **Configure HCM Data Loader** task will now be audited.

To review the audit report for Configure HCM Data Loader:

1. Navigate to **Tools > Audit Reports**.
2. Select the **Product** as **HCM Common Architecture** and the **Business Object Type** as **Configure HCM Data Loader Parameters**.
3. Select a date and click **Search**.
4. On the table toolbar, check the **Show Attribute Details** and **Show Additional Object Identifier Columns** checkboxes.
5. Select **All Attributes** within the **Show Attribute Details** choice list.

How You Define Referenced Objects

Business objects that you're loading may reference a few business objects that you can't load using HCM Data Loader because they're not integration enabled. You define or review these objects in the target Oracle HCM Cloud environment before you load data that references them.

You may have performed this step during implementation of Oracle HCM Cloud. This topic identifies how you define these referenced objects.

This table identifies the main objects of this type. It lists the tasks that you use to manage them and the functional area to which each task belongs in the Setup and Maintenance work area.

Business Object	Functional Area	Task
Application Reference Data Set	Enterprise Profile	Manage Reference Data Sets
Assignment Status Type	Workforce Information	Manage Assignment Status
Business Unit	Organization Structures	Manage Business Unit
Content Type	Workforce Profiles	Manage Profile Content Types
Currency Code	Financial Reporting Structures	Manage Currencies
Element Type	Elements and Formulas	Manage Elements
Legal Entity	Legal Structures	Manage Legal Entity
Official Language Code	Application Extensions	Manage Languages
Profile Type	Workforce Profiles	Manage Profile Types
Schedule	Workforce Information	Manage Work Schedules

Foreign-Object References

When you refer to these objects from objects that you're loading, you use their user keys. (Alternatively, you can use their surrogate IDs, if available.) HCM Data Loader provides business-object documentation for all supported objects. This documentation identifies the user key that you can use for foreign-object references. For example, the Position object includes a reference to the Business Unit object, which isn't integration enabled. The position documentation identifies the business-unit name as its user key. Therefore, when loading a Position object, you can refer to the associated business unit using the business-unit name. To see the supplied documentation for a business object, click the object name on the View Business Objects page.

How You Review Lists of Values

The permitted values of many object attributes are defined in lists of values. Some lists of values are predefined and can't be updated in any way. Others contain some values, but you can also add your own.

In some cases, you can edit or remove predefined values. In Oracle HCM Cloud, lists of values are defined as lookup types. You're recommended to review the predefined lookup types and make any updates before you attempt to load data that uses them. You may have completed this process during implementation of Oracle HCM Cloud.

Reviewing Lookup Types

To manage lookup types, perform relevant tasks in the Setup and Maintenance work area. For example, the Workforce Information functional area contains tasks for managing many lookup types, including person lookups, termination lookups, document lookups, and checklist lookups. To manage person lookup types, perform the **Manage Person Lookups** task. On the Manage Person Lookups page, select a lookup type to edit. The list includes:

- Address types
- Contact relationships
- Ethnicity
- Highest education level
- Honors
- Marital status
- Military rank
- Phone types
- Religion

Ensure that the defined lookup values support the data that you're planning to load.

Related Topics

- [Overview of Lookups](#)

Define the Source-System Owner

If you plan to use source keys, then you must add your source-system owner values to the HRC_SOURCE_SYSTEM_OWNER lookup type before you load data.

You must have the Application Implementation Consultant or Human Capital Management Application Administrator job role or privileges to perform this task.

Use the **Manage Common Lookups** task in the Setup and Maintenance work area. Follow these steps:

1. On the Manage Common Lookups page, search for the lookup type **HRC_SOURCE_SYSTEM_OWNER**.
2. In the search results, select the lookup type to display its lookup codes.
3. In the Lookup Codes section of the page, click the **New** icon to add a lookup-code row.
4. Complete the fields in the row. Ensure that the new code is enabled and the dates are valid for the data that you're loading.
5. Repeat steps 3 and 4 for additional source-system owner values.
6. Click **Save and Close**.

Note: The source-system owner values are saved in uppercase by default.

Related Topics

- [Source Keys](#)

How You Test the HCM Data Loader Process Flow and Connections

In a new environment, you may want to test HCM Data Loader end-to-end processing without creating unwanted data. Once you're sure that HCM Data Loader processing and connections are working as expected, you can load your own data with confidence.

This topic describes how to test HCM Data Loader end-to-end processing.

Testing End-to-End Processing

HCM Data Loader provides:

- A process, **Test HCM Data Loader Process Flow and Connections**
- An object, Example Object

When you submit the **Test HCM Data Loader Process Flow and Connections** process, set the **Action** parameter to **Test HCM Data Loader**. The process:

1. Generates a file of MERGE instructions for the Example Object
2. Adds the ExampleObject.dat file to the **ExampleObject||<date and time>||.zip** file
3. Uploads the .zip file to the **hcm/dataloader/import** directory on the Oracle WebCenter Content server

4. Initiates HCM Data Loader to import and load the file
5. Provides a log file

You can review progress and results on the Import and Load Data page.

The Example Object

When you run the **Test HCM Data Loader Process Flow and Connections** process, the Example Object data is saved to its own tables. No data is saved to the application tables. You can review the structure and attributes of the Example Object on the View Business Objects page in the Data Exchange work area. The example object:

- Has child and grandchild components and a translation object
- Supports all key types
- Supports Create and Delete actions
- Has attributes of most data types
- Has lookup-validated attributes

You can generate a template file for the Example Object.

Note: Although an Example Object Translation object exists, no translation data is generated when you run the **Test HCM Data Loader Process Flow and Connections** process.

Deleting the Test Data

You're recommended to run the **Test HCM Data Loader Process Flow and Connections** process periodically to delete the test data and associated data sets. When you submit the process, set the **Action** parameter to **Delete test data**. In this case, the process generates a file of DELETE instructions, adds it to the **DeleteExampleObject||<date and time>|.zip** file, and uploads it.

How You Audit Objects Loaded in Bulk

You can audit changes made to objects when you bulk load them using either HCM Data Loader or HCM Spreadsheet Data Loader. When audit is enabled, the creation, update, and deletion of objects are audited.

This topic describes how to enable and manage audit for bulk-loaded objects.

Tip: Enable audit only for incremental updates and not during data migration.

Enabling Audit for Bulk-Loaded Objects

Auditing is optional. To audit bulk loading of objects, you must:

1. Perform the **Manage Audit Policies** task in the Setup and Maintenance work area to select objects and their attributes for audit.
2. Enable auditing in general on the Manage Audit Policies page.
3. Enable auditing of bulk-loaded data, either for the enterprise or in individual .dat files and spreadsheet templates.

Enabling Audit for HCM Data Loader

The **Enable Audit Data** parameter on the HCM Data Loader parameters tab of the Configure HCM Data Loader page controls audit for the enterprise. By default, **Enable Audit Data** is set to **No**, but you can override that setting. Alternatively, you can enable audit selectively by including the following SET instruction in relevant .dat files:

```
SET ENABLE_AUDIT_DATA Y
```

Enabling Audit for HCM Spreadsheet Data Loader

The **Enable Audit Data** parameter on the HCM Spreadsheet Data Loader parameters tab of the Configure HCM Data Loader page controls audit for the enterprise. By default, the **Enable Audit Data** parameter is set to **No**, but you can override that setting. Alternatively, you can enable audit selectively by setting the **Enable Audit Data** parameter in the relevant spreadsheet template to **Yes**.

When to Enable Audit for the Enterprise

Setting the **Enable Audit Data** configuration parameters to **Yes** for the enterprise may affect performance. Consider enabling audit in this way only for ongoing integrations.

Purging Audit Data

You may want to purge audit data, typically when disposing of personally identifiable information in worker records. By default, the purging of audit data is disabled for the enterprise and you can't override this setting at the enterprise level. However, you can include the following SET instruction in individual .dat files:

```
SET PURGE_AUDIT_DATA Y
```

In spreadsheet templates, you can set the **Purge Audit Data** parameter to **Yes**. These instructions purge the audit data for the business objects in the .dat file or spreadsheet.

Restrictions exist on enabling audit and purging data at the same time. You can't include both of the following SET instructions in a single .dat file:

```
SET ENABLE_AUDIT_DATA Y
SET PURGE_AUDIT_DATA Y
```

Similarly, you can't include both of the following parameter values in a spreadsheet template:

Parameter	Value
Enable Audit Data	Yes
Purge Audit Data	Yes

To purge audit data when auditing is enabled for the enterprise, you must disable audit in the same .dat file or spreadsheet template. That is, you must do one of the following:

- Include a `SET ENABLE_AUDIT_DATA N` instruction in the same .dat file.
- Set the **Enable Audit Data** parameter in the spreadsheet template to **No**.

Correcting Load Errors for Audited Objects

When audit is enabled for an object, you can't correct load errors for it in the HCM Data Loader user interface. For example, your .zip file may contain both Worker and Job objects. If you include the `SET ENABLE_AUDIT_DATA Y` instruction in the Worker.dat file only, then you can't correct load errors from that file in the user interface. However, you can correct errors in the Job objects, as audit isn't enabled in the Job.dat file.

Related Topics

- [Set Spreadsheet Template Parameters](#)
- [Enable Audit for Oracle HCM Cloud Business Objects](#)
- [The SET Instruction](#)
- [HCM Data Loader Configuration Parameters](#)

FAQs for Preparing to Use HCM Data Loader

Can I load data in multiple languages?

Yes. Language packs for the required languages must be installed. For cloud implementations, requested language packs may be installed before the environment is delivered. Alternatively, you can request language packs by raising a service request (SR).

4 Data Preparation

Guidelines for Preparing the Source Data

This topic describes some ways in which you can prepare your data for loading to Oracle HCM Cloud and ensure that it loads successfully.

Reviewing and Cleansing the Source Data

Identify the business objects that you're planning to upload to Oracle HCM Cloud and their source systems. Review this source data, and verify that it's both accurate and current. Correct any problems before you attempt to extract it. For example:

- Ensure that a manager is identified for every worker and that the information is accurate.
- For jobs and positions, ensure that correct job codes and titles exist in the source systems.
- For worker history, establish the accuracy of any historical data. Understand whether all historical data must be uploaded or just key events, such as hire, promotion, and termination.

Preparing the source data in this way minimizes the problems that can occur when you upload data to Oracle HCM Cloud. It also makes it less likely that you load inaccurate data to the new environment.

Extracting Data from the Source System

You must define mappings between your source data and the Oracle HCM Cloud business-object model by comparing source and target attributes. To review the structure of a business object in Oracle HCM Cloud:

1. Open the **View Business Objects** page in the **Data Exchange** work area.
2. Search for the object by entering its name in the **Business Object** query-by-example field and pressing **Enter**.
3. Click the object name to open the Business Object Details page for the object. This page provides comprehensive information about the object's structure and attributes.

You must also define the transformation logic and build extraction routines. You can use tools that are native to the source system, such as PL/SQL in Oracle E-Business Suite or SQR in Oracle PeopleSoft. Alternatively, you can use an Extract, Transform, and Load tool, such as Oracle Data Integrator or PowerCenter Informatica.

Validating the Source Data Before Upload

HCM Data Loader validates your data during both the import and load phases of data loading. The Data File Validator tool, available from My Oracle Support, can perform most of the data-formatting validations before you attempt to load the data. You run this utility in your source environment to test your generated .dat files. The utility generates a list of validation errors in HTML format. You can correct the errors in the .dat files before you load them.

You can download the Data File Validator tool from the My Oracle Support document, Data File Validator Tool for HCM Data Loader (document ID 2022617.1).

Here's a tutorial on creating and maintaining data with HCM Data Loader [Create and Maintain Data with HDL](#)

General Data Considerations for HCM Data Loader

This topic describes some general aspects of data preparation. Follow these rules to ensure successful data upload.

Unchanged Attributes

When you update existing data in Oracle HCM Cloud, you supply:

- A unique identifier for the record being updated
- The attributes that have changed

Any attributes that you omit retain their current values. For performance reasons, you're recommended not to include attributes that haven't changed.

Note: For date-effective records, all attributes that you supply are updated for the specified date range.

Null Attribute Values

To set an attribute value explicitly to null, you must supply the #NULL token as the attribute value. You can't simply leave the attribute blank.

Lookup-Validated Attributes

For nonflexfield attributes that are defined in Oracle HCM Cloud as lookup types, you can specify either the lookup code or its meaning. For example, you can specify the sex of a person using either the lookup codes or the lookup meanings shown in this table.

Lookup Code	Lookup Meaning
M	Male
F	Female

You're recommended to use the lookup code, as the lookup meaning can be translated. The language of the lookup meaning must match the language of the user who's uploading the data.

Different rules apply to flexfield attributes that are defined as lookup types.

Note: You're recommended to use the lookup code, as you must supply the lookup meaning in the language configured for the FUSION_APPS_HCM_ESS_LOADER_APPID elevated user, which may differ from language of the session user.

Numeric Values

For numeric values, only the decimal separator is supported. Don't include currency symbols, scientific notation, or thousands separators. To set an existing numeric value to null, supply the #NULL token as the attribute value.

Date and Time Attributes

The expected formats for date and time values are shown in this table.

Date or Time	Format
Date	YYYY/MM/DD
Time	YYYY/MM/DD HH24:MI:SS

For example: 2016/11/05 14:20:00

To set an existing date or time value to null, supply the #NULL token as the attribute value.

Related Topics

- [Guidelines for Loading Flexfield Data](#)

How You Load Images, Attachments, and Large Strings

You can load both Character Large Objects (CLOB) and Binary Large Objects (BLOB) using HCM Data Loader. However, the way in which you provide data for these objects is unique to these attributes.

Instead of supplying the data directly in the data (.dat) file, you supply it in a separate file. You supply the name of this file as the value of the relevant attribute in the data file.

This example shows a data file for the Document Attachment component. The **File** attribute references a text attachment file in each MERGE line.

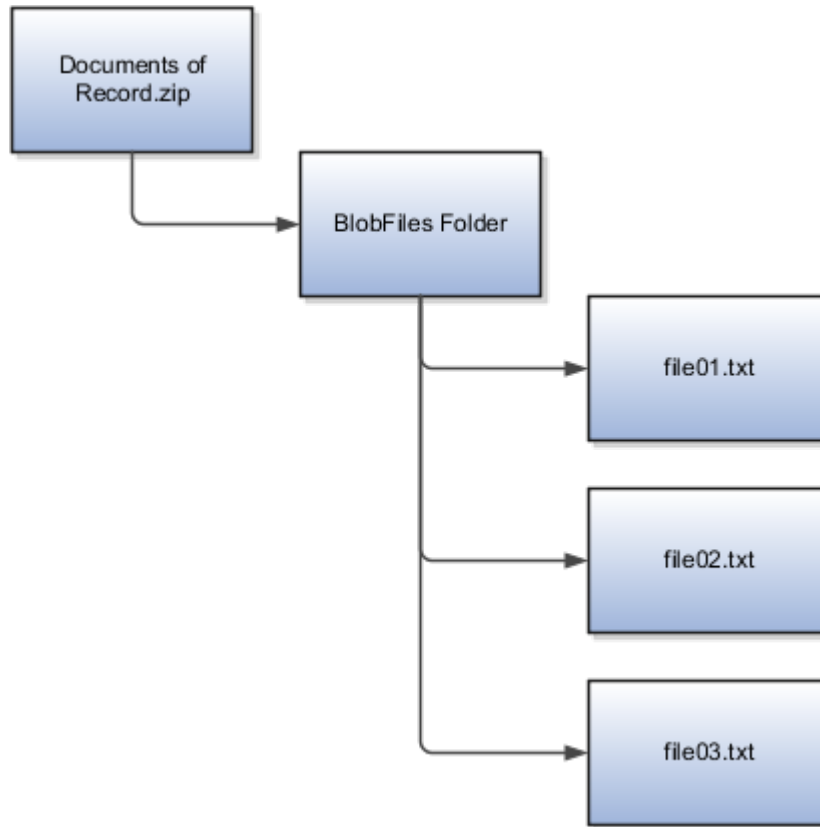
```
METADATA|DocumentAttachment|DocumentType|File|PersonNumber|...  
MERGE|DocumentAttachment|Drivers License|file01.txt|23901|...  
MERGE|DocumentAttachment|Drivers License|file02.txt|64235|...
```

This approach is used because data for these data types can be very large. Also, data to be loaded directly rather than by attachment may need new-line characters, making it complex to include in the business-object data file.

How to Supply CLOB and BLOB Files

To load data into a CLOB attribute, you supply the data in a separate file. You place the file in a **ClobFiles** folder in the same .zip file as the business-object data file. Similarly, to load data or upload an attachment to a BLOB attribute, you supply the data or file to attach in a **BlobFiles** folder. The data type of the attribute that's used to load a large object or attachment determines which folder to use.

The business-object documentation specifies the data types of all attributes. For example, the File attribute of the Document Attachment component has the BLOB data type. Therefore, you place referenced files in the BlobFiles folder,



as shown in this figure.

File names in the **ClobFiles** and **BlobFiles** folders can include UTF-8 single-byte characters. For example, file names can include the alphanumeric characters a through z, A through Z, and 0 through 9, underscore (_), hyphen (-) and parentheses (). Most file-name extensions are supported, though CLOB data is usually supplied in text (.txt) files.

Here's a tutorial on loading images and attachments using HDL: [Loading Images with Attachments with HCM Data Loader \(HDL\)](#)

How You Supply Source Keys

The source key is made up of two attributes, SourceSystemOwner and SourceSystemId. If you use a source key to identify a record being merged, then you can also use source keys to identify integration-enabled foreign objects.

For example, you can identify the parent record of a child component using a source key. You can't use source keys for foreign-object references if you aren't also using a source key for the local record.

This topic describes how to supply source keys for both local records and foreign-object references. It also describes how default source keys are constructed.

Defining Source-System Owner Values

The SourceSystemOwner attribute is common to all source keys supplied in a record. Therefore, all foreign objects that you identify using source keys must have the same SourceSystemOwner value as the record being merged. You must

define the SourceSystemOwner value by updating the HRC_SOURCE_SYSTEM_OWNER lookup type before you use source keys.

Specifying Source Keys for Local Records

To identify a local record uniquely using a source key, you supply values for both the SourceSystemId and SourceSystemOwner source-key attributes. This example shows how to identify a job object using a source key.

```
METADATA|Job|SourceSystemId|SourceSystemOwner|JobCode|JobName|SetCode|EffectiveStartDate|EffectiveEndDate  
MERGE|Job|12349|EBS-UK|SE|Software Engineer|COMMON|2010/01/01|4712/12/31
```

Specifying Source Keys for Foreign-Object References

To use a source key for a foreign-object reference, you append the hint (SourceSystemId) to the surrogate ID attribute for the foreign object. This example shows how to identify a Job object in an assignment record using a source key. JobId is the surrogate ID attribute for the Job object, to which you append the hint (SourceSystemId) in the METADATA line. You must have created the Job object using HCM Data Loader with the supplied source key.

```
METADATA|Assignment|SourceSystemId|SourceSystemOwner|JobId(SourceSystemId)|EffectiveStartDate|  
EffectiveEndDate  
MERGE|Assignment|234234|EBS-UK|12349|2013/01/01|4712/12/31
```

Note: You can use source keys for foreign objects only if those objects are integration-enabled. The integrated business-object documentation identifies which objects are integration enabled.

If your foreign-object has a different source system owner for the local record, you must also supply the source system owner of the foreign-object reference. To do this, you append the hint (SourceSystemOwner) to the surrogate ID attribute for the foreign object. This example extends the example above and shows how to identify a Job object in an assignment record, where the source system owner for the Job and Assignment are different.

```
METADATA|Assignment|SourceSystemId|SourceSystemOwner|JobId(SourceSystemId)|JobId(SourceSystemOwner)|  
EffectiveStartDate|EffectiveEndDate MERGE|Assignment|234234|EBS-UK|12349|VISION|2013/01/01|4712/12/31
```

Default Source Keys

If you specify no source key when you create a record using HCM Data Loader, then a default source key is generated. The default SourceSystemOwner value is FUSION and the SourceSystemId is the surrogate ID. You can use source keys to identify records with default source keys. You can also update existing source keys, including default source keys, using the Source Key object. However, no process exists for extracting source-key information.

Related Topics

- [Source Keys](#)
- [Define the Source-System Owner](#)
- [How You Update Source Keys](#)

How You Update Source Keys

When you load data using HCM Data Loader, you can provide a source key. The source key is a value, usually generated from a legacy environment, that identifies the record uniquely in that environment.

You can use the source key to refer to the object both when maintaining it and when referring to it from another object. If you specify no source key, then a default source key is generated. You can update both default and locally defined source keys for integration-enabled objects. This topic describes how to update source keys.

Loading Updates to Source Keys

To update the source key associated with any record, you load a SourceKey.dat file. In the file, you supply both a reference to the record to update and the new source-key value.

This example SourceKey.dat file updates the source key for a person address that's identified by its existing source key. The **BusinessObject** and **Component** attributes are used to identify the type of record being updated. For both attribute values, you supply the relevant file discriminator.

```
METADATA|SourceKey|BusinessObject|Component|OldSourceSystemId|OldSourceSystemOwner|NewSourceSystemId|
NewSourceSystemOwner
MERGE|SourceKey|Worker|PersonAddress|2342|FUSION|1422-HOME|VISION
```

How You Supply User Keys

User keys are specific to the business object component that you're loading. The Business Object Details page in the View Business Objects task identifies the user-key attributes available for all business-object components and foreign-object references.

The user-key attributes are required when you create a record. They're also required when you update it, unless you supply a different key type to identify the record uniquely. This topic describes how to use user keys to identify local records and refer to foreign objects. It also describes how changes to user-key values affect their usefulness as keys.

Specifying User Keys for Local Records

A user key can be made up of multiple attributes. You must supply them all if you're not using another key type, such as a source key, to identify the record. This example shows how to identify a Job object by its user key, which is made up of the JobCode and SetCode attributes.

```
METADATA|Job|JobCode|JobName|SetCode|EffectiveStartDate|EffectiveEndDate
MERGE|Job|SE|Software Engineer|COMMON|2010/01/01|4712/12/31
```

Specifying User Keys for Foreign-Object References

In this example, an Assignment object is uniquely identified by its source key. However, a user key is used for the foreign-object reference to the associated Job object.

```
METADATA|Assignment|SourceSystemId|SourceSystemOwner|JobCode|SetCode|EffectiveStartDate|EffectiveEndDate
MERGE|Assignment|234234|EBS-UK|SE|COMMON|2013/01/01|4712/12/31
```

Managing Changed User-Key Values

The values of some user keys aren't fixed. For example, you can change the name of an organization or location. As user keys can change, using them for historical references is difficult. If you're loading date-effective history for a business-object component where the user key has changed, then you must also supply a source key. This approach enables HCM Data Loader to identify related date-effective records correctly to form the object that you're loading.

Related Topics

- [User Keys](#)
- [Sources of Business-Object Information](#)

Examples of Supplying Oracle Fusion Surrogate IDs

An Oracle Fusion surrogate ID is assigned automatically to a new record when you save it to the database. Surrogate IDs are specific to the business-object component that you're loading.

The integrated business-object documentation identifies the surrogate ID attribute for all business-object components and foreign-object references. This topic describes how to use surrogate IDs to identify local records and provide foreign-object references.

Note: Oracle HCM Cloud customers are unlikely to have access to the Oracle Fusion surrogate ID.

Supplying Surrogate IDs for Local Records

This example shows how to use a surrogate ID to identify a Job component. For this component, the **JobId** attribute is its surrogate ID.

```
METADATA|Job|JobId|JobName|EffectiveStartDate|EffectiveEndDate  
MERGE|Job|13413|Software Engineer - Java|2013/01/01|4712/12/31
```

Supplying Surrogate IDs for Foreign-Object References

This example shows an Assignment component that's uniquely identified using a source key. The record includes a foreign-object reference to the associated Job object, which is identified by its surrogate ID, JobId.

```
METADATA|Assignment|SourceSystemId|SourceSystemOwner|JobId|EffectiveStartDate|EffectiveEndDate  
MERGE|Assignment|234234|EBS-UK|13413|2013/01/01|4712/12/31
```

Related Topics

- [Oracle Fusion GUIDs and Surrogate IDs](#)

Examples of Supplying Oracle Fusion GUIDs

The Oracle Fusion GUID (Globally Unique Identifier) is a hexadecimal value that's assigned automatically to a record when it's saved to the database. This topic shows how to use GUIDs to identify local records and provide foreign-object references.

Supplying GUIDs for Local Records

When you supply a GUID value to identify the record being merged or deleted, you use the attribute name GUID, regardless of the business-object component. This example shows how to specify a GUID value to identify a Job component.

```
METADATA|Job|GUID|JobName|EffectiveStartDate|EffectiveEndDate  
MERGE|Job|2342UJFHI2323|Software Engineer - Java|2013/01/01|4712/12/31
```

Supplying GUIDs for Foreign-Object References

To use a GUID for a foreign-object reference, you append the hint (GUID) to the surrogate ID attribute for the object to which you're referring. This example shows an Assignment component that's identified using source keys. The record includes a foreign-object reference to the associated Job object, which is identified by its GUID. JobId is the surrogate ID attribute for the Job object.

```
METADATA|Assignment|SourceSystemId|SourceSystemOwner|JobId (GUID) |EffectiveStartDate|EffectiveEndDate  
MERGE|Assignment|234234|EBS-UK|2342UJHFI2323|2013/01/01|4712/12/31
```

You can use GUIDs for foreign objects only if they're integration enabled. The integrated business-object documentation identifies which foreign objects are integration enabled.

Related Topics

- [Oracle Fusion GUIDs and Surrogate IDs](#)

How You Manage Reserved Characters

A few characters are reserved in the context of an HCM Data Loader data file. That is, they have a specific meaning by default, and you can't include them in attribute values unless you identify them explicitly as data.

This topic explains how to use the reserved characters in data that you're uploading. It also explains how to override the default reserved characters.

Using Reserved Characters in Attribute Values

By default, these characters are reserved:

- Delimiter (vertical bar |)
- Newline character (n)
- Escape (backslash \)

To include the newline (n) and vertical bar (|) characters in attribute values, you precede them immediately with the escape character (backslash \). For example:

```
METADATA|Address|AddressLine1  
MERGE|Address|The Stables\|Main Allan
```

This entry enables the vertical bar to appear in the attribute value for address line 1:

The Stables|Main Allan

To include the newline character in a value, you specify \n. For example:

```
METADATA|Address|AddressLine1
MERGE|Address|The Stables\nMain Allan
```

This entry results in the following value for address line 1:

The Stables

Main Allan

Overriding the Reserved Characters

You can override the reserved characters for a file using the SET file-line instruction, which must appear before any METADATA lines in the file. The SET commands for overriding reserved characters have this format:

```
SET FILE_ESCAPE <new_value>
SET FILE_DELIMITER <new_value>
SET FILE_NEW_LINE <new_value>
```

The new value can be up to 10 characters. For example, you could set the newline character to newline and the file delimiter to comma (,) using the following SET commands:

```
SET FILE_DELIMITER ,
SET FILE_NEW_LINE newline
```

In this case, METADATA and MERGE lines could appear as follows:

```
METADATA,Address,AddressLine1
MERGE,Address,TheSteading\nnewlineKier Allan
```

Tip: You can also override the reserved characters for the enterprise by setting relevant configuration parameters. For example, set the **File Delimiter** configuration parameter to override the default file delimiter for the enterprise.

Related Topics

- [The SET Instruction](#)

How You Load and Maintain Translated Objects

In environments where multiple languages are enabled, you can use HCM Data Loader to upload translated objects.

You specify the character set of the data file by naming any Java-supported character set on the **File Character Set** configuration parameter. The default character set is UTF-8.

Loading Translated Objects

Loading translated objects is a two-stage process:

1. You create the object by loading the base-language version. At this stage, translation records are created for all enabled languages, but they hold base-language versions of any translatable values. For example, if US English is your base language, then translation records hold US English versions of translatable values.
2. You load the translated values as corrections to the base-language object. To perform this step, you use data-file templates that are provided specifically for translation. One translation data-file template is provided for each business-object component that includes translatable values.

For example, you may create the Sales Manager job in an environment where US English is the base language. If French, German, and Spanish are also enabled, then the object is created as shown in this table.

Language	Source Language	Job Name
US English	US English	Sales Manager
French	US English	Sales Manager
German	US English	Sales Manager
Spanish	US English	Sales Manager

Once this object exists, you can load a single translation data file (JobTranslation.dat) to correct the French, German, and Spanish versions of the job name. Alternatively, you can load a translation file for each language if you prefer. You can deliver translation files either in the same .zip file as the original object or separately. However, you can't deliver them before the base-language object exists.

Updating Translated Data

To update existing translated data, you can upload just the relevant translation file, provided that the object isn't date effective.

You must also update the base-language object if:

- The object is date effective.
- The object currently has no date-effective record with the same effective start date as the new translation value.

This requirement exists to ensure that the effective dates of the base-language and translation objects remain the same.

You can't delete translation objects in isolation. Translation objects are deleted automatically when you delete the associated object. For example, when you delete a Job Family object any associated translation objects are deleted automatically.

Translation-File Discriminators

Unique file discriminators exist for the translation files and are identified in relevant files. For example, the file discriminator for the file JobTranslation.dat is JobTranslation.

This example shows a Job.dat file followed by the associated JobTranslation.dat file.

```
METADATA | Job | SourceSystemOwner | SourceSystemId | EffectiveStartDate | EffectiveEndDate | JobCode | Name | ActiveStatus
MERGE | Job | EBS-UK | JB2ACC44 | 2010/01/01 | 2014/04/04 | ACADM | Accounts Administrator | A
MERGE | Job | EBS-UK | JB2ACC44 | 2014/04/05 | 4712/12/31 | ACADM | Accounts Clerk | A

METADATA | JobTranslation | SourceSystemOwner | SourceSystemId | EffectiveStartDate | EffectiveEndDate | Language | Name
MERGE | JobTranslation | EBS-UK | JB2ACC44 | 2010/01/01 | 2014/04/04 | ES | Administrador de Cuentas
MERGE | JobTranslation | EBS-UK | JB2ACC44 | 2014/04/05 | 4712/12/31 | ES | Cuentas Clerk
```

Related Topics

- [HCM Data Loader Configuration Parameters](#)

How You Include Reference Values in Data Files

You can include reference values in each data line in a file.

They enable you to record reference data for the file line, which could include values already supplied on the file line or within the object, or source-system information, such as the database-table name, column names, and attribute values from the source system. The Object Status page within the Import and Load Data interface, displays the reference values supplied against the top-level record in your object. Navigate to Object Details page to view the reference information supplied against each file line that constructs your object. You can create OTBI reports to include these values using the HCM Integrations Real Time subject area. The reference names and values are available in the Imported Objects sub-folder within the HCM Data Loader folder.

For example, when loading element entries, you may want to filter and reconcile by the assignment number, element name and amount. You can supply these values against the ElementEntry record, even though the amount is loaded using the ElementEntryValue record type. This would then allow you to filter by the assignment number and element name, to check the status of the element entry, while reviewing the amount specified for the element entry.

You aren't restricted as to the values you supply, there is no validation of these attributes. The data supplied to them in held only the staging tables, it is not saved to the application tables.

References are optional. This topic describes how to construct reference attributes, which are made up of a name and a value.

Reference Names

You specify reference context and column names in the METADATA line. To specify a source-system database-table name or reference context, you add the following entry to the METADATA line:

```
SourceRefTableName=<table name>
```

You can specify up to 10 reference column names in the same METADATA line using the SourceRef001 to SourceRef010 tags.

Examples:

To supply details of the source record on your legacy system:

```
METADATA|Job|SourceRefTableName=PER_JOBS|SourceRef001=JOB_ID|SourceRef002=EFFECTIVE_START_DATE|  
SourceRef003=EFFECTIVE_END_DATE
```

To supply reconciliation and reporting information:

```
METADATA|ElementEntry|SourceRef001=Assignment Number|SourceRef002=Element Name|SourceRef003=Effective Start  
Date|SourceRef004=Amount
```

Reference attributes can appear anywhere in the METADATA line after the instruction and discriminator values.

Reference Values

Supply the reference values on each data line, ensuring that they appear in the order specified on the METADATA line. In data lines, you must leave the SourceRefTableName attribute blank. This value appears in the METADATA definition only. For example:

```
METADATA|Job|SourceRefTableName=PER_JOBS|SourceRef001=JOB_ID|SourceRef002=EFFECTIVE_START_DATE|  
SourceRef003=EFFECTIVE_END_DATE  
MERGE|Job||135|2010/01/01|4712/12/31  
MERGE|Job||136|2010/01/01|4712/12/31
```

Related Topics

- [How You Review and Correct Errors](#)

Data Deletion Using HCM Data Loader

You can delete many objects using HCM Data Loader, regardless of whether they were loaded using HCM Data Loader. This topic explains how to delete an object or its components and identifies any restrictions.

Tip: As you can't recover a deleted object, try to correct your data instead of deleting and recreating it.

What You Can and Can't Delete

You can delete:

- Most complete business objects, except Worker.
- Most individual business-object components.
- Some child components of the Worker object. For example, you can delete the Person Email component of a worker object.

When you delete a parent object, its child components and any translation objects are also deleted. For example, to delete a Grade object and its child components, you create a DELETE instruction for the Grade discriminator. To delete only a Grade Rate Value child component, you create a DELETE instruction for the GradeRateValue discriminator.

You can't delete:

- Individual date-effective records
- Individual translation objects
- Worker objects

Note: Before you delete an object, ensure that other business objects don't refer to it.

For information about whether you can delete a component, see the business-object details on the View Business Objects page in the Data Exchange work area.

DELETE Instructions

To delete an object, include the DELETE instruction in the relevant data file. For example, you could delete a Job Family object by including these lines in the JobFamily.dat file:

```
METADATA|JobFamily|EffectiveStartDate|EffectiveEndDate|JobFamilyName  
DELETE|JobFamily|2012/10/01|4712/12/31|Sales01
```

These rules apply:

- You can't include DELETE instructions in translation data files.
- You must not supply a DELETE instruction for a record that has a MERGE instruction in the same file. HCM Data Loader doesn't know which instruction to process first.

Deleting Date-Effective Objects

To delete a date-effective object identified by user keys, you must supply both the effective start and effective end dates. If the object is identified using any of the other key types, then the effective start and end dates are optional.

Here's a tutorial on creating and maintaining data with HCM Data Loader [Create and Maintain Data with HDL](#)

5 Loading Date-Effective Data

Overview of Loading Date-Effective Objects

Many Oracle HCM Cloud objects are date-effective. That is, they retain a history of changes, each of which has effective start and end dates. Professional users can retrieve and edit the version of an object as of a specified date.

You can load date-effective objects and their date-effective history using HCM Data Loader. This topic provides some general rules about loading date-effective objects.

Attributes of Date-Effective Records

This table introduces the attributes that occur in date-effective records.

Attribute	Description
EffectiveStartDate	The start date for the attribute values. This value is required for all date-effective records.
EffectiveEndDate	The end date for the attribute values. If you leave this value blank, then the date-effective record continues by default to the end of time.
EffectiveSequence	When multiple changes per day occur, this attribute identifies their order.
EffectiveLatestChange	For objects with multiple changes per day, this attribute identifies the latest record for the effective start date.

Supplying Date-Effective History

You can decide how much history to load for new objects, but the history that you provide must be complete and valid. Although you can supply date-effective records in any order in the .dat file, no break in the dates can occur. When you create a date-effective object, only the first date-effective record is processed as a new record. Later-dated records are updates to the first record. To retain values from the previous date-effective record, you can either leave those attribute values blank or repeat the value that's to continue.

Note: Leaving an attribute value blank doesn't set the attribute to null. To change a value to null, you must supply the #NULL token as the attribute value.

Key Values

HCM Data Loader groups records into logical objects, where a logical object is one occurrence of the business object, such as a Worker. The records belonging to a logical object are identified by the unique key for the component. Therefore, the key value must be the same throughout the date-effective history. You can use any of the four key types.

Creating Date-Effective Objects

When you create business objects containing date-effective components using HCM Data Loader:

- Include the `SET PURGE_FUTURE_CHANGES Y` instruction at the start of the data file. This instruction sets the HCM Data Loader date-effective maintenance mode to Replace. In this mode, which is the default, the contents of the data file replace any existing data.
- If you're including object history, then supply the `#NULL` token for all attributes that must have null values.

Updating Date-Effective Objects

When you load updates to objects that contain date-effective components:

- Include the `SET PURGE_FUTURE_CHANGES N` instruction at the start of the data file. This instruction sets the HCM Data Loader date-effective maintenance mode to Retain. In this mode, existing future-dated changes are retained. If you don't include this instruction, then any existing future-dated changes for the date range of your updates are purged.
- Always supply the effective start date of each change. The effective end date is optional. However, you must set the effective end date to `#RETAIN` if you don't want to correct any future-dated records. Leave the effective end date null if you want a change to apply until the end of time.

Note: When you update date-effective objects, the values that you supply are applied to every record in the specified date-range. However, some attributes, such as **ActionCode**, are protected from update.

Related Topics

- [How You Load Objects with Multiple Changes per Day](#)
- [Options for Updating Date-Effective Objects with Future-Dated Records](#)
- [Options for Carrying Forward Changed Attribute Values in Retain Mode](#)
- [Examples of Changing First Effective Start Date and Last Effective End Date](#)

How You Load Objects with Multiple Changes per Day

For most date-effective objects, only one record exists for a specific date. Therefore, the object history shows only the last update to each attribute on any day.

However, for some objects, such as Assignment, a record exists for each update in a day. Such objects include an effective sequence number to show the update order. Changes are cumulative, and the record with the highest effective sequence number is available to transactions for that date. This topic describes how to load and maintain objects that support multiple changes per day.

Effective Sequence Number

When you load date-effective history for an object with multiple changes on a single date, the **EffectiveSequence** value must start at 1 and increase sequentially. You can't repeat an **EffectiveSequence** value for the same logical object on the same date, nor can gaps exist in the sequence. If only one change exists for an effective start date, then **EffectiveSequence** must be 1.

Note: Don't leave **EffectiveSequence** blank when you're loading multiple changes for a single day. Without this information, the order in which records with the same effective start date are processed is undefined.

Effective Latest Change

When you report multiple changes on the same effective start date, the **EffectiveLatestChange** attribute of the last record in the sequence must be **Y**. For all earlier records, the attribute must be **N**. **EffectiveLatestChange** is a required attribute for records that support multiple changes per day. If only one change exists on an effective start date, then **EffectiveLatestChange** must be **Y**.

Effective Start and End Dates

The effective end date of any record with **EffectiveLatestChange** set to **N** must be the same as its effective start date.

