# Contents

## Preface

## 1 Overview

- Overview of Configuration
  - 1
- Configurations and Extensions
  - 2
- Personalization
  - 6
- Context Layers
  - 6
- Business Process Models
  - 15

## 2 Configuration Life Cycle

- Overview of Configuration Life Cycle
  - 17
- Configurations and Extensions
  - 18
- Sandboxes
  - 19
- Migration
  - 34
- Configuration Management
  - 46

## 3 Page Modification

- Overview of Page Modification
  - 49
- Overview of Using Page Composer
  - 49
- Modify Page Content Using Page Composer
  - 52
- Configure the Global Page Template
  - 60
- Saved Search Management
  - 61
- Infolet Management
  - 64
- Configure Infotiles on a Page
  - 68
- New Page Creation
  - 69
7 Configuration of Home Page and Navigation

Overview of Configuring Home Page and Navigation 197
Configure Navigation 198
Configure Home Page 206
Deep Links 209

8 Help Content Management

How You Manage Different Types of Help 211
Manage Help for Fields and Other UI Elements 212
Why can't I see the Manage Help Content or Edit Getting Started link? 212
Help Windows 212
Getting Started Work Area 214
Manage All Added Help Content 215
Page or Section Values 216
Copy Added Help for Migration 217
Preface

This preface introduces information sources that can help you use the application.

Using Oracle Applications

Help

Use help icons ? to access help in the application. 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. Not all pages have help icons. You can also access the Oracle Help Center to find guides and videos.

Watch: This video tutorial shows you how to find and use help.

You can also read about it instead.

Additional Resources

- Community: Use Oracle Cloud Customer Connect to get information from experts at Oracle, the partner community, and other users.

- Training: Take courses on Oracle Cloud from Oracle University.

Conventions

The following table explains the text conventions used in this guide.

<table>
<thead>
<tr>
<th>Convention</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>boldface</td>
<td>Boldface type indicates user interface elements, navigation paths, or values you enter or select.</td>
</tr>
<tr>
<td>monospace</td>
<td>Monospace type indicates file, folder, and directory names, code examples, commands, and URLs.</td>
</tr>
<tr>
<td>&gt;</td>
<td>Greater than symbol separates elements in a navigation path.</td>
</tr>
</tbody>
</table>
Documentation Accessibility

For information about Oracle's commitment to accessibility, visit the Oracle Accessibility Program website. Videos included in this guide are provided as a media alternative for text-based help topics also available in this guide.

Contacting Oracle

Access to Oracle Support

Oracle customers that have purchased support have access to electronic support through My Oracle Support. For information, visit My Oracle Support or visit Accessible Oracle Support if you are hearing impaired.

Comments and Suggestions

Please give us feedback about Oracle Applications Help and guides! You can send an e-mail to: oracle_fusion_applications_help_ww_grp@oracle.com.
Overview

Overview of Configuration

Oracle applications by default provide robust functionality, tailored to support most of the business requirements an organization could have. But you can still make changes to your application to best fit your specific business or personal needs.

Types of Changes

You can make 2 types of changes, based on who’s making the change and for whom.

- Configurations: These are changes made by administrators, and these changes affect many users. For example, you hide a page entry in the springboard for specific job roles, or create a new page for all users.
- Personalization: These are changes made by individual users, and they affect only the users that made them. For example, you change the column width of a table for yourself.

Here's a visual representation of how changes are categorized.

An administrator can't make personalizations for any specific user other than themselves. Personalizations are also limited to certain types of changes such as springboard and infolet personalizations, and resizing columns and tables.

Configurations are preserved when the application is updated to a newer release.
What You Can Change

You can configure many aspects of the application, for example the user interface, business intelligence, and data model.

The application is built using a common data model. So, when you make an application change in one area, that change is available to all objects in the application. For example, if you add an attribute to an object, you can easily add that attribute to these related artifacts:

- Web-based view page
- Associated mobile page
- Associated reports

Generally, you use the same tools and processes to configure all applications. For more information on configuring business intelligence, see the Creating and Editing Analytics and Reports guides relevant to your products.

Related Topics
- Overview of Configuration Life Cycle

Configurations and Extensions

Examples of Configurations and Extensions

You can make configurations and extensions using browser-based composers and other tools. All users or a subset of users can view and use these configurations and extensions. If your role has an administrative privilege, you can access most configuration tools to modify the user interface (UI), create and modify objects, and so on. Some configuration tools, such as Application Composer, are available only for specific product families.

Modifying the UI

To modify the UI, use:

- The User Interface Text tool to edit text that appears on multiple pages. For example, you can change the term, "buyer" to "customer" if that is your preferred term, and the change affects all pages where the term is displayed.
- The Appearance work area to change the look and feel of the application pages.
- The Announcements work area to create, edit, and delete announcements displayed on the home page.
- Page Composer to configure application pages for other users. For example, you can:
  - Add fields
  - Add validation
  - Change default content
  - Rearrange regions
  - Add external content
Save queries

Tip: In Page Composer, you can make changes using the WYSIWYG view. However, in some cases, you can also use the Source view.