Loading Date-Effective History with Multiple Changes Per Day: Example

This example loads date-effective history for an Assignment object.

```
METADATA|Assignment|SourceSystemId|EffectiveStartDate|EffectiveSequence|EffectiveLatestChange|EffectiveEndDate|...
MERGE|Assignment|2724|2010/06/08|1|Y|2012/03/03|...
MERGE|Assignment|2724|2012/03/04|1|N|2012/03/04|...
MERGE|Assignment|2724|2012/03/04|2|N|2012/03/04|...
MERGE|Assignment|2724|2012/03/04|3|Y|2012/06/01|...
MERGE|Assignment|2724|2012/06/02|1|Y||...
```

This example observes these rules:

- No gaps exist in the dates.
- The unique key is identical for all date-effective records.
- **EffectiveSequence** starts at **1** for all new effective start dates and is sequential for matching effective start dates.
- **EffectiveLatestChange** is **Y** when the record is the last change on an effective start date. Otherwise, **EffectiveLatestChange** is **N**.
- The **EffectiveEndDate** matches the **EffectiveStartDate** when **EffectiveLatestChange** is **N**.

Updating Objects with Multiple Changes Per Day

When supplying a single update to an existing object that support multiple changes per day, you may not know the next available sequence number. In this case, leave the **EffectiveSequence** attribute blank. The next number in the existing sequence is generated automatically.

To correct an existing record, you must supply the existing values for these attributes to identify the record correctly:

- Effective Start Date
- Effective End Date
- Effective Sequence
- Effective Latest Change

For example:

```
METADATA|Assignment|SourceSystemId|EffectiveStartDate|EffectiveSequence|EffectiveLatestChange|  
EffectiveEndDate|...  
MERGE|Assignment|2724|2010/06/08|3|N|4712/12/31|...
```

Note: Don't leave **EffectiveSequence** blank when you're loading multiple changes for a single day. Without this information, the order in which records with the same effective start date are processed is undefined. However to leave the **EffectiveSequence** blank, ensure that you include the SET PURGE_FUTURE_CHANGES N instruction in the data file. For more information on this instruction, refer Example of Retaining Attribute Values in Future-Dated Records. Additionally, for details on supplying values when updating Date-effective objects, refer Updating Date-Effective Objects.

Here's a tutorial on supplying date-effective history, corrections and updates. :[Making Date-Effective Changes with HCM Data Loader \(HDL\)](#)

Options for Updating Date-Effective Objects with Future-Dated Records

When you update a date-effective object, your changes may affect more than one of its existing date-effective records. Also, some of the affected records may have effective start dates that are later than the effective start date of the update.

Such records are known as future-dated records. This topic describes how updates to future-dated records are processed in each of the maintenance modes. It also shows how to set the maintenance mode.

This table shows the date-effective history of a Job object.

Effective Start Date	Effective End Date	Job Code	Job Name	Regular Temporary	Full Part Time	Active Status
2010/06/08	2012/01/09	ACC1	Accounts Clerk	T	PART_TIME	A
2012/01/10	2012/03/03	ACC1	Accounts Clerk	T	FULL_TIME	A
2012/03/04	4712/12/31	ACC1	Accounts Administrator	T	FULL_TIME	A

If you update the object specifying an effective start date of 2011/01/01, then a new date-effective record is generated for 2011/01/01. This update changes the effective end date of the 2010/06/08 record. It also affects both the 2012/01/10 and 2012/03/04 records. The exact effect on these existing future-dated records depends on:

- The maintenance mode, which can be either Retain or Replace
- The effective end date value, if any, that you supply

Retain Mode or Replace Mode

This table summarizes the combined effects of the maintenance mode and effective end date on existing future-dated records.

Mode	Effective End Date Value	Description
Retain	#RETAIN	Existing attribute values in future-dated records are retained. Your updates don't affect these values. All existing records are retained.
Retain	#ALL	Changed attribute values are carried forward to all future-dated records until the end of time. All existing records are retained. Tip: Leaving the effective end date blank or setting it to 4712/12/31 has the same effect as specifying the #ALL token. However, you must use the #ALL token to carry forward changes for objects that have a specific end date.
Retain	Specific date other than 4712/12/31	Changed attribute values are carried forward to future-dated records until the specified end date. All existing records are retained.
Replace	Specific date or blank	A single record replaces existing date-effective records for the date range specified on the update.

Retain mode is recommended when you're applying an incremental update to an existing record. Replace mode, which is the default mode, is useful during data migration, when you're uploading the complete data for a record.

Note: Replace mode may have unexpected consequences if you upload less than the complete data for a record.

In both modes:

- A new date-effective record is generated for the object if you specify an effective start or end date for which no date-effective record currently exists.
- Only the attributes for which you supply values are updated.

Setting the Maintenance Mode

Use the SET command to set the maintenance mode for a date-effective object. This table shows the set command to include in the data file.

Mode	SET Command
Retain	SET PURGE_FUTURE_CHANGES N
Replace	SET PURGE_FUTURE_CHANGES Y

If you include neither of these SET commands in a data file, then the default mode is Replace. The default mode can be overridden for the enterprise using the **Date-Effective Update Mode** configuration parameter.

Related Topics

- [Example of Retaining Attribute Values in Future-Dated Records](#)
- [Options for Carrying Forward Changed Attribute Values in Retain Mode](#)
- [Example of Replacing Attribute Values in Future-Dated Records](#)

Example of Retaining Attribute Values in Future-Dated Records

When you update a date-effective object, you can leave attribute values in future-dated records unchanged. To retain these values, you set the maintenance mode to Retain and specify the #RETAIN token instead of an effective end date.

This token ensures that your changes take effect from the supplied effective start date to the start of the next record. If no future-dated records exist, then your changes apply until the end date of the record. This topic provides an example showing the effects of Retain mode.

Retaining Future-Dated Attribute Values in an Assignment Object

Consider the date-effective history of the Assignment object shown in this table.

Effective Start Date	Effective Sequence	Effective End Date	Action Code	Job Code	Grade	Location	Hours
2010/06/08	1	2012/03/03	HIRE	ACC1	IC2	HQ	40
2012/03/04	1	2012/03/04	PROMOTION	ACC1	IC4	HQ	40
2012/03/04	2	2012/03/04	TRANSFER	ACC1	IC4	LVP	40
2012/03/04	3	2012/06/01	JOB_CHANGE	ACC3	IC4	LVP	40
2012/06/02	1	4712/12/31	JOB_CHANGE	ACC2	IC4	LVP	40

To change the working hours to 37.5 from 2012/01/10 without affecting future-dated records, you use this data file:

```
SET PURGE_FUTURE_CHANGES N
METADATA|Assignment|SourceSystemId|EffectiveStartDate|EffectiveSequence|EffectiveLatestChange|
EffectiveEndDate|ActionCode|NormalHours
MERGE|Assignment|2724|2012/01/10|Y|#RETAIN|ASG_CHANGE|37.5
```

After update, the Assignment object is as shown in this table.

Effective Start Date	Effective Sequence	Effective End Date	Action Code	Job Code	Grade	Location	Hours
2010/06/08	1	2012/01/09	HIRE	ACC1	IC2	HQ	40
2012/01/10	1	2012/03/03	ASG_CHANGE	ACC1	IC2	HQ	37.5
2012/03/04	1	2012/03/04	PROMOTION	ACC1	IC4	HQ	40
2012/03/04	2	2012/03/04	TRANSFER	ACC1	IC4	LVP	40
2012/03/04	3	2012/06/01	JOB_CHANGE	ACC3	IC4	LVP	40
2012/06/02	1	4712/12/31	JOB_CHANGE	ACC2	IC4	LVP	40

Attributes in records with effective start dates after 2012/01/10 retain their current values.

Note: As recommended, the data file includes no effective sequence number. Therefore, the next available number in the sequence is allocated automatically.

Related Topics

- Options for Updating Date-Effective Objects with Future-Dated Records
- Options for Carrying Forward Changed Attribute Values in Retain Mode

Options for Carrying Forward Changed Attribute Values in Retain Mode

When you update a date-effective object in Retain mode, you may want to carry changes forward to future-dated records. In this case, you can either specify an effective end date or carry the changes forward to all existing future-dated records.

This topic describes how to achieve each outcome and what to consider when deciding how to proceed.

Specifying an Effective End Date

This table shows the date-effective history of a Job object.

Effective Start Date	Effective End Date	Job Code	Job Name	Regular Temporary	Full Part Time	Active Status
2010/06/08	2012/01/09	ACC1	Accounts Clerk	T	PART_TIME	A

Effective Start Date	Effective End Date	Job Code	Job Name	Regular Temporary	Full Part Time	Active Status
2012/01/10	2012/03/03	ACC1	Accounts Clerk	T	FULL_TIME	A
2012/03/04	4712/12/31	ACC1	Accounts Administrator	T	FULL_TIME	A

To change the **Regular Temporary** value from **T** to **R** for the period from 2011/03/04 to 2014/04/04, you include these instructions in the .dat file:

```
SET PURGE_FUTURE_CHANGES N
METADATA|Job|SourceSystemId|EffectiveStartDate|EffectiveEndDate|RegularTemporary
MERGE|Job|45346|2011/03/04|2014/04/04|R
```

After update, the Job object is as shown in this table.

Effective Start Date	Effective End Date	Job Code	Job Name	Regular Temporary	Full Part Time	Active Status
2010/06/08	2011/03/03	ACC1	Accounts Clerk	T	PART_TIME	A
2011/03/04	2012/01/09	ACC1	Accounts Clerk	R	PART_TIME	A
2012/01/10	2012/03/03	ACC1	Accounts Clerk	R	FULL_TIME	A
2012/03/04	2014/04/04	ACC1	Accounts Administrator	R	FULL_TIME	A
2014/04/05	4712/12/31	ACC1	Accounts Administrator	T	FULL_TIME	A

A new date-effective record is created for both the start and end dates of the change. The **Regular Temporary** attribute is updated for the specified period. All other attributes keep their existing values.

Applying Changes to All Future-Dated Records

You can use the #ALL token in place of an effective end date. This token specifies explicitly that your changes apply to all future-dated records, regardless of the object's end date.

For example, you want to make these changes to the Job object:

- Change the **Regular Temporary** value from **T** to **R**.
- Set the effective start date of the change to 2011/03/05.
- Apply the changes to all future-dated records.

You include these instructions in the data file:

```
SET PURGE_FUTURE_CHANGES N
METADATA|Job|SourceSystemId|EffectiveStartDate|EffectiveEndDate|RegularTemporary
```



```
MERGE | Job | 45346 | 2011/03/04 | #ALL | R
```

After update, the Job object is as shown in this table.

Effective Start Date	Effective End Date	Job Code	Job Name	Regular Temporary	Full Part Time	Active Status
2010/06/08	2011/03/03	ACC1	Accounts Clerk	T	PART_TIME	A
2011/03/04	2012/01/09	ACC1	Accounts Clerk	R	PART_TIME	A
2012/01/10	2012/03/03	ACC1	Accounts Clerk	R	FULL_TIME	A
2012/03/04	4712/12/31	ACC1	Accounts Administrator	R	FULL_TIME	A

You can achieve the same result by leaving the effective end date in the .dat file blank or setting it explicitly to 4712/12/31. However, you must use the #ALL tag for any object that has an explicit end-date.

Related Topics

- [Options for Updating Date-Effective Objects with Future-Dated Records](#)
- [Example of Retaining Attribute Values in Future-Dated Records](#)

Example of Replacing Attribute Values in Future-Dated Records

When you update a date-effective object in Replace mode, you replace all existing, future-dated records with the uploaded data. You're recommended to use Replace mode only when the contents of the .dat file must replace the existing data rather than update it.

Replace mode is the default maintenance mode. However, you can use this SET command to select Replace mode explicitly and record the setting in the .dat file:

```
SET PURGE_FUTURE_CHANGES Y
```

Updating Only Selected Attributes of an Object in Replace Mode

This table shows the date-effective history of an Assignment object.

Effective Start Date	Effective Sequence	Effective End Date	Action Code	Job	Grade	Location	Normal Hours
2010/06/08	1	2012/03/03	HIRE	ACC1	IC2	HQ	40
2012/03/04	1	2012/03/04	PROMOTION	ACC1	IC4	HQ	40

Effective Start Date	Effective Sequence	Effective End Date	Action Code	Job	Grade	Location	Normal Hours
2012/03/04	2	2012/03/04	TRANSFER	ACC1	IC4	LVP	40
2012/03/04	3	2012/06/01	JOB_CHANGE	ACC3	IC4	LVP	40
2012/06/02	1	4712/12/31	JOB_CHANGE	ACC2	IC4	LVP	40

To update the working hours to 37.5 from 2012/01/10 in Replace mode, you create the following data file:

```
SET PURGE_FUTURE_CHANGES Y
METADATA|Assignment|SourceSystemId|EffectiveStartDate|EffectiveSequence|EffectiveLatestChange|
EffectiveEndDate|ActionCode|NormalHours
MERGE|Assignment|2724|2012/01/10|Y|ASG_CHANGE|37.5
```

After update, the Assignment object is as shown in this table.

Effective Start Date	Effective Sequence	Effective End Date	Action Code	Job	Grade	Location	Normal Hours
2010/06/08	1	2012/01/09	HIRE	ACC1	IC2	HQ	40
2012/01/10	1	4712/12/31	ASG_CHANGE	ACC1	IC2	HQ	37.5

In Replace mode, records with effective start dates that are later than the effective start date of the update are purged. Attribute values that predate the change and for which you supply no values in the data file appear in the new record. This update isn't reversible.

CAUTION: You're recommended to supply values for all attributes that currently exist on the object when updating in Replace mode. In this example, you could have provided values for job, grade, and location. As you didn't supply those values, the values that are current on the effective start date of the update remain in effect.

Related Topics

- [Options for Updating Date-Effective Objects with Future-Dated Records](#)

Examples of Changing First Effective Start Date and Last Effective End Date

For some objects, you can change the first effective start date or last effective end date. This topic provides examples showing how to make these changes.

To change an object's first effective start date, you include the **ReplaceFirstEffectiveStartDate** attribute in the .dat file and set it to **Y**. To change an object's last effective end date, you include the **ReplaceLastEffectiveEndDate** attribute

in the .dat file and set it to **Y**. You supply the revised dates on the **EffectiveStartDate** or **EffectiveEndDate** attribute, as appropriate. You can make these changes at the same time as other changes. For example, you can change both the start date of an object and its name.

Note: The **ReplaceFirstEffectiveStartDate** and **ReplaceLastEffectiveEndDate** parameters don't appear in object template files or on the Attributes tab of the Business Object Details page. However, you can add them to .dat files for date-effective objects that support these changes.

These restrictions apply to changing first effective start date and last effective end date:

- For some objects, the last effective end date must be 4712/12/31. Any attempt to change this date causes an error.
- You can't change the first effective start date of the Worker object in this way. Instead, you must use the **NewStartDate** attribute.
- When you include the **ReplaceLastEffectiveEndDate** attribute in a .dat file, you must specify a date value for the **EffectiveEndDate** attribute. You must not supply a #RETAIN, #ALL, or #NULL token in place of the effective end date.

Tip: On the Component Details tab of the Business Object Details page, identify whether you can change a component's first effective start or last effective end date. For example, in the Date Effectivity section on the Component Details tab for the Job object, the **First start date** option is selected. Therefore, you can include the **ReplaceFirstEffectiveStartDate** parameter in the .dat file.

Changing the First Effective Start Date of a Job Object

This example shows how to include the **ReplaceFirstEffectiveStartDate** attribute in a .dat file.

```
METADATA|Job|JobId|EffectiveStartDate|EffectiveEndDate|ReplaceFirstEffectiveStartDate
MERGE|Job|23452|1950/01/01|4712/12/31|Y
```

Changing the Last Effective End Date of a Recurring Element Entry

This example shows how to include the **ReplaceLastEffectiveEndDate** attribute in a .dat file.

```
METADATA|ElementEntry|ElementEntryId|EffectiveStartDate|EffectiveEndDate|ReplaceLastEffectiveEndDate
MERGE|ElementEntry|4634|2014/01/01|2014/04/05|Y
```

Related Topics

- [Overview of Loading Date-Effective Objects](#)

6 Loading Flexfield Data

Guidelines for Loading Flexfield Data

Using HCM Data Loader, you can load data for both descriptive flexfields and extensible flexfields. This topic explains concepts that are common to both types of flexfields.

Configuring Flexfields

Before you generate template files for objects with flexfields, you must configure the flexfields in Oracle HCM Cloud. After the flexfields are configured, the METADATA lines in generated template files include all attributes that are required to load the flexfield data. You also see a Flexfield Attributes tab for relevant object components on the Business Object Details page in the Data Exchange work area.

Flexfield Codes

When loading flexfield data, you must supply the flexfield code in the METADATA line in this format:

```
FLEX:<flexfield code>
```

For example, to load data for the job descriptive flexfield, the flexfield code is:

```
FLEX:PER_JOBS_DFF
```

Relevant flexfield codes appear in the generated template file and on the Flexfield Attributes tab for the business object.

Flexfield Attributes

Flexfield attribute names are those that you specify when configuring the flexfield. Both descriptive and extensible flexfields have one or more contexts. When you include a flexfield attribute name on the METADATA line for an object, you must also identify the context. You provide a hint value in the following format to identify both the flexfield and the context to which the attribute belongs:

```
<flexfield attribute name>(<flexfield code>=<context code>)
```

For example, for the _MAIN_CONTRACT attribute of the PER_CONTRACT_LEG_DDF descriptive flexfield with the CN context code, the entry on the METADATA line is:

```
_MAIN_CONTRACT (PER_CONTRACT_LEG_DDF=CN)
```

Business Objects with Multiple Descriptive Flexfields

Some business object components support multiple descriptive flexfields. You can include all descriptive flexfield attributes for every flexfield and configured context on a single METADATA line. This approach is possible because you supply both the flexfield code and the context for each attribute.

Lookup and Value Set Validated Flexfield Values

For flexfield values that are validated by lookups, you can supply either the lookup code or the lookup value. To supply the value, you must append the suffix `_Display` to the attribute name in the METADATA line.

For example, for a configured gender flexfield attribute, you can supply either the lookup code (M or F) or the value (Male or Female). To supply the value, use the attribute name `gender_Display`, as shown in this example:

```
METADATA|Job|FLEX:PER_JOBS_DFF|gender_Display(PER_JOBS_DFF=JOBCONTEXT1)|...  
MERGE|Job|JOBCONTEXT1|Male|...  
MERGE|Job|JOBCONTEXT1|Female|...
```

The same is true for flexfield segments validated by table-validated value sets. You can supply either the ID or value. To supply the value, you must append the suffix `_Display` to the attribute name in the METADATA line.

Note: You're recommended to use the lookup code or value set ID, as the lookup meaning and value set value can be translated. The language of the value provided to the `_Display` attribute must match the language of the user who's uploading the data. For HCM Data Loader that's the elevated `FUSION_APPS_HCM_ESS_LOADER_APPID` user, whose language may differ from language of the session user.

Secured Value Sets

A value set is a group of values that validate the content of a flexfield segment. If the value set is security enabled, then it's available to specific users or processes only. You must ensure that the predefined `FUSION_APPS_HCM_ESS_LOADER_APPID` user has access to any security-enabled value sets that your flexfields reference. Otherwise, the load fails. For more information about enabling access to secured value sets, see the *Securing HCM* guide.

Deleting Flexfield Segments

Let's say, for GDPR or other compliance purposes, you need to remove values from a flexfield segment and disable it. First, enable the segment if it's currently disabled. Next, change the values for the segment in your HCM Data Loader file to empty. Don't use spaces, and make sure the values are empty. Upload your HCM Data Loader file and confirm that the segments appear empty. Finally, disable the segment.

Here's a tutorial on loading flexfield segments using HDL: [Loading Flexfield Segments with HCM Data Loader \(HDL\)](#)

Here's a tutorial to add preconfigured and custom flexfield segments to your HCM Spreadsheet Data Loader templates [Including Flexfield Segments in HCM Spreadsheet Data Loader Templates](#)

Related Topics

- [Generate and Use Template Files](#)
- [How You Load Data for Descriptive Flexfields](#)
- [How You Load Data for Extensible Flexfields](#)
- [Enable Security for Value Sets](#)
- [Secure Access to Value Sets](#)

How You Load Data for Descriptive Flexfields

This topic describes how to construct METADATA and MERGE lines in a .dat file to supply descriptive flexfield data.

Constructing the METADATA Line

Descriptive flexfields extend a business object. You can supply the attributes of the descriptive flexfield on the same MERGE line as the core attributes of the business-object component. On the METADATA line, you must include both the flexfield code and the descriptive flexfield attributes for which you want to load data. In a generated business-object template, the METADATA line automatically includes the flexfield code and attributes for configured descriptive flexfields.

This example shows the METADATA line for the Job business object with entries for the PER_JOBS_DFF descriptive flexfield:

```
METADATA|Job|FLEX:PER_JOBS_DFF|JobCode|ActiveStatus|FullPartTime|MedicalCheckupRequired|RegularTemporary|
EffectiveStartDate|EffectiveEndDate|Name|SetCode|JobFamilyName|_JOB_LEVEL(PER_JOBS_DFF=US)
```

The descriptive flexfield attributes can appear anywhere on the line. You don't have to append them at the end.

Constructing the MERGE Lines

For each descriptive flexfield for a business-object component, a single MERGE record can have only one context. You specify the context value on the MERGE line against the flexfield code for the descriptive flexfield in the METADATA line.

For example:

```
METADATA|Job|FLEX:PER_JOBS_DFF|JobCode|ActiveStatus|FullPartTime|MedicalCheckupRequired|RegularTemporary|
EffectiveStartDate|EffectiveEndDate|Name|SetCode|JobFamilyName|_JOB_LEVEL(PER_JOBS_DFF=US)
MERGE|Job|US|ACC|A|F|N|R|2000/01/01|4712/12/31|Accountant|COMMON|Finance|1
```

On the MERGE line, the context value (US) appears in the same position as the flexfield code (FLEX:PER_JOBS_DFF) on the METADATA line.

Supplying MERGE Lines for Multiple Flexfield Codes and Contexts

When a single business object component supports multiple descriptive flexfields, you can load all of the flexfield data at once. The following example shows METADATA and MERGE lines for the Contract component of the Worker object. Each MERGE line is for a single context.

```
METADATA|Contract|AssignmentId|ContractId|EffectiveStartDate|EffectiveEndDate|
FLEX:PER_CONTRACT_DF|FLEX:PER_CONTRACT_LEG_DDF|_CONTRACT_GLB(PER_CONTRACT_DF=Global Data
Elements)|_Currency(PER_CONTRACT_DF=CONTRACT_DF)|_MAIN_CONTRACT(PER_CONTRACT_LEG_DDF=CH)|
_CONST_PROB_DATE(PER_CONTRACT_LEG_DDF=CN)|_NDA(PER_CONTRACT_LEG_DDF=CN)|
_COMPETITION_CLAUSE(PER_CONTRACT_LEG_DDF=CN)|_NOTICE_DURATION_UNIT(PER_CONTRACT_LEG_DDF=CN)
MERGE|Contract|E8732|39987|2013/12/14|2014/03/04|CONTRACT_DF|CN|Contract Glb value|USD|Contract Data|||
MERGE|Contract|E8732|39987|2014/03/05|4712/12/31|CONTRACT_DF|CH|Contract Glb value|USD||31/03/2015|NDA
Value|Competition Clause Value
```

Updating Global Segments

You can configure both global and context-specific segments for a flexfield. When updating global segments in records that have context-specific segments, set the context value to the existing context value, not Global Data Elements. If you set the context value to Global Data Elements, then existing context-specific values are set to null because the context has changed. For example, consider these two segments of a flexfield record:

Segment	Global	Context
Segment 1	Yes	Not applicable

Segment	Global	Context
Segment 2	No	CN

If you update segment 1, which is a global segment, then you must set the context to CN, not Global Data Elements. Otherwise, the existing contents of segment 2 are set to null.

Related Topics

- [Guidelines for Loading Flexfield Data](#)
- [Example of Loading Data for Descriptive Flexfields](#)

Example of Loading Data for Descriptive Flexfields

This topic shows how to construct the METADATA line for the Contract component of the Worker object with multiple configured descriptive flexfields. The METADATA line appears automatically in the generated worker template.

The contract component supports these descriptive flexfields:

- PER_CONTRACT_DF
- PER_CONTRACT_LEG_DDF

Constructing the METADATA Line for the Worker Contract Component

The PER_CONTRACT_DF descriptive flexfield is configured in Oracle HCM Cloud with the context and attribute values shown in this table. The **Hint** column shows the resulting hint for each attribute.

Context	Attribute	Hint
Global	CONTRACT_GLB	(PER_CONTRACT_DF=Global Data Elements)
CONTRACT_DF	Currency	(PER_CONTRACT_DF=CONTRACT_DF)

The PER_CONTRACT_LEG_DDF descriptive flexfield is configured in Oracle HCM Cloud with the context and attribute values shown in this table. The **Hint** column shows the resulting hint for each attribute.

Context	Attribute	Hint
CH	_MAIN_CONTRACT	(PER_CONTRACT_LEG_DDF=CH)
CN	_CONST_PROB_DATE	(PER_CONTRACT_LEG_DDF=CN)
CN	_NDA	(PER_CONTRACT_LEG_DDF=CN)

Context	Attribute	Hint
CN	_COMPETITION_CLAUSE	(PER_CONTRACT_LEG_DDF=CN)
CN	_NOTICE_DURATION_UNIT	(PER_CONTRACT_LEG_DDF=CN)

In the generated Worker.dat template file, the Contract METADATA line includes these attribute definitions for the flexfield attributes:

```
METADATA|Contract|...|_CONTRACT_GLB(PER_CONTRACT_DF=Global Data Elements) |  
_Currency(PER_CONTRACT_DF=CONTRACT_DF) | _MAIN_CONTRACT(PER_CONTRACT_LEG_DDF=CH) |  
_CONST_PROB_DATE(PER_CONTRACT_LEG_DDF=CN) | _NDA(PER_CONTRACT_LEG_DDF=CN) |  
_COMPETITION_CLAUSE(PER_CONTRACT_LEG_DDF=CN) | _NOTICE_DURATION_UNIT(PER_CONTRACT_LEG_DDF=CN)
```

As each attribute has a hint that identifies both the descriptive flexfield and the context, you can define attributes for multiple flexfields on the same METADATA line.

Related Topics

- [How You Load Data for Descriptive Flexfields](#)
- [Guidelines for Loading Flexfield Data](#)

How You Load Data for Extensible Flexfields

This topic describes how to construct METADATA and MERGE lines in a data file to supply extensible flexfield data.

Category Code

Extensible flexfields have both a flexfield code and a category code. The category code is provided automatically in generated template files and has the attribute name EFF_CATEGORY_CODE.

Constructing the METADATA Line

Unlike descriptive flexfields, extensible flexfields aren't an extension of a business-object component. Rather, they're separate components in the business object hierarchy. To supply data for an extensible flexfield, you simply include the METADATA line for the extensible flexfield component in the data file. Remove the names of any attributes for which you aren't supplying data.

This example shows a METADATA line for the JobLegislative extensible flexfield.

```
METADATA|JobLegislative|EFF_CATEGORY_CODE|FLEX:PER_JOBS_LEG_EFF|JobId(SourceSystemId)|JobCode|  
SetCode|EffectiveStartDate|EffectiveEndDate|SourceSystemOwner|SourceSystemId|LegislationCode|  
_EEO1_CATEGORY(PER_JOBS_LEG_EFF=HRX_US_JOBS) | _FLSA_STATUS(PER_JOBS_LEG_EFF=HRX_US_JOBS) |  
_INSEE_PCS_CODE(PER_JOBS_LEG_EFF=FR) | _ECAP_JOB(PER_JOBS_LEG_EFF=FR) |  
_LINE_OF_PROGRESSION(PER_JOBS_LEG_EFF=HRX_US_JOBS)
```

Constructing the MERGE Lines

Each MERGE line for an extensible flexfield record can have only one context. In the METADATA line, you specify the context value against the flexfield code for the extensible flexfield.

This example shows a METADATA line and two MERGE lines for the JobLegislative extensible flexfield:

```
METADATA|JobLegislative|EFF_CATEGORY_CODE|FLEX:PER_JOBS_LEG_EFF|JobId(SourceSystemId)|JobCode|
SetCode|EffectiveStartDate|EffectiveEndDate|SourceSystemOwner|SourceSystemId|LegislationCode|
_EEO1_CATEGORY(PER_JOBS_LEG_EFF=HRX_US_JOBS)|_FLSA_STATUS(PER_JOBS_LEG_EFF=HRX_US_JOBS)|
_INSEE_PCS_CODE(PER_JOBS_LEG_EFF=FR)|_ECAP_JOB(PER_JOBS_LEG_EFF=FR)|
_LINE_OF_PROGRESSION(PER_JOBS_LEG_EFF=HRX_US_JOBS)
MERGE|JobLegislative|JOB_LEG|HRX_US_JOBS|OCT18EFF1|OCT18EFF1|COMMON|1990/01/01|4712/12
MERGE|JobLegislative|JOB_LEG|FR|OCT18EFF1|OCT18EFF1|COMMON|1990/01/01|4712/12/31|VISION|OCT18EFF1_LEG2|FR|||
387b|N|
```

Note: Unlike other components of the business-object hierarchy, extensible flexfields can't be supplied in isolation. They must be accompanied by a parent record.

Multirow Extensible Flexfield Contexts

You can configure an extensible flexfield context with multiple rows. In this case, you configure one or more of the flexfield segments as the key that uniquely identifies a single row of the multirow context. HCM Data Loader treats these flexfield segments as user-key attributes. Therefore, you can uniquely identify a single flexfield record using user keys.

You must supply a value for the user-key flexfield-segment attributes when creating an extensible flexfield record. When updating a multirow extensible-flexfield record, you must supply either the source key or the complete user key, including the flexfield segments defined as the unique key. You can identify which flexfield segments form the unique key by reviewing the flexfield-segment configuration on the Manage Extensible Flexfields page. All flexfield segments for which **Unique Key** is selected are user-key attributes.

Here's a tutorial on loading flexfield segments using HDL: [Loading Flexfield Segments with HCM Data Loader \(HDL\)](#)

Here's a tutorial to add preconfigured and custom flexfield segments to your HCM Spreadsheet Data Loader templates [Including Flexfield Segments in HCM Spreadsheet Data Loader Templates](#)

Related Topics

- [Guidelines for Loading Flexfield Data](#)
- [Example of Loading Data for Extensible Flexfields](#)

Example of Loading Data for Extensible Flexfields

This topic shows how the METADATA line is constructed for a Job business object component with the configured JobLegislative extensible flexfield. The METADATA line appears automatically in the generated job template.

Constructing the METADATA Line for the JobLegislative Extensible Flexfield

The PER_JOBS_LEG_EFF extensible flexfield is configured in Oracle HCM Cloud with the context and attribute values shown in this table. The **Hint** column shows the resulting hint for each attribute.

Context	Attribute	Hint
CA	_EEOG	(PER_JOBS_LEG_EFF=CA)
CA	_NOC_CODE	(PER_JOBS_LEG_EFF=CA)
CH	_POSITION_TYPE	(PER_JOBS_LEG_EFF=CH)
FR	_ECAP_JOB	(PER_JOBS_LEG_EFF=FR)
FR	_INSEE_PCS_EXT_CODE	(PER_JOBS_LEG_EFF=FR)

In the generated Job.dat template file, the JobLegislative METADATA line includes these attribute definitions for the flexfield attributes:

```
METADATA|JobLegislative|...|EFF_CATEGORY_CODE|FLEX:PER_JOBS_LEG_EFF|_EEOG(PER_JOBS_LEG_DFF=CA)|  
_NOC_CODE(PER_JOBS_LEG_DFF=CA)|_POSITION_TYPE(PER_JOBS_LEG_DFF=CH)|_ECAP_JOB(PER_JOBS_LEG_DFF=FR)|  
_INSEE_PCS_EXT_CODE(PER_JOBS_LEG_DFF=FR)
```

Related Topics

- [Guidelines for Loading Flexfield Data](#)
- [How You Load Data for Extensible Flexfields](#)

7 Importing and Loading Data Using HCM Data Loader

How Data Is Imported and Loaded

HCM Data Loader imports data initially to stage tables, where some validation occurs. HCM Data Loader then loads valid logical objects from the stage tables to the application tables. You can perform the import and load steps either separately or as one step.

This topic summarizes the processing that occurs in each step. It also describes how you specify import and load options.

Importing Records to the Stage Tables

In the import stage, HCM Data Loader:

- Extracts the business-object data from your .zip file on the Oracle WebCenter Content server and decrypts it, if the file is encrypted.
- Validates the business-object file name and METADATA definitions and raises any errors.

Note: Files for objects such as User Defined Tables, Schedule Request, Organization Tree, Department Tree, Grade Ladder, and Progression Grade Ladder often contain large volumes of child records associated with a single parent record. An additional validation is done to detect files with more than 2000 child records per parent. If the file surpasses this threshold, an error will be raised during the import process to prevent performance degradation. To resolve this error, split the data into two separate files: one containing the parent record and the other containing only the child records, and then load them.

- Imports records from the data files into the HCM Data Loader stage tables.
- Validates the data against attribute data types and raises any errors.
- Groups the data by local key values to form logical records of related date-effective file lines. For example, all date-effective file lines supplied for a Job object are grouped into a logical occurrence of a job, such as the Accountant job.
- Forms logical objects by resolving references to parent components. For example, the logical record for a valid grade is associated with its parent Job object.
- Records the status of all imported objects. View the status when you click the Total Objects count from the Import and Load status page.

Loading Objects to the Application Tables

HCM Data Loader doesn't load valid logical objects directly into the application tables. Instead, it passes valid object data to business-object-specific services. For example, to load Worker objects, HCM Data Loader passes them to the worker service. These services perform the validation that's specific to the business object, raise any errors, and load valid objects to the application tables.

Setting the File Action Parameter

When you import a file on the Import and Load data page, you can set the **File Action** parameter on the Schedule Request page. The value that you specify here overrides the enterprise default setting. This table describes how the **File Action** parameter controls import and load.

File Action Value	Description
Import and load	HCM Data Loader imports the contents of the .zip file to the stage tables and calls object services to load valid logical objects to the application tables.
Import only	HCM Data Loader imports the contents of the .zip file to the stage tables but doesn't pass valid logical objects to the relevant object service. Instead, you perform that step separately on the Import and Load data page by clicking Submit and selecting the Load action for each business object.

You may want to select **Import only** when you start to load data so you can fix import errors before attempting to load objects. Once your data loading is routine, use **Import and load** to run them both with one job submission.

Tip: You can validate your data file before you load it. Use the Data File Validator tool for HCM Data Loader. For more information, see Data File Validator Tool for HCM Data Loader (2022617.1) on My Oracle Support at <https://support.oracle.com>.

Automating Data Loading

When you automate data loading using the HCM Data Loader web service, you can set **File Action** to **Import only** or **Import and load**. If you select **Import only**, then you load the imported objects manually on the Import and Load Data page.

Reviewing Object History

For some object components, such as Assignment, you can review the object history to see who last updated the object. When the object was last updated using HCM Data Loader, the **Last Updated By** value is FUSION_APPS_HCM_ESS_LOADER_APPID. You don't see the user name of the user who uploaded the changes.

Uploading and Securing Files on the Web Center Content Server

You can import files directly from your desktop, alternatively, you can import files already loaded to the Oracle Web Center Content server.

Upload a File to the Web Center Content server manually

Do the following to upload the file to the Web center content server:

1. Navigate to Tools > File Import and Export.

2. Click the create icon on the table toolbar.
3. Choose the file to upload.
4. Select an account to upload the file to `hcm/dataloader/import`.
5. Click Save and Close.

Import and Load Data

The file of data that you're preparing to import and load using HCM Data Loader may exist locally on your desktop. Or, you may already have loaded it to the Oracle WebCenter Content server.

This topic provides steps for importing and loading your data interactively on the Import and Load Data page. It also describes how to load data to the application tables if you imported it separately to the stage tables.

Import and Load a Local File

1. In the **Data Exchange** work area, select **Import and Load Data** to open the Import and Load Data page.
2. Click **Import File**.
3. In the Import File dialog box, click **Browse** to search for and select your file. Alternatively, drag the file from your local file browser to the **Browse** button.
4. Click **Submit Now** to upload your file to Oracle WebCenter Content server to immediately initiate processing of your file.

The file is loaded automatically to the HCM Data Loader import account `hcm/dataloader/import` and allocated a unique content ID.

5. Alternatively, you can click **Review Parameters** to review and edit the default values of parameters or to schedule the submission of the file for a later date or time.

Clicking **Review Parameters** navigates you to the Schedule Request, Parameters, and Schedule sections.

In the Parameters section of the Import and Load Data page, review the parameter values and update them as required. Most of these values are set for the enterprise on the Configure HCM Data Loader page and can be left unchanged. The parameters are shown in this table.

Parameter	Description
File Name	The name of the file on the WebCenter Content server.
Content ID	The WebCenter Content ID for the file on the WebCenter Content server.
File Action	<p>The Import only option imports the business object data into the stage tables but doesn't load it to the application tables. You can start the load process manually from the user interface.</p> <p>The Import and load option automatically loads to the application tables any business object that loads successfully to the stage tables.</p>
File Encryption	The encryption type for the file.

Parameter	Description
Maximum Percentage of Import Errors	The percentage of file lines in error that can occur in a business object before the import process stops for the object.
Maximum Percentage of Load Errors	The percentage of business-object instances in error that can occur for a business object before the load process stops.
Delete Source File	Indicates whether the source file is to be purged from the WebCenter Content server after processing.
Maximum Concurrent Threads for Import	The maximum number of concurrent process threads to use for importing your data set.
Maximum Concurrent Threads for Load	The maximum number of concurrent process threads to use for loading your data set.
Load Group Size	<p>The number of objects to process at a time on each thread. The record counts are updated only when processing completes for each group.</p> <p>Note: This parameter appears on the Schedule Request page only if the Enable Load Group Size Calculation configuration parameter is set to No. Otherwise, the load-group size is calculated automatically.</p>

6. Click **Submit**. A confirmation message showing the process ID appears.
7. Note the process ID and click **OK** to close the Confirmation dialog box.

Import a File from the Oracle WebCenter Content Server

You can load a .zip file to the Oracle WebCenter Content server if you prefer. Select **Navigator > Tools > File Import and Export** and upload the file to the **hcm/dataloader/import** account. To import and load a file that's already on the WebCenter Content server:

1. In the **Data Exchange** work area, select **Import and Load Data** to open the Import and Load Data page.
2. On the Import and Load Data page, click **Import File**.
In the Import File dialog box, select **Oracle WebCenter Content Files**. Any files in the **hcm/dataloader/import** account that haven't been processed are listed. You can filter the list to reduce it. For example, you can enter the file creation date.
3. Select your file in the Import File dialog box and click **Submit Now** to immediately initiate the file processing.
4. Alternatively, you can click **Review Parameters** to review and edit the default values of parameters before submitting.

Clicking **Review Parameters** navigates you the Schedule Request and Parameters sections.

In the Parameters section of the Import and Load Data page, review the parameters, edit them as necessary, and click **Submit**.

Creating a One-Time Schedule

You can delay the submission of your file to be processed at a later date or time. This is useful for processing large files outside of working hours.

1. In the **Data Exchange** work area, select **Import and Load Data** to open the **Import and Load Data** page.
2. Click **Import File**.
3. In the **Import File** dialog box, select your file, either locally or from the Oracle WebCenter Content server.
4. Click **Review Parameters**.
5. In the Schedule section, specify **Run** as **Using as schedule** and specify the Start Date.
6. Click **Submit**.

Here's a tutorial on creating and maintaining data with HCM Data Loader [Create and Maintain Data with HDL](#)

Related Topics

- [Example of using HCM Data Loader](#)

Example of using HCM Data Loader

In this example, you learn how to import and load objects manually to Oracle HCM Cloud. Import and load are performed separately in this example.

This table summarizes key decisions for this scenario.

Decisions to Consider	In This Example
Which business object am I loading?	Job. The child components of the Job object aren't being loaded. These components are: <ul style="list-style-type: none">• Job Evaluation• Job Extra Information• Job Legislative Extra Information• Job Valid Grade
What's the template name?	Job.dat
What's the name of the file that I am uploading?	Sales_Jobs.zip
What's the value of the File Action parameter?	Import only . You have decided to manage the load step separately.

Summary of Tasks

Upload the jobs data by:

1. Generating the jobs template
2. Editing the template file
3. Importing jobs to the stage tables
4. Loading jobs to the application tables

Generate the Job Template

1. In the **Data Exchange** work area, select **View Business Objects**.
2. On the **View Business Objects** page, enter **Job** in the **Business Object** query-by-example field and press **Enter**. Select the row for the Job object in the Search Results section.
3. Click **Refresh Object** on the table toolbar.
4. In the Schedule Request dialog box, click **Submit**.
5. Click **OK** to close the confirmation message.
6. Click **Refresh** on the table toolbar to view the **Download File** icon in the **File** column for the job row.

You may have to refresh more than once before the icon appears.

Tip: If the icon already exists, then you can download the existing template. The **Last Refreshed** column shows when the template was generated.

7. Click the **Download File** icon in the **File** column.
8. Save the Job.dat template locally.

Note: You can also download the object definition by clicking the download file icon on the View Business Objects page for your object. You will need to log in when requested to build the object definition spreadsheet.

Edit the Template File

1. Open the Job.dat template.
2. Edit the template file as follows:
 - o Remove METADATA lines for these unwanted child components: Job Evaluation, Job Extra Information, Job Legislative Extra Information, and Job Valid Grade.
 - o Edit the job METADATA line to remove unwanted attributes.
 - o Make any required changes to the attribute order.

3. Enter jobs data in the configured template. For example:

```
METADATA|Job|SourceSystemOwner|SourceSystemId|EffectiveStartDate|SetCode|JobCode|Name|ActiveStatus|
FullPartTime|RegularTemporary|JobFamilyId(SourceSystemId)|JobFunctionCode|ManagerLevel
MERGE|Job|VISION|SALES_CONS|2000/01/01|COMMON|SALES_CONS|Sales Consultant|A|FULL_TIME|R|CONSULT||9
MERGE|Job|VISION|SALES_MGR|2000/01/01|COMMON|SALES_MGR|Sales Manager|A|FULL_TIME|R|MANAGEMENT|MANAG|6
MERGE|Job|VISION|SALES_DIR|2000/01/01|COMMON|SALES_DIR|Sales Director|A|FULL_TIME|R|MANAGEMENT|MANAG|1
MERGE|Job|VISION|MRKT_CONS|2000/01/01|COMMON|MRKT_CONS|Marketing Consultant|A|FULL_TIME|R|CONSULT||9
MERGE|Job|VISION|MRKT_MGR|2000/01/01|COMMON|MRKT_MGR|Marketing Manager|A|FULL_TIME|R|MANAGEMENT|MANAG|6
MERGE|Job|VISION|MRKT_DIR|2000/01/01|COMMON|MRKT_DIR|Marketing Director|A|FULL_TIME|R|MANAGEMENT|MANAG|1
```

4. Save and close the template file.
5. Compress the template file, name it **Sales_Jobs.zip**, and save it locally.

Note: Ensure that the SourceSystemOwner supplied in your file exists in the HRC_SOURCE_SYSTEM_OWNER lookup. For details, refer How You Supply Source Keys.

Import Data to the Stage Tables

1. In the **Data Exchange** work area, select **Import and Load Data**.
2. On the Import and Load Data page, select **Import File**.
3. In the Import File dialog box, click **Browse**.
4. Select the **Sales_Jobs.zip** file.
5. Click **Review Parameters**.
6. On the Schedule Request page, select **Import only** in the **File Action** field.
7. Click **Submit**.
8. Click **OK** to close the confirmation message.
9. On the Import and Load Data page, click **Refresh** in the **Data Sets** table.
10. Review the status and details of the **Sales_Jobs.zip** data set that you uploaded. When the Total Objects count is greater than zero you can click the new count to review the objects found in your file and their status. Refresh the page if the data set is still in progress. Click Object Details to review the object hierarchy and the values of attributes supplied for the logical object.

Return to the Data Sets table on Load and Import Data page. When the **Import Status** column shows **Success** and the **Import Success (%)** column displays **100**, you're ready to load the data to the application tables.

Load Data to the Application Tables

1. In the Data Sets table on the Import and Load page, select the **Sales_Jobs.zip** data set.
2. In the Business Objects table on the Import and Load Data page, select the Job object row and click **Submit**.
3. In the Schedule Request dialog box, ensure that the **Action** is set to Load and click **Submit**.
4. Click **OK** to close the confirmation message.
5. Click **Refresh** on the Business Objects table toolbar to check the status of the load.

When the **Load Status** column shows **Success** and the **Load Success (%)** column displays 100, the job data has loaded successfully to the application tables.

Extending the Retention Period of a Data Set

You can extend the retention period of a data set, if it contains data that can be validated or rolled back.

1. In the Business Objects table on the Import and Load Data page, select the business object that supports being validated or rolled back and click Submit.
2. On the Schedule Request page specify a File Action of Retain Data Set for an Extended Period and click Submit.

This option is only available when the data set is not already retained for an extended period. You can also extend data set retention when importing and loading the file using a SET instruction.

Related Topics

- [How You Supply Source Keys](#)
- [Import and Load Data](#)
- [The SET Instruction](#)
- [HCM Data Loader Configuration Parameters](#)

How You Review and Correct Errors

Errors that occur during the import and load stages of data loading are reported automatically on the Import and Load Data page. This topic describes how to review and correct reported errors.

Reviewing Import Errors

The **Failed Lines** column in both the Data Sets and Business Objects tables shows the number of records that failed to import. When the number in the **Failed Lines** column is greater than zero, clicking it opens the Failed Lines page. When you open the Failed Lines page from the Data Sets table, it lists:

- The METADATA line for each component
- All messages and the file lines in error for the selected data set

You can:

- Filter the list to show the METADATA lines and the file lines in error for the components of a single business object.
- Use Query by Example fields to show specific file lines or messages.

When you open the Failed Lines page from the Business Objects table, you see details for the selected object by default. You can filter the list to show failed lines for different objects from the same data set.

Examples of physical row errors are:

- The instruction or discriminator isn't recognized.
- The number of values in a data line doesn't match the number in the METADATA line.
- You didn't supply a unique reference for the component.
- The parent of a child component wasn't found, either in the file or in Oracle HCM Cloud.
- A required attribute is missing.

To correct import errors, you correct the source data before importing and loading it again. You need to add the corrected data file to a new .zip file. You can't process the existing file again.

Generating Failed Lines File

You can use the **Extract Failed Lines** option in both the Data Sets and Business Objects tables toolbar generate a file of failed lines for either a data set or a specific business object within the data set.

To generate a failed lines file:

1. On the Import and Load Data page, select the Data Set or Business Object for which you want to generate the failed lines file.
2. On the Schedule Request page, based on the type of file you want to generate, set one of these values in **Actions**:
 - Extract failed lines with messages
 - Extract failed lines

3. Specify the language in which the messages must be displayed when extracting the messages.
4. Click **Submit** to initiate the file generation.

Use the **Failed Lines File** column to monitor the progress of the file generation by clicking **Refresh**. Once the file download icon appears, click the icon to access the generated file.

The Extract Failed Lines option is not enabled by default. To enable it:

1. Go to **Setup and Maintenance > Configure HCM Data Loader** and override the default for the **Enable File Generation for Failed File Lines** parameter and set it to one of the following:
 - Extract failed lines - includes only the failed lines
 - Extract failed lines with messages - includes all the messages against a file line or the object it forms immediately before the file line or group of files lines to which the message relates to.

Selecting either of these values enables the Extract Failed Lines option on the Import and Load Data page. The value supplied to this parameter is the default value when generating a file however, you can override the default value when requesting a failed lines file generation.

Accessing the Failed Lines File from the Oracle WebCenter Content Server

When error files are generated, they are encrypted by HCM Data Loader and can only be decrypted when downloading the file from the **Import and Load Data** page, as described above.

To enable generated error files to be accessed directly from the Oracle WebCenter Content Server:

1. Go to **Setup and Maintenance > Configure HCM Data Loader** and override the default for the **Enable Oracle WebCenter Content Access to Error Files** parameter and set it to one of the following:
 - Yes, with no encryption
 - PGP - Signed encryption
 - PGP - Unsigned encryption

When you request a generation of the error file, the **File Encryption** parameter will be available and defaulted to value selected here.

Note: When choosing to encrypt the file you will have to specify the encryption verification key.

Reviewing Load Errors

The **Failed Objects** column in both the Data Sets and Business Objects tables shows the number of objects that failed to load. When the number in the **Failed Objects** column is greater than zero, clicking it opens the Failed Objects Messages page. When you open the Failed Objects Messages page from the Data Sets table, it includes for each object in error:

- Its user keys and any source key
- The component in the object hierarchy for which the error is reported
- The error message
- A link to the object details, which opens the Object Hierarchy page

You can:

- Filter the list to show error details for a specific object in the data set.
- Use Query by Example fields to show specific keys or messages.

When you open the Failed Objects Messages page from the Business Objects table, you see details for the selected object by default. You can filter the list to show messages for different objects from the same data set.

Reviewing Messages

To review the messages for an import or load failure, click the icon in the **Messages** column of the Data Sets or Business Objects table. When you open the Messages page for a data set, you see for each message:

- The process where the message occurred, which can be Import or Load
- The message location
- The message itself, with the number of occurrences of the message

When you click the message link, you open the Message Details page. It provides details of the message with the information from the Failed Lines or Failed Objects page, as appropriate.

Tip: Messages appear in the session language if the HCM Data Loader configuration parameter **Enable UI Message Translation** is set to **Yes** in your environment.

Correcting Objects in Error

On the Object Hierarchy page, which you open from the Failed Object, Message Detail or Object Status page, you can expand object folders to see error details. When you click the name of a component entry, the Attribute Details dialog box opens. It shows the original and current values of each attribute. For components in error, you can edit an attribute to correct it before loading the object again on the Import and Load Details page. You can see how many objects you corrected in the **Objects Corrected** columns of the tables on the Import and Load Data page. When you click the number in the **Objects Corrected** column, the Failed Objects page opens for the corrected object.

If you prefer to correct object errors in the source data, then you need to import and load the corrected data again. You must add the corrected data file to a new .zip file. You can't process the existing file again.

Related Topics

- [Import and Load Data](#)

Start or Resubmit a Load

You can start or resubmit the load process for a data set or directly for a business object.

You may want to start a load if you've submitted your data set in **Import only** mode or resubmit when the business objects in the data set have failed to load successfully.

To Start or Resubmit Load for a Data Set

1. In the **Data Sets** table on the **Import and Load Data** page, select the data set for which you want to start or resubmit the load process, and click **Submit**.
2. On the **Schedule Request** page, select **Load** as the **Action**.
3. Review the parameter values for **Maximum Percentage of Load Errors**, **Maximum Concurrent Threads for Load**, edit them if necessary.
4. Select the suitable run schedule between: **As soon as possible** and **Using a schedule**.
5. Click **Submit** to submit the process.

To Start or Resubmit Load for a Business Object

1. In the **Business Object** table on the **Import and Load Data** page, select the business object for which you want to start or resubmit the load, and click **Submit**.
2. On the **Schedule Request** page, select **Load** as the **Action**.
3. Review the parameter values for **Maximum Percentage of Load Errors**, **Maximum Concurrent Threads for Load**, edit them if necessary.
4. Select the suitable run schedule between: **As soon as possible** and **Using a schedule**.
5. Click **Submit** to submit the process.

Resubmitting Load Processing from the Object Status Page

You can also resubmit the load process from the **Object Status** page.

- Click the **Total Objects** link on the **Import and Load Data** page for the desired data set in the **Data Sets** table or corresponding business object from the **Business Objects** table.
- On the **Object Status** page for the data set or business object, you can review the imported business objects and click **Resubmit**.
- On the **Submit Request** page,
 - Select **All** to load all the business objects in the data set or any of the individual business objects listed under **Business Object**.

Note: If you clicked the **Total Objects** link from the **Business Objects** table, only the records for that business object will be listed.

- Review and set the **Maximum Concurrent Threads**, **Maximum Percentage of Errors** parameters.
- Select **Load** from **Actions**.
- Click **Submit**.

Roll Back HCM Data Loader Data Set or Business Object

You can roll back the successfully loaded records of a business object if the business object supports being rolled back and there are loaded objects to roll back. You can also request that the retention period of a data set in the staging tables be extended, allowing greater time to perform roll back of successfully loaded objects.

Note: Review the business object details in the **View Business Objects** task to determine the business objects which support being rolled back and the level of roll back available.

To Roll Back the load or extend the retention period for a Data Set

- In the **Data Sets** table on the **Import and Load Data** page, select the data set with the business object for which the data needs to be rolled back or the data set retention period needs to be extended, and click **Submit**.
- On the **Schedule Request** page, select **Roll Back** or **Retain Data Set for an Extended Period** as the **Action**.
- Review the parameter values for **Maximum Percentage of Load Errors** and **Maximum Concurrent Threads for Load**, edit them if necessary.
- Select a suitable run schedule between: **As soon as possible** and **Using a schedule**.
- Click **Submit** to submit the process.

To Roll Back the load or extend the retention period for a Business Object

- In the **Business Objects** table on the **Import and Load Data** page, select the business object for which the data needs to be rolled back or the data set retention period needs to be extended, and click **Submit**.
- On the **Schedule Request** page, select **Roll Back** or **Retain Data Set for an Extended Period** as the **Action**.
- Review the parameter values for **Maximum Percentage of Load Errors** and **Maximum Concurrent Threads for Load**, edit them if necessary.
- Select a suitable run schedule between: **As soon as possible** and **Using a schedule**.
- Click **Submit** to submit the process.

Stop HCM Data Loader Processing

You can stop the processing of either a data set or an individual business object. This topic explains how to stop processing.

Stop the Processing of a Data Set

Follow these steps:

1. In the Search Results section of the Import and Load Data page, select the data set that you want to stop.
2. Click **Refresh** to update the status and confirm that the data set is still processing.
3. Click **Stop**.

Tip: The **Stop** button is active only if either the import or the load is still in progress.

4. In the Warning dialog box, click **Yes** to confirm the stop request.

The status of the import or load process that was stopped is updated to **Stopped**. A large data set may take a few minutes to stop processing completely. For efficiency in import and load processing, HCM Data Loader checks at intervals for stop requests.

Stop the Processing of a Business Object

Follow these steps:

1. In the Details section of the Import and Load Data page, select the business object that you want to stop processing.
2. Click **Refresh** to update the status and confirm that the business object is still processing.
3. Click **Stop**.
4. In the Warning dialog box, click **Yes** to confirm the stop request.

The status of the import or load process that was stopped is updated to **Stopped**.

Restart Stopped Processes

You can't restart a stopped import process for a business object. However, you can restart a stopped load process for a business object. Select the stopped business object in the Details section of the Import and Load Data page and click **Load**. The status of both the business object and the associated data-set load processes changes to **In Progress**.

Overview of Automating Data Loading

You can automate HCM Data Loader processes to load data to Oracle HCM Cloud and extract both status and error information. This topic provides an introduction to the automation processes.

The processes that you can automate are:

- Delivery of .zip and .csv files to the Oracle WebCenter Content server
- Initiation of HCM Data Loader or HCM Spreadsheet Data Loader to process files and monitor progress on the WebCenter Content server
- Initiation of the HCM Data Loader Data Set Summary extract and retrieval of its output

Delivering Files to Oracle WebCenter Content Automatically

To automate the delivery of .zip files to the WebCenter Content server, you can use:

- The WebCenter Content Document Transfer Utility, a feature-rich Java library that provides content import and export capabilities.
- The **dataLoadDataSets** REST resource provides an uploadFile method for uploading smaller Base64 encoded files.

Initiating the HCM Data Loader and HCM Spreadsheet Data Loader using REST API

You can automate the HCM Data Loader and HCM Spreadsheet Data Loader integrations using **dataLoadDataSets** REST resource to upload files to the Oracle WebCenter Content server to initiate HDL and HSDL and monitor the progress and status of your data sets.

The following table lists the REST API custom actions:

Action	Method	Description
Upload a File	POST	Uploads a file to the HDL import directory of the Oracle WebCenter Content server (UCM). Note: Due to the Base64 encoding of the files, the uploadFile action is suitable for uploading smaller files.
Create a File Data Set	POST	Creates a file based data set for a file preloaded to the UCM, importing valid data into staging tables. Optionally loading valid objects into the application.
Create a Spreadsheet Data Set	POST	Creates a spreadsheet based data set, transferring the UCM file and importing valid data into staging tables using the named HSDL template code. Optionally loading valid objects into the application.
Get all Data Sets	GET	Provides summary information for HCM Data Loader and HCM Spreadsheet Data Loader data sets available in the staging tables.
Get an individual Data Set	GET	Requests the staging table data for data sets identified by the parameter values supplied.
Get Business Object Information	GET	Provides summary information for business objects within a data set.
Get Data Set Messages	GET	Provides details of the messages raised against a specific data set.
Get Data Set Process Information	GET	Provides details of the processes submitted to upload a specific data set.
Stop a running Data Set	POST	Requests that processing of the data set be ceased.
Submit Load or Rollback for a Data Set	POST	Submits the load or rollback processing of a data set.
Delete a Data Set's Staging Table Data	POST	Deletes the staging table data of a non-processing data sets identified by the input parameters.
Stop a running Data Set Business Object	POST	Stops the processing of an in-progress data set.
Submit Load or Rollback for a Business Object within a Data Set	POST	Submits the load or rollback of a business object within a data set.
Retrieve details of the sub-processes submitted by HCM Data Loader	GET	Provides details of the sub-processes submitted by a specific HCM Data Loader data set process
Initiate the generation of a failed lines file for your data set, or an individual business object within your data set	POST	Submits a process to generate a file of the failed lines and METADATA for the business objects in error.

Granting Access to the HDL REST resource

To enable access to initiate processing and monitor your own data sets you'll need this role hierarchy:

Role Name	Role Code
Use REST Service – Data Load Data Set	ORA_HRC_REST_SERVICE_ACCESS_DATA_LOAD_DATA_SETS

To only have access to preassigned HSDL template by default and to also have access to initiate HDL via REST you need to add the following function security privilege:

Role Name	Role Code
Create HCM Data Loader Data Set Using REST Service	HRC_CREATE_FILE_DATA_SET_USING_REST_PRIV

Note:

- Unless this new privilege is granted, customers who've created custom roles to grant access to initiate HDL from REST will stop working.
- It's recommended to enable and configure role access to HDL business objects to control which business objects your roles can upload data with. Without this, user's with createFileDataSet custom access action can upload data for any supported business object.

To upload file to the Oracle WebCenter you'll also need access to the HCM Data Loader import account. Assign this role:

Role Name	Role Code
Upload Data for Human Capital Management file based import	HCM_DATALOADER_IMPORT_RWD

To enable access to initiate process and monitor all data sets, regardless of who created them, you'll need this role hierarchy:

Role Name	Role Code
Load All Data Sets using Data Load Data Sets REST Service	ORA_HRC_LOAD_ALL_DATA_SETS_USING_REST

Set up access to the HCM Data Loader REST API using the steps listed in the [Configure Access to HCM Data Loader](#) tutorial.

To run the REST API custom actions, see the [Using the HCM Data Loader REST API](#) tutorial.

Initiating the HCM Data Loader Data Set Summary Extract Automatically

HCM Extracts is a tool for reporting and outbound integrations. When you use HCM Extracts as part of an integrated process, you can initiate it automatically from an automated flow. The Payroll Flow Engine is the generic processing engine that you use for this purpose.

When you save a copy of the predefined HCM Data Loader Data Set Summary extract, you also create a payroll flow of type Extract. You can initiate this payroll flow using the **FlowActionsService**, which you use to automate and monitor the extract. You use the **payrollProcessingActionService.fetchExtractOutput** method to download the extract output. When the extract is associated with a BI template, the payroll flow ensures that the BI report is generated and written to the BI server.

Initiating HCM Data Loader for HCM Extract Generated Files

You can enable **Payroll flow** tasks to seamlessly submit **HCM Data Loader** or **HCM Spreadsheet Data Loader** for your extract generated files.

- Create a file extract using a new delivery option called the **Data Loader Interface** in the **HCM Extract: Extract Definitions** page.
- Configure and submit a **Payroll flow** by sequencing the created extract and a process called the **Run HCM Data Loader** to process the extract file by submitting the HDL or HSDL process based on the file type and parameters defined in the delivery option.
- You can create a delivery option for both HCM Data Loader and HCM Spreadsheet Data Loader files. For HCM Spreadsheet Data Loader, you can use different spreadsheet templates by configuring HSDL parameters for the delivery option. You can also create multiple delivery options for the same extract.

This tutorial explains in detail the steps to generate HCM Data Loader (HDL) and HCM Spreadsheet Data Loader (HSDL) formatted data files using HCM Extracts and how to automate the import and load of those files: *[Initiate HCM Data Loader for HCM Extract Generated Files](#)*.

Related Topics

- [Overview of HCM Data Loader Extracts](#)
- [How You Configure HCM Data Loader Business Object Access](#)

How to Review the Objects Processed in a Data Set

The **Total Objects** column in both the **Data Sets** and **Business Objects** tables shows the number of processed business objects. When the number is greater than zero, clicking it opens the **Object Status** page.

The **Object Status** page provides the following information for the business objects and component in the data set:

- **User Key, Source System Identifier** and **Source System Owner**
- **Reference Names** and **Reference Values**
- **Import Status**
- **Load Status**
- **Roll Back Status**
- **Object Details** which opens the **Object Hierarchy** page.
- You can filter the list by:
 - Specified business object in the data set
 - Specified Load Status

- Specified reference value

You can generate and download a spreadsheet for HCM Spreadsheet Data Loader data sets on the **Import and Load Data** page.

- Click the spreadsheet icon in the **Review** column of the **Data Sets** table.

Note: You can only download those data sets for which you have the access to view.

Maintaining Oracle Search Indexes

Capabilities such as Global Search and Connections use Oracle Search indexes to expedite responses to your queries.

While loading data, HCM Data Loader captures the Fusion surrogate ID of all the records successfully created, updated, and deleted by your data set.

After which, it automatically submits the **Update Oracle Search with HCM Data Loader Changes** process to maintain the Oracle Search indexes with the changes made.

You can review the status of this process by clicking **Review Processes** for your data set in the **Import and Load Data** task.

Note: Any records that fail to be bulk-ingested into the Oracle Search indexes will be automatically retried for loading at an hourly interval.

Disabling Oracle Search Bulk Ingestion

If you frequently load multiple small files for an object, you can consider disabling the automatic submission of the **Update Oracle Search with HCM Data Loader Changes** process.

Note: It should only be considered when the frequency of loading data for a business object could cause a backlog of process submissions.

To disable the **Update Oracle Search with HCM Data Loader Changes** process include the following instruction at the top of your file:

```
SET DISABLE_POST_PROCESS_TASKS OracleSearchBulkIngestion
```

HCM Data Loader will continue to capture the details needed for the bulk ingestion of Oracle Search indexes despite disabling the **Update Oracle Search with HCM Data Loader Changes** process.

You must manually submit the **Update Oracle Search with HCM Data Loader Changes** process to keep the Oracle Search Indexes updated.

Initial Ingestion of Oracle Search Index

If the update you're making impacts the vast majority of records for an Oracle Search index, you may want to consider disabling bulk ingestion and reinitializing the Oracle Search index instead.

Note: As the initial ingestion of an index can take many hours, it should not be used if only a few records within the indexes are impacted by your update.

To disable bulk-ingestion include the following instruction at the top of your file:

```
SET ORACLE_SEARCH_UPDATE_MODE INITIAL
```

It disables the capture of the Fusion surrogate IDs for Oracle Search bulk-ingestion and the automated submission of the **Update Oracle Search with HCM Data Loader Changes** process.

Manually Submitting the Bulk-Ingestion Process for Oracle Search Indexes

You can manually submit the **Update Oracle Search with HCM Data Loader Changes** process when the automatic submission is disabled.

To submit the bulk-ingestion process, click **Update Oracle Search Indexes** within the **Import and Load Data** task.

You can submit the process:

- For all changes made by an individual data set identified by its Content ID.
- For all unprocessed changes for a specific HCM Data Loader business object, such as Worker.

How to Enable Bulk Ingestion for Oracle Search Indexes

This functionality is enabled by default and controlled by the **Enable Bulk-Ingestion of Oracle Search Indexes** parameter in the **Configure HCM Data Loader** task, available from **Setup and Maintenance**.

Note: Bulk-Ingestion of Oracle Search Indexes is only available if the **Enable/Disable Search Ext Framework** (ORA_FND_SEARCH_EXT_ENABLED) profile is enabled.

FAQs for Importing and Loading Data Using HCM Data Loader

What's an HCM Data Loader data set?

A set of data supplied together to be loaded into Oracle HCM Cloud. The source of a data set is either a compressed data file containing delimited business-object-specific data files or an HCM Spreadsheet Data Loader spreadsheet.

What happens in the import stage ?

The application imports individual file lines from your compressed data file to the HCM Data Loader stage tables.

Related file lines are grouped to form logical business objects, and some preliminary validation occurs. Errors reported at this stage are specific to the object itself. The object's dependencies on other objects aren't assessed.

Why do I get an error about exceeding child record limit during import stage?

Files for objects such as User Defined Tables, Schedule Request, Organization Tree, Department Tree, Grade Ladder, and Progression Grade Ladder often contain large volumes of child records associated with a single parent record.

An additional validation is done to detect files with more than 2000 child records per parent. If the file surpasses this threshold, an error will be raised during the import process to prevent performance degradation.

To resolve this error, split the data into two separate files: one containing the parent record and the other containing only the child records, and then load them.

What happens in the load stage?

The application validates foreign-object references and applies business-object-specific validation rules. If the object passes all validation, then it's loaded to the Oracle HCM Cloud application tables. If any component of the object fails validation, then the whole logical object fails to load.

You can monitor object status on the Import and Load Data page. Errors reported at this stage may relate to either the object itself or its dependencies on other objects.

How can I review the status of the objects in my file?

When you import data into the stage tables, HCM Data Loader records the status of all imported objects. On the Import and Load status page, the Total Objects count increases as the import process progresses.

Click the count to open the Object Status page and review your objects. If the process hasn't finished, refresh the Object Status page until all imported objects appear.

How can I view the data in error for a data set or business object?

If your data file import encounters errors, then the Failed Lines and Failed Objects columns on the Import and Load Data will become active links.

Click the number to open the Failed Lines or Failed Objects page, as appropriate, to see error details. From the Failed Objects page, you can link to the Object Hierarchy page and correct individual attribute values.

Or, you can click the **Messages** icon on the Import and Load Data page to open the Messages page. From the Messages page, you can navigate to error details. For failed objects, you can open the Object Hierarchy page.

How do I fix HCM Data Loader errors?

You need to fix import errors in the source data and upload the corrected data again in a new .zip file. You can't process the existing .zip file again.

You can also fix load errors in the source data and upload the corrected data again. Or, you can correct attribute values in the Attribute Details dialog box on the Object Hierarchy page for a failed object. In this case, you can resubmit the load process for corrected objects on the Import and Load Data page.

How do I keep up the bulk-data loading performance?

If it takes longer to load data using HCM Data Loader, then it's likely that your staging tables are near capacity. You can review the stage table capacity used on the Recent File Loads task.

For information on how to clean up your staging tables, see the Maintaining Stage Tables chapter.

Why do I get a warning when loading security related data?

You don't have the access to bulk-load data using security related objects.

Currently, the enhancement to restrict access to security related objects is disabled, so you can still successfully use these objects but a warning is raised. You must request access to bulk-load security related data before the functionality is enabled, after which attempts to bulk-load data for security related objects will fail.

Related Topics

- [How You Enable Access to HCM Data Loader](#)
- [HCM Data Loader Configuration Parameters](#)

Why can't I roll back data I previously loaded?

You may not be able to roll back data you previously loaded for these reasons: the business object may not support rollback, there may be no loaded objects available to roll back, or the records may have been updated since they were initially loaded.

To check if a business object can be rolled back, review the business object details in the **View Business Objects** task.

FAQs for Bulk Ingestion for Oracle Search Indexes

Why is the Update Oracle Search with HCM Data Loader Changes process not submitted after my data set?

The **Update Oracle Search with HCM Data Loader Changes** process is only submitted after data sets which successfully load data for business objects referenced by Oracle Search Indexes.

8 Initiate File Processing on the Go

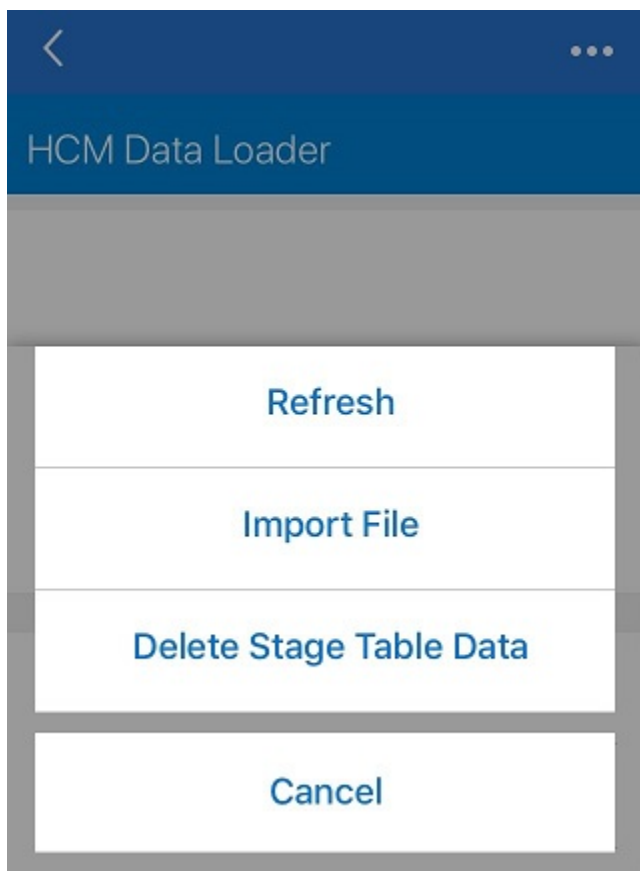
Using the Recent File Loads Page

You can now Import and Load Data from your mobile device. The Recent File Loads page shows you all the files that were submitted for processing by HCM Data Loader and enables you to import and load files as well.

You can choose to view specific uploaded files by using the filters on the page. For example, to only view files that failed to load successfully, use the Status filter. This page displays a subset of the information available on the Import and Load Data page, but unlike the Import and Load Data, you can access this page from your mobile devices.

On mobile devices, you can initiate the import and load of files that are available on the Oracle WebCenter Content server, by clicking the Import File action. Like the Import and Load Data page, you can optionally override the default parameters when submitting a file, refer to Import and Load Data for details of the parameters. You can also access the Delete Stage Table Data task from the Recent File Load page on your mobile devices. For more information, refer to Maintaining Stage Tables. You can access the other tasks from your desktop browsers using the Actions Menu.

The images show the new mobile user interface for importing and deleting stage table data.



9 Data Loading Using Spreadsheets

Spreadsheet Data Loaders

In this topic, you learn about using the spreadsheet data loader provided with Oracle HCM Cloud. To use this tool, you must install the Oracle ADF Desktop Integration Add-in for Excel.

HCM Spreadsheet Data Loader

You can use HCM Spreadsheet Data Loader to load most objects that HCM Data Loader supports. The exceptions are components, such as Document Record Attachment and Person Image, that load content like PDF files and images into the Cloud. Here's how the spreadsheet gets processed:

- Rows in the spreadsheet are initially saved to the HCM Spreadsheet Data Loader stage tables
- Stage table data is reformatted to the hierarchical shape supported by HCM Data Loader
- Reformatted stage table data gets saved to the HCM Data Loader staging table

You can find HCM Spreadsheet Data Loader in the Data Exchange work area.

Single-Object Spreadsheet Loaders

Oracle Fusion HCM provides many spreadsheet loaders for specific object types.

In most cases, you:

- Create the spreadsheet on the page where you manage the object.
- Load the objects directly to the application tables.
- Correct any errors in the spreadsheet.

In many cases, you can both create and update the object in a spreadsheet.

Single-object spreadsheets are most useful for users such as human resource specialists and benefits specialists, who are creating objects of a specific type in bulk. You're less likely to use these types of spreadsheet loaders for integrations.

This table shows some key objects that you can create in bulk using this type of spreadsheet. It also identifies the page or tab where you generate the spreadsheet.

Note: Some of these are HCM Spreadsheet Data Loader spreadsheets embedded in the relevant page or tab.

Business Object	Page or Tab
Benefit Enrollment	Evaluation and Reporting work area, Enrollment Uploads tab
Benefit Plan	Manage Benefit Plans page

Business Object	Page or Tab
Benefit Program	Manage Benefit Programs page
Competency	Manage Content Items page
Department	Manage Departments page
Job	Manage Jobs page
Location	Manage Locations page
Performance Rating	Manage Content Items page
Postal Code Range and Service Area	Manage Postal Code Ranges and Service Areas page
Regulation	Manage Regulations page
Reporting Group	Manage Reporting Groups page
Stock Grant	Manage Stock Grants page
Variable Rate Profile	Manage Benefit Rates page
Work Pattern	Manage Work Schedules page
Work Schedule Assignment	Manage Work Schedule Assignment Administration page
Work Schedule	Manage Work Schedules page
Worker Goal	Administer Goals page

For some objects, such as Benefit Rate and Assignment, you can perform a mass update in a spreadsheet. To update these objects, you download existing objects to a generated spreadsheet, edit the objects, and upload the changes.

Set Up Desktop Integration for Excel

You can create or edit records and upload them to the application using integrated Excel workbooks. To use these workbooks, you must install an Excel add-in.

Note: Every release of Oracle Fusion Cloud Applications is likely to have a new version of the Oracle ADF Desktop Integration add-in. When prompted, install the latest version of the add-in.

- For information on the new version, see the relevant *What's New* document.
- For information on the upgrade policy for the Oracle ADF Desktop Integration add-in, see: *Add-In Upgrade Policy for ADF Desktop Integration* (DOC ID 2242879.1) on *My Oracle Support*.

Install or Upgrade the Desktop Integration Add-In

To install or upgrade the Oracle ADF Desktop Integration Add-In for Excel, you must have the appropriate Excel and Windows version. See: *Supported Platforms for ADF Desktop Integration* (DOC ID 2242428.1) on *My Oracle Support*. Also, make sure that you have the Download Desktop Integrator (ATK_DOWNLOAD_DESKTOP_INTEGRATOR_PRIV) privilege to access the add-in installer.

1. Sign into your computer with your account.
2. Close Excel.
3. Click **Navigator > Tools > Download Desktop Integration** to download the installer file.
4. Run the `adfdi-excel-addin-installer-current-user.msi` installer just like you would run any program that you install on your computer.
5. Run Excel once to complete the installation.

Note: If you have administrator privileges, you can also install the add-in for all users. For more information, see: *How to install the ADF Desktop Integration Add-in for Excel* (DOC ID 2681794.1) on *My Oracle Support*.

Related Topics

- [Troubleshoot Desktop Integration for Excel](#)
- [Guidelines for Using Desktop Integrated Excel Workbooks](#)
- [Business User's Guide for ADF Desktop Integration](#)

Troubleshoot Desktop Integration for Excel

If you run into any issues with the integration, use the Client Health Check tool first to find out what integration issues you might have and how to resolve them.

For more information, see: *How to use the ADF Desktop Integration Client Health Check Tool* (Doc ID 2010222.1) on *My Oracle Support*.

If the health check tool doesn't resolve your issue, you can uninstall and reinstall the Oracle ADF Desktop Integration add-in. For more information, see: *Information Center: Troubleshooting Oracle ADF Desktop Integration* (DOC ID 2012600.2) on *My Oracle Support*.

Note: To check the version of Oracle ADF Desktop Integration, see: *ADFDi Version Check* (Doc ID 2012570.1) on *My Oracle Support*.

Uninstall the Desktop Add-In

Use the Microsoft Windows Settings to remove the ADF Desktop Integration add-in. After removing the add-in, you can use the integrated Excel workbooks on this system only after you reinstall the ADF Desktop Integration add-in.

1. Open **Windows Settings** and go to **Apps & features**.
2. Select **Oracle ADF Desktop Integration Add-In for Excel** from the list of programs.
3. Click **Uninstall** from the toolbar at the top of the program list.
4. When prompted, click **Uninstall** to remove the Oracle ADF Desktop Integration add-in.

Note: If you have installed the Oracle ADF Desktop Integration add-in on multiple user profiles using the current-user installer, you must remove it from each user profile.

Related Topics

- [Set Up Desktop Integration for Excel](#)

10 Introduction to HCM Spreadsheet Data Loader

Guidelines for Using HCM Spreadsheet Data Loader

You can load data to the HCM Data Loader stage tables from either a data (.dat) file or a spreadsheet. Your approach depends largely on how often you load data, the complexity of the data that you're loading, who performs the bulk update.

HCM Data Loader is designed for the integration specialist, having access to HCM Data Loader allows the user to load data for supported objects. HCM Spreadsheet Data Loader is designed with the business user in mind. You can secure the spreadsheets to implement the data security policies of the spreadsheet user. This topic describes some key features of data loading from spreadsheets. This information helps you to identify the best approach for your business case.

Spreadsheet Templates

You must generate the spreadsheet from a spreadsheet template. Some spreadsheet templates are predefined. To create templates, you can either copy a predefined template and edit it or create a template from scratch.

Tip: You can access Oracle By Example tutorials for HCM Data Loader in the Oracle Learning Library. Search for HCM Spreadsheet Data Loader to find HCM Spreadsheet Data Loader content.

When to Use Spreadsheets

Spreadsheets offer many advantages. For example:

- Spreadsheets are easy to use.
- You can create spreadsheet templates and generate spreadsheets tailored to your business case. For example, you can omit attributes, change attribute order, change attribute labels, and add help text. You can also define different templates for creating and maintaining the same business objects.
- You can enable business users to load data for selected spreadsheet templates.
- You can save data periodically from a spreadsheet to the stage tables.
- Errors are reported in the spreadsheet, and you can correct them there.
- You can mix your use of spreadsheets with use of .dat files. For example, you can load data from a spreadsheet but maintain it using .dat files.
- You can use spreadsheets to delete data.

- In specific cases, spreadsheets are the simpler option. For example, to load an element entry for an element with 5 lines from a .dat file, you supply:
 - 1 record for the element entry
 - 1 record for each name and value

In a spreadsheet, you supply just 1 spreadsheet line where the values are individual columns on the same line.

- You can restrict the job roles that can access a spreadsheet. For example, you can make the spreadsheet to create new hires available only to the Human Resource Specialist.

Spreadsheet Restrictions

When loading data from a spreadsheet:

- You can't upload data to attributes whose data type is binary large object or character large object, like images and file attachments.
- HCM Spreadsheet Data Loader doesn't recognize source keys. Therefore, objects that you load from a spreadsheet have default source keys. To maintain such objects using .dat files, you supply user keys or you can retrieve the source key values to use, using the Integration Object User Key Map Extract.
- You can't create objects or object components, such as Profile Item, that don't supply user keys. However, you can update such objects if you define a list of values on the local surrogate ID attribute to identify the record to update.
- You can't update the values of the user-key attributes if they identify the record to update. Therefore, you're recommended to define separate spreadsheets for creating and updating objects. In the spreadsheet for updating an object, you can identify the record to update in one of these ways:
 - Include the local surrogate ID attribute, which has a predefined LOV.
 - Include the primary user key attribute, but configure an LOV for the attribute value. With this approach, the user sees a recognizable value in the spreadsheet.

Note: Whether you load data from a spreadsheet or a .dat file, no approval processes exist for loaded objects.

Setup Requirements

All users must set up desktop integration for Microsoft Excel to use HCM Spreadsheet Data Loader.

Excel Security Requirements

You don't need to enable macros or change your ActiveX settings for HCM Spreadsheet Data Loader.

Related Topics

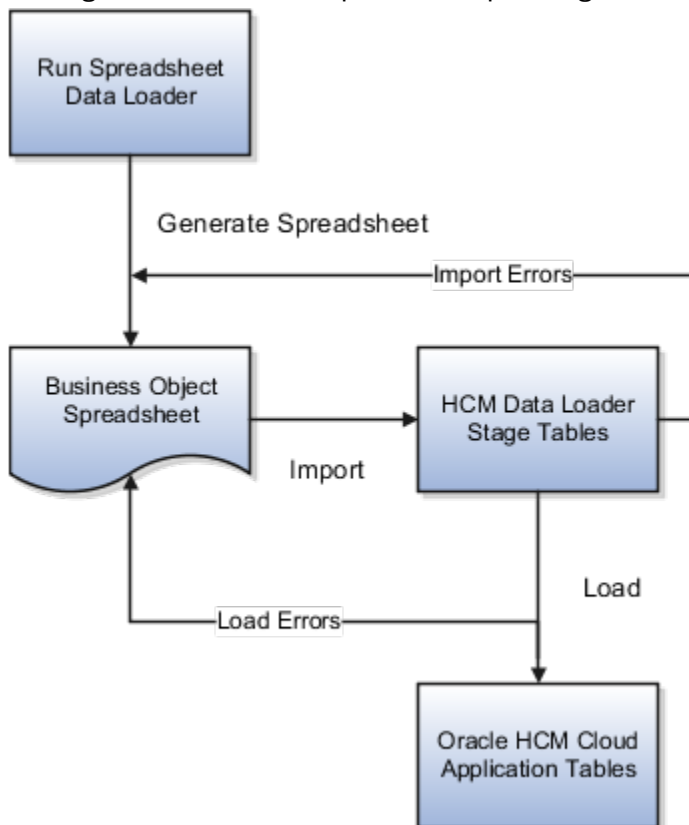
- [HCM Spreadsheet Data Loader Templates](#)
- [Set Up Desktop Integration for Excel](#)
- [Configure Lists of Values for Attribute Validation](#)

How Data Is Uploaded Using HCM Spreadsheet Data Loader

When you click Upload in a spreadsheet, your data is uploaded to Oracle HCM Cloud. Behind the scenes, upload is a two-stage process.

In the first stage, HCM Spreadsheet Data Loader imports the data to the HCM Data Loader stage tables. In the second stage, HCM Data Loader loads valid business objects to the application tables.

This figure summarizes the process of uploading data using HCM Spreadsheet Data Loader.



How Spreadsheet Data Is Loaded

The details of this process are:

1. You use the **Run Spreadsheet Data Loader** task in the Data Exchange work area to generate a spreadsheet for a business object from a spreadsheet template directly. You can also access spreadsheet templates directly from the business objects pages, such as Job, Location, Position.
2. You click **Create Data Set** in the spreadsheet to create a data set for the data that you want to load.
3. You enter data in the spreadsheet and click **Upload**. This action imports the spreadsheet data set to the HCM Data Loader stage tables.

4. Some validation occurs in the stage tables. Any import errors are reported in the **Progress** and **Message** columns of the **Spreadsheet Line Status** section of the spreadsheet when you click **Refresh**.

Errors at this stage are rare, as the spreadsheet format enforces business-object structure and rules.

5. As each object loads successfully to the stage tables, HCM Data Loader calls the relevant object service to load the object to the application tables.
6. The object service validates the object. Any errors are reported in the **Progress** and **Message** columns of the **Spreadsheet Line Status** section of the spreadsheet when you click **Refresh**.
7. You correct any import or load errors in the spreadsheet and click **Upload** again. This action imports the corrected data to the stage tables again.

Tip: You can review the status of an HCM Spreadsheet Data Loader data set on the HCM Data Loader Import and Load Data page. Search for data sets by file type to list those loaded from spreadsheets.

The Instruction worksheet of the spreadsheet may contain instructions on how to use the spreadsheet, while the Messages worksheet holds a summary of all import and load messages generated for this data set.

Related Topics

- [Upload Data Using HCM Spreadsheet Data Loader](#)

How You Test the HCM Spreadsheet Data Loader Process Flow and Connections

Before you attempt to load your own data, you may want to test HCM Spreadsheet Data Loader end-to-end processing without creating unwanted data. This topic describes how to test HCM Spreadsheet Data Loader end-to-end processing.

Testing End-to-End Processing

To test end-to-end processing, you use:

- A process, **Test HCM Data Loader Process Flow and Connections**
- An object, Example Object

When you run the **Test HCM Data Loader Process Flow and Connections** process, you set the **Action** parameter to **Test HCM Spreadsheet Data Loader**. The process:

1. Generates a CSV file of data for the example object
2. Starts HCM Spreadsheet Data Loader to import and load the file
3. Creates a log file, which you can access on the Scheduled Processes page

About 200 hundred records of example data are imported and loaded to the object's own tables. No data is saved to the application tables.

The Example Object

The example object:

- Has child and grandchild components
- Supports create and update actions
- Has attributes of most data types

A predefined spreadsheet template is provided for the example object. The generated CSV file is based on this template.

Deleting the Test Data

You're recommended to run the **Test HCM Data Loader Process Flow and Connections** process periodically to delete the test data and associated data sets. When you run the process, you set the **Action** parameter to **Delete test data**. The process deletes all example data, regardless of whether the data was created by HCM Data Loader or HCM Spreadsheet Data Loader. It generates a file of HCM Data Loader DELETE instructions, adds the file to the **DeleteExampleObject||<date and time>|.zip** file, and uploads it.

Spreadsheet Business Object Access

Users who have the predefined Human Capital Management Integration Specialist job role can create and maintain spreadsheet templates for any business object. They can also generate spreadsheets from templates and perform all data-set operations, such as create and upload.

Note: The predefined Human Capital Management Integration Specialist job role also provides access to HCM Data Loader and other tools which you may not want your users to have access to. You're encouraged to create custom roles to grant access to just the tools your users need.

Some other business-user roles inherit HCM the ability to load data using HCM Spreadsheet Data Loader, but these need to be assigned to pre-configured and custom templates. This topic describes how to enable other roles to:

- Create and import spreadsheet templates.
- Maintain spreadsheet templates.
- Generate spreadsheets and create, save, and upload data sets from those spreadsheets.
- View all data sets in spreadsheets generated from a specific template.

It also introduces some security setup that you must perform to enable users to access relevant tasks in the Data Exchange work area.

Tip: You can refer to the Configure Access to HCM Spreadsheet Data Loader (HSDL) tutorial, available in the *Oracle Learning Library*.

Creating and Importing Spreadsheet Templates

To enable users to create and import spreadsheet templates, you perform the **Spreadsheet Business Object Access** task. Using this task, you can:

- Select a role and assign one or more business objects to it.

For example, you can select a custom Compensation Administrator job role and assign all compensation objects to the role. Users who have the job role can create and import spreadsheet templates for all compensation objects.

- Select a single business object and assign one or more custom roles to it.

For example, you can select the Job object and assign custom Human Resource Specialist and Human Capital Management Application Administrator job roles to it. Users who have either role can create and import spreadsheet templates for the Job object.

In both cases, users can't create and import spreadsheet templates for other business objects, unless they have other custom roles that provide that access.

Tip: When you assign business objects individually, you must assign the translation objects explicitly, if required. They're not assigned by default with the base object.

Maintaining Spreadsheet Templates

While spreadsheet templates are in **Draft** status, the users who created them can maintain them. When a spreadsheet template is in other statuses, then only the roles listed in the Template Maintenance section of the spreadsheet template can maintain it.

Generating Spreadsheets and Performing Data Set Operations

Users who can create and import spreadsheet templates determine who can generate spreadsheets and perform data set operations. On the Manage Role Access tab of the spreadsheet template, they identify the roles that can:

- Create, save, and upload data sets from spreadsheets generated from the template.
- View all data sets in spreadsheets generated from the spreadsheet template.

For example, a user with a custom Compensation Administrator job role may create a spreadsheet template named US Compensation Grades for loading Grade objects. On the Manage Role Access tab of the template, that user selects a custom Compensation Analyst role and selects the **Create**, **Save**, and **Upload** options. Therefore, any user with the custom Compensation Analyst job role can:

- Generate spreadsheets for the Grade object from the US Compensation Grades spreadsheet template.
- Create, save, and upload data sets from the generated spreadsheets.

Note: The Role Access tab of the spreadsheet template is enabled only if the **Assign Roles to Template** field is set to **Enabled**. This field appears in the Template Security section of the Define page of the spreadsheet template.

Enabling Task Access

Users who have the predefined Human Capital Management Integration Specialist job role or a HSDL administrator role can access the Data Exchange work area and perform these tasks:

- Spreadsheet Templates
- Run Spreadsheet Data Loader

Users who have one of the following job roles can access the Data Exchange work area and perform the Run Spreadsheet Data Loader task:

- Compensation Analyst
- Compensation Manager
- Human Resource Specialist
- Human Capital Management Application Administrator
- Payroll Manager
- Payroll Administrator
- Payroll Interface Coordinator
- Time and Labor Manager

Other users need access to the same tasks so that they can:

- Create, import, and maintain spreadsheet templates.
- Generate spreadsheets and perform permitted data set operations.

To provide this access, you add relevant privileges to the custom roles that are authorized to use HCM Spreadsheet Data Loader. You must have the IT Security Manager job role or privileges to perform this task.

Related Topics

- [Manage Spreadsheet Business Object Access](#)
- [How You Enable Access to HCM Spreadsheet Data Loader Tasks](#)
- [How You Associate a User Name with a Data Upload](#)
- [How You Enable Administrator Access to HCM Spreadsheet Data Loader](#)

Manage Spreadsheet Business Object Access

Perform the Spreadsheet Business Object Access task to enable selected roles to create and import spreadsheet templates for selected business objects. You can secure this access by either role or business object:

- To secure access by role, you select a single role and assign one or more business objects to it.
- To secure access by business object, you select a single business object and assign one or more roles to it.

This topic describes both methods. In the **Setup and Maintenance** work area, go to the following for your offering:

- Functional Area: HCM Data Loader

- Task: Spreadsheet Business Object Access

Securing Access by Role

To secure access by role, follow these steps:

1. On the Spreadsheet Business Object Access page, select the **By Role** option in the page title.
2. In the Job and Abstract Roles section of the page, enter the name of the role in the **Role** query-by-example field and press **Enter**. For example, enter **Compensation Administrator Custom**.
3. In the results, select the role.

The Assigned Business Objects section title is updated automatically to include the role name.

4. In the Assigned Business Objects section, select an entry in the **Assign** menu.

If you select **Assign Individual Business Objects**, then:

- a. Search for and select the business objects in the **Search and Select Business Objects** dialog box. For example, search by the **Compensation** product area and select specific objects.
- b. Click **Add** to add the selected business objects to the role. An entry appears in the Assigned Business Objects section for each selected business object.

5. If you select **Assign All Business Objects in a Product Area**, then:

- a. Select the product area in the Select Product Area dialog box. For example, select **Compensation**.
- b. Click **Add**. A single entry appears for all compensation objects in the Assigned Business Objects section.

6. If you select **Assign All Business Objects**, then a warning message appears. Click **Add** to close the warning and continue. A single entry appears for all business objects in the Assigned Business Objects section.

7. Click **Save**.

Users who have the selected role can now create and import spreadsheet templates for the selected business objects.

Securing Access by Business Object

To secure access by business object, follow these steps:

1. On the Spreadsheet Business Object Access page, select the **By Business Object** option in the page title.
2. In the Business Object section of the page, enter the name of the business object in the **Business Object** query-by-example field and press **Enter**. For example, enter **Job**.
3. In the results, select the business object.

The Assigned Roles section title is updated automatically to include the object name.

4. In the Assigned Roles section, click the **Add** icon.
5. In the Search and Select Roles dialog box, search for and select one or more roles. For example, select **Human Resource Specialist Custom**.
6. Click **Add**.

An entry appears in the Assigned Roles section for each of the selected roles.

7. Click **Save**.

Users who have the selected roles can now create and import spreadsheet templates for the selected business object.

Related Topics

- [Spreadsheet Business Object Access](#)

How You Enable Access to HCM Spreadsheet Data Loader Tasks

You can perform the following tasks in the Data Exchange work area using the predefined job roles Human Resource Specialist and Human Capital Management Integration Specialist:

- Spreadsheet Templates
- Run Spreadsheet Data Loader
- Delete Spreadsheet Stage Table Data

This topic explains how to give other roles access to these tasks. You must have the IT Security Manager job role or privileges.

Access to the Spreadsheet Templates Task

On the Spreadsheet Business Object Access page, you can authorize selected custom roles to create and import spreadsheet templates for selected business objects. You must also enable those custom roles to perform the **Spreadsheet Templates** task in the Data Exchange work area. To enable this access, you edit the custom roles to add the following aggregate privilege:

Privilege Name	Privilege Code
Manage HCM Spreadsheet Data Loader Templates	ORA_HRC_MANAGE_HSDL_TEMPLATES

In addition, if the Manage Data Exchange Work Area (`HRC_MANAGE_DATA_EXCHANGE_PRIV`) function security privilege isn't already granted to the role, then you must grant it.

If you want the users of this role to upload data for testing purposes using the Spreadsheet Templates preview functionality, you should also grant this aggregate privilege:

Aggregate Privilege Name	Aggregate Privilege Code
Load Data Using HCM Spreadsheet Data Loader	ORA_HRC_LOAD_DATA_USING_HSDL

Access to the Run Spreadsheet Data Loader Task

On the Manage Role Access tab of a spreadsheet template, you can authorize selected custom roles to generate spreadsheets and perform data-set actions. You must also enable those custom roles to perform the **Run Spreadsheet Data Loader** task in the Data Exchange work area. To enable this access, you edit the custom roles to add the following aggregate privileges:

Aggregate Privilege Name	Aggregate Privilege Code
Access HCM Spreadsheet Data Loader Templates	ORA_HRC_ACCESS_HSDL_TEMPLATES
Load Data Using HCM Spreadsheet Data Loader	ORA_HRC_LOAD_DATA_USING_HSDL

In addition:

- Roles that are authorized to maintain a spreadsheet template must have the Manage HCM Spreadsheet Data Loader Templates (`ORA_HRC_MANAGE_HSDL_TEMPLATES`) aggregate privilege. You authorize a role to maintain a spreadsheet template on the Manage Role Access tab of the template.
- If the Manage Data Exchange Work Area (`HRC_MANAGE_DATA_EXCHANGE_PRIV`) function security privilege isn't already granted to the role, then you must grant it.

Access to the Delete Spreadsheet Stage Table Data Task

To enable custom roles to perform the Delete Spreadsheet Stage Table Data task in the Data Exchange work area, add the following function security privilege:

Function Security Privilege Name	Function Security Privilege Code
Delete HCM Spreadsheet Data Loader Stage Data	HRC_DELETE_HSDL_STAGE_DATA_PRIV

In addition, if the Manage Data Exchange Work Area (`HRC_MANAGE_DATA_EXCHANGE_PRIV`) function security privilege isn't already granted to the role, then you must grant it.

Here's a tutorial on configuring roles to access HCM Spreadsheet Data Loader functionality [Configure Roles to Access HCM Spreadsheet Data Loader](#)

Here's a tutorial configure access for your business users and inbound integrations to use HCM Spreadsheet Data Loader templates [Grant Access to Run HCM Spreadsheet Data Loader](#)

Related Topics

- [Spreadsheet Business Object Access](#)
- [Edit Job Role and Abstract Role](#)

How You Enable Administrator Access to HCM Spreadsheet Data Loader

There are two administrator aggregate privileges available for HCM Spreadsheet Data Loader, which allow you to give access to:

- Maintain all HSDL templates
- Access all data set and data set actions

Access to Maintain All Templates

If you want to have a single role responsible for maintaining all HSDL templates, add this aggregate privilege to your custom role:

Aggregate Privilege Name	Aggregate Privilege Code
HCM Spreadsheet Data Loader Template Administration	ORA_HRC_HSDL_TEMPLATE_ADMINISTRATION

This administration role provides the following access:

- The **Spreadsheet Templates** task
- Create spreadsheet templates for any object that supports HSDL
- Maintain all existing templates

Access to Assign Roles to All Templates

To allow the user to assign roles to access any **HCM Spreadsheet Data Loader** template, including those based on secured business objects that don't implement data security in HSDL, add this privilege to your custom role:

Privilege Name	Privilege Code
Assign Roles to any HCM Spreadsheet Data Loader Template	HRC_ASSIGN_ROLES_TO_ANY_HSDL_TEMPLATE_PRIV

Access to All Data Sets and Data Set Actions

Apart from assigning roles to access the templates, you can configure the data set access for each role, such as create, save, upload, roll back and view all.

For business object templates where the object's application data security isn't extended to HCM Spreadsheet Data Loader, you won't be able to assign the role access. In such cases, you can create a custom role with this administrator access to grant data set permissions on all HSDL templates.

Aggregate Privilege Name	Aggregate Privilege Code
HCM Spreadsheet Data Loader Data Set Administration	ORA_HRC_HSDL_DATA_SET_ADMINISTRATION

This administration role provides the following access:

- The **Run Spreadsheet Data Loader** task
- Generate a spreadsheet from any active template
- Create, save, upload, roll back and view all data sets within a generated spreadsheet, where the business object supports the capability.

However, in order to be able to upload data using the template you should grant this aggregate privilege

Aggregate Privilege Name	Aggregate Privilege Code
Load Data Using HCM Spreadsheet Data Loader	ORA_HRC_LOAD_DATA_USING_HSDL

Here's a tutorial on configuring roles to access HCM Spreadsheet Data Loader functionality [Configure Roles to Access HCM Spreadsheet Data Loader](#)

How You Enable Audit of Role Configuration for a Spreadsheet Template

You can configure the data-set operations a role has for HCM Spreadsheet Data Loader templates using the **Spreadsheet Templates** task. Any changes made to the role configuration can be audited and reviewed using the **Audit Reports** task found in the **Tools** area.

Follow these steps to enable the audit of HSDL Template Roles:

1. Perform the **Manage Audit Policies** task in the **Setup and Maintenance** work area.
2. Select **Audit Level** as **Auditing** for **Oracle Fusion Applications**.
3. Click on **Configure Business Object Attributes**.
4. Select the product as **HCM Common Architecture**.

5. Select **HSDL Template Roles** node under **HCM Spreadsheet Data Loader** in the hierarchy.

Configure Business Object Attributes ?

Product HCM Common Architecture ▼

Objects

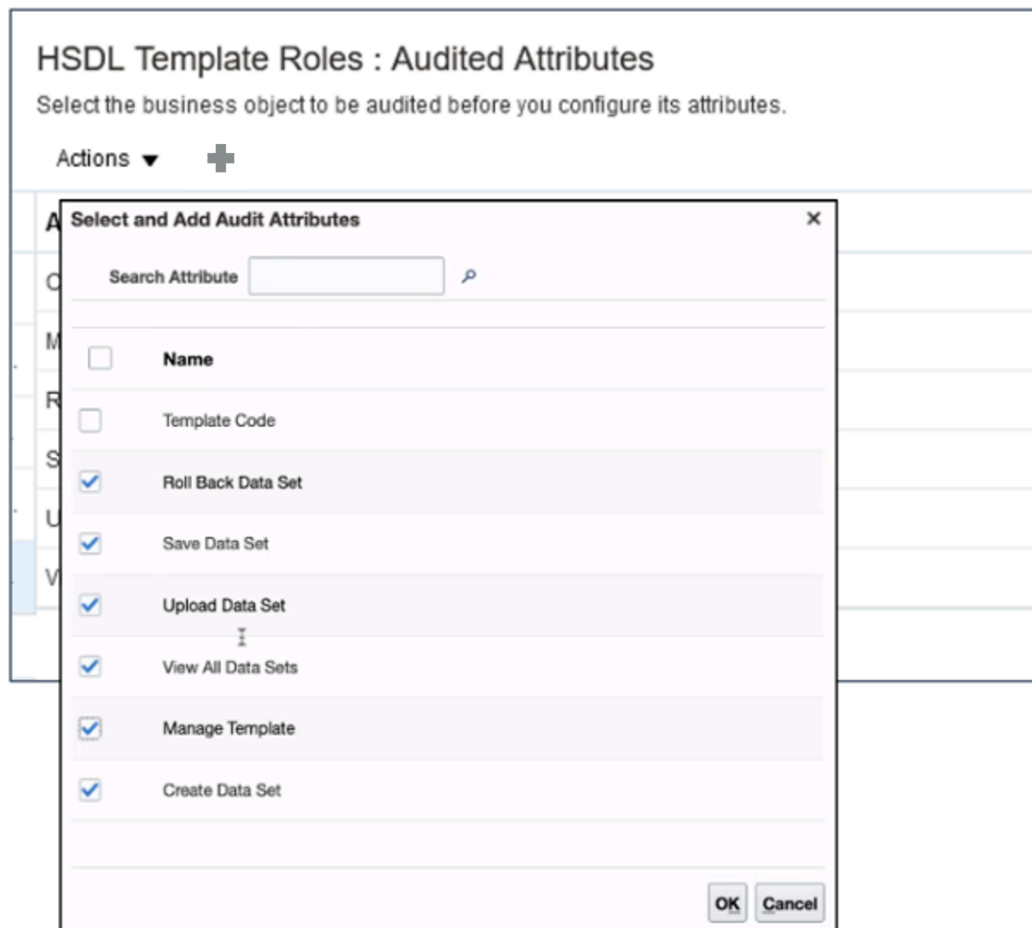
Select the product to be audited.

Actions ▼ View ▼

Audit	Name
<input type="checkbox"/>	▲ Audit Top Node
<input type="checkbox"/>	▲ HCM Data Loader
<input type="checkbox"/>	Configure HCM Data Loader Parameters
<input type="checkbox"/>	▲ HCM Spreadsheet Data Loader
<input checked="" type="checkbox"/>	HSDL Template Roles

6. Click **Create** under the **HSDL Template Attributes: Audited Attributes**

7. In the **Select and Add Audit Attributes** dialog, select the following:
 - a. Roll Back Data Set
 - b. Save Data Set
 - c. Upload Data Set
 - d. View All Data Sets
 - e. Manage Template
 - f. Create Data Set



8. Click **OK**.
9. Click **Save**.

Related Topics

- [How You Audit Objects Loaded by HCM Spreadsheet Data Loader](#)

How You Audit Objects Loaded by HCM Spreadsheet Data Loader

You can audit changes made to objects when you bulk load them using HCM Spreadsheet Data Loader. When audit is enabled, the creation, update, and deletion of objects are audited.

Enabling Audit for HCM Spreadsheet Data Loader

The **Enable Audit Data** parameter on the HCM Spreadsheet Data Loader parameters tab of the Configure HCM Data Loader page controls audit for the enterprise. By default, the **Enable Audit Data** parameter is set to **No**, but you can override that setting. Alternatively, you can enable audit selectively by setting the **Enable Audit Data** parameter in the relevant spreadsheet template to **Yes**.

Related Topics

- [How You Audit Objects Loaded in Bulk](#)
- [Enable Audit for Oracle HCM Cloud Business Objects](#)

FAQs for HCM Spreadsheet Data Loader

Why can't I access HCM Spreadsheet Data Loader?

You may not have the correct permissions.

You must grant access to either develop or use HCM Spreadsheet Data Loader templates.

Related Topics

- [How You Enable Access to HCM Spreadsheet Data Loader Tasks](#)
- [Spreadsheet Business Object Access](#)

What do cells highlighted in red indicate?

When you leave the required attributes blank, Spreadsheet highlights those cells in red. This is to indicate that you need to provide a value in those cells.

In some cases, you may be able to successfully upload the spreadsheet rows without providing a value for the required attributes. However, you are recommended to provide value for all the required attributes to successfully upload the spreadsheet rows.

Why can't I see the user name of the user who created or last updated the record?

If the spreadsheet template is configured to upload data as a session user, you can see the user's name in the Created By or Last Updated By attribute.

Can I delete data using a spreadsheet?

Yes. In the spreadsheet template, you set the Supported Action value to Delete only. If the action isn't available, then the selected object doesn't support deletion of any of its components.

On the Design page of a template for deleting objects, only those components that can be deleted appear.

When should I use HSDL instead of HDL?

HSDL is designed to support business users in performing bulk updates, while HCM Data Loader provides bulk loading capabilities for many HCM objects using pipe-delimited flat file format. HDL is recommended for data migration and ongoing automated integrations.

The intended user for HDL is a technical professional, specifically an integration specialist. Data files are typically generated by an extract routine, but they can also be manually created for simpler business objects.

HCM Spreadsheet Data Loader (HSDL) utilizes all the HDL processes with a spreadsheet front end. The user first creates an object-specific template that defines the structure of the spreadsheet for data entry.

A spreadsheet used for data entry can provide date pickers, choice lists and searchable LOVs to validate cell values. The template that defines the spreadsheet shape can default attribute values, define the order of the columns and customize the column headings and help text.

This template can then be shared with less technical users or business users to streamline the data entry and perform bulk updates. The user initiates the data upload from the spreadsheet, and errors are reported against the failed lines in the spreadsheet. The data can then be corrected in the spreadsheet, and the upload can be retried.

11 HCM Spreadsheet Data Loader Templates

HCM Spreadsheet Data Loader Templates

HCM Spreadsheet Data Loader loads HCM business objects from spreadsheets, which you generate from spreadsheet templates. A spreadsheet template for a business object or business object component contains selected attributes in a specified order.

Every spreadsheet generated from the template has the same structure and organization as the template. This topic describes spreadsheet templates.

Spreadsheet Template Features

You can configure many components of the spreadsheet template. Therefore, you can create spreadsheet templates to suit business requirements. For example, you can:

- Select attributes to include.
- Specify attribute order and column headings.
- Enable and disable attributes for data entry.
- Hide attributes that are included.
- Provide default values.
- Make optional values required.
- Create instruction text for the spreadsheet.
- Configure Lists of Values to provide attribute validation.

Managing Spreadsheet Templates

To create a spreadsheet template, you can:

- Copy an existing spreadsheet template and edit the copy.
- Create a spreadsheet template from scratch.
- Import a spreadsheet template that was exported from a different environment or from Oracle Cloud Customer Connect.

Predefined Spreadsheet Templates

Predefined spreadsheet templates may exist for commonly loaded objects. These templates contain columns for all supported components and attributes. Search for predefined spreadsheet templates, which have codes beginning **ORA_**, on the Spreadsheet Templates page.

Tip: Copy the predefined templates and edit the copies to suit business requirements.

Spreadsheet Template Status

The status of a spreadsheet template determines whether you can edit it, delete it, or generate spreadsheets from it. This table summarizes the template status values.

Spreadsheet Template Status	Edit	Delete	Generate Spreadsheets
Draft	Yes	Yes	No
Active	Yes See note.	No	Yes
Inactive	Yes	No	No

Note: You can disable attributes in an active spreadsheet template, but you can't remove them.

To make an active spreadsheet template unavailable to users, you change its status to **Inactive**. You can't change it back to **Draft**.

Access to Templates and Generated Spreadsheets

Users who have a HSDL template administrator role can create and maintain spreadsheet templates for any business object. They also perform the **Spreadsheet Business Object Access** task to enable other roles to create and import spreadsheet templates for selected objects.

On the **Manage Role Access** tab of the template, users who can create and import spreadsheet templates identify roles that can:

- Create, save, and upload data sets from spreadsheets generated from the template.
- View all data sets in spreadsheets generated from the template.
- Maintain the template.

Note: The **Manage Role Access** tab is enabled in spreadsheet templates for unsecured objects and objects where the application data security is available in HSDL. To allow the user to assign roles to access any HCM Spreadsheet Data Loader template, including those based on secured business objects that don't implement data security in HSDL, grant the `HRC_ASSIGN_ROLES_TO_ANY_HSDL_TEMPLATE` privilege to your custom role. When maintaining templates, users with this privilege will now see:

- **Assign Roles to Template** on the **Define** page is always **Enabled**
- The **Create** button on the **Role Access** page is always **Enabled**

Create role mappings for business HCM Spreadsheet Data Loader users to access templates.

In generated spreadsheets:

- Users see only data sets that they created, unless their role is associated with the spreadsheet template with the **View All** option.
- Users can perform the data-set operations that are enabled for their roles.

- Users with the **Human Capital Management Integration Specialist** job role or a HSDL data set administrator role can perform all actions and view all data sets.

Note: It's strongly recommended that you avoid granting access to the Human Capital Management Integration Specialist job role. This is a very potent role that exposes the user to many other features that they may not need.

Translating Spreadsheet Templates

You can provide translated versions of these text entries in a spreadsheet template:

- The template name and description
- Template instructions
- Column headings and descriptions

Set the session language to the target language before editing the spreadsheet template to enter translated text.

Note: By default, column headings are based on the attribute label and column help text is based on the attribute description. You can overwrite the column headings and help text as per your use case. To allow this customization, column headings and help text aren't automatically translated for you.

Here's a tutorial to create a HCM Spreadsheet Data Loader template. This tutorial explains the common tasks while creating a template to upload new hires [Creating HCM Spreadsheet Data Loader Templates](#)

Related Topics

- [Create and Edit Spreadsheet Templates](#)
- [Export and Import Spreadsheet Templates](#)
- [Copy Spreadsheet Templates](#)
- [Define Categories for Spreadsheet Templates](#)
- [Spreadsheet Business Object Access](#)

Define Categories for Spreadsheet Templates

If you define many spreadsheet templates, then you may want to group them by category to make retrieval easy. No categories are predefined. This topic describes how to define template categories.

Follow these steps:

1. In the Setup and Maintenance work area, go to the Manage Common Lookups task.
2. On the Manage Common Lookups page, search for the lookup type **ORA_HRC_DL_TEMPLATE_CATEGORY**.
3. In the search results, select **ORA_HRC_DL_TEMPLATE_CATEGORY** to display its lookup codes.
4. Click the **New** icon to add a row to the lookup codes table.
5. Define the lookup codes for your categories. For example, to create a category for worker templates:
 - a. Enter **WORKER_TEMPLATES** in the **Lookup Code** field.

- b.** Enter **Worker** in the **Meaning** field. This value appears in the list of categories on the Spreadsheet Templates page, for example.

Tip: The **Display Sequence** value is a number that specifies the position of this category in the list of categories.

- 6.** Click **Save and Close**.

You can now select your category on the Spreadsheet Templates and Template Definition pages. You can also search by category when you perform an advanced search for a spreadsheet template on the Run Spreadsheet Data Loader page.

Here's a tutorial to create a HCM Spreadsheet Data Loader template. This tutorial explains the common tasks while creating a template to upload new hires [Creating HCM Spreadsheet Data Loader Templates](#)

Related Topics

- [HCM Spreadsheet Data Loader Templates](#)

Create and Edit Spreadsheet Templates

Creating a spreadsheet template is a five-step process. You:

1. Define general information for the template.
2. Design the template's contents and layout.
3. Include template instructions, if required.
4. Specify who can perform data set operations and who can maintain the template.
5. Review the design and generate a test spreadsheet from the template.

This topic provides instructions for the five steps of this process and describes how to edit an existing spreadsheet template.

Define the Template

Follow these steps:

1. In the Data Exchange work area, select **Spreadsheet Templates** to open the Spreadsheet Templates page.
2. Click **Create**.
3. In the Create Template dialog box:
 - a.** Enter a name for the template. By default, data sets created with the template will have the name of the template with a date and time stamp.
 - b.** The **Code** value is generated automatically based on the **Name** value. You can overwrite the code.
 - c.** Select the business object for which you're creating the template.

Note: If you click **Search** at the bottom of the business object list, it opens the **Search and Select: Business Object** window which provides two free-texts fields for **Product Area** and **Business Object** to support wild card searching.
 - d.** Select a legislative data group (LDG), if appropriate. Typically, the LDG is relevant for Global Payroll objects.
 - e.** Select a value in the **Supported Action** field. If you select a business object that doesn't support deletion for any of its components, then **Create and update** appears by default.

- f. Enter a description and select a category, if required.
- g. Click **OK**.

The Define page opens. Some fields contain values that you entered in the Create Template dialog box. The template is in draft status by default. Spreadsheet users can generate spreadsheets from active templates only. Leave the template in draft status until it's ready for use.

4. If the spreadsheet template is for an unsecured object, such as Job, or for an object where data security is supported by HCM Spreadsheet Data loader, then set the **Upload Data As** value in the Template Security section. This value determines which user name appears in the object's history. You can't set this value for secured objects. This step is optional.
5. In the Template Parameters section, set parameter values to apply to all spreadsheets generated from this template. This step is optional.
6. Click the Design tab to open the Design page.

Design the Template

The Available Attributes panel lists all attributes for the top-level component of the object. Use the **Available Attributes** menu to select a different component. If you selected the **Delete only** value in the **Supported Action** field, then only those components that can be deleted appear.

Follow these steps:

1. Add attributes from the Available Attributes panel to the Selected Attributes panel. When you select any attribute, all attributes that are required to identify the object move automatically to the Selected Attributes panel.
2. Either edit individual attributes, as required, or manage all attributes at once by selecting **Advanced Options for Attribute Definition**.
3. When the template design is complete, click the Add Instructions tab to enter instructions, if required. Otherwise, click the Manage Role Access tab.
4. Reorder the attributes in the Selected Attributes panel to specify the order you require in generated spreadsheets.

Enter Template Instructions

On the Add Instructions page, you can enter instructions to appear on a separate Instructions worksheet in spreadsheets generated from the template. These instructions could describe how loaded objects are to be validated, for example. To enter instructions:

1. Click **Add**.
2. In the Create Instruction dialog box, enter a heading and the instruction text.
3. Repeat steps 1 and 2 to create additional sections of instruction text, as required. You can reorder and delete sections.
4. Click the Manage Role Access tab. The Manage Role Access page opens.

Manage Role Access

On the **Manage Role Access** tab, you specify which roles can:

- Create, save, or upload data sets created in spreadsheets generated from the template.
- View all data sets created in spreadsheets generated from the template.
- Maintain the template.

Note: **Manage Role Access** actions are enabled only if the **Assign Roles to Template** field is set to **Enabled**. **Assign Roles to Template**, which is read only, appears in the **Template Security** section of the **Define** page. To allow the user to assign roles to access any HCM Spreadsheet Data Loader template, including those based on secured business objects that don't implement data security in HSDL, grant the `HRC_ASSIGN_ROLES_TO_ANY_HSDL_TEMPLATE` privilege to your custom role. When maintaining templates, users with this privilege will now see:

- **Assign Roles to Template** on the **Define** page is always **Enabled**
- The **Create** button on the **Role Access** page is always **Enabled**

CAUTION: When data is uploaded using template where the **Upload Data As** is set to **Elevated User**, any data security provided in the application isn't honored by the HSDL spreadsheet. Data is uploaded as the superuser.

Create role mappings for business HCM Spreadsheet Data Loader users to access templates.

Follow these steps:

1. In the **Permitted Data Set Operations** section of the **Manage Role Access** tab, click the **Add** icon.
2. In the **Search and Select Roles** dialog box, search for and select one or more custom roles and click **Add**.

The roles appear in the **Permitted Data Set Operations** section. By default, the selected roles can create and save data sets. You can select the **Upload** and **View All** options for any of the roles, if required.

Tip: The actions that you authorize on the **Manage Role Access** tab are enabled by default. You can deselect the **Enabled** option for any role to disable the role's access temporarily.

3. In the **Template Maintenance** section of the page, click the **Add** icon.
4. In the **Search and Select Roles** dialog box, search for and select one or more custom roles and click **Add**.

The roles appear in the **Template Maintenance** section.

5. Click **Save**.
6. Click the **Review** tab. The **Review** page opens.

Review the Template

On the **Review** page, your template design appears.

1. To validate the template, click **Validate and Save**.
2. To generate a sample spreadsheet, click **Preview**.

Tip: You can click **Preview** on any page to generate a sample spreadsheet. However, the spreadsheet shows only the configuration that you have saved so far.

The default file name is **GenericHdlSpreadsheet.xlsx**.

3. Either save the file locally or open it. When you open it, you're prompted to sign in.
4. Review the spreadsheet.
5. Save the template.

When you're ready to make the template available, edit it and set its status to **Active**.

Only active templates are available on the **Run Spreadsheet Data Loader** page.

Note: You can only save valid active templates. If your template has validation errors, you must resolve them before saving. Alternatively, set the status to **Draft** or **Inactive** and save without validation.

Edit the Template

To edit a spreadsheet template, search for it on the **Spreadsheet Templates** page and click its name in the search results. You can edit an existing spreadsheet template in any status, but you can't change its code or type. In addition, you can't remove attributes from an active template, but you can disable them.

As an example, here's a tutorial to define a bulk-loading spreadsheet to activate and deactivate user accounts: [HCM Spreadsheet Data Loader Template to Activate-Deactivate a User Account](#)

Here's a tutorial to create a HCM Spreadsheet Data Loader template. This tutorial explains the common tasks while creating a template to upload new hires [Creating HCM Spreadsheet Data Loader Templates to Upload New Hires](#)

Here's a tutorial to add preconfigured and custom flexfield segments to your HCM Spreadsheet Data Loader templates [Including Flexfield Segments in HCM Spreadsheet Data Loader Templates](#)

Related Topics

- [HCM Spreadsheet Data Loader Templates](#)
- [Spreadsheet Template Design](#)
- [Set Spreadsheet Template Parameters](#)
- [Spreadsheet Business Object Access](#)
- [How You Associate a User Name with a Data Upload](#)

How You Associate a User Name with a Data Upload

For some object components, you can review the object history to see who last updated the object. This topic explains how to specify which user name appears in the object history when you upload data using HCM Spreadsheet Data Loader.

Setting Template Security

When you create a spreadsheet template, you select a value for the **Upload Data As** field in the Template Security section of the Define page. This value determines what appears in an object's history when data is uploaded from spreadsheets generated from the template.

When you create a spreadsheet template for an object that doesn't use data security in the application, or the application data security is available for spreadsheet loading:

- You can select either **Session User** or **Elevated User** in the **Upload Data As** field. When you select **Session User**, the user name of the signed-in user who uploaded the data appears in the object history. If you select **Elevated User**, then the user name is always **FUSION_APPS_HCM_ESS_LOADER_APPID**. For objects that don't use data security, or where data security is available for spreadsheet bulk data loading, **Upload Data As** is set to **Session User** by default.

When you create a spreadsheet template for an object that uses data security in the application, but that data security isn't available yet for spreadsheet loading:

- **Upload Data As** is set to **Elevated User** and can't be changed. The user name in object history is always **FUSION_APPS_HCM_ESS_LOADER_APPID**.
- It isn't possible to share such templates with other roles. Only users who have an administrator role can upload data from a spreadsheet for objects that use data security in the application where that data security isn't available for spreadsheet data loading.

Here's a tutorial to define a bulk-loading spreadsheet to activate and deactivate user accounts: [HCM Spreadsheet Data Loader Template to Activate-Deactivate a User Account](#)

Related Topics

- [Create and Edit Spreadsheet Templates](#)
- [Spreadsheet Business Object Access](#)
- [How You Assign Access to a Spreadsheet Template](#)
- [How You Enable Administrator Access to HCM Spreadsheet Data Loader](#)

Set Spreadsheet Template Parameters

In a spreadsheet template, you can set parameters that control the processing of all spreadsheets generated from the template. Any values that you specify in a spreadsheet template override equivalent enterprise settings. This topic explains how to set spreadsheet template parameters.

Follow these steps:

1. Open the Define page in the Manage Spreadsheet Templates task flow.
2. In the Template Parameters section, click the **Add** icon.
3. In the Search Template Parameters dialog box, either enter a parameter name or select a category and click **Search**. This table lists the parameters and their categories.

Parameter	Category	Description	Enterprise Default Value
Enable Audit Data	Availability	Specifies whether audit data is captured when audit is configured for business objects loaded from generated spreadsheets.	No
Enable Incremental Load Events	Availability	Enable events to be triggered for the data supplied.	No

Parameter	Category	Description	Enterprise Default Value
		CAUTION: Enabling load events will significantly increase the time it takes for your data to load.	
Enable Automatic Extended Data Set Retention	Scheduling Default	Determines whether data sets for business objects that support validate or rollback are automatically retained beyond the standard retention period.	No
Purge Audit Data	File Definition	Specifies whether audit data is purged for objects successfully loaded or updated from the spreadsheet.	No
Maximum Concurrent Threads for Load	Performance	The maximum number of threads to be used concurrently to load spreadsheet data.	8
Calculate Worker Full-Time Equivalent	Scheduling Default	Specifies whether to calculate automatically the full-time equivalent value of the working hours on a worker assignment. This parameter applies to templates for the Worker object only.	No
Create Worker Default Work Hour Pattern	Scheduling Default	Specifies whether to create default working hour patterns automatically for workers loaded without working hour patterns. This parameter applies to templates for the Worker object only.	No
Date-Effective Update Mode	Scheduling Default	When set to Replace , specifies that the data in spreadsheets replaces any existing data in Oracle HCM Cloud. When set to Retain , specifies that any	Retain

Parameter	Category	Description	Enterprise Default Value
		existing future-dated changes are retained.	
Initiate Business Object Post Processing	Scheduling Default	Specifies whether to run processes that are configured to run automatically after the object is loaded.	Yes
Maximum Percentage of Load Errors	Scheduling Default	The percentage of business object instances in error that can occur for a business object before the validation and load processes stop.	100
Oracle Search Index Update Mode	Scheduling Default	Determines how changes are ingested by the Oracle Search indexes. Caution: Only set this to Real-time ingestion when spreadsheets generated from the template load small volumes that require an incremental update of the Oracle Search indexes.	Bulk ingestion

4. In the search results, select a parameter and click **OK**. The selected parameter appears in the Template Parameters section.
5. Enter or select a value for the parameter, as appropriate.
6. Repeat from step 2 for additional parameters.
7. Either click **Save and Close** or select a different page in the task flow.

You can set parameters to support each template's use case. For example, you can set **Date-Effective Update Mode** to:

- **Replace**, in a spreadsheet template for loading new hires
- **Retain**, in a spreadsheet template for updating worker records

Tip: You can set spreadsheet parameter values for the enterprise on the HCM Spreadsheet Data Loader Parameters tab of the Configure HCM Data Loader page.

If you'd like to import and activate spreadsheet templates from the HCM Resource Center on Cloud Customer Connect, please see this tutorial: [Import and Activate Spreadsheet Templates from the HCM Resource Center on Cloud Customer Connect](#),

Related Topics

- [HCM Data Loader Configuration Parameters](#)

Spreadsheet Template Design

Creating a spreadsheet template is a five-step process. You:

1. Define general information for the template.
2. Design the template's contents and layout.
3. Specify instruction text, if required.
4. Identify the roles that can perform data set operations, such as create and upload, and the roles that can maintain the template
5. Review the design and generate a test spreadsheet from the template.

This topic describes the design step of this process.

Available Attributes Panel

The Available Attributes panel lists all attributes of the selected component of the object. You can:

- Set the **Show** field to filter the attributes.
- Select a different component or all components.
- Switch to a tree view to see the component hierarchy. For each component of the object, you see separate entries in the hierarchy for attributes, flexfield attributes, and child components.
- Click an attribute name to view the attribute details.
- Move attributes from the Available Attributes panel to the Selected Attributes panel. When you move the first attribute, any required attributes for the component move automatically to the Selected Attributes panel. Attributes that are required conditionally don't move. You must add them explicitly to include them.

Note: On occasion, you may see two attributes with the same name in the Available Attributes panel. Typically, this duplication occurs for foreign object references. It enables you to choose between entering free text and selecting a value from an LOV. The icons that precede each name show the attribute type.

Selected Attributes Panel

In the Selected Attributes panel, attributes appear in the order in which you added them. You can:

- Edit the details of an attribute. For example, you can:
 - Change the column heading.
 - Make an optional value required.
 - Specify a default value. You can also make attributes, such as those with default values, read-only.
 - Specify whether the attribute is visible in the generated spreadsheet.
 - Disable optional attributes. These attributes appear but can't be updated.
 - Enter a description to appear as help text in the generated spreadsheet.
 - Configure a list of values to validate the attribute.

- Create **Note** attributes. Use **Note** attributes to return values from a list of values for information only. For example, you may have a list of values for the **PersonNumber** attribute that returns the person number to this attribute. It can also return the person's last and first names in the **Note** column. You can make the **Note** column read-only, and multiple **Note** columns can have values returned from the same list of values. Data in **Note** columns isn't passed to the application but can be used in the data set for information purposes or coordination of multiple contributors.

You can also use **Note** attributes to communicate information that doesn't need to be uploaded. For example, the Human Resources Specialist may have submitted overtime information in a spreadsheet for the Payroll Administrator to action.

- Change the attribute order. After you determine if any attributes are hidden, click the **Reorder** icon to open the Reorder Attributes dialog box.
- Duplicate a component. This option exists for components that can appear more than once in an object, such as Person Phone or Location Legislative Extra Information.

Tip: Edit the selected attributes of the component before creating additional instances.

Note: You can only duplicate components present at the bottom of the object hierarchy with no child components.

When you add an attribute to the Selected Attribute panel, some of the following attributes are automatically added for you:

- Attributes that are used to uniquely identify the record you've selected an attribute from.
- Attributes that uniquely identify the parent record when you select an attribute from a child object.
- Attributes from the parent record if you duplicate the selected attributes for a child object or include attributes from a different child object.

Note: It's important to include attributes from the parent object when supplying data to different child objects, or duplicated child objects. The parent record groups all the related data and ensures the spreadsheet row is loaded together.

Advanced Settings

To manage all attributes at once, click the **Advanced Options for Attribute Definition** icon. On the Advanced Options for Attribute Definition page, you can manage:

- Basic information, flexfield attributes, and lists of values.
- Attribute keys. When you import data to a spreadsheet from a file, you can include a header. The attribute names used in the import file can be different from those in the spreadsheet. In this case, the **Attribute Key** fields must include the names from the import file. For example, you can import a location code to a spreadsheet from a file where it's known as the location ID. To ensure that the data imports successfully, you must set the **Attribute Key** value to **Location ID** for the **Location Code** attribute.

Related Topics

- [Create and Edit Spreadsheet Templates](#)
- [How You Configure Default Values for Spreadsheet Attributes](#)
- [Configure Lists of Values for Attribute Validation](#)

Guidelines for Designing Spreadsheet Templates

Spreadsheet templates are highly configurable. Therefore, you can design them to support specific business use cases. This topic provides some guidelines to help you create spreadsheet templates that minimize the scope for error and are easy to use.

Supported Actions

Depending on the business object, you can use HCM Spreadsheet Data Loader to either create and update or delete object components. You can design a single spreadsheet for both creating and updating an object. However, spreadsheet templates that support a single action are better for users and easier for you to manage. For example:

- In templates for creating objects, you can mark as required all attributes that must be supplied for new objects. Typically, the same attributes aren't required when updating objects, so you can mark different attributes as required in templates for updating objects.

Tip: To mark an attribute as required, edit the attribute in the Selected Attributes panel on the Design page and select **Required**.

- In templates for updating objects, you can configure lists of values to identify records to update.

Lists of Values

For many attributes, you can ensure that users enter only valid data by providing choice lists and searchable lists of values. Some attributes have lists of values by default. In particular:

- Some lookup-validated attributes are automatically configured with choice lists. A list of values is displayed when the user clicks in the cell.
- To reference a foreign object from your template, you can include the Oracle Fusion Surrogate ID attribute. Attributes like this are automatically configured with searchable lists of values, and the selected attribute is returned to the spreadsheet. While a surrogate ID itself may not be meaningful, you can also configure your own lists of values for attributes included in a spreadsheet. These attributes can be validated using a lookup type, value set, or view object.

For example, you can return additional values to other attributes, such as Person Number, instead of just returning the person ID. You can also copy the list of values configuration to another attribute that is typically used to identify a record, such as the Name or Number identifier. You configure the list of values to return the recognizable number.

Tip: Attributes with lists of values have the **List of values** icon before their names in the Available Attributes panel of the spreadsheet template.

Default Values

You can configure default values for attributes in spreadsheet templates. You may want to exclude attributes with default values so that you can enforce their values. Alternatively, you can include the attribute so that users can enter a value, which overrides the default value.

Date Editors

A date editor is provided automatically for all date attributes. However, the user must click in the spreadsheet row to initialize date editors. Therefore, you're recommended not to make a date value the first column in the row. This approach ensures that date editors in the row are initialized before they're used.

Related Topics

- [How You Configure Default Values for Spreadsheet Attributes](#)
- [Configure Lists of Values for Attribute Validation](#)
- [Create and Edit Spreadsheet Templates](#)

How You Configure Default Values for Spreadsheet Attributes

When you design a spreadsheet template for bulk data loading using HCM Spreadsheet Data Loader, you can specify default values for attributes. This topic describes the ways in which you can specify default values.

Specifying Default Values for Date Attributes

This table describes the options for specifying default date values.

Option	Description
Specified date	Using the date editor, select a date.
Load date	Use the date when the spreadsheet row is loaded.
Existing selected attribute	Select another date attribute to supply the default value. The attribute must be in the Selected Attributes section of the Design page.
Use expression	Supply a Groovy expression to derive the default value.

For example, to set the default value of a date attribute to the load date:

1. Select the attribute in the Selected Attributes section of the Design page of Spreadsheet Templates and click **Edit**.
2. In the Edit dialog box, set **Default Value** to **Value** and select **Load date** from the list.

Specifying Default Values for Attributes Other Than Dates

This table describes the options for specifying default values for attributes other than dates.

Option	Description
Constant value	Enter the default value.
Existing selected attribute	Select another attribute of the same data type to supply the default value. The attribute must be in the Selected Attributes section of the Design page.
Use expression	Supply a Groovy expression to derive the default value.

Supplying Groovy Expressions

Using Groovy expressions enables you to manipulate other attribute values to provide a default value. To specify a Groovy expression to derive a default value:

1. Select the attribute in the Selected Attributes section of the Design page and click **Edit**.
2. In the Edit dialog box, set **Default Value** to **Value** and select **Use expression** from the list.
3. Enter the expression in the text area and click **Validate** to ensure the expression is valid.

Hiding Attributes

You may want to hide attributes that have default values so that spreadsheet users aren't aware of them. To hide an attribute, edit it in the Selected Attributes panel on the Design page of the template. In the Edit dialog box, deselect the **Visible** option.

Configure Lists of Values for Attribute Validation

Some types of attributes are configured automatically with choice lists or searchable lists of values in spreadsheets. For such attributes, spreadsheet users can select only valid values, which simplifies data loading.

You can configure lists of values for other attributes to make populating spreadsheets easier. This topic describes how to configure lists of values.

Select the LOV Source for an Attribute

To validate an attribute using a list of values, you must identify the list source. It can be one of:

- Lookup

- Value set
- View object

Select **View object** to include user key values in a spreadsheet in place of a surrogate ID to reference a foreign object. Having user keys helps the spreadsheet user, who sees recognizable values in the spreadsheet. For example, in a spreadsheet to create positions you include a reference to a department. You could just include the **Department ID** attribute in the spreadsheet, as it's automatically configured with a searchable list of values. However, when the user selects a value from the list, the surrogate ID is returned to the spreadsheet. Instead, you could include the **Department** user key attribute in the spreadsheet. Using a view object, you configure the **Department** attribute with a searchable list of values to return the department name to the spreadsheet.

Validate Attributes Using Lookups or Value Sets

Follow these steps:

1. On the Design page of the Manage Spreadsheet Templates task flow, move attributes to the Selected Attributes pane.
2. In the Selected Attributes pane, select the attribute to be validated with a lookup type or value set and click **Edit**.
3. Complete the details on the main tab of the Edit dialog box, as appropriate.

When you're editing the details of an existing list of values, you can change the list display type on the main tab, if required.

4. Click the List of Values tab.
5. Set the **LOV Source** value to **Lookup** or **Value set**, as appropriate.
6. Select the name of the lookup type or value set.
7. Specify whether the LOV is to appear as a choice list or a searchable list of values in generated spreadsheets.

Tip: If the list of values has more than 20 entries, then a searchable list of values is better for the spreadsheet user. Use choice lists only when the number of choices is limited.

8. Select attributes to appear in the LOV.
9. Identify the values to be returned to the spreadsheet from the choice list or search.

If the LOV is a searchable list of values, then you can add filter and search criteria.

- Add filter criteria to enable the user to filter the returned list of attribute values.
- Add search criteria to enable the user to search the returned list of attribute values.

Note: When creating a value set to use as the source of values for an HSDL template column, ensure that the bind variable names don't include spaces.

To add filter criteria, in the Filter Criteria section:

1. Select a value in the **Filter Criteria** field and click **Add**. The **View Criteria** and **Bind Variable** values are completed automatically.
2. Set the **Expression Type** value for each of the view criteria to one of the following values:
 - Constant
 - Expression
 - Attribute

3. If the **Expression Type** is set to **Constant** or **Expression**, then enter the constant or expression in the **Value** field. If the **Expression Type** is set to **Attribute**, then select the attribute in the **Value** field.

To add search criteria, in the Search Criteria section:

1. Select a value in the **Search Criteria** field and click **Add**. The **Search Criteria** and **Attribute** fields are completed automatically.
2. Repeat step 1 for additional search criteria.

Click **OK** to close the Edit dialog box.

Validate Attributes Using View Objects

Follow these steps:

1. On the Design tab of the Manage Spreadsheet Templates task flow, move both the surrogate ID and user key attributes to the Selected Attributes pane.
2. Edit the surrogate ID attribute. On the List of Values tab in the Edit dialog box, the **LOV Source** field shows the full path to the view object. For example, for the **Department ID** attribute you would see the following value:

```
oracle.apps.hcm.organizations.publicView.DepartmentPVO
```

Copy this path and close the Edit dialog box.

3. Edit the primary user-key attribute and set **LOV Source** to **View object**.
4. In the **LOV Source** field, paste the full path to the view object that you copied from the surrogate ID attribute.
5. Leave **Displayed As** set to **Searchable list of values**.
6. Select the attributes to appear in the searchable list.
7. Identify the attribute values to be returned to the spreadsheet from the search. Usually, you include both the surrogate ID and one or more user key values. The user key values are for the spreadsheet user, and the surrogate ID is used to load the data. You can hide the surrogate ID attribute in generated spreadsheets by deselecting the **Visible** option in the Edit dialog box for the attribute.
8. Enter filter and search criteria, as required.
9. Click **OK** to close the Edit dialog box.

Related Topics

- [Spreadsheet Template Design](#)

Autogenerated Attribute Values in Spreadsheet Templates

In spreadsheet templates for the Worker object you can specify that values for some attributes, such as Person Number and Assignment Number, are generated automatically.

For example, on the Design page in a Worker spreadsheet template, you edit the **Person Number** attribute in the Selected Attributes panel and select **Autogenerate**. In this case, users don't have to supply the person number value in generated spreadsheets. However, so that related records can be identified and processed together, a unique identifier must still be supplied. Therefore, **Person Number** must appear in generated spreadsheets and its value must be a unique identifier. This value isn't used as the person number, which is generated on successful creation of the worker record.

To ensure that users supply this unique identifier for a logical object, these rules apply to the template definition:

- If you select the **Autogenerate** option for an attribute, then the **Required** option is selected automatically and made read-only.
- If the **Required** option is selected and the **Default Value** field is set to **None**, then the **Visible** option is selected automatically and made read-only.
- A default column heading is defined. For example, for the **Person Number** attribute the column heading is **Person Unique Identifier**. For the **Assignment Number** attribute, the column heading is **Assignment Unique Identifier**. You can change the default column heading.
- The default description of the attribute explains that a unique identifier must be supplied. You can edit this description.

Spreadsheets generated from the template include the **Person Number** or **Assignment Number** attribute, as appropriate, with its new column heading and description. For the attribute value, users must enter a unique identifier to ensure that records are loaded successfully.

Include Reconciliation and Reporting Information

You can include reference attributes for your spreadsheet rows.

They enable you to record reference data for the spreadsheet row, which could include values already supplied on the spreadsheet row, or source-system information, such as the database-table name, column names, and attribute values from the source system. The **Object Status** page within the **Import and Load Data** interface, displays the reference values supplied against your spreadsheet row. You can create OTBI reports to include these values using the **HCM Integrations Real Time** subject area. The reference names and values are available in the **Imported Objects** sub-folder within the **HCM Data Loader** folder.

For example, when loading element entries, you may want to filter and reconcile by the assignment number, element name and amount. This would then allow you to filter by the assignment number and element name, to check the status of the element entry, while reviewing the amount specified for the element entry.

To add a reference attribute to your spreadsheet template:

1. On the **Data Exchange** work area, select **Spreadsheet Templates**.
2. Search and click the name of the template to edit.
3. On the **Design** Tab, click the **Add Reference Attribute** icon on the **Selected Attributes** panel toolbar.
4. Specify which of the 10 reference attributes you want to use and define a reporting name, which will be passed to HCM Data Loader and a column heading to display in the spreadsheet if the attribute is visible.
5. Optionally configure a default value. For example, you can default the value of the reference attribute to the value of another selected attribute in the template, such as **Assignment Number**, or **Element Name**.
6. Click **OK**.

You aren't restricted to the values you supply as there is no validation of these attributes. The data supplied to them is stored only in the staging tables. It is not saved to the application tables.

Note: While defaulting values for reference attributes from other selected attributes, ensure that the attribute selected for copying has the same value across all records for the logical object created. Examples of a logical object are new hire, element entry, or salary. Reference attributes are stored only against the top-level record in the object hierarchy. Hence, the upload will fail if the value is sourced from an attribute that changes values across those rows. For example, consider that your template maintains phone numbers for your employees and includes multiple phone records such as work, home, or mobile for the same employee. You can safely default a reference attribute from the Person Number attribute, which will be the same for all entries. However, defaulting a reference attribute to the phone number or phone type will cause three different values to be stored, causing the upload to fail. It is only a concern for templates having duplicate attributes for child records or where multiple spreadsheet rows are supplied for the same record, for example, for date-effective history.

Non-reference attributes specific to the object hierarchy are stored in anonymous stage table columns. It's not possible to retrieve the specific attribute information. Reference attribute values are visible on the **Import and Load Data** user interface, available in **HCM Extracts** and **OTBI**. Using the reference attributes, you can easily create reconciliation reports to filter and report on the attribute values required for your use case.

Copy Spreadsheet Templates

Copying a spreadsheet template and editing the copy is an efficient way to create a new spreadsheet template. This topic describes how to copy spreadsheet templates. You perform this task in the Data Exchange work area.

Follow these steps:

1. In the Data Exchange work area, select **Spreadsheet Templates**.
2. On the Spreadsheet Templates page, search for the spreadsheet template to copy.
3. Select the spreadsheet template in the search results and select **Copy** from the template **Actions** menu.
4. In the Copy Template dialog box:
 - a. Enter a unique name for the spreadsheet template in the **Name** field.
 - b. A **Code** value is created automatically based on the name value. You can overwrite the code value.
 - c. Select a legislative data group, if the template is legislation-specific.

Note: You can't change the **Supported Action** value. For example, if **Supported Action** is set to **Delete only** in the source spreadsheet, then copies of that spreadsheet have the same value.

 - d. Enter a description and select a category, if required.
 - e. Click **OK**.
5. Click **OK** to close the Confirmation dialog box.
6. On the Spreadsheet Templates page, search for the new spreadsheet template.
7. In the search results, click the name of the spreadsheet template to open it for editing.

You can copy a spreadsheet template in any status, but the status of the new spreadsheet template is always **Draft**. Change the status to **Active** when the template is ready to use. Active spreadsheet templates are available on the Run Spreadsheet Data Loader page.

Related Topics

- [HCM Spreadsheet Data Loader Templates](#)

Export and Import Spreadsheet Templates

You can import spreadsheet templates to other environments to avoid having to recreate them. For example, you can export spreadsheet templates from a test environment and import them to a production environment.

This topic describes how to export and import spreadsheet templates. You perform these tasks in the Data Exchange work area.

Export Spreadsheet Templates

Follow these steps in the source environment:

1. On the **Data Exchange** work area, select **Spreadsheet Templates**.
2. On the **Spreadsheet Templates** page, search for and select the spreadsheet template that you want to export.
3. From the **Actions** menu for the spreadsheet, select **Export**.
4. In the opening dialog box, select **Save file** and click **OK**.
5. In the file-name dialog box, select a location for the file and click **Save**.

The file is exported to the specified location in XML format. Deliver the file to the target environment.

Import Spreadsheet Templates

Follow these steps in the target environment:

1. On the **Data Exchange** work area, select **Spreadsheet Templates**.
2. On the **Spreadsheet Templates** page, click **Import**.
3. In the **Import** dialog box:
 - a. Enter a unique name for the spreadsheet template. This name applies in the target environment.
 - b. The **Code** field is completed automatically based on the name value. You can overwrite the value.
 - c. If your template is legislation-specific, then select the legislative data group.
 - d. Click **Choose File** to select the template file.
 - e. Enter a description and select a category, if required.
 - f. Click **OK** in the Import dialog box to import the template.

Click **OK** to close the Confirmation dialog box.

4. On the Spreadsheet Templates page, search for the imported template.

The status of an imported spreadsheet template is always **Draft**. Change the status to **Active** when the template is ready for use.

Here's a tutorial to define a bulk-loading spreadsheet to create multiple child records on a single spreadsheet row
Defining HSDL Templates to Create Multiple Child Records in a Row

Related Topics

- [HCM Spreadsheet Data Loader Templates](#)

How You Enable Business Users with Custom Roles to Load Spreadsheet Data

Business users can load data for objects such as jobs and locations from HCM Spreadsheet Data Loader spreadsheets in their work areas. These business users can generate spreadsheets and load data from them because they have predefined roles that provide the access they need.

This topic explains how to give the same access to business users with custom roles.

For example, let's say that your human resource specialists create job definitions in bulk. They load them from a spreadsheet on the Manage Jobs page in the Workforce Structures work area. They can generate the spreadsheet and load the data because:

- They have the predefined Human Resource Specialist job role. This job role inherits the Workforce Structures Management duty role, which inherits the Load Data Using HCM Spreadsheet Data Loader aggregate privilege.
- The Job (ORA_PER_JOB) spreadsheet template behind the Manage Jobs page lists the predefined Human Resource Specialist job role on its Manage Role Access page. Users with the role can create, save, and upload data.

Human resource specialists who use a custom role can't load data from a spreadsheet by default. To let them load data, you:

- Add the Load Data Using HCM Spreadsheet Data Loader aggregate privilege to their custom role. You need a template administration role to update custom roles.
- List the custom role on the Manage Role Access page in the predefined spreadsheet template for the object. You need a template administration role to update predefined templates.

You can make these changes for custom versions of any predefined role that can load data from HCM Spreadsheet Data Loader spreadsheets.

Related Topics

- [Add a Role to a Predefined Spreadsheet Template](#)
- [How You Enable Administrator Access to HCM Spreadsheet Data Loader](#)
- [Edit Job Role and Abstract Role](#)

Add a Role to a Predefined Spreadsheet Template

Your business users may need to load data in bulk from HCM Spreadsheet Data Loader spreadsheets in their work areas, or from the **Run Spreadsheet Data Loader** task. You need to assign which roles can generate and use the spreadsheets.

This topic describes how to add roles to predefined spreadsheet templates.

Suppose you want users with the **Human Resource Specialist** role to load grades on the Manage Grades page in the Workforce Structures work area. Follow these steps:

1. Sign in with a HCM Spreadsheet Data Loader template administrator job role or privileges.

2. Select **Navigator > My Client Groups > Data Exchange**
3. In the **Data Exchange** work area, select **Spreadsheet Templates**.
4. On the Templates page, search for the **Grade** (ORA_PER_GRADE) spreadsheet template.

Tip: The codes of predefined spreadsheet templates begin with the characters ORA_.

5. Click the template name in the search results to open the spreadsheet template for editing.
6. Click the Manage Role Access tab in the spreadsheet template.
7. Click the **Add** icon in the Permitted Data Set Operations section of the Manage Role Access page.
8. Select the **Human Resource Specialist** role in the **Search and Select Roles** dialog box and click **Add**.

Your role appears in the Permitted Data Set Operations section. The **Create** and **Save** options are selected automatically. They mean that users with this role can create data sets in spreadsheets generated from the template and save them.

9. Select **Upload** to let users upload their data. Select **View All** to let users view spreadsheets generated by other users from the template.
10. Click **Save** then **Done**.

Tip: You can't change the attributes in a predefined spreadsheet template, but you can set the template parameters and manage role access.

Note:

- You may need to define a custom role to access spreadsheet templates if your preconfigured role doesn't grant access to the **Load Data using HCM Spreadsheet Data Loader** (ORA_HRC_LOAD_DATA_USING_HSDL) function security privilege.
- To allow the user to assign roles to access any HCM Spreadsheet Data Loader template, including those based on secured business objects that don't implement data security in HSDL, grant the **HRC_ASSIGN_ROLES_TO_ANY_HSDL_TEMPLATE** function security privilege to your custom role.

CAUTION: When data is uploaded using such a template, any data security provided in the application isn't honored by the HSDL spreadsheet. Data is uploaded as the superuser.

Related Topics

- [How You Enable Business Users with Custom Roles to Load Spreadsheet Data](#)

How You Assign Access to a Spreadsheet Template

When you create a spreadsheet template it isn't available immediately for use. This topic explains how to associate which roles can use the template and the data set access that each role has.

Assigning Role Access

You can assign job roles to a template using the **Spreadsheet Templates** task:

- Navigate to the **Role Access** tab of the spreadsheet template designer.
- Click **Add**.
- Search and select the job role that needs access to the template.
- Use the following checkboxes to configure the data set access for the role.
 - **Create** : Create data sets for data entry within the spreadsheet.
 - **Save**: Save data set data to the stage tables.
 - **Upload**: Initiate the upload of data sets into the application tables.
 - **Roll Back** : Initiate the roll back of data sets. This checkbox is available only if the roll back is supported by the object.
 - **View All**: View all data sets and their content, including the ones created by other users.
- Repeat for any other roles that require access to the template.
- For the spreadsheet to be available to users with the assigned roles, you must set the template's **Status** as **Active** on the **Define** tab.

Note: When you create a spreadsheet template, the **Template Security** section on the **Define** tab indicates whether roles can be assigned to the template or not. To allow the user to assign roles to access any HCM Spreadsheet Data Loader template, including those based on secured business objects that don't implement data security in HSDL, grant the `HRC_ASSIGN_ROLES_TO_ANY_HSDL_TEMPLATE` function security privilege to your custom role.

Enabling Audit of Role Configuration

You can enable audit of Spreadsheet Template Roles under HCM Spreadsheet Data Loader using **Manage Audit Policies** and review changes using the existing **Audit Reports** task found in the **Tools** area.

If you'd like to import and activate spreadsheet templates from the HCM Resource Center on Cloud Customer Connect, please see this tutorial: [Import and Activate Spreadsheet Templates from the HCM Resource Center on Cloud Customer Connect](#),

Related Topics

- [How You Enable Audit of Role Configuration for a Spreadsheet Template](#)

FAQs for HCM Spreadsheet Data Loader Templates

Can I hide attributes in spreadsheet templates?

Yes. In the Selected Attributes panel on the Design tab of the template, edit the attribute. In the Edit dialog box, deselect the Visible option.

How can I manage role access for some objects?

Some objects provide data security validation within the application but that isn't available when bulk loading with HSDL.

Templates based on these objects restrict roles from being assigned to them.

To allow the user to assign roles to access any HCM Spreadsheet Data Loader template, including those based on secured business objects that don't implement data security in HSDL, grant the `HRC_ASSIGN_ROLES_TO_ANY_HSDL_TEMPLATE` privilege to your custom role. When maintaining templates, users with this privilege will now see:

- **Assign Roles to Template** on the **Define** page is always **Enabled**
- The **Create** button on the **Role Access** page is always **Enabled**

CAUTION: Data loaded with any template where **Upload Data As** is set to **Elevated User** will not evaluate data security. Records will be seen as being created and updated by the elevated user.

You can create role mappings for business HCM Spreadsheet Data Loader users to access templates.

Can I load data from a spreadsheet as the session user?

Yes. In the spreadsheet template from which you generate the spreadsheet, **Upload Data As** must be set to **Session User**. This setting is for unsecured objects and those secured objects where data security is implemented for spreadsheet data loading.

Why can't I save my template changes?

Once a template has an Active status, the template must be valid before you save the changes. You'll only see the **Validate and Save** option to save.

To save your changes without validating the template, you can change the status of your template to Inactive.

Since active templates are available for use to other users, they can't be saved without being validated first.

Why can't I delete an existing spreadsheet template?

You can only delete a template if it has a draft status and has no data sets associated with it.

12 Importing and Loading Data Using HCM Spreadsheet Data Loader

Generate Spreadsheets

To load data using HCM Spreadsheet Data Loader, you must first generate a spreadsheet from a spreadsheet template. This topic describes how to generate a spreadsheet and save it locally.



You perform this task in the **Data Exchange** work area, or directly in the page that maintains the object, such as **Jobs**.

Follow these steps:

1. In the Data Exchange work area, select **Run Spreadsheet Data Loader**.
2. On the Run Spreadsheet Data Loader page, the saved search runs automatically. If the template that you want to use doesn't appear, then click **Show Filters**, enter search terms for the template, and click **Search**.

Tip: In the search results, click the **Template Overview** icon for a template. The Template Overview dialog box provides a list of the attributes that the template supports.

3. Click the name of the template that you want to use.
4. In the Opening GenericHdlSpreadsheet.xlsx dialog box, you will be prompted to log in. You should log in before saving the file, otherwise the **Save File** won't have the columns generated for the template shape.
5. In the file-name dialog box, select a location and save the spreadsheet.

You can now enter data in the spreadsheet and upload it.

Here's a tutorial to shape and import files using a HCM Spreadsheet Data Loader template: [Import Files into HCM Spreadsheet Data Loader Spreadsheets](#)

Related Topics

- [HCM Spreadsheet Data Loader Templates](#)

Import Data to a Spreadsheet

To populate a spreadsheet with data for upload, you can either enter the data manually or import it from a file. This topic describes how to import data to a spreadsheet from a file.

Follow these steps:

1. If the spreadsheet isn't already open, then open it and sign in when prompted.

2. You import data to a data set. You can either create the data set or use an existing data set.
 - o To create a data set, click **Create Data Set** on the Spreadsheet Loader toolbar. A name is assigned automatically to this new data set. You can overwrite the data set name.
 - o To use an existing data set, select its name in the Search section of the spreadsheet and select a **Search Mode** value. Click **Search** in the Spreadsheet Loader toolbar.
3. On the Spreadsheet Loader toolbar, click **Import File**. The Import from File dialog box opens.
4. In the Import from File dialog box, select one of **File Type** values shown in this table.

File Type	Description
Comma separated values	Comma-separated (.csv) file
Pipe separated values	Delimited data (.dat) file
Other file type	Text file (.txt) with a user-specified separator
XML file type	XML file

Tip: You can download CSV and XML file templates from a spreadsheet template. Use them to load data to a spreadsheet generated from the same template.

5. For file types other than **XML file type**, indicate whether the file contains a header.

The header provides the names of the attributes for which the file contains values. In this case, the attribute order in the file doesn't have to match the spreadsheet column order. If you deselect **Header included**, then the order of the values in the file must match the order of the columns in the spreadsheet.

In an XML file, the XML tags used for the attribute values must use the attribute keys configured in the spreadsheet template.
6. Click **Browse** to select the file to import.
7. For file types other than **XML file type**, expand the Advanced File Options section. Complete the options as shown in this table for the selected file type. You can use the default values where supplied.

File Type	File Separator	Escape Indicator	Newline Indicator
Comma separated values	Not applicable	1 - 10 characters	1 - 10 characters
Pipe separated values	Not applicable	1 - 10 characters	1 - 10 characters
Other file type	1 - 10 characters	1 - 10 characters	1 - 10 characters

The **Newline Command** field shows the newline command constructed from the supplied **Escape Indicator** and **Newline Indicator** values.

8. Click **OK** to import the data from the selected file. After the data is imported, you can edit any of the imported attribute values.

Note: Some validation of the imported data occurs. Any errors appear automatically in the **Progress** and **Message** columns of the **Spreadsheet Line Status** section of the spreadsheet. Correct any errors before you attempt to upload the data.

Here's a tutorial to shape and import files using a HCM Spreadsheet Data Loader template: [Import Files into HCM Spreadsheet Data Loader Spreadsheets](#)

Manage Spreadsheet Data Sets

A data set is all data loaded from an instance of the same spreadsheet and identified by the same data-set name. You can load the data from a spreadsheet all at once.



Watch video

Alternatively, you can add data rows to an existing spreadsheet and upload them to the existing data set. This topic describes how to create and manage spreadsheet data sets.

Create a Data Set

Follow these steps:

1. Open the generated spreadsheet.
2. In the Connect dialog box, click **Yes** and sign in when prompted. After a few seconds, the spreadsheet shows all attributes from the selected spreadsheet template.
3. Click the Spreadsheet Loader tab in the spreadsheet toolbar.
4. Click **Create Data Set** and click **OK** to close the dialog box. A data set name is generated automatically, based on the template name, date, and time, but you can overwrite it. The name must be unique for the spreadsheet template.
5. Click **OK** to close the confirmation dialog box.

You can either enter data manually in the spreadsheet or import it from a file.

Update a Data Set

To retrieve an existing data set, follow these steps:

1. Either open the original spreadsheet or generate a new spreadsheet from the same template as the original spreadsheet.
2. In the **Data Set Name** field under the Search header, select the name of the data set that you want to update. Available data sets:
 - Were uploaded from a spreadsheet that was generated from the same spreadsheet template as the current spreadsheet.
 - Still exist in the stage tables. You can't update a data set after you have deleted it from the stage tables.
3. Select a **Search Mode** value. You can retrieve either all records in the data set or only those that failed to load.

4. In the Spreadsheet Loader toolbar, click **Search**. The specified data-set records are downloaded to the spreadsheet.

You can correct failed rows and add new rows, but you can't update any row that loaded successfully. Rows that you add belong to the data set that you're updating. When you upload the data, only new and updated rows are imported to the stage tables.

Save a Data Set

You can import a data set periodically to the HCM Data Loader stage tables by clicking **Save** on the Spreadsheet Loader toolbar. The data isn't loaded to the application tables until you click **Upload**. The **Save** action lets you:

- Import data to the stage tables over a period of time. For example, you can enter and save payroll adjustments throughout the week but upload them only at the end of the week.
- Create large volumes of data that you save regularly to a single data set. You don't have to manage those volumes all at once in the spreadsheet.
- Allow multiple users to supply data for processing in the same data set.

Related Topics

- [Import Data to a Spreadsheet](#)
- [Maintain the HCM Spreadsheet Data Loader Stage Tables](#)

Upload Data Using HCM Spreadsheet Data Loader

In this example, you learn how to upload jobs for the Sales Department using HCM Spreadsheet Data Loader.

This table summarizes key decisions for this scenario.

Decisions to Consider	In This Example
Which business object am I loading?	Job
Which spreadsheet template am I using?	Sales Jobs
How do I populate the spreadsheet?	Import the file Sales_Jobs.csv , which is a comma-separated file that contains a header and uses the \n newline command.
Will I create a data set or import the file to an existing data set?	Create a data set.

Summary of the Tasks

Upload the jobs by:

1. Generating the spreadsheet

2. Importing Job objects to the spreadsheet
3. Uploading jobs and correcting errors

Prerequisites

Before you can complete this task:

1. You must have installed the desktop client Oracle ADF 11g Desktop Integration Add-in for Excel.
2. You must have enabled the Trust Center setting **Trust access to the VBA project object model** in Microsoft Excel.
3. The Sales Jobs spreadsheet template must exist and be active.
4. The **Sales_Jobs.csv** file must exist.

Generating the Spreadsheet

1. In the **Data Exchange** work area, select **Run Spreadsheet Data Loader**.
2. On the **Run Spreadsheet Data Loader** page, click **Show Filters**.
3. Enter **Sales Jobs** in the **Name** field and click **Search**.
4. In the search results, click the name of the **Sales Jobs** template.
5. In the spreadsheet dialog box, select **Save File** and click **OK**.
6. In the file-name dialog box, select a location. Enter **SalesJobs.xlsx** in the **File name** field and click **Save**.

Importing Job Objects to the Spreadsheet

1. Open the **SalesJobs.xlsx** spreadsheet and sign in when prompted.
2. On the Spreadsheet Loader toolbar, click **Create Data Set**. Click **OK** to close the warning message, and click **OK** to close the confirmation message.
The data set is created with a generated name.
3. On the Spreadsheet Loader toolbar, click **Import File**.
4. Complete the fields of the Import from File dialog box as shown in this table.

Field	Value
File Type	Comma separated values
Header Included	Yes
File	Sales_Jobs.csv
Escape Indicator	\
Newline Indicator	n

5. Click **OK**.
6. Correct any Import File Errors reported in the **Progress** and **Message** columns of the Spreadsheet Line Status section of the spreadsheet.

Uploading Jobs and Correcting Errors

1. On the Spreadsheet Loader toolbar, click **Upload**.
Data is imported to the stage tables. Objects that import successfully are loaded automatically to the application tables.
2. Click **Refresh**.
Errors from both import and load are reported automatically in the **Progress** and **Message** columns of the Spreadsheet Line Status section of the spreadsheet.
3. Correct any errors in the spreadsheet.
4. Repeat from step 1 of Uploading Jobs and Correcting Errors until all rows are imported and loaded successfully.
5. Close the spreadsheet.

Related Topics

- [Set Up Desktop Integration for Excel](#)

Roll back Data using HCM Spreadsheet Data Loader

For objects that support being rolled back, you can roll back successfully loaded data for a data set using HCM Spreadsheet Data Loader.

Scenario

Follow these steps:

1. Generate a spreadsheet from the template used to create the data set you want to roll back.
2. Select the data set name to roll back from the list of data sets created by the spreadsheet.
3. Click the Spreadsheet Loader tab in the spreadsheet toolbar.
4. Click Search.
5. Click More > Roll Back.
6. Refresh the data set to monitor the status and any errors raised by roll back.

Related Topics

- [Sources of Business-Object Information](#)

How to Set a Value to Null

To explicitly set an attribute value to null, the value you must supply is determined by the data type of the value.

Data Type	Value to Supply
String/Text	#NULL

Data Type	Value to Supply
Number	-999999991
Date	4712/01/01
Time Stamp	4712/01/01 23:59:59

Review Existing Data Sets

The **Recent Spreadsheet Loads** task allows you to review the details of data sets recently processed using the HCM Spreadsheet Data Loader.

This mobile responsive user interface displays the following summary information:

- The data set name
- The template name
- The data set status
- The business object
- The number of failed spreadsheet rows
- The percentage of rows that loaded successfully

You can expand a data set to display:

- The total number of spreadsheet rows
- The number of spreadsheet rows that were uploaded
- The number of objects that were uploaded
- Who created the data set
- The last updated date of the data set

By default, only data sets last updated by the session user in the last 7 days are displayed. Click the **Show Filters** link to change the filter criteria.

Click on the spreadsheet icon to generate a spreadsheet and download the data set you're reviewing.

Note: You can only download those data sets for which you have the access to view.

Related Topics

- [How You Assign Access to a Spreadsheet Template](#)

How You Delete Data Using HCM Spreadsheet Data Loader

You can delete business objects using HCM Spreadsheet Data Loader, regardless of how they were created. This topic describes how to delete data using HCM Spreadsheet Data Loader.

Designing the Spreadsheet Template

To delete data, you create a spreadsheet template and set the **Supported Action** value to **Delete only**. You can use spreadsheets generated from that template to delete specified objects. The **Delete only** option for the **Supported Action** field is available only for objects with at least one component that can be deleted. The object components that appear on the Design tab of the Spreadsheet Templates page are those that you can delete. For example, for the Job object you see the Job Evaluation, Job Valid Grade, Job Extra Information, and Job Legislative Extra Information components. You don't see the parent Job component, because you can't delete it.

When you move an attribute to the Selected Attributes panel of the Design page, all attributes required for deleting the component move across simultaneously.

Copying Spreadsheet Templates

When you copy a spreadsheet template, you can't change the supported action. Therefore, if the source template has **Delete only** as its supported action, then the copy has the same supported action.

Uploading the Spreadsheet

To delete object components using a spreadsheet, you enter attribute values to identify the components and upload the spreadsheet. When you delete a parent component, its child components are deleted automatically.

Tip: You can monitor progress on the Import and Load Data page. Search for data sets by file type to list those loaded from spreadsheets.

FAQs for Importing and Loading Data Using HCM Spreadsheet Data Loader

How do I fix HCM Spreadsheet Data Loader errors?

You fix errors in the spreadsheet from which you uploaded the data. Errors are reported in the Progress and Message columns of the Spreadsheet Line Status section of the spreadsheet when you click Refresh. After correcting any errors, click Upload again.

Can I load date-effective data?

Yes. You can create, update, and correct date-effective objects. To load date-effective history, you provide a separate row for each date range.

Can I change first and last effective dates?

Yes. Include the Replace First Effective Start Date and Replace Last Effective End Date attributes in the spreadsheet template, as appropriate. Spreadsheet users set these attributes to Y and provide revised dates on the Effective Start Date and Effective End Date attributes.

For some objects, you can change only one of these dates.

Can I load translated objects?

Yes. You don't have to create a separate spreadsheet template for the translation object. Instead, include translation attributes, such as Language, in the spreadsheet template for the base object. You can use spreadsheets generated from the template to maintain both base-object and translation data.

13 Postload Processing

Processes to Run After Loading Data

After a successful data migration or incremental update, you must run processes to complete the setup of some business objects. This topic describes these processes.

Processes to Run After Loading Worker

This table identifies processes to run after loading Worker objects. The table also suggests the order in which to run the processes when you're running them all manually.

Process	Order
Synchronize Person Records	1
Refresh Manager Hierarchy	2
Update Person Search Keywords	3
Optimize Person Search Keywords Index	4
Autoprovision Roles for All Users	5
Send Pending LDAP Requests	6
Send Personal Data for Multiple Users to LDAP	7
Apply Name Formats to Person Names	Not applicable

Also, there are two required processes that are always run automatically that can't be disabled:

- EMEA Localization Postprocessing
- Global Tax Reporting Unit Maintenance

By default, these two processes also run automatically when a load of Worker objects completes:

- Refresh Manager Hierarchy
- Update Person Search Keywords

You can prevent either or both of these processes from running automatically using a SET instruction in the Worker.dat file. Or, you can prevent the processes from running for the enterprise by setting configuration parameters.

Run these post-processes in the Scheduled Processes work area.

Select **Navigator > Tools > Scheduled Processes**.

Process Descriptions

Apply Name Formats to Person Names

Applies the Display Name, List Name, Full Name, or Order Name format to denormalized person-name data when you add or update a name format. Optionally, you can run the process for a single country.

Autoprovision Roles for All Users

Generates requests to provision and remove roles based on current role-provisioning rules.

Optimize Person Search Keywords Index

Optimizes the index of the PER_KEYWORDS table to improve search performance. By default, the process runs for a maximum of 180 minutes and performs a full optimization. You can enter a different **Maximum Optimization Time**. If you select the **Rebuild the index** option, then the **Maximum Optimization Time** value is ignored. Rebuilding the index can be time consuming.

Run this process whenever you run **Update Person Search Keywords**. Run it after that process completes.

Refresh Manager Hierarchy

For performance reasons, the complete manager hierarchy for each person is extracted from active data tables. The hierarchy is stored in a separate manager hierarchy table, known as the denormalized manager hierarchy (PER_MANAGER_HRCHY_DN). This approach ensures that a person's manager hierarchy is both easily accessible and up to date. The **Refresh Manager Hierarchy** process populates the denormalized manager hierarchy table with latest information after each data load.

If this process isn't enabled to run automatically, then run it manually after all data loads.

Send Pending LDAP Requests

Sends user-account and role-provisioning requests to your LDAP directory.

Send Personal Data for Multiple Users to LDAP

Ensures that personal data held in your LDAP directory matches that held by Oracle HCM Cloud.

Synchronize Person Records

Notifies consuming Oracle Cloud applications, such as Oracle Fusion Trading Community Model, of changes to person and assignment details since the last data load.

For a once-only load, submit this process with the **After Batch Load** parameter set to **Yes**. In the **From Date** field, enter the start date of the load. In the **To Date** field, enter the end date of the load. For ongoing updates with daily changes to person records, schedule the process without specifying start and end dates.

Update Person Search Keywords

Several attributes of person, employment, and profile records are used as person-search keywords. This process copies keyword values in all installed languages from the originating records to the PER_KEYWORDS table, where they're indexed to improve search performance.

Select a person name in the **Name** field to run the process for a single worker. If you leave the field blank, then the process runs for all workers, which can be time consuming. To avoid performance problems, submit the process with the

After Batch Load parameter set to **Yes**. In this case, the process creates keywords for new records but doesn't update those for existing records. Leave the **Batch ID** field blank.

Automated Post-Processing for Other Objects

Other objects automate the submission of post processing. See Business Object Post Processes topic for further information.

Actions to Perform After Loading Trees

After loading department or organization tree nodes, you must activate your tree version. For more information, see the Implementing Global Human Resources guide.

Process Guidelines

Send Pending LDAP Requests:

You must always run **Send Pending LDAP Requests** after loading worker records. When you run **Send Pending LDAP Requests**:

- User accounts are created for new workers.
- Roles are provisioned to user accounts as specified by current role mappings.

If appropriate role mappings don't exist when you load new workers, then user accounts are created but no roles are provisioned. User accounts without roles are automatically suspended when **Send Pending LDAP Requests** completes. Also, subsequent role-provisioning requests to the suspended user accounts may be slow to process. To avoid this scenario:

1. Create a role mapping to provision at least one role, for example, Employee, to every current worker before you load worker records.
2. Prevent the creation of user accounts for terminated workers. Set the **User Account Creation for Terminated Workers** option to **No** using the following in the Setup and Maintenance work area:
 - Functional Area: Workforce Structures
 - Task: Manage Enterprise HCM Information
3. Load worker records.

Note: Always create a role mapping for the workers you're loading before you load them. Having this role-provisioning rule in place prevents new user accounts from being suspended unnecessarily.

Autoprovision Roles for All Users:

You don't have to run this process after the initial or only load of worker records if you create role mappings to autoprovision roles before you load workers. However, if you load the workers before creating the role mapping, then you must run this process followed by **Send Pending LDAP Requests** for the loaded workers.

You don't have to run this process after an incremental update. However, you're recommended to run this process whenever a change is made to the autoprovisioning rules.

Send Personal Data for Multiple Users to LDAP:

Run this process after the initial or only load of worker records. Run it after an incremental load:

- For changed users only

- If you're loading workers separately from their managers
- If updates were made to personal data by means of SQL scripts

Related Topics

- [Why You Should Run the Send Pending LDAP Requests Process](#)
- [Autoprovisioning](#)
- [The SET Instruction](#)

How You Validate Loaded Data

Once HCM Data Loader loads your data successfully, you're recommended to validate that the loaded objects appear as expected. This topic identifies some of the ways in which you can validate loaded data. These include use of:

- Manual testing
- Oracle Transactional Business Intelligence (OTBI) reports
- HCM Extracts

Using Manual Testing

You can:

- Validate data interactively by searching for a representative sample. For example, you can search for workers in the Directory or Manage Person work area. If a worker is also a manager, then you can explore the manager hierarchy.
- Validate work structures by searching for them. For example, you can search for jobs on the Manage Job page and locations on the Manage Locations page. Alternatively, you can test that they're available for selection. For example, you can update a worker's assignment to select a different job or location.

Using Oracle Transactional Business Intelligence Reports

You can use predefined OTBI reports to validate some uploaded data. To run these reports:

1. Open the Reports and Analytics work area.
2. In the Contents pane of the work area, open the folder **Shared Folders > Human Capital Management**. Many subfolders of the **Human Capital Management** folder contain a **Transactional Analysis Samples** folder, which contains the predefined reports.

To list the predefined reports for an area, select the relevant **Transactional Analysis Samples** folder in the Contents pane and click the **Browse Catalog** icon. A list of predefined reports for the selected area, such as Workforce Management, appears in the work area. For example, you can run the Job Details report to list active jobs and the Location Details report to list active locations.

You can also design your own reports using Oracle BI Publisher. To open Oracle BI Publisher, select **Create > Report** in the Contents pane of the Reports and Analytics work area.

Using HCM Extracts

You can use HCM Extracts to extract data from Oracle HCM Cloud and deliver it in various formats. These formats include CSV, XML, Microsoft Excel, HTML, RTF, and PDF.

How You Review Automatically Submitted Post Processes

After a successful data migration or incremental update, post processes are configured to run and complete the setup of some business objects. This topic explains how to monitor and review the statuses of the business object post processes.

Monitor and review the status of the business object post processes

After you submit a data set through the **Import and Load Data** page, select the data set and click **Review Processes**.

On the **Processes** page:

- For each of the business object, the corresponding post processes, configured on the **Business Object Post Processes** page, is listed as an individual line item under **Manage HCM Data Loader Post Processes** entry.
- You can view the processing status, submission time, finish time, elapsed time, and access the log file for the post process.
- If a business object has more than one configured post processes and even if one of them has ended in an error, then the post processing status for the business object as a whole is Error.

The overall post process processing status for an import is indicated by the **Status** column in the **Data Sets** and **Business Objects** tables on the **Import and Load Data** page.

The table here lists the various post processes status and its meaning:

Post Process Status for the Business Object	Meaning
In Progress	If any of the post processes for a business object is in progress.
Error	If all the post processes for a business object have completed and one or more of those ended in error.
Success	If all the post processes for a business object have completed successfully.
Not Applicable	If there are no post processes for the business object.

Name	Business Object	Process ID	Status	Submission Time	Finish Time	Elapsed Time	Log
▶ Import and Load Data		76957	Succeeded	09/27/22 12:13 PM	09/27/22 12:16 PM	00:02:43	
▲ Manage HCM Data Loader Post Processes	Worker	76960	Succeeded	09/27/22 12:13 PM	09/27/22 12:15 PM	00:02:32	
▶ Refresh Manager Hierarchy	Worker	76962	Succeeded	09/27/22 12:13 PM	09/27/22 12:14 PM	00:00:45	
▶ Update Person Search Keywords	Worker	76970	Succeeded	09/27/22 12:14 PM	09/27/22 12:14 PM	00:00:22	
▶ Worker HDL Localization Post Processing	Worker	76985	Succeeded	09/27/22 12:14 PM	09/27/22 12:15 PM	00:01:00	

Columns Hidden 1

Monitor and review the status of the data set post processes

After you submit a data set through the **Import and Load Data** page, select the data set file and click **Review Processes**.

On the **Processes** page:

- The post processes configured on the **Data Set Post Processes** page is listed as an individual line item. For example: You can find the **Oracle Search Bulk Ingestion** post process under **Update Oracle Search Indexes with HCM Data Loader Changes** entry.
- You can view the processing status, submission time, finish time, elapsed time, and access the log file for the post process.
- Click **Refresh** to see updated status of the data set post process.

<

Processes: JobFamily.zip

View ▾

Export

Refresh

Detach

	Name	Business Object	Process ID	St
	▲ Import and Load Data		409372	SU
	▲ Import and Load Business Object	Job Family	409373	SU
	Transfer Business Object	Job Family	409374	SU
	Import Business Object	Job Family	409375	SU
	▲ Load Business Object	Job Family	409377	SU
	Load Business Object: Subprocess	Job Family	409378	SU
	▲ Update Oracle Search Indexes with HCM Data Loader Changes		409379	SU
	Oracle Search Bulk Ingestion		409380	SU

Related Topics

- [Data Set Post Processes](#)
- [Business Object Post Processes](#)
- [How You Manage Business Object Post Processes](#)

Business Object Post Processes

Enabling Business Object Post Processes

This functionality is enabled by default and controlled by the **Enable HCM Data Loader Post Process Configuration** parameter, with the default value set to **Yes** in the **Configure HCM Data Loader** task, available from **Setup and Maintenance**.

Configuring Automatic Submission of Post Processes for Business Objects

The **Business Object Post Processes** tab on the **Configure HCM Data Loader** page is used to view and configure automatic submission of all the post processes for the business objects.

In the **Setup and Maintenance** work area, select **Configure HCM Data Loader** task in the **HCM Data Loader** functional area. On the **Configure HCM Data Loader** page, select the **Business Object Post Processes** tab.

- You can review the business object post processes available for automated submission by HCM Data Loader.
- For supporting post processes, you can enable or disable the automatic submission for all the data loads for a business object.

Configure HCM Data Loader ?

←

HCM Data L...

HCM Spread...

Business O...

Business Object Post Processes

Business Object

View

<div>Worker</div>	<div></div>	<div></div>	<div></div>	<div></div>	<div></div>
Business Object	Process Order	Process	Default	Override	Description
Worker	1	Refresh Man...	Yes	<div></div>	Re-creates the st...
Worker	2	Update Pers...	Yes	<div></div>	Updates the pers...
Worker	3	EMEA Locali...	Yes	<div></div>	Payroll postproce...
Worker	4	Global Tax R...	Yes	<div></div>	Copies TRU inform...

Process

Process Name

Refresh Manag

Description

Re-creates the

Task Name

RefreshManag

Parameters

Parameter Name	Default	Override	Descrip
Updated Within the Last N Days	1	<div></div>	Number

This table explains the set-up for the business object post processes on this page:

Name	Description
Business Object	Name of the business object.
Process Order	Preset order of processing of the business object post process.

Name	Description
	Note: For multiple post processes for a business object: <ul style="list-style-type: none">• If the processing order is the same for one or more post processes, then they'll be processed concurrently.• If the processing order is incremental, then they'll be processed sequentially.
Process	Name of the business object post process.
Default	Default running state of the business object post process.
Override	If enabled, reset the default running state of the business object post process. Values are Yes and No.
Description	Describes the function of the business object post process.

This table explains the **Process** section on this page:

Name	Description
Process Name	Name of the business object post process.
Description	Describes the function of the business object post process.
Task Name	Name of the task for the business object post process. Use this to disable the process for a specific file.

This table explains the set-up for the business object post processes parameters on this page:

Name	Description
Parameter Name	Name of the parameter of the business object post process.
Default	Default value of the parameter.
Override	If enabled, reset the default value of the parameter.
Description	Describes the parameter.

To manage the automatic submission of post process for a business object, in the **Business Object Post Processes** page:

1. Filter by the desired **Business Object**.

2. Select the post process and review the value under the **Default** column.
 - The value: **Yes**, indicates that the post process for the business object will be automatically submitted whenever HCM Data Loader or HCM Spreadsheet Data Loader successfully loads data for it.
 - The value: **No**, indicates that the post process for the business object will never be submitted.
 - If the **Override** column is enabled for the selected post process, you can overwrite the default value by resetting it with the desired option.
 - The parameters for the post process appear below the post process list for the business object.
 - If the **Override** column is enabled for the parameters, you can overwrite the default value and redefine it with the desired value.

Disabling Business Object Post Processes

To disable the capability of configuring the automatic submission of business object post processes, set the override value of the **Enable HCM Data Loader Post Process Configuration** parameter to **No** in the **Configure HCM Data Loader** task, available from **Setup and Maintenance**.

CAUTION: This change is not recommended as it disables the ability to configure post processes for HCM Data Loader business objects. On the **Processes** page for a submitted data set, you will not be able to: view the individual line items of the post processes, track their statuses and view their completion times.

Related Topics

- [How You Review Automatically Submitted Post Processes](#)
- [How You Manage Business Object Post Processes](#)

Data Set Post Processes

Use the **Data Set Post Processes** tab on the **Configure HCM Data Loader** to review and configure automatic submission of the data set level post processes.

Configuring Automatic Submission of Post Processes for Data Sets

In the **Setup and Maintenance** work area:

- Select **Configure HCM Data Loader** task in the **HCM Data Loader** functional area.
- On the **Configure HCM Data Loader** page, select the **Data Set Post Processes** tab.
- Review the data set post processes available for automated submission by HCM Data Loader.
- For supporting data set post processes, enable or disable the automatic submission.

Configure HCM Data Loader ?

←

HCM Data Lo...

HCM Spread...

Business Obj...

Data Set Po...

Data Set Post Processes

Review and configure processes that are automatically submitted after your data set has loaded.

View

	Process Order	Process	De
	1	Cancel Failing Delete HCM Data Loader Stage Data Schedules	Yes
	1	Update Oracle Search Indexes with HCM Data Loader Changes	Yes

Process

Details of the selected process. If the task name is available, use it to disable this process in individual files.

Process Name

Cancel Faili

Description

Cancels Fa

Task Name

CancelFaile

This table explains the set-up for the data set post processes on this page:

Name	Description
Process Order	<div>Preset order of processing of the data set post process.</div> <div>Note:<ul style="list-style-type: none">If the processing order is the same for one or more post processes, then they'll be processed concurrently.If the processing order is incremental, then they'll be processed sequentially.</div>
Process	<div>Name of the data set post process.</div>

Name	Description
Default	Default running state of the data set post process.
Override	If enabled, reset the default running state of the data set post process. Values are Yes and No.
Description	Describes the function of the data set post process.

This table explains the **Process** section on this page:

Name	Description
Process Name	Name of the data set post process.
Description	Describes the function of the data set post process.
Task Name	Name of the task for the data set post process. Use this to disable the process for a specific file.

To manage the automatic submission of post process for data sets, in the **Data Set Post Processes** page: Select the post process and review the value under the **Default** column.

- The value: **Yes**, indicates that the data set post process will be automatically submitted whenever HCM Data Loader or HCM Spreadsheet Data Loader successfully loads data for it.
- The value: **No**, indicates that the data set post process will never be submitted.
- If the **Override** column is enabled for the selected post process, you can overwrite the default value by resetting it with the desired option.

Related Topics

- [How You Manage Data Set Post Processes](#)
- [How You Review Automatically Submitted Post Processes](#)
- [How You Manage Business Object Post Processes](#)

How You Manage Business Object Post Processes

This topic explains how to disable non-mandatory post processes through the file you're uploading.

Disable Business Object Post Processing

To disable the running of all the non-mandatory post processes of a business object, include the following instruction at the top of your file:

```
SET INVOKE_POST_PROCESS N
```

Alternatively, to disable a post-process for all files, go to the **Business Object Post Processes** tab in the **Configure HCM Data Loader Parameters** page, filter by the desired **Business Object**. Select from the configured post processes for the business object and if it's enabled, change the **Override** value to **No**.

To disable the running of all selected post processes of a business object, include the following instruction at the top of your file:

```
SET DISABLE_POST_PROCESS_TASKS <process 1,...,process N>
```

The post processes not included in the `DISABLE_POST_PROCESS_TASKS` statement will be invoked and these post processes will appear under the **Manage HCM Data Loader Post Processes** entry in the **Processes** page for a submitted data set.

Related Topics

- [How You Review Automatically Submitted Post Processes](#)
- [Business Object Post Processes](#)
- [The SET Instruction](#)

How You Manage Data Set Post Processes

This topic explains how to disable the data set level post processes.

Disable Data Set Post Processing

- To disable a post-process for all data sets select the **Data Set Post Processes** tab in the **Configure HCM Data Loader Parameters** page.
- If the data set post process is enabled, you can review, and if enabled, change the **Override** value to **No**.

Note: The SET instruction provided in the .dat file takes precedence over the value set in the **Data Set Post Processes** tab.

For example: If you've set **Update Oracle Search Indexes with HCM Data Loader Changes** data set process to **No** on the **Data Set Post Processes** tab, and if you've provided the following SET instruction in your data set file:

```
SET ORA_ENABLE_OS_BULK_INGEST Y
```

The **Update Oracle Search Indexes with HCM Data Loader Changes** process would still be automatically submitted.

14 Removing Person Information

Remove Person Information

You can remove data related to terminated employees or canceled work relationships from the Oracle HCM Cloud. This solution is available in the Data Exchange work area with its own dedicated tasks.

Additionally using the remove person information feature you can:

- Purge data from HCM Data Loader staging tables for terminated employees.
- Perform staggered disposal of data for a terminated employee. For example, you can now purge Global HR data first while retaining the payroll data for a longer time and purge payroll data at a later date in the future using the same person number reference.
- Delete workers with canceled work relationships (rather than just terminated).

To remove a person information, you need to have access to the following pages:

Remove Person Information



Remove Person Information



Configure Person Information Removal Policies

- **Configure Person Information Removal Policies:** Configure the business objects and its components that you want to remove and mask mandatory attributes. To access this page, you must have the Configure Person Information Removal Policies privilege `HRC_CONFIGURE_PERSON_INFO_REMOVAL_POLICIES_PRIV`.
- **Remove Person Information:** You can remove person information for specified people from this page. When you submit the person information removal process, the data is removed as per the configuration defined in the associated person information removal policy.

To grant access to a user to configure Remove Person Information policies:

- On the Create Role page of the **Security Console (Navigator > Security Console > Roles > Create Role)**, create a job role.
- On the Function Security Policy page, add the following privileges:
 - Configure Person Information Removal Policies `HRC_CONFIGURE_PERSON_INFO_REMOVAL_POLICIES_PRIV`
 - Manage Data Exchange Work Area `HRC_MANAGE_DATA_EXCHANGE_PRIV`

To grant access to a user for removing person information:

- On the Create Role page of the Security Console (**Navigator > Security Console > Roles > Create Role**), create a job role
- On the Function Security Policy page, add the following privileges:
 - Manage Data Exchange Work Area `HRC_MANAGE_DATA_EXCHANGE_PRIV`
 - Upload Data for Human Capital Management file based Import `HCM_DATALOADER_IMPORT_RWD`

- c. On the Role Hierarchy page (**Navigator > Security Console > Roles > Create Role**), add this role:
 - i. Load Data using HCM Data Loader `ORA_HRC_LOAD_DATA_USING_HDL` and one of these roles:
 - a. Remove Person Information `ORA_HRC_REMOVE_PERSON_INFORMATION` role membership to access both Report and Remove modes
 - b. Generate Remove Person Information `ORA_HRC_GENERATE_REMOVE_PERSON_INFORMATION_FILE` role membership to access Report mode alone
- d. On the Data Roles and Security Profiles page (**Navigator > My Client > Workforce Structures > Data Roles and Security Profiles**), create a data role for the job role that you created.

Note: Remove Person Information security is person-based therefore, to restrict access for removing person information, you can create a security profile on the Person Security profiles page and add it to the data role. If you want users to access and remove all person records, use the **View All People** security profile.

- e. Add the data role to appropriate users.

Note: For more information about Security Roles, refer the HCM Security Guide.

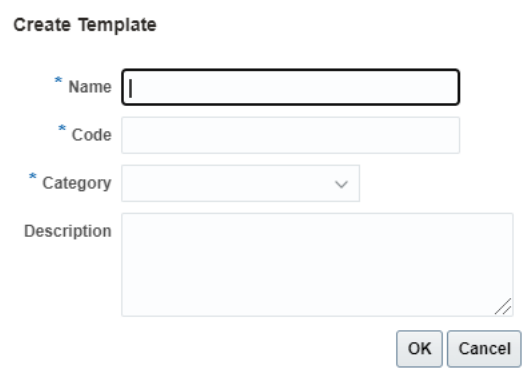
How Person Removal Policies Work

Before you submit the Remove Person Information process, you must configure a policy that defines which business objects will have data removed or obfuscated. The Configure Person Information Removal Policies user interface includes business objects that are supported by this feature. You will see that records you can delete are highlighted and can only be selected for deletion in the policy configuration. Records that you can't delete, you can configure at the attribute level, and identify the attributes that should be set to null or how an attribute should be masked (for attributes that cannot be set to NULL). Here are the attributes displayed on the page:

- CATEGORY: The Business Object the template was created for (for example: Worker).
- STATUS: The status of the template (Draft, Active, Inactive). Only a template in Draft status can be modified. An active template can be only processed. An inactive template can't be processed nor can it be modified.
- CREATED BY: The user who created the template.
- LAST UPDATE DATE: The last date and time the template was updated.

To create templates and configure removal policies:

1. Click **Create** to open the **Create Template** window. Required fields are:
 - NAME: The name of the template.
 - CODE: The template code, defaults to the name of the template.
 - CATEGORY: The Business Object the template is created for: the options are Worker or Candidate.
 - DESCRIPTION (optional): The user can also enter a description.



Create Template

* Name

* Code

* Category

Description

OK Cancel

2. Select one or more Business Objects.

3. Under the Business Objects section, search for a specific business object and select objects to process. The Components section, select the child objects under the selected business object.

Note:

- When you select a component with associated dependent objects, you're recommended to select the dependent objects to avoid orphan data.

*Name

Remove worker

Code

Remove worker

Status

Draft

Description

Category

Worker

Last Update Date

5/17/19

Last Updated By

HCM_USER10

Business Objects

View 

Object Name	Enabled
Worker	<input checked="" type="checkbox"/>
Salary	<input checked="" type="checkbox"/>

Components

View 

Worker

☐

☒

External Identifier

☐

☒

Person Address

☐

☒

Person Citizenship

☐

☒

Person Delivery M

☐



☒






Person Drivers Lic

4. Use the Rules tab to configure policies. Most objects are configured for deletion, so no specific rules are needed for those objects. You can configure rules for objects if they contain attributes that can only be obfuscated and cannot be deleted. If you specify a generic value to include in a field, you can use any value except null. For example, the Person Last Name can't be deleted, so it needs to be replaced by a static value. ZZZZZ is the default. The Person Last Name will be replaced by this static value appended by an internal identifier, to



maintain the uniqueness of this attribute and thus the data integrity. Notice the Search boxes for Attribute Names and Description.

Attribute Rules: Worker

View  

Attribute Name	Description	Operation	Value
 Blood type	The person's blood type.	Delete	
 Correspondence L...	The person's correspondence language.	Delete	
 Country of Birth	The person's country of birth.	Delete	
 Date of Birth	The person's date of birth.	Delete	
 Date of Death	The person's date of death.	Delete	

Flexfield Attribute Rules: Worker

View  

Flexfield	Flexfield Context Attribute	Code	Value
No required flexfield global segments are available for this component.			
Columns Hidden: 1			

Note: You can remove flexfields that are registered in HCM Data Loader registered like any business object. Upon deletion, only required global segments of flexfield attributes remain and the rest is removed automatically.

Remove Person Information

When you submit the Remove Person Information process, you will need to identify the population of workers to discard and select the data removal policy to use. The policy determines which business objects are removed and how. Surrogate IDs are used to identify the records to discard, but the Person Information Removal framework will also include any other attributes that are mandatory when disposing data, such as ActionCode.

Points to consider:

- Ensure that the process is run on terminated workers only, with a termination date in the past. Deleted data can't be recovered so proceed with caution.
- The process details are stored (for example, templates and person number) but the worker data as per the template definition is removed.
- A user can delete data for a person if they can access the person's data. For example, a user must have the following privileges to dispose Person Email and Person Phone data:
 - Manage Person Phone - PER_MANAGE_PERSON_PHONES_DATA
 - Manage Person Email - PER_MANAGE_PERSON_EMAIL_DATA

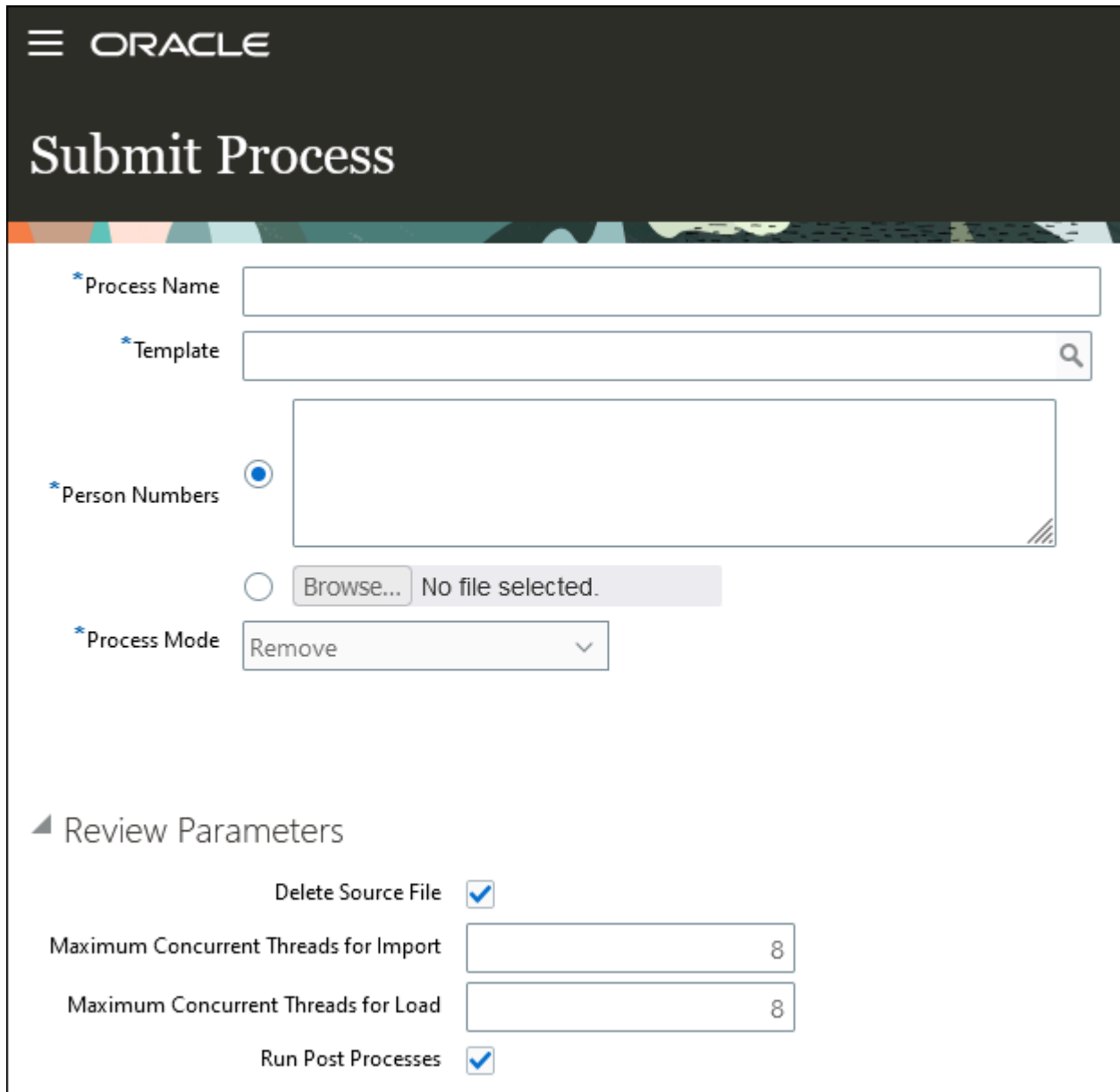
To start the **Remove Person Information** process, perform these steps:

1. Click **Submit Process**
2. Enter a process name and select a template from the searchable list.
3. Enter the **Person Numbers**, separated by commas, which are to be deleted or obfuscated.

Note: You can also upload a file containing a list of Person Numbers separated by commas.

4. Select a **Process Mode: Report** or **Remove**.

Note: The **Remove** mode is an ESS process that extracts key information for the records to be deleted or masked and generates an HCM Data Loader business object .dat file for each business object in the selected policy. The generated .dat files are zipped together and uploaded with HCM Data Loader. The **Report** mode only produces the .dat file that is zipped and can be downloaded locally for review. HCM Data Loader does not process the .dat file and no information is deleted or obfuscated when run in **Report** mode.



Submit Process

*Process Name

*Template

*Person Numbers ☐

☐ No file selected.

*Process Mode

Review Parameters

Delete Source File ☒

Maximum Concurrent Threads for Import

Maximum Concurrent Threads for Load

Run Post Processes ☒

- a. In the **Remove** mode, you can set the following additional runtime parameters to override the default HCM Data Loader values in the **Review Parameters** section:
 - i. **Delete Source File:** Deletes the data set file from UCM.
 - ii. **Maximum Concurrent Threads for Import:** Maximum number of concurrent threads to assign to the import process. The default is defined by the Maximum Concurrent Threads for Import parameter in the Configure HCM Data Loader task.

- iii. **Maximum Concurrent Threads for Load:** Maximum number of concurrent threads to assign to the load process. The default is defined by the Maximum Concurrent Threads for Load parameter in the Configure HCM Data Loader task.
- iv. **Run Post Processes:** Submits the business object post processes and the Oracle Search Bulk Ingestion process.

b. In the **Report** mode, you can only set the **Run Post Processes** review parameter.

Note: If you deselect the **Run Post Processes** parameter while clicking Submit, then a warning message displayed to indicate that post processes, including Oracle Search Bulk Ingestion, will not be executed for this Data Disposal process.

Here's what you will see on the page after you submit the process:

- STATUS: Success or Error.
- PROCESS MODE: Remove mode or Report mode.
- SUBMITTED BY: The user who submitted the process.
- SUBMISSION DATE: The date the process was submitted.

You can also use the View Process Details icon to view different stages of the Person Information Removal process, as well as the corresponding log files:

- The log file details all steps taken in the process and the number of workers processed (for example, No of Employees Matching the Termination Criteria).
- If the Report mode was selected, you can use the Download Data File icon to download the .zip file containing the .dat file, although this file can't be used in another HDL process.

When to Create Templates for Effective Data Disposal

You must create new templates to prevent components from being missed out or to avoid errors during data disposal.

These are some of the scenarios when you must create new templates:

- Changes to Business Object Components or Attributes:
 - Adding a new Business Object Component. For example, adding a new business object component Person Phone in the Worker object.
 - Modifying components in a business object hierarchy. For example, changing the entity type from DELETE to UPDATE.
 - Deleting an existing component from the business object hierarchy. For example, deleting the Person Email component of the Worker object.
 - Adding a new Attribute in a Business Object Component. For example, adding a new attribute Person First Name to the list of Person Name component of the Worker object.
 - Changing the default value for an existing attribute. For example, updating the default value of Person First Name from ZZZZZ to WWWW.
- Changes in Flexfield structure:
 - Adding a new Mandatory Global Segments. For example, adding a new Global Mandatory Segment for a worker descriptive flexfield in the Worker business object hierarchy. This would cause the existing templates in Worker to fail due to missing mandatory segments. You must create a new template to include the latest flexfield definition.
 - Supporting new flexfields for a business-object component in a new release.

How to Enable Access to the Data Disposal Folder for Reporting

To enable access to the Data Disposal folder to report on Data Disposal Processes in the Data Exchange work area, add the following function security privilege:

Functional Privilege Name	Functional Privilege Code
Report Person Information Removal Processes	ORA_HRC_REPORT_PERSON_INFO_REMOVAL_PROCESSES

This functional security privilege allows reporting on person information removal process submissions.

To enable access to the Data Disposal folder to report on Data Disposal Templates in the Data Exchange work area, add the following function security privilege:

Functional Privilege Name	Functional Privilege Code
Report Person Information Removal Templates	ORA_HRC_REPORT_PERSON_INFO_REMOVAL_TEMPLATES

This functional security privilege allows reporting on the templates that are used by the person information removal process.

Remove Person Payroll Information

Use the **Remove Person Information** task in the Data Exchange work area to remove payroll data related to terminated employees or canceled work relationships from the Oracle HCM Cloud.

You can only remove payroll data from employees who don't have Cloud Payroll results. For example, when you use a third-party payroll product, you can delete the payroll data captured in Oracle HCM Cloud.

Note: You can't remove payroll data for employees with payroll results, such as employees included in the Calculate Payroll process.

The employee payroll data you can remove include:

- Element Entries: Includes entries created by other products such as Compensation and Benefits.
- Calculation Entries and Calculation Cards: Includes time, absence, and involuntary deductions cards.
- Payroll Relationships and Payroll Assignments
- Payrolls: This is the payroll record assigned to the employee, such as Weekly payroll.
- Person Costing
- Personal Payment Methods

Consider the following when you configure the person information removal policy for payroll objects:

- In the **Business Objects** section, you can select the check box next to either the standalone **Payroll** node or the **Worker > Work relationship** node. When you select either of these nodes, all the employee payroll data mentioned previously gets deleted.
- Unlike the Worker node, where you can view the components on the page, for the Payroll node, components or child nodes are not displayed on the page.
- You cannot configure any rules for payroll because the payroll node doesn't include attributes for masking.

Note: Run the process only for terminated employees with a termination date in the past. Once deleted, you can't recover the deleted data. You can delete data for a person only if you have access to the person's data.

Related Topics

- [Remove Person Information](#)

Initiating Remove Person Information using REST API

You can automate the Remove Person Information integration using **removePersonInformation** REST resource .

This resource allows you to retrieve details of the process submissions and provides the following custom actions:

Action	Method	Description
Get all Remove Person Information processes	GET	Provides details of Remove Person Information processes.
Get an individual Remove Person Information process	GET	Provides details of Remove Person Information process for the specified process id.
Submit RPI	POST	Submit Remove Person Information process to remove the data that's configured by the named template, for the population specified.
Report Person Information	POST	Submits Remove Person Information to generate a HDL file that would remove the data configured by the named template, for the population specified, but it doesn't submit HDL. It's stops after generating the file, allowing you to review the impact.
Stop Remove Person Information process	POST	Requests that Remove Person Information processing ceases.
Download File	POST	Downloads the HCM Data Loader generated report file to the Oracle WebCenter Content server.
Delete Report Process	POST	Delete the processes that are run in report mode.

Granting Access to the RPI REST resource

To enable access to initiate remove person information processing, you'll need this role hierarchy:

Role Name	Role Code	Access Granted
Use REST Service - Remove Person Information	ORA_HRC_REST_SERVICE_ACCESS_DATA_LOAD_DATA_SETS	removePersonInformation REST resource
Upload data for Human Capital Management file based Import	HCM_DATALOADER_IMPORT_RWD	HCM Data Loader import account on the Oracle WebCenter
Load Data using HCM Data Loader	ORA_HRC_LOAD_DATA_USING_HDL	Submit HDL to upload the Remove Person Information generated file.

On the **Data Roles and Security Profiles** page (**Navigators > My Client > Workforce Structures > Data Roles and Security Profiles**), create a data role for the job role that you created.

Users granted this data role will be able to submit and monitor RPI using the **removePersonInformation** REST resource.

15 Purging Person Data from Test Environments

Overview of the Purge Person Data in Test Environments Process

During implementation and testing, you may want to load, test, purge, and reload HCM data. To purge person data from test environments, use the Purge Person Data in Test Environments process.

This topic explains how to enable the process and identifies the data that it purges.

Enabling the Purge Process for Person Data

The **Purge Person Data in Test Environments** process is now enabled by default on the test or stage environments.

Note: You can't run the **Purge Person Data in Test Environments** process in production environments.

Purging Person Data

The **Purge Person Data in Test Environments** process purges person records, including passports, citizenship records, licenses, emails, addresses, national identifiers, and extra information type data. It also purges dependent data of the following types:

- Absences
- Assignments, including extra information type data
- Benefits
- Calculation cards
- Compensation
- Contacts and contact relationships
- Documents and attachments
- Element entries
- Goals
- Payroll relationships
- Performance records
- Roles and user details
- Talent profiles
- Work relationships

Deletion requests are generated for user accounts in the test environment. To process these requests, you must run the **Send Pending LDAP Requests** process when **Purge Person Data in Test Environments** completes.

Exclusions from the Purge Process

The **Purge Person Data in Test Environments** process doesn't purge data of the following types:

- Predefined person data.
- Contacts enrolled in a Benefits program.
- Oracle Trading Community Architecture (TCA) data.

The process inactivates dependent TCA records in the TCA tables but doesn't purge them.

If processed payroll actions are associated with a worker, then the worker isn't purged. You must roll back relevant payroll processes before you can purge such workers.

Related Topics

- [Purge Person Data from Test Environments](#)
- [Tables Purged by the Purge Person Data in Test Environments Process](#)

Purge Person Data from Test Environments

This topic describes how to run the Purge Person Data in Test Environments process. The process must be enabled for use in your test environment.

This topic also describes how to run **Send Pending LDAP Requests** to process user-deletion requests generated by the **Purge Person Data in Test Environments** process.

Run the Purge Person Data in Test Environments Process

You must have the `HRC_PURGE_PERSON_PROCESS_PRIV` function security privilege to run this process.

Sign in and follow these steps:

1. Select **Navigator > Tools > Scheduled Processes**.
2. In the Scheduled Processes work area, click **Schedule New Process**.
3. In the Schedule New Process dialog box, search for and select the **Purge Person Data in Test Environments** process.
4. In the Process Details dialog box, set one of the following parameters to identify the person data to purge.

Parameter	Description
Person ID	Provide one or more person IDs. Separate multiple IDs with commas, for example, <code>100010034148157,100010034148169,100010034148181,100010034148192</code> .
Person Number	Provide one or more person numbers. Separate multiple numbers with commas. You can use the wildcard character to supply partial values, for example, <code>PER%,Test%,%abc%</code> .
Person Query	Enter a SQL query to return the person IDs of all persons whose data you want to purge.

Parameter	Description

To return the person IDs of all person records where the last name starts with the characters `John`, you could use the following SQL query:

```
SELECT papf.person_id FROM per_all_people_f papf, per_person_names_f ppnf WHERE  
ppnf.last_name like 'John%' and ppnf.person_id = papf.person_id
```

To return all person IDs in the test environment, use the following SQL query:

```
SELECT person_id FROM per_all_people_f
```

5. Set the **Save** parameter, as follows.

Save Value	Description
N	Lists the persons whose data would be purged by the process. No data is purged.
Y	Purges the person data.

6. Click **Submit**.

Tip: The time this process takes depends on the amount of data to be purged. Don't cancel the process, as its final stages include clean-up operations. If you cancel the process, then the clean-up operations resume when you next run the process.

Run the Send Pending LDAP Requests Process

Purge Person Data in Test Environments generates requests to delete user accounts from your LDAP directory. To process these requests, run the **Send Pending LDAP Requests** process. If this process isn't scheduled to run in the test environment, then follow these steps when **Purge Person Data in Test Environments** completes:

1. In the Scheduled Processes work area, click **Schedule New Process**.
2. In the Schedule New Process dialog box, search for and select the **Send Pending LDAP Requests** process.
3. In the Process Details dialog box:
 - a. Set **User Type** to **All**.
 - b. Leave **Batch Size** set to **A**, which means that the processing batch size is calculated automatically.
4. Click **Submit**.

Related Topics

- [Overview of the Purge Person Data in Test Environments Process](#)
- [Tables Purged by the Purge Person Data in Test Environments Process](#)

Tables Purged by the Purge Person Data in Test Environments Process

This topic lists the tables that are purged when you run the Purge Person Data in Test Environments process.

Product	Table
Absence Management	<ul style="list-style-type: none">• ANC_PER_ABS_CERTS• ANC_PER_ABS_ENTRIES• ANC_PER_ABS_ENTRY_DTLS• ANC_PER_ABS_MATERNITY• ANC_PER_ABS_PLAN_ENTRIES• ANC_PER_ABS_PLN_SUMM_ENT• ANC_PER_ABS_TYP_LNKG• ANC_PER_ABS_TYPE_ENTRIES• ANC_PER_ACCRUAL_ENTRIES• ANC_PER_ACRL_ENTRY_DTLS• ANC_PER_EVENTS• ANC_PER_PLAN_ENROLLMENT
Benefits	<ul style="list-style-type: none">• BEN_BATCH_ACTN_ITEM_INFO• BEN_BATCH_BNFT_CERT_INFO• BEN_BATCH_COMMU_INFO• BEN_BATCH_DPNT_INFO• BEN_BATCH_ELCTBL_CHC_INFO• BEN_BATCH_ELIG_INFO• BEN_BATCH_LER_INFO• BEN_BATCH_RATE_INFO• BEN_BENEFIT_ACTIONS• BEN_BENEFIT_RELATIONS_F• BEN_CVRD_DPNT_CTFN_PRVDD• BEN_ELCTBL_CHC_CTFN• BEN_ELIG_CVRD_DPNT• BEN_ELIG_DPNT• BEN_ELIG_FLX_CRDT_POOL_CHC• BEN_ELIG_PER_ELCTBL_CHC• BEN_ELIG_PER_F• BEN_ELIG_PER_OPT_F

Product	Table
	<ul style="list-style-type: none"> BEN_ELIG_RSLT_F BEN_ENRT_BNFT BEN_ENRT_RT BEN_EXTRACT_DETAILS_ARCH BEN_EXTRACT_REQ_DETAILS BEN_FLEX_CRDT_LDGR BEN_PER_BENEFIT_GROUP_F BEN_PER_BNF_ORG BEN_PER_BNFTS_BAL_F BEN_PER_IN_LER BEN_PER_LE_HABITS_COV_F BEN_PERSON_ACTIONS BEN_PIL_ELCTBL_CHC_POPL BEN_PL_BNF BEN_PL_BNF_CTFN_PRVDD BEN_PRTT_ENRT_ACTN BEN_PRTT_ENRT_CTFN_PRVDD BEN_PRTT_ENRT_RSLT BEN_PRTT_LEG_DISCMR_ACTN BEN_PRTT_RT_VAL BEN_PTNL_LER_FOR_PER BEN_REL_ASSIGNMENTS_F BEN_REPORTING
Compensation Management	<ul style="list-style-type: none"> CMP_BUDGET_AUDIT CMP_BUDGET_USAGES CMP_CWB_ALERTS CMP_CWB_APPROVAL_HISTORY CMP_CWB_AUDIT CMP_CWB_FEEDBACK CMP_CWB_HRCHY CMP_CWB_INTCPT_APPR_DTLS CMP_CWB_MODEL_B CMP_CWB_MODEL_DTL CMP_CWB_MODEL_TL CMP_CWB_PERF_RATINGS CMP_CWB_PERSON_DEFAULTS CMP_CWB_PERSON_ELEMENTS CMP_CWB_PERSON_INFO

Product	Table
	<ul style="list-style-type: none"> • CMP_CWB_PERSON_RATES • CMP_CWB_POST_LOG • CMP_CWB_POST_PERSON • CMP_CWB_PROMOTIONS • CMP_CWB_RANKS • CMP_CWB_SUMMARY • CMP_CWB_SWITCH_MANAGERS • CMP_DYNAMIC_FILTER • CMP_INTERCEPT_APPROVERS • CMP_PERSON_BUDGETS • CMP_PROCESS_DATA • CMP_PROCESS_RUN_INFO • CMP_SALARY • CMP_SALARY_COMPONENTS • CMP_SALARY_EXPORT • CMP_STOCK_DETAILS • CMP_TCS_PER_ITEM_VALUE • CMP_TCS_PER_PERD • CMP_TCS_PER_PERD_STMT • CMP_TCS_PER_PERD_STMT_CAT • CMP_TCS_PER_ROW_VALUES • CMP_TCS_REPORT_DETAILS • CMP_VC_PER_ELIG_PLANS
Oracle Middleware Extensions for Applications	<ul style="list-style-type: none"> • FND_ATTACHED_DOCUMENTS • FND_DOCUMENTS • FND_DOCUMENTS_TL • FND_SESSION_USERS
Workforce Modeling	<ul style="list-style-type: none"> • HMO_MODEL_PLAN_DETAILS • HMO_MODEL_PLANS_B • HMO_MODEL_PLANS_TL
Performance Management	<ul style="list-style-type: none"> • HRA_EVAL_ITEMS • HRA_EVAL_PARTICIPANTS • HRA_EVAL_RATINGS • HRA_EVAL_ROLE_ACTIONS • HRA_EVAL_ROLES • HRA_EVAL_SECTIONS • HRA_EVAL_STEPS

Product	Table
	<ul style="list-style-type: none"> HRA_EVALUATIONS HRA_CHECK_IN_MEETINGS HRA_DISCUSSION_TOPICS
Goal Management	<ul style="list-style-type: none"> HRG_GOAL_ACCESS HRG_GOAL_ACTIONS HRG_GOAL_ALIGNMENTS HRG_GOAL_TARGET_OUTCOMES HRG_GOALS
Succession Management	<ul style="list-style-type: none"> HRM_PLANS HRM_PLAN_CANDIDATES HRM_PLAN_OWNERS
Talent Review	<ul style="list-style-type: none"> HRR_DASHBOARDS HRR_MEETINGS HRR_MEETING_FACILITATORS HRR_MEETING_PARTICIPANTS HRR_MEETING_REVIEWEES
Questionnaire	<ul style="list-style-type: none"> HRQ_QSTN_RESPONSES HRQ_QSTNR_PARTICIPANTS HRQ_QSTNR_RESPONSES
Profile Management	<ul style="list-style-type: none"> HRT_NOTES HRT_POOL_MEMBERS HRT_POOL_OWNERS HRT_PROFILE_INTERESTS HRT_PROFILE_ITEMS HRT_PROFILE_KEYWORDS HRT_PROFILE_MATCH_CRIT HRT_PROFILE_RELATIONS HRT_PROFILES_B HRT_PROFILES_TL
Workforce Management	<ul style="list-style-type: none"> HWM_TM_REC HWM_TM_REC_GRP HWM_TM_REC_GRP_SUM HWM_TM_REC_GRP_USAGES HWM_TM_REP_ATRB_USAGES
Workforce Predictions	<ul style="list-style-type: none"> HWP_ARCH_DATA_MINING_REASONS

Product	Table
	<ul style="list-style-type: none"> • HWP_ARCH_DATA_MINING_RESULTS • HWP_DATA_INPUT_VALUES • HWP_DATA_MINING_REASONS • HWP_DATA_MINING_RESULTS • HWP_WHATIF_PRED_DETAILS
Global Payroll	<ul style="list-style-type: none"> • PAY_ASSIGNED_PAYROLL • PAY_ASSIGNED_PAYROLLS_DN • PAY_ASSIGNED_PAYROLLS_DN_ • PAY_ASSIGNED_PAYROLLS_F • PAY_ASSIGNED_PAYROLLS_F_ • PAY_DATES • PAY_DIR_CARD_COMPONENTS_F • PAY_DIR_CARD_COMPONENTS_F_ • PAY_DIR_CARDS_F • PAY_DIR_CARDS_F_ • PAY_DIR_COMP_DETAILS_F • PAY_DIR_COMP_DETAILS_F_ • PAY_DIR_REP_CARD_USAGES_F • PAY_DIR_REP_CARD_USAGES_F_ • PAY_DIR_REP_CARDS_F • PAY_DIR_REP_CARDS_F_ • PAY_ELEMENT_ENTRIES_F • PAY_ELEMENT_ENTRIES_F_ • PAY_ELEMENT_ENTRY_VALUES_F • PAY_ELEMENT_ENTRY_VALUES_F_ • PAY_ENTRY_PCT_DIST • PAY_ENTRY_PROC_DETAILS • PAY_ENTRY_USAGES • PAY_PAY_RELATIONSHIPS_DN • PAY_PAY_RELATIONSHIPS_F • PAY_PERSON_PAY_METHODS_F_ • PAY_RANGE_ITEMS_F • PAY_RANGE_ITEMS_F_ • PAY_REL_GROUPS_DN • PAY_REL_GROUPS_F • PAY_RETRO_ENTRIES • PAY_VALUE_DEFINITIONS_F • PAY_VALUE_DEFINITIONS_F_

Product	Table
Global Human Resources	<ul style="list-style-type: none"> • HR_DOCUMENT_DELIVERY_PREFS • HR_DOCUMENTS_OF_RECORD • PER_ABSENCE_CASES • PER_ACTION_OCCURRENCES • PER_ADDRESSES_F • PER_ALL_ASSIGNMENTS_M • PER_ALL_PEOPLE_F • PER_ALLOCATED_CHECKLISTS • PER_ALLOCATED_TASKS • PER_ASG_RESPONSIBILITIES • PER_ASSIGN_GRADE_STEPS_F • PER_ASSIGN_WORK_MEASURES_F • PER_ASSIGNMENT_EXTRA_INFO_M • PER_ASSIGNMENT_SUPERVISORS_F • PER_CITIZENSHIPS • PER_CONTACT_EXTRA_INFO_F • PER_CONTACT_RELSHIPS_F • PER_CONTRACTS_F • PER_DELEGATED_ROLES • PER_DISABILITIES_F • PER_DRIVERS_LICENSE_TYPES • PER_DRIVERS_LICENSES • PER_EMAIL_ADDRESSES • PER_ETHNICITIES • PER_IMAGES • PER_KEYWORDS • PER_LDAP_REQUESTS • PER_LDAP_ROLE_MEMBERSHIPS • PER_LDAP_USERS • PER_MANAGER_HRCHY_DN • PER_NATIONAL_IDENTIFIERS • PER_PASSPORTS • PER_PEOPLE_EXTRA_INFO_F • PER_PEOPLE_LEGISLATIVE_F • PER_PERIODS_OF_SERVICE • PER_PERSON_ADDR_USAGES_F • PER_PERSON_DLVRY_METHODS • PER_PERSON_NAMES_F

Product	Table
	<ul style="list-style-type: none">• PER_PERSON_TYPE_USAGES_M• PER_PERSONS• PER_PHONES• PER_PORTRAIT_CARD_CONFIG• PER_PORTRAIT_PREFERENCES• PER_RELIGIONS• PER_SCHEDULE_ASSIGNMENTS• PER_SHARE_INFORMATION• PER_USER_HISTORY• PER_USER_ROLES• PER_USERS• PER_VISAS_PERMITS_F

Related Topics

- [Overview of the Purge Person Data in Test Environments Process](#)
- [Purge Person Data from Test Environments](#)

16 Maintaining Stage Tables

Overview of Maintaining the Stage Tables

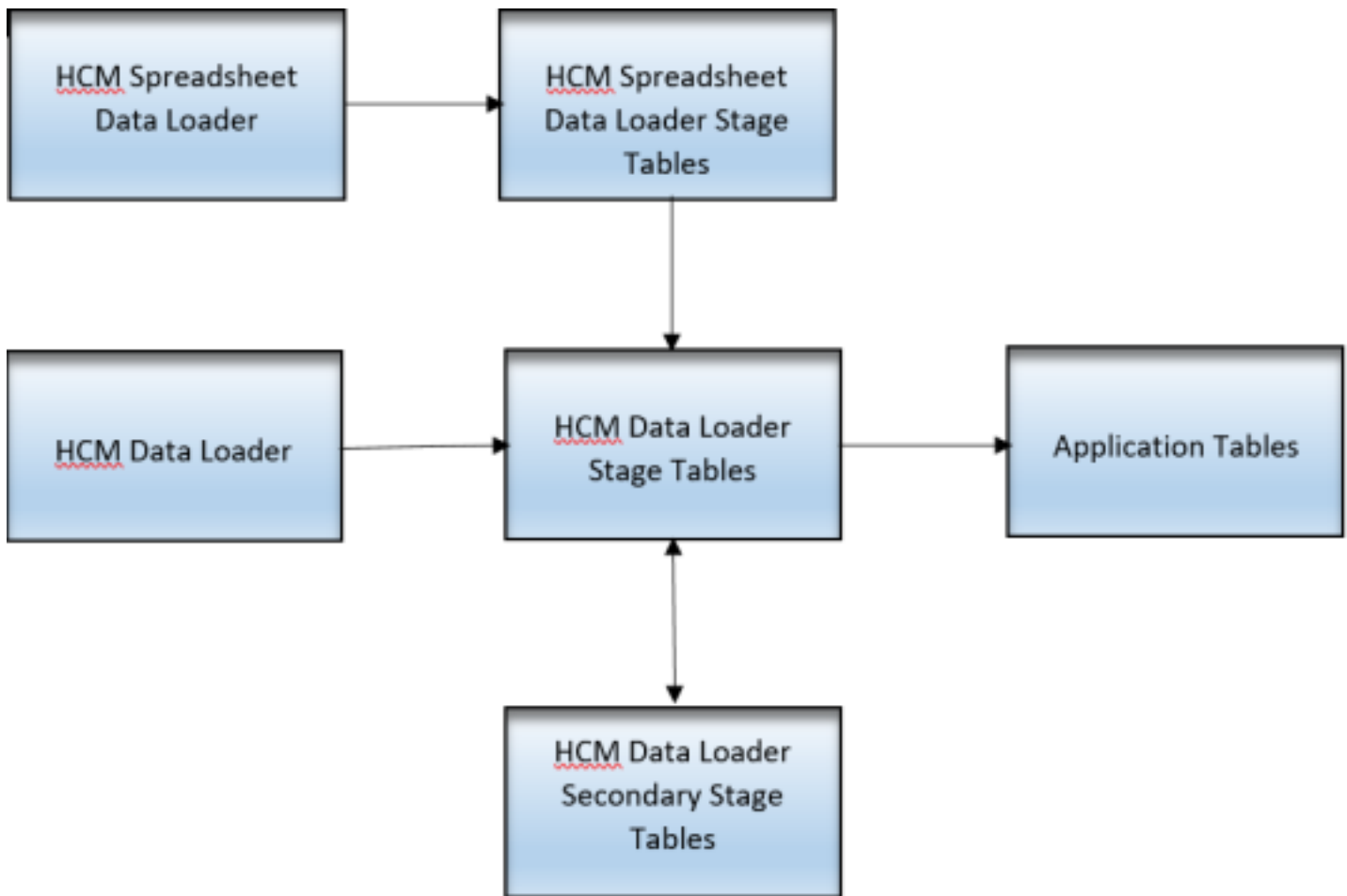
When you load large volumes of data into Oracle HCM Cloud, the stage tables can grow rapidly.

You're recommended to create a schedule to delete processed data sets that you no longer need in the stage tables. How often you delete processed data sets from the stage tables depends on the volume and frequency of your data loads. During data migration, consider deleting every large data set as its processing completes. This topic describes the role of stage tables in data loading and explains how the deletion process maintains the stage tables automatically.

How Data Is Loaded to Oracle HCM Cloud

This figure shows how data is loaded to the application tables.

Data that you load using either HCM Data Loader or HCM Spreadsheet Data Loader is imported first to stage tables, where validation occurs. HCM Spreadsheet Data Loader has its own stage tables, from where valid data is loaded to the HCM Data Loader stage tables. Valid data is loaded from the HCM Data Loader stage tables to the application tables. When a data set contains data for an object that supports roll back, such as element entry, the data for the entire data set can be moved to secondary stage tables, ensuring it's not purged prematurely and doesn't negatively impact the bulk load processing which occurs from the main stage tables



Note: In the Import and Load Data page, data sets that are retained for an extended period in the secondary stage table can be identified by the green tick that are retained for an extended period in the secondary stage table can be identified by the green tick in the Extended Retention column.

How to Extend the Period of Retention for a Data Set

The Enable Automatic Extended Data Set Retention configuration parameter determines if data sets containing data that can be validated or rolled back are automatically moved to the secondary staging tables. The extended retention period is defined by the Days to Preserve Extended Retention Data Sets configuration parameter. You can override the default behavior at the data set level by using the `EXTEND_DATA_SET_RETENTION SET` instruction. If you have loaded a data set and it is held for the standard retention period within the primary staging tables, you can extend the data set retention by moving it to the secondary staging tables from the Import and Load Data page.

How Processed Data Sets Are Deleted from Stage Tables

You can delete processed data sets from three sets of stage tables.

- To delete HCM Spreadsheet Data Loader data sets from all stage tables, use the **Delete Spreadsheet Stage Table Data** task. It runs the **Delete HCM Spreadsheet Data Loader Stage Table Data** process.

- To delete data sets from the HCM Data Loader extended retention stage tables, use the **Delete Stage Table Data** task. It runs the **Delete HCM Data Loader Stage Table Data** process.

Alternatively, you can delete individual data sets from these stage table using the Delete button on the Data Sets table within the Import and Load Data interface.

Perform all tasks in the Data Exchange work area.

You can delete:

- Individual data sets
- All data sets that currently match specified criteria

Tip: You can schedule a recurring deletion request. For example, you can schedule weekly deletion of all data sets that haven't been updated in the last 7 days.

When you run the **Import and Load Data** process, it determines whether a schedule exists for the **Delete HCM Data Loader Stage Table Data** process. If no schedule exists, then the process creates one to run **Delete HCM Data Loader Stage Table Data** nightly.

Note: If the **Import and Load Data** process identifies that an existing schedule is failing because the initiating user is no longer having the permissions to run the process, the schedule will be automatically deleted with a warning.

When you delete extended retention data sets from the HCM Spreadsheet Data Loader stage tables, they're deleted automatically from the HCM Data Loader stage tables. You can also delete spreadsheet data sets explicitly from the HCM Data Loader stage tables. In this case, they remain available in the HCM Spreadsheet Data Loader stage tables. You can't recover deleted data sets. Don't delete data sets that you want to report on or that you want to extract error information for.

When you submit the HSDL, it determines whether a schedule exists for **Delete HCM Spreadsheet Data Loader Stage Table Data** process. If none exists, then the process creates one to run **Delete HCM Spreadsheet Data Loader Stage Table Data** weekly.

Note:

- When the HSDL data set is removed from the HDL staging tables, it can no longer be used for processing. It's only useful for reporting purposes when retained beyond its expiry period.
- If the HSDL purge process fails more than five consecutive times, the **Cancel Failing Delete HCM Data Loader Stage Data Schedules** process will cancel the **Delete HCM Spreadsheet Data Loader Stage Table Data** schedule.

How Delete HCM Data Loader Stage Table Data Works

The **Delete HCM Data Loader Stage Table Data** process deletes data sets from the HCM Data Loader and extended retention stage tables. You can delete from one set of stage tables at a time and this is determined by the data set type. Whenever the process runs for standard data sets, the process:

1. Purges automatically from the stage tables any data set that hasn't been updated in the last 30 days. You can specify a different number of days by setting the **Days to Preserve Standard Data Sets Before Deleting** configuration parameter.

2. Reviews the number of data lines still in the stage tables. If the number exceeds a specified maximum value, then data sets may be archived automatically. You specify:
 - The maximum number of data lines on the **Maximum Data Lines Before Archiving** configuration parameter.
 - The number of days since a data set was last updated before it can be backed up. Specify this value on the **Days to Preserve Standard Data Sets Before Archiving** configuration parameter.

Therefore, a data set is a candidate for automatic archiving only if:

- The number of data lines in the stage tables exceeds the value specified on the **Maximum Data Lines Before Archiving** parameter.
- The data set hasn't been updated in the number of days specified on the **Days to Preserve Standard Data Sets Before Archiving** parameter.

The archive starts with the oldest data set and continues until the number of data lines in the stage tables drops below the specified maximum.

The same process is used to delete extended retention data sets. The retention period of extended retention data sets is controlled by the *Days to Preserve Extended Retention Data Sets*.

How HCM Spreadsheet Data Loader Stage Table Data Works?

When you run HCM Spreadsheet Data Loader (HSDL), the system checks whether a schedule exists for the **Delete HCM Spreadsheet Data Loader Stage Table Data** process. If none exists, a new schedule is automatically created to run weekly.

This scheduled process uses the **Days to Retain Data Sets Before Deleting** parameter to purge data sets that haven't been processed within the specified time. The default value of this parameter is 60 days, but you can customize it on the **HCM Spreadsheet Data Loader Parameters** tab of the **Configure HCM Data Loader** page available in the **Data Exchange** work area.

Archived Data Sets

These values remain in the stage tables when a data set is archived:

- The statuses of the data set and business objects
- The record count
- Any error messages

All other data from the data set is deleted. After a data set is archived

- You can't open error management pages for that data set.
- It doesn't appear in search results when you search for data sets to purge.
- You can access the data using a custom query only.

You can archive data sets manually by running the **Delete HCM Data Loader Stage Table Data** process with the **Action** parameter set to **Archive**. You can't create a schedule when the action is **Archive**.

Related Topics

- [Maintain the HCM Data Loader Stage Tables](#)
- [Maintain the HCM Spreadsheet Data Loader Stage Tables](#)
- [HCM Data Loader Configuration Parameters](#)
- [The SET Instruction](#)
- [Example of using HCM Data Loader](#)

Example of the Automatic Deletion and Backup of Stage Table Data

When you perform the Delete Stage Table Data task, it runs the Delete HCM Data Loader Stage Table Data process. This process maintains the HCM Data Loader stage tables automatically.

This example shows how automatic maintenance of the stage tables works in conjunction with a user request to delete stage table data.

Maintaining the Stage Tables Automatically

The user performs the **Delete Stage Table Data** task on 9 May 2018 to purge any data sets:

- That weren't updated in the last 5 days
- Whose **Import** and **Load** statuses are both **Success**

Configuration parameters have their default values, as follows:

Configuration Parameter	Value
Days to Retain Data Sets Before Archiving	5
Days to Retain Data Sets Before Deleting	30

The following table shows the status of recent import and load processes when the user performs **Delete Stage Table Data**.

Data Set	Imported Status	Loaded Status	Last Updated
1	Success	Success	9 May 2018
2	Success	Error	7 May 2018
3	Success	Success	1 May 2018

Data Set	Imported Status	Loaded Status	Last Updated
4	Success	Error	28 April 2018
5	Success	Success	8 April 2018
6	Success	Error	8 April 2018

These data sets are either purged or backed up as follows:

1. Data sets 3 and 5 are purged by the user's process, based on the user's parameters.
2. Data set 6 is purged automatically because it hasn't been updated in the last 30 days.
3. Data set 4 is backed up automatically. The backup occurs because the number of data lines in the stage tables still exceeds the configured **Maximum Data Lines Before Archiving** value.

Related Topics

- [Overview of Maintaining the Stage Tables](#)

Maintain the HCM Data Loader Stage Tables

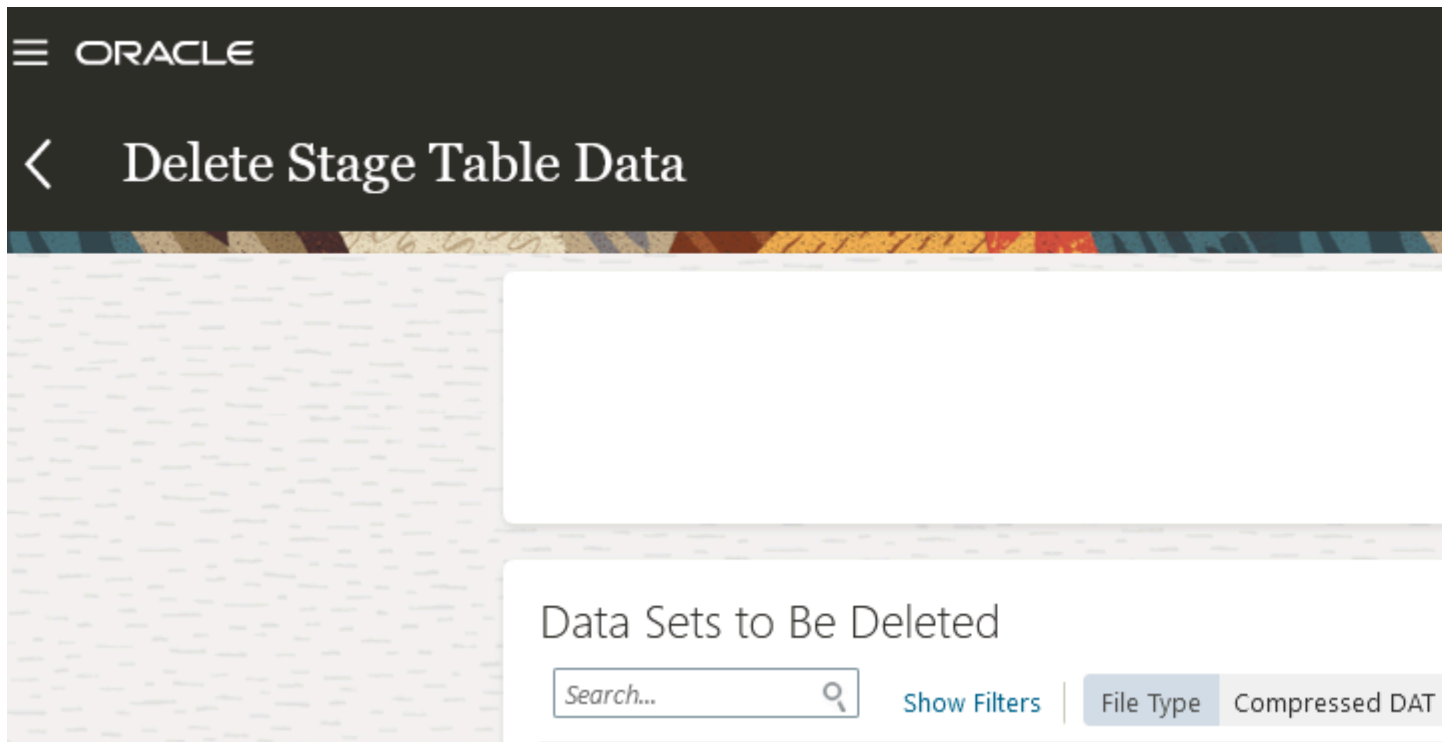
This topic explains how to maintain the stage tables for HCM Data Loader by viewing the stage table status and then either deleting data sets immediately or scheduling a recurring deletion request.

How to Check the Stage Table Status

Follow these steps to check how full the stage tables are:

1. In the **Data Exchange** work area, click **Delete Stage Table Data**.

2. Check the value on the **Stage Table Capacity Used** section.



This percentage helps to know the capacity left on the stage tables. Based on this value, you can decide to delete data sets.

Tip: The stage table capacity information is also available on the **Recent File Loads** task.

Delete a Single Data Set on the Import and Load Data Page

Follow these steps to delete a single data set from the stage tables on the Import and Load Data page:

1. In the **Data Exchange** work area, select **Import and Load Data**.
2. Search for and select the data set that you want to delete.
3. In the Search Results section, click the **Delete** icon for the selected data set.

Note: You can only delete protected data sets if you have the `Delete HCM Data Loader Protected Data-Sets` function security privilege.
4. In the Schedule Request dialog box, set the **Delete Source File** parameter to **Yes** to delete the source file from the Oracle WebCenter Content server.

Note: You can delete the data from the stage tables without also deleting the source file. If you loaded this data set with the **Delete Source File** parameter set to **Yes**, then the source file has already been deleted. In this case, selecting **Yes** in the Schedule Request dialog box has no effect.
5. Click **Submit**. The ID of the deletion process appears.

6. Click **OK** to close the Confirmation dialog box.

Delete Multiple Data Sets on the Delete Stage Table Data Page

You can delete one or more data sets immediately from the stage tables. Alternatively, you can schedule a recurring deletion process. Follow these steps:

1. In the **Data Exchange** work area, select **Delete Stage Table Data**.
2. On the Delete Stage Table Data page, set the filter criteria to identify the data sets you that you want to delete. This table describes the main criteria.

Field	Description
Days Since Last Updated	Specify the number of days since the data set was last updated. For example, to identify any data set not updated in the last week, set this field to 7 . The default value is 30, the maximum value is 60, and the minimum value is 1.
Type	Select any one of the following: <ul style="list-style-type: none">○ Standard○ Extended <p>This field is hidden unless you have the additional privilege (Delete HCM Data Loader Extended Retention Data sets) required to delete extended retention data sets.</p>
Created By	Enter the user name of the user who created the data set.
File Type	Select one of: <ul style="list-style-type: none">○ Compressed DAT file○ Spreadsheet
Import Status	Select a status. For example, set to Success to list data sets that imported successfully.
Load Status	Select a status. For example, set to Success to list data sets that loaded successfully.
Data Set	Specify a particular data set.
Content ID	Enter the Oracle WebCenter Content ID.
Retention Period	By default, only data sets that have the standard retention period will be deleted. When set to Extended , data sets that are held for an extended period in the staging tables are deleted. This field is hidden unless you have the additional privilege (Delete HCM Data Loader Extended Retention Data Sets) required to delete extended retention data sets.

3. To delete all data sets that match the criteria specified:

- a. Click **Actions > Schedule Deletion**.
- b. On the **Parameters** tab of the **Schedule Request** page, the data set criteria from the **Delete Stage Table Data** page appear. You can't edit these values on the **Schedule Request** page.

Tip: To change the data set criteria, edit them on the **Delete Stage Table Data** page, review the results, and click **Actions > Schedule Deletion** again. You must refresh the search results before clicking **Actions > Schedule Deletion**. Otherwise, the parameters aren't updated on the **Parameters** tab of the **Schedule Request** page.

- c. Review and set the **Action** parameter.
- d. Review and set the **Delete Oracle Search Bulk Import Records** parameter.
- e. Review and set the **Initiate Multithreaded Processing** parameter.

Note: This parameter ensures that the elapsed time for deleting large volumes of data is reduced by submitting the **Delete HCM Data Loader Stage Table Data** process as a multi-threaded process. Only if the volume of data to be deleted warrants it will the process be multi-threaded.

- f. Source files for the data sets are deleted from the Oracle WebCenter Content server by default. To retain the source files, set the **Delete Source File** parameter to **No**.

4. In the **Schedule** section,

- a. To submit the delete process immediately, set **Run** to **As soon as possible**.
- b. To schedule a recurring delete process for all data sets that match the specified criteria:
 - i. Set **Run** to **Using a schedule**
 - ii. Specify the frequency of the deletion process. For example, you could schedule the process to run weekly to delete any data set that hasn't been updated in the last 7 days.
 - iii. Enter the **Start Date** and **End Date**

Note: If the start date is tomorrow's date, then the process runs tomorrow and at the specified interval (for example, weekly) until the end date. If the start date is today's date, then the process runs for the first time in one week's time, assuming a weekly frequency.

5. Click **Submit**.

You can review the status of any deletion process by clicking **Actions > Review Processes** on the **Delete Stage Table Data** page. On the **Recent Processes** page:

- To view the delete process parameters such as **Status**, **Submission Time**, **Scheduled Time**, and **Elapsed Time**, action icon for the process and select **Parameters**. If the process created a schedule, then the **Parameters** dialog box also includes the schedule details. Scheduled processes are identified by an icon before the process name.
- You can cancel a schedule for any scheduled process by clicking the action icon for the process and selecting **Cancel Schedule**.
- Recurring deletion requests appear on the **Recent Processes** page with a parent process in Wait status. The parent process starts a child process for each scheduled deletion request. The ID of the parent process appears in the Scheduling Process ID column.
- For deletions involving 500 or fewer data sets, the process log file includes the request parameters and the details of each data set. If more than 500 data sets are deleted, the log file provides the total number of deleted data sets along with an SQL statement to retrieve the details of the purged data.

To access the log file, click the action icon for the process and select **Log**.

Delete Spreadsheet Data Sets

On the Delete Stage Table Data page, your search results include spreadsheet data sets if you set **File Type** to **Spreadsheet**. You can delete these data sets from the HCM Data Loader stage tables. But the spreadsheet data sets remain in the HCM Spreadsheet Data Loader stage tables. So, you can continue to edit and upload these spreadsheet data sets after you have deleted them from the HCM Data Loader stage tables. Such a spreadsheet is reloaded from the HCM Spreadsheet Data Loader stage tables when you next save or upload it. To delete spreadsheet data sets from both sets of stage tables, use the **Delete Spreadsheet Stage Table Data** task.

Related Topics

- [Overview of Maintaining the Stage Tables](#)

Maintain the HCM Spreadsheet Data Loader Stage Tables

This topic explains how to maintain the stage tables for HCM Spreadsheet Data Loader. You can either delete data sets immediately or schedule a recurring deletion request.

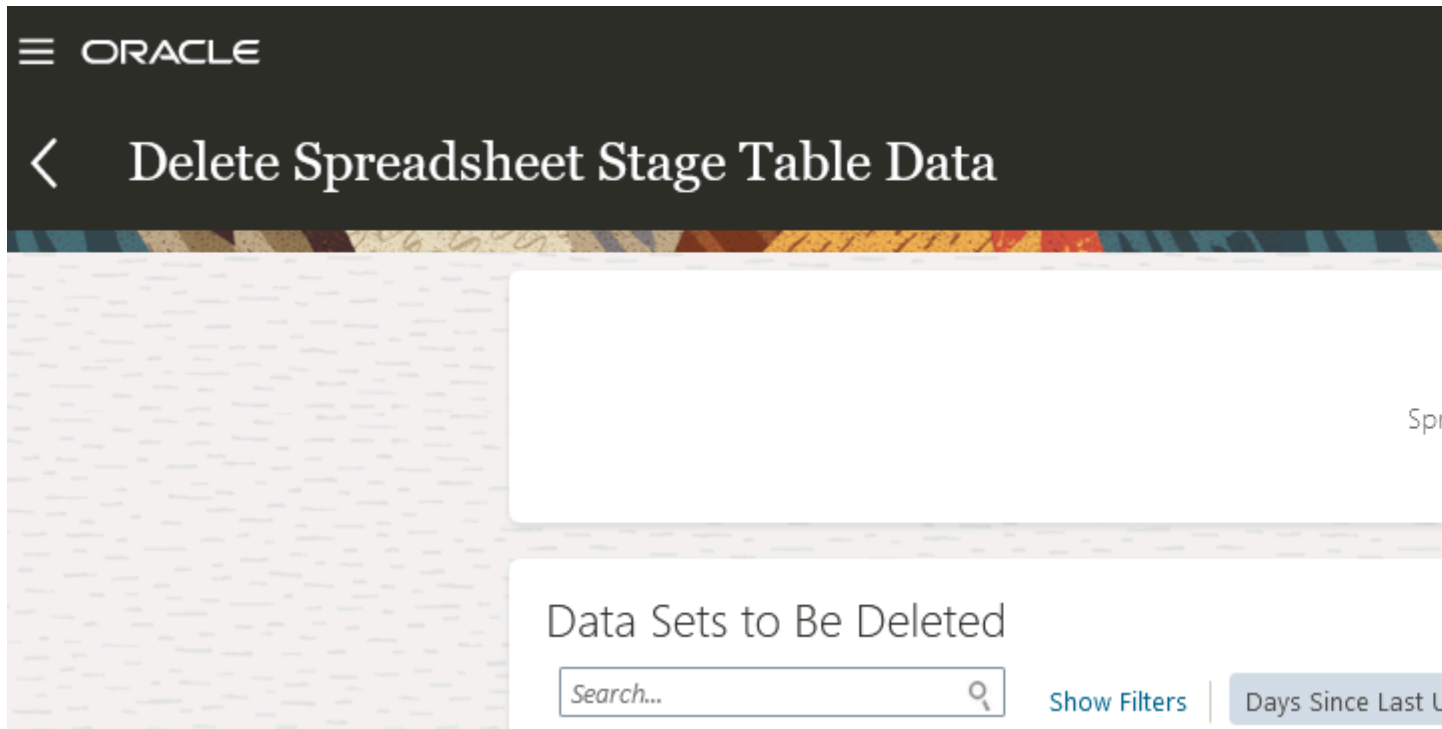
Note: When you delete HCM Spreadsheet Data Loader data sets, they are deleted from all stage tables.

How to Check the Stage Table Status

Follow these steps to check how full the stage tables are:

1. In the **Data Exchange** work area, select **Delete Spreadsheet Stage Table Data**.

2. Check the value on the **Spreadsheet Stage Table Capacity Used** section.



This percentage helps to know the capacity left on the stage tables. Based on this value, you can decide to delete data sets.

Delete Data Sets from the HCM Spreadsheet Data Loader Stage Tables

Follow these steps:

1. In the **Data Exchange** work area, select **Delete Spreadsheet Stage Table Data**.
2. On the **Delete Spreadsheet Stage Table Data** page, set the filter criteria to identify the data sets that you want to delete. This table describes the criteria:

Field	Description
Days Since Last Updated	Specify the number of days since the data set was last updated. For example, to identify any data set not updated in the last week, set this field to 7 . The default value is 30, the maximum value is 60, and the minimum value is 1.
Template	Select an HCM Spreadsheet Data Loader template to identify any data set created in spreadsheets generated from the template.
Data Set	Specify a particular data set. If multiple spreadsheets have been used to load the data set, then they're all listed.

Field	Description
Status	Select one of: <ul style="list-style-type: none">○ Import completed with errors○ Load completed with errors○ Load successful○ Ready to process
Last Updated By	Enter the user name of the user who last updated the data set.
Created By	Enter the user name of the user who created the data set.

3. To delete all data sets that match the criteria specified:

- a. Click **Actions > Schedule Deletion**.
- b. On the **Parameters** tab of the **Schedule Request** page, the data set criteria from the **Delete Spreadsheet Stage Table Data** page appears. You can't edit these values on the **Schedule Request** page.

Tip: To change the data set criteria, edit them on the **Delete Spreadsheet Stage Table Data** page, review the results and click **Actions > Schedule Deletion** again. You must refresh the search results before clicking **Actions > Schedule Deletion**. Otherwise, the parameters won't be updated on the **Parameters** tab of the **Schedule Request** page.

- c. Click **Submit**.

4. To schedule a recurring deletion process for all data sets that match the specified criteria:

- a. Click **Actions > Schedule Deletion** on the **Delete Spreadsheet Stage Table Data** page.
- b. Set **Run** to **Using a schedule**.
- c. Specify the frequency of the deletion process. For example, you could schedule the process to run weekly to delete any data set that hasn't been updated in the last 7 days.
- d. Enter start and end dates.

Note: If the start date is tomorrow's date, then the process runs tomorrow and at the specified frequency (for example, weekly) until the end date. If the start date is today's date, then the process runs for the first time in one week's time, assuming a weekly frequency.

- e. Click **Submit**.

You can review the status of any deletion process by clicking **Actions > Review Processes** on the **Delete Spreadsheet Stage Table Data** page. On the **Processes** page:

- You can see the parameters for any process by clicking the contextual action icon next to the process name. If the process was scheduled, then the **Parameters** dialog box also includes the schedule details. Scheduled processes are identified by an icon in the **Schedule** column.

- Recurring deletion requests appear on the **Processes** page with a parent process in **Wait** status. The parent process starts a child process for each scheduled deletion request. The ID of the parent process appears in the **Scheduling Process ID** column.
- For deletions involving 500 or fewer data sets, the process log file includes the request parameters and the details of each data set. If more than 500 data sets are deleted, the log file provides the total number of deleted data sets along with an SQL statement to retrieve the details of the purged data.

To access the log file, click the action icon for the process and select **Log**.

When you delete data sets from the **HCM Spreadsheet Data Loader** stage tables, they're also deleted from the **HCM Data Loader** stage tables.

Related Topics

- [Overview of Maintaining the Stage Tables](#)

17 Best Practices for HCM Data Loader and HCM Spreadsheet Data Loader

HCM Data Loader Best Practices

This topic highlights some best practices for data loading. Follow these recommendations to ensure successful creation and maintenance of data using HCM Data Loader.

Source Keys

The source key is the recommended key type to use for all implementations. If you supply a source key when you create your data, then you can also update or simply reference that data using source keys.

You can't update some objects if you identify them using only user keys. For example, you can't update most components of the Worker object if you supply only a user key. The attribute that you want to change is often the attribute that's used to identify the record. For example, in the Person Address component, the **AddressLine1** attribute is used both to identify the address to update and to supply the new value. Therefore, you're recommended to use source keys whenever possible. You can also update source keys. You may want to do this when the source key value is auto-generated.

Data Migration

When creating business objects using HCM Data Loader:

- Deliver one object type in each .zip file. For example, create one .zip file for jobs, one for grades, one for workers, and so on. To avoid data-reference errors, you must ensure that you load .zip files in the correct order and fix any errors before loading the next file.
- Purge the stage table data regularly for data sets that you no longer need, especially if you're loading large volumes of data.
- Understand and follow the rules for creating date-effective data.

Tip: HCM Data Loader can load very large volumes of data from a single file. However, do consider the practical challenges, such as error management and the length of the transition period, associated with large data loads. Best practice is to split large loads into logical groupings so that you can extract, transform, and load the data more easily. For example, when loading workers you could load ex-employees separately from current, active employees. If this grouping still leaves you with very large data loads, then you could define subgroups of records based on other significant criteria.

Updating Objects

When updating business objects using HCM Data Loader:

- Understand and follow the rules for maintaining date-effective data. In particular, always include the **SET PURGE_FUTURE_CHANGES N** instruction in the .dat file to ensure that any existing future-dated changes are retained.

- Supply all business-object files in the same .zip file. HCM Data Loader processes them in the correct order. Referenced data is loaded before the data that references it.
- Supply any changed attribute values and a unique reference to the record that you're updating. HCM Data Loader validates every attribute name in the METADATA line.
- Never include both DELETE and MERGE instructions for the same record in the same file. HCM Data Loader doesn't guarantee the order in which the file lines are processed.

Related Topics

- [Maintain the HCM Data Loader Stage Tables](#)
- [Source Keys](#)
- [How You Update Source Keys](#)
- [Overview of Loading Date-Effective Objects](#)

18 Reporting and Diagnostics

Overview of HCM Data Loader Extracts

HCM Extracts provides four extracts for HCM Data Loader, as shown in this table.

Extract Name	Description
HCM Data Loader Data Set Summary	Provides status values and record counts for both a specified data set and each business object in the data set. It reports all error messages and identifies all records affected by each error.
Compensation Changes Extract	For a specified compensation run, extracts assignment, salary, and bonus changes made by the run. This extract is most useful if you're implementing a coexistence scenario.
Integration Object User Key Refresh	Retrieves latest user key values. Run this extract before running the Integration Object User Key Map Extract.
Integration Object User Key Map Extract	Extracts GUIDs, source keys, surrogate IDs, and user keys for a specified business object hierarchy.

The output from all extracts is in XML format by default. You can provide a BI Publisher layout to reformat the output. You can also define options to automate delivery of the output.

Before users can run any of these extracts, you:

1. Copy the predefined extract, specify a delivery option, and compile the formulas for your copy of the extract.
2. Configure your copy of the extract.

Once the extract is configured, users can run it and view the output.

To learn how to create a HCM Extract delivery option to format your output into a HDL file and define a payroll flow pattern to initiate the extract and submit HDL, see the tutorial: [Initiate HCM Data Loader for HCM Extract Generated Files](#)

Related Topics

- [Manage an HCM Data Loader Extract Definition](#)
- [Configure a Copied HCM Data Loader Extract](#)

Setting Up HCM Data Loader Extracts

Manage an HCM Data Loader Extract Definition

HCM Extracts provides four predefined extracts for HCM Data Loader:

- HCM Data Loader Data Set Summary
- Compensation Changes Extract
- Integration Object User Key Refresh
- Integration Object User Key Map Extract

This topic explains how to copy a predefined extract, specify a delivery option, and compile the formulas for the copied extract. You can follow these instructions for any of the HCM Data Loader extracts.

Copy the Extract

Follow these steps to copy an HCM Data Loader extract:

1. In the Data Exchange work area, click **Extract Definitions**.
2. On the Extract Definitions page, search for the extract.
3. Select the extract in the search results and click **Actions > Copy or Rename**.
4. In the Copy Extract Definition dialog box, enter a name for your copy of the extract and click **OK**.

A confirmation message appears when the copy is complete.

Specify a Delivery Option for the Copied Extract

Follow these steps to specify a delivery option:

1. On the Extract Definitions page, search for your copy of the extract.
2. Click the extract name in the search results to display the extract definition on the Extract Definitions page.
3. Click **Deliver**.
4. In the Extract Delivery Options section, click **Add**.
5. Specify a delivery option. To keep the default XML output, set the **Output Type** to **Data**.
6. For the XML delivery option, specify **Delivery Option Name**, **Output Name**, and **Delivery Type** values.

Tip: The values that you must specify vary by output type.

7. Click **Save**.

For more information, see Oracle Fusion HCM Extracts: Delivery Options (document ID 1573099.1) on My Oracle Support at <https://support.oracle.com>.

Compile the Formulas for the Copied Extract

Follow these steps to compile the formulas:

1. On the Extract Definitions page, click **Validate**.
2. In the Extract Execution Tree section, click **Compile All Formula** to compile the formulas.
3. Refresh the table periodically until all formulas are successfully compiled. A check mark means success.
4. Click **Submit**.

Once these steps are complete, perform the **Refine Extracts** task in the Data Exchange work area to configure the copied HCM Data Loader extract.

Related Topics

- [Overview of HCM Data Loader Extracts](#)
- [Configure a Copied HCM Data Loader Extract](#)

Configure a Copied HCM Data Loader Extract

HCM Extracts provides four predefined extracts for HCM Data Loader:

- HCM Data Loader Data Set Summary
- Compensation Changes Extract
- Integration Object User Key Refresh
- Integration Object User Key Map Extract

This topic explains how to configure an extract after copying it, specifying a delivery option, and compiling its formulas. Follow these steps:

1. On the Tasks panel tab of the Data Exchange work area, select **Refine Extracts** to open the Payroll Flow Patterns page.
2. On the Payroll Flow Patterns page, enter the name of your copy of the extract in the **Flow Pattern** field and click **Search**.
3. In the search results, select your extract and click **Edit**.
4. Click the Parameters tab to review the parameter definitions.
5. Edit the **Display**, **Sequence**, **Parameter Basis**, and **Basis Value** for each parameter. You can also edit the parameter names, if required.
6. Save your changes.

Users can now run the extract.

Related Topics

- [HCM Data Loader Data Set Summary Extract Parameters](#)
- [Compensation Changes Extract Parameters](#)
- [Integration Object User Key Refresh and Integration Object User Key Map Extract Parameters](#)

HCM Data Loader Data Set Summary Extract Parameters

This topic describes how to configure the parameters of the HCM Data Loader Data Set Summary extract.

To configure the parameters, you perform the **Refine Extracts** task in the Data Exchange work area.

Configuring the Parameters

This table shows the parameters that you must edit and the required values for the HCM Data Loader Data Set Summary extract. You can ignore parameters not shown here.

Parameter	Display	Sequence	Parameter Basis	Basis Value
Effective Date	No	3	Context Binding	System Date
Start Date	No	9	Not applicable	Not applicable

Parameter	Display	Sequence	Parameter Basis	Basis Value
Content ID	Yes	12	Not applicable	Not applicable
Data Set Name	Yes	13	Not applicable	Not applicable
Process ID	Yes	11	Not applicable	Not applicable
Show Messages	No	14	Constant Bind	Y
Show Stack Trace	No	15	Constant Bind	N

Related Topics

- [Configure a Copied HCM Data Loader Extract](#)
- [Overview of HCM Data Loader Extracts](#)

Compensation Changes Extract Parameters

This topic describes how to configure the parameters of the Compensation Changes Extract for HCM Data Loader.

To configure the parameters, you perform the **Refine Extracts** task in the Data Exchange work area.

Configuring the Parameters

This table shows the parameters that you must edit and the required values for the Compensation Changes Extract. You can ignore parameters not shown here.

Parameter	Display	Sequence	Parameter Basis	Basis Value
Effective Date	No	3	Context Binding	System Date
Start Date	No	9	Not applicable	Not applicable
Extract Target	No	15	Constant Bind	HDL
Period Name	Yes	12	Not applicable	Not applicable
Plan	Yes	11	Not applicable	Not applicable
Process Id	Yes	13	Not applicable	Not applicable
Run Id	Yes	14	Not applicable	Not applicable

Parameter	Display	Sequence	Parameter Basis	Basis Value

Related Topics

- [Configure a Copied HCM Data Loader Extract](#)
- [Overview of HCM Data Loader Extracts](#)

Integration Object User Key Refresh and Integration Object User Key Map Extract Parameters

This topic describes how to configure the parameters of these HCM Data Loader extracts:

- Integration Object User Key Refresh
- Integration Object User Key Map Extract

To configure the parameters, you perform the **Refine Extracts** task in the Data Exchange work area.

Configuring the Parameters

This table shows the parameters that you must edit and the required values for the Integration Object User Key Refresh extract.

Parameter	Display	Sequence	Parameter Basis	Basis Value
Effective Date	Mandatory	3	Context Binding	System Date
Start Date	No	9	Not applicable	Not applicable
Process Configuration Group	Yes	17	Not applicable	Not applicable
Business Object	Yes	76	Not applicable	Not applicable
Component Discriminator	Yes	81	Not applicable	Not applicable
Include Child Components	Yes	91	Not applicable	Not applicable

Note: You're recommended to change the name of the **Business Object** parameter to **Business Object Discriminator**.

This table shows the parameters that you must edit and the required values for the Integration Object User Key Map Extract.

Parameter	Display	Sequence	Parameter Basis	Basis Value
Effective Date	Mandatory	3	Context Binding	System Date
Start Date	No	9	Not applicable	Not applicable
Process Configuration Group	Yes	17	Not applicable	Not applicable
Business Object	Yes	22	Not applicable	Not applicable
Component Discriminator	Yes	23	Not applicable	Not applicable
Include Child Components	Yes	24	Not applicable	Not applicable
Source System Owner	Yes	25	Not applicable	Not applicable
Baseline Only	No	81	Not applicable	Not applicable
Changes Only	No	91	Not applicable	Not applicable
Delete Archive	No	101	Not applicable	Not applicable

Note: You're recommended to change the name of the **Business Object** parameter to **Business Object Discriminator**.

Related Topics

- [Configure a Copied HCM Data Loader Extract](#)
- [Overview of HCM Data Loader Extracts](#)

Running HCM Data Loader Extracts

Submit an HCM Data Loader Extract

HCM Extracts provides four predefined extracts for HCM Data Loader:

- HCM Data Loader Data Set Summary

- Compensation Changes Extract
- Integration Object User Key Refresh
- Integration Object User Key Map Extract

This topic explains how to submit a configured HCM Data Loader extract.

Select the Extract

Follow these steps:

1. On the Tasks panel tab of the Data Exchange work area, select **Submit Extracts** to open the Submit a Process or Report page.
2. Using the query by example fields, find your extract and click **Next**.
The Enter Parameters page opens.
3. In the **Payroll Flow** field, enter a flow name for this run of the extract.
4. Enter the parameter values for your extract.

Enter Parameters for the HCM Data Loader Data Set Summary Extract

This table shows the parameters for the HCM Data Loader Data Set Summary extract. Enter any one of the **Content ID**, **Process ID**, and **Data Set Name** parameters. You can find all of these values for a specific data set on the Import and Load Data page.

Parameter	Description
Content ID	The content ID associated with the data set. This ID was assigned when you uploaded your source file to the Oracle WebCenter Content server.
Data Set Name	The data set name, which is also your source file name.
Process Configuration Group	Leave blank.
Process ID	The process ID of the process that created the data set. This ID was assigned when you submitted your source file for processing by HCM Data Loader.

Enter Parameters for the Compensation Changes Extract

This table shows the parameters of the Compensation Changes Extract. To identify the compensation run to report on, use one of the following parameters or pairs of parameters:

- Plan and Period Name
- Process Id
- Run Id

Parameter	Description
Plan	The compensation plan name

Parameter	Description
Period Name	The period name for the compensation run
Process Id	The process ID for the compensation run
Run Id	The run ID for the compensation run

Enter Parameters for the Integration Object User Key Refresh Extract

This table shows the parameters of the Integration Object User Key Refresh Extract.

Parameter	Description
Business Object	The file discriminator of the business object for which you want to refresh keys. For example, for some or all components of the Worker object, set this parameter to Worker . You can find the file discriminator values on the Business Object Details page for the object in the Data Exchange work area.
Component Discriminator	The file discriminator of the business object component. For example, to refresh keys for the Work Relationship component of the Worker object, set this parameter to WorkRelationship . You can find the file discriminator values on the Business Object Details page for the object in the Data Exchange work area.
Include Child Components	Set to Yes to include the child components of the business object or component in the refresh.
Process Configuration Group	Leave blank.

Enter Parameters for the Integration Object User Key Map Extract

This table shows the parameters of the Integration Object User Key Map Extract.

Parameter	Description
Business Object	The file discriminator of the business object for which you want to extract keys. For example, for some or all components of the Worker object, set this parameter to Worker . You can find the file discriminator values on the Business Object Details page for the object in the Data Exchange work area.
Component Discriminator	The file discriminator of the business object component. For example, to extract keys for the Work Relationship component of the Worker object, set this parameter to WorkRelationship . You can find the file discriminator values on the Business Object Details page for the object in the Data Exchange work area.
Include Child Components	Set to Yes to include the child components of the business object or component in the extract.

Parameter	Description
	Tip: If you're extracting a large number of records, then you're recommended to set this parameter to No and extract each component separately.
Process Configuration Group	Leave blank.
Source System Owner	If required, select a value to extract the keys for a specified source system owner. The extract includes all types of keys, but only for those object instances with this source system owner.

Submit the Extract

Follow these steps:

1. Click **Next** until you reach the Review page.
2. Click **Submit**.

A confirmation message appears when your extract is submitted successfully.

3. Close the Confirmation dialog box.

You can now view the extract output.

To learn how to create a HCM Extract delivery option to format your output into a HDL file and define a payroll flow pattern to initiate the extract and submit HDL, see the tutorial: [Initiate HCM Data Loader for HCM Extract Generated Files](#)

Related Topics

- [Overview of HCM Data Loader Extracts](#)
- [View the Results of an HCM Data Loader Extract](#)

View the Results of an HCM Data Loader Extract

HCM Extracts provides four predefined extracts for HCM Data Loader:

- HCM Data Loader Data Set Summary
- Compensation Changes Extract
- Integration Object User Key Refresh
- Integration Object User Key Map Extract

This topic explains how to view the output from an HCM Data Loader extract. Follow these steps:

1. Click **Navigator > My Client Groups > Data Exchange**.
2. In the Data Exchange work area, click **View Extract Results**.
3. On the View Extract Results page, enter the extract instance name or extract name, submission date range, and its process status for which you want the results.
4. Click **Search**.

5. In the Search Results section, click the extract instance name link for which you want to view the extract result details.
6. On the extract details page, click the Extract Delivery Options tab to download the extract results output file.

To learn how to create a HCM Extract delivery option to format your output into a HDL file and define a payroll flow pattern to initiate the extract and submit HDL, see the tutorial: *Initiate HCM Data Loader for HCM Extract Generated Files*

Related Topics

- [Overview of HCM Data Loader Extracts](#)
- [Submit an HCM Data Loader Extract](#)

Diagnostic Tests for HCM Data Loader

HCM Data Loader Error Analysis Report

The HCM Data Loader Error Analysis Report is a diagnostic test. It provides technical information that can help to resolve data-loading errors. Run this diagnostic test when error messages don't supply enough information for you to identify a solution.

You can send the test output to Oracle Support, if necessary.

To run the HCM Data Loader Error Analysis Report test, select **Settings and Actions > Run Diagnostic Tests**.

Diagnostic Test Parameters

Content ID

A unique reference to a single data set. You can find the content ID in the **Content ID** column of the Search Results table on the Import and Load Data page. You must specify either the content ID or the process ID to identify the data set.

Process ID

A unique reference to a single data set. You can find the process ID in the **Process ID** column of the Search Results table on the Import and Load Data page. You must specify either the content ID or the process ID to identify the data set.

Business Object Name

The name of a business object in the data set, such as Worker or Job. You can find the business object name in the **Business Object** column of the Details table on the Import and Load Data page. This parameter is optional, but you're recommended to supply it if your data set contains multiple business objects.

Detailed Mode

Set this parameter to **Y** to include the details of any reported errors in the output, including message stack trace for the records in error.

Diagnostic Test Results

The diagnostic test output is in user-readable HTML format. It provides:

- Status information

- Details of every process submitted to process your data set
- All errors raised for each business object in the data set
- Message stack trace and nonsensitive attribute values for the records in error, if you set the **Detailed Mode** parameter to **Y**

No sensitive data appears in the test results.

Related Topics

- [Run the HCM Data Loader Error Analysis Report](#)

Run the HCM Data Loader Error Analysis Report

The HCM Data Loader Error Analysis Report is a diagnostic test that provides technical information about data loads to help you resolve errors. To submit this diagnostic test, you must have access to the Diagnostic Dashboard.



Follow these steps:

1. Select **Settings and Actions > Run Diagnostic Tests** to open the Diagnostic Dashboard.
2. In the Search for Tests section of the Diagnostic Dashboard, enter **HCM Data Loader Error Analysis Report** in the **Test Name** field and click **Search**.
3. In the search results, select the check box next to the test name and click **Add to Run**. The test appears in the Choose Tests to Run and Supply Inputs section of the page.
4. To review the parameters, click the **Click to Supply or Edit Input Parameters** icon in the **Status** column.
5. In the Input Parameters dialog box, select parameters to include in this test run and supply values for the selected parameters.

Note: Use the parameters to restrict the output from the test to those errors that need investigation. For example, don't run the test for an entire data set if the errors relate to a single business object.
6. Click **OK**.
7. If required, enter a name for the test run in the **Run Name** field. The name can help you to locate your results easily. If you leave the field blank, then a name is generated.
8. Click **Run** to run the test and click **OK** to close the confirmation dialog box.
9. In the Diagnostic Test Run Status section, click the **Display Latest Test Run Status Information** icon to see the results of this test.
10. Expand the test hierarchy for your run to see the results. Click the **Report** icon to open the report.

You can save the output and attach it to a service request if required.

Related Topics

- [HCM Data Loader Error Analysis Report](#)

Diagnostic Tests for HCM Spreadsheet Data Loader

HCM Spreadsheet Data Loader Diagnostic Report

The HCM Spreadsheet Data Loader Diagnostic Report provides a detailed analysis of a spreadsheet template or associated data sets to help you resolve template errors. Run this report when error messages don't supply enough information for you to resolve any errors.

You can send the report output to Oracle Support, if necessary.

To run the HCM Spreadsheet Data Loader Diagnostic Report, select **Settings and Actions > Run Diagnostic Tests**.

Diagnostic Test Parameters

Data Set Name

To report on a specific data set, enter its name. You must provide a value for either the **Data Set Name** or the **Template Name** parameter.

Template Name

To report on a spreadsheet template, enter its name. You must provide a value for either the **Data Set Name** or the **Template Name** parameter.

Number of Data Sets to Include

Enter the number of data sets most recently created in spreadsheets generated from the spreadsheet template to include in the report. The default value is **5**. This parameter applies only when you also specify a value for the **Template Name** parameter.

Include Error Details

Set this parameter to **Y** to provide details, including stack trace, of the most frequent import and load errors in the report output.

Diagnostic Report Contents

The report is in user-readable HTML format. It includes the following sections:

Report Section	Section Content
Parameters	Lists all report parameters with the values that you supplied
Template Details	The values held for the template definition
Template Business Objects	Details of the business objects that the template supports
Template Attribute Details	Details of the attributes selected in the template, regardless of whether they're visible in generated spreadsheets

Report Section	Section Content
Map Table	Details of the template map
Attribute Map Table	The business object map information
Flexfield Contexts	Flexfield contexts in the spreadsheet template
Flexfield Attributes	Flexfield attributes in the spreadsheet template
Data set Name: <data set name>	One section for each data set in the report

The Data Set Name: <data set name> section includes the following information for each data set in the report:

- A summary of all errors raised for the data set
- The status and record counts
- Details of the **Import and Load** process that created the data set
- All parameters to the **Import and Load** process and its child processes
- Detailed information about each business object in the spreadsheet, including:
 - Object status
 - Details of the import and load processes
 - SET commands
 - The 25 most frequent import errors
 - The 25 most frequent load errors
 - Object and component counts

Related Topics

- [Run the HCM Spreadsheet Data Loader Diagnostic Report](#)

Run the HCM Spreadsheet Data Loader Diagnostic Report

The HCM Spreadsheet Data Loader Diagnostic Report is a diagnostic test that enables you to analyze problems in custom spreadsheet templates. To run this report, you must have access to the Diagnostic Dashboard.

Follow these steps:

1. Select **Settings and Actions > Run Diagnostic Tests** to open the Diagnostic Dashboard.
2. In the Search for Tests section of the Diagnostic Dashboard, enter **HCM Spreadsheet Data Loader Diagnostic Report** in the **Test Name** field and click **Search**.
3. In the search results, select the check box next to the test name and click **Add to Run**. The test appears in the Choose Tests to Run and Supply Inputs section of the page.
4. To review the parameters, click the **Click to Supply or Edit Input Parameters** icon in the **Status** column.

5. In the Input Parameters dialog box, select parameters to include in this test run and supply values for the selected parameters.

Note: Use the parameters to restrict the output from the test to those errors that need investigation. For example, don't run the test for multiple data sets if the errors relate to a single data set.

6. Click **OK**.
7. If required, enter a name for the test run in the **Run Name** field. The name can help you to locate your results easily. If you leave the field blank, then a name is generated.
8. Click **Run** to run the test and click **OK** to close the confirmation dialog box.
9. In the Diagnostic Test Run Status section, click the **Display Latest Test Run Status Information** icon to see the results of this test.
10. Expand the test hierarchy for your run to see the results. Click the **Report** icon to open the report.

You can save the output and attach it to a service request if required.

Related Topics

- [HCM Spreadsheet Data Loader Diagnostic Report](#)

Overview of OTBI Reporting

The following folders are available in the **HCM Integrations Real Time** subject area. Use these folders to create your reports within OTBI and BI Publisher.

HCM Data Loader

The HCM Data Loader area contains folders relating to HCM Data Loader (HDL) and HCM Spreadsheet Data Loader (HSDL) data sets and processes and HSDL templates.

Folder Name	Description
Data Sets	HCM Data Loader and HCM Spreadsheet Data Loader data set details, such as created by, last update date, and import and load statuses.
Data Set Business Objects	Status information for the business objects processed within HCM Data Loader and HCM Spreadsheet Data Loader data sets.
Data Set Business Object Parameters	Details of the overridden parameters for a business object within a data set.
Data Set Post Processes	Details of the automatically submitted processes for your data set.
Data Set Business Object Post Processes	Details of post processes that are automatically submitted after data for the business object has been successfully loaded.
Facts	Numeric facts for a business object's files, lines, and imported objects.
Imported Objects	Data related to the objects imported for a business object within a data set. It includes the identifying key, reference attribute values with the line, and object messages. This data is only available while the data set exists in the stage tables.

Folder Name	Description
Processes	Information relating to the processes used by a data set.
Process Parameters	Parameter values used when processing a data set.
Spreadsheet Data Sets	HCM Spreadsheet Data Loader data sets details and facts.
Spreadsheet Templates	HCM Spreadsheet Data Loader template information.
Spreadsheet Template Roles	Roles associated with HCM Spreadsheet Data Loader templates.
Spreadsheet Template Parameters	Overridden parameters for HCM Spreadsheet Data Loader templates.

HCM Data Disposal

The HCM Data Disposal area contains folders relating to the Remove Person Information functionality.

Folder Name	Description
Remove Person Information	Information relating to Remove Person Information templates and processes.
Remove Person Information Facts	Number of processes generated by the Remove Person Information functionality.

19 Transferring Encrypted Data

Set up Encryption for File Transfer

You use encryption keys to encrypt files for secure transfer between Oracle HCM Cloud and your own servers through the Oracle WebCenter Content server. This PGP-based encryption support is available for secure file transfer using HCM Data Loader and HCM Extracts.

The process for inbound files (into Oracle HCM Cloud) is:

1. You encrypt files using the Oracle HCM Cloud public key.
2. The data-loading process decrypts files using the Oracle HCM Cloud private key.

The process for outbound files (generated by Oracle HCM Cloud) is:

1. Oracle HCM Cloud encrypts files using your public key.
2. You decrypt files using your private key.

Therefore, before you can encrypt or decrypt files, you must:

1. Import your public key to Oracle HCM Cloud.
2. Generate the PGP encryption key pair and download the Oracle HCM Cloud public key.

You can also sign the files, as follows:

- Outbound files are signed using the HCM Cloud private key. You verify these files using the HCM Cloud public key.
- Inbound files are signed using your private key. The data-loading process verifies inbound files using your public key.

This topic describes how to set up encryption, decryption, and signing of files.

Encryption and Signature Keys

This table shows you the keys that are used for encryption, decryption, signing, and verification in each supported encryption mode.

Encryption Mode	Encryption Key	Decryption Key	Signing Key	Verification Key
Outbound PGP Signed	customer-key_pub	customer-key_priv	fusion-key_priv	fusion-key_pub
Outbound PGP Unsigned	customer-key_pub	customer-key_priv	N/A	N/A
Inbound PGP Signed	fusion-key_pub	fusion-key_priv	customer-key_priv	customer-key_pub
Inbound PGP Unsigned	fusion-key_pub	fusion-key_priv	N/A	N/A

Importing Your Public Key

You use your public key (customer-key_pub) for encrypting outbound files. You can decrypt the files using your private key (customer-key_priv). If you also want outbound files to be signed, then use the HCM Cloud private key (fusion-key_priv) for signing. You can verify signed outbound files using the HCM Cloud public key (fusion-key_pub).

To import the customer public key:

1. Sign in to Oracle HCM Cloud with the IT Security Manager job role or privileges.
2. Select **Navigator > Tools > Security Console** to open the Security Console.
3. Click the Certificates tab to open the Certificates page.
4. Click **Import** to open the Import page.
5. Set **Certificate Type** to **PGP**.
6. In the **Alias** field, enter a name to uniquely identify your key.
7. Click **Browse** to identify the location of the customer public key.
8. Click **Import and Close** to import the public key into the Oracle HCM Cloud keystore.

Your public key now appears on the Security Console Certificates page.

Generating the PGP Encryption Key Pair

You generate the PGP key pair on the Security Console. You download the public key to encrypt files that are inbound into HCM Cloud (for example, input data files for HCM Data Loader). To sign these inbound files, you can use your private key (customer-key_priv), which is verified using your public key (customer-key_pub) in Oracle HCM Cloud. You must have imported the customer public key.

To generate the PGP Encryption Key Pair:

1. Sign in to Oracle HCM Cloud with the IT Security Manager job role or privileges.
2. Select **Navigator > Tools > Security Console** to open the Security Console.
3. Click the Certificates tab to open the Certificates page.
4. Click **Generate** to open the **Generate** dialog box.
5. In the **Generate** dialog box, set **Certificate Type** to **PGP**.
6. In the **Alias** field, enter **fusion-key**.

Note: You must enter **fusion-key** in this field. Otherwise, the encryption APIs can't use this key to decrypt all encrypted inbound files.

7. In the **Passphrase** field, enter a passphrase for the private key. This passphrase is needed when you edit, delete, or download the private key.

Note: If you forget the passphrase, then you may have to raise a service request for help to delete the private key. Once the old key is deleted, you can generate a new key using the process described here.

8. In the **Key Algorithm** field, select **RSA**.
9. In the **Key Length** field, select either **1024** or **2048**.
10. Click **Save and Close**. The fusion-key pair is generated and ready for download. You can see the fusion-key pair on the Certificates page of the Security Console.
11. In the **Status** actions for the fusion-key pair on the Certificates page, select **Export > Public key**. Save the HCM Cloud public key (**fusion-key_pub.asc**) to your desktop. Use the downloaded key to encrypt files that are inbound to Oracle HCM Cloud.

Encrypt and Upload Files Automatically

Encrypt files of data with PGP encryption and transfer them automatically between your servers and Oracle WebCenter Content using APIs and web services.

You write programs to collect the encrypted files from your file server. You then place them on the Oracle WebCenter Content server and call a data loader to decrypt and load the data to Oracle HCM Cloud.

Note: You must set up your encryption keys before you perform these tasks.

Supported Encryption Algorithms

Oracle HCM Cloud supports the following encryption algorithms. You must ensure that you use only supported encryption algorithms.

- Cipher: AES-128, Blowfish, CAST5, 3DES, AES-256

Note: These cipher algorithms aren't supported: Twofish, IDEA, AES-192

- Compression: bzip2, zlib, .zip, uncompressed
- Hash: SHA-1, SHA-256, SHA-224, SHA-512, MD5, SHA-384, RIPEMD-160

Encrypting Files

This section provides the commands to encrypt files in Microsoft Windows and Linux environments using the GnuPG encryption tool. For other tools and platforms, work with your suppliers to find the necessary commands for setting the cipher algorithm.

- Gpg4win, the official GnuPG distribution for Microsoft Windows, provides both a command-line interface and a graphical user interface for encryption, decryption, signing, and verification. For encryption, use the command-line interface. You can find Gpg4win here: <https://www.gpg4win.org/about.html>.
- You can download GnuPG for Linux from various sources, depending on the Linux distribution that you're using. Commonly used GnuPG versions can be found here: <https://www.gnupg.org/index.html>.

After installing the Gpg4win or GnuPG tool, follow these steps to encrypt or encrypt and sign a file:

1. Import the HCM Cloud public key (downloaded from the Security Console) using this command at the command prompt:

```
gpg --import <fusion public key file>
```

2. Perform one of these steps.

- To encrypt a file without signing, use this command:

```
gpg --cipher-algo <one_of_the_supported_algorithms> -r fusion-key --encrypt <inbound_file_name>
```

- To both encrypt and sign a file, use this command:

```
gpg --cipher-algo <one_of_the_supported_algorithms> -r fusion-key -se <inbound_file_name>
```

Note: When signing files, ensure that your private key is imported into the keystore that's used for signing.

Loading Encrypted Files

Perform the following steps to load encrypted files to Oracle HCM Cloud from the Oracle WebCenter Content server.

1. Write programs to send your encrypted files to Oracle WebCenter Content, using the Oracle WebCenter Content Web Services. If your home page is: **https://Hostname/homePage/faces/AtkHomePageWelcome**, then the Oracle WebCenterContent Server WSDL is: **https://Hostname/idcws/GenericSoap?wsdl**.
2. Call the loader program to pass the encryption parameter with other required parameters. The loaderIntegrationService uses the submitEncryptedBatch method, which has an additional parameter named encryptType. This parameter has the following values, which are defined in the ORA_HRC_FILE_ENCRYPT_TYPE lookup type:
 - NONE
 - PGPSIGNED
 - PGPUNSIGNED

Related Topics

- [Set up Encryption for File Transfer](#)