Configuring Navigation
Use the Structure work area to configure the Navigator and springboard. On the Navigator, select Configuration > Structure.

Adding User-Defined Attributes to Business Components Using Flexfields
Most business components, except those in Oracle Engagement Cloud products, support using flexfields to add attributes to objects. Use flexfields to create your own attributes without programming. A flexfield captures data that is related to a specific purpose, such as information about job positions or inventory items. Each attribute is a segment of a flexfield, and corresponds to a reserved column in the application database.

Modifying Reports and Analytics
Predefined analyses, dashboards, and reports help in meeting business intelligence requirements. You can modify them to fit specific business needs, for example, change the layout.
For more information, see the Creating and Administering Analytics and Reports guides relevant to your products.

Managing Help
Use the Manage Help Content page to:
- Add and edit help files in the application Help
- Determine which help files to show in specific help windows

You can open the Manage Help Content page from any help window, or from the help site.

Note: You must have the appropriate job roles to add and edit help.

Related Topics
- Overview of Flexfields

Tools for Configurations and Extensions
You can configure and extend your application to suit your business needs.
Choose an appropriate tool based on the types of configurations and extensions to make, such as:
- Page modifications
- Branding modifications
- Object modifications
- Security modifications
- Business intelligence modifications
- Help content management

Note: The following tables present only the key tasks for application changes, not all tasks.
## Page Modifications

This table shows some types of modifications that you can make to pages, and the corresponding tools to use. You can modify only certain pages in Page Composer.

<table>
<thead>
<tr>
<th>Modification Task</th>
<th>Tool</th>
</tr>
</thead>
<tbody>
<tr>
<td>Add, move, delete, show, or hide components on a page</td>
<td>Page Composer</td>
</tr>
<tr>
<td>Change a page layout</td>
<td>Page Composer</td>
</tr>
<tr>
<td>Create a site-level search for all users</td>
<td>Page Composer</td>
</tr>
<tr>
<td>Change a page title</td>
<td>Structure</td>
</tr>
<tr>
<td>Modify dialog box content</td>
<td>Page Composer</td>
</tr>
<tr>
<td>Modify attributes for a flexfield on a page</td>
<td>Page Composer</td>
</tr>
<tr>
<td>Change properties for user interface (UI) components on a standard page</td>
<td>Page Composer</td>
</tr>
<tr>
<td>Configure the UI Shell template</td>
<td>Page Composer</td>
</tr>
<tr>
<td>Update a text string wherever it appears across all pages</td>
<td>User Interface Text</td>
</tr>
<tr>
<td>Change the look and feel of application pages</td>
<td>Appearance page</td>
</tr>
<tr>
<td>Change the announcements on the home page</td>
<td>Announcements page</td>
</tr>
</tbody>
</table>

## Branding Modifications

This table shows some types of modifications that you can make to use your own branding logo, and the corresponding tools to use.

<table>
<thead>
<tr>
<th>Modification Task</th>
<th>Tool</th>
</tr>
</thead>
<tbody>
<tr>
<td>Configure the UI Shell template</td>
<td>Page Composer</td>
</tr>
</tbody>
</table>
Object Modifications
This table shows some types of modifications that you can make to objects, and the corresponding tools to use.

<table>
<thead>
<tr>
<th>Modification Task</th>
<th>Tool</th>
</tr>
</thead>
<tbody>
<tr>
<td>Change the logo and application name in the UI</td>
<td>Appearance page</td>
</tr>
<tr>
<td>Change report layouts</td>
<td>Layout editor in the BI application or external applications such as Microsoft Word</td>
</tr>
</tbody>
</table>

Security Modifications
This table shows a security modification that you can make to objects, and the corresponding tool to use.

<table>
<thead>
<tr>
<th>Modification Task</th>
<th>Tool</th>
</tr>
</thead>
<tbody>
<tr>
<td>Add data security to a custom object</td>
<td>Setup and Maintenance work area</td>
</tr>
</tbody>
</table>

Business Intelligence Modifications
This table shows some types of modifications that you can make to business intelligence (BI) *analytics* and *reports*, and the corresponding tools to use. For more information, see the Creating and Administering Analytics and Reports guides relevant to your products.

<table>
<thead>
<tr>
<th>Modification Task</th>
<th>Tool</th>
</tr>
</thead>
<tbody>
<tr>
<td>Create report layout</td>
<td>Layout editor in the BI application or external applications such as Microsoft Word</td>
</tr>
<tr>
<td>Change report layouts</td>
<td>Layout editor in the BI application or external applications such as Microsoft Word</td>
</tr>
<tr>
<td>Create a report</td>
<td>The BI application</td>
</tr>
<tr>
<td>Modify analyses</td>
<td>Reports and Analytics work area or the BI application</td>
</tr>
</tbody>
</table>
Help Content Management
This table shows some types of changes that you can make to help, and the corresponding tools to use.

<table>
<thead>
<tr>
<th>Modification Task</th>
<th>Tool</th>
</tr>
</thead>
<tbody>
<tr>
<td>Modify text that is displayed when the user hovers over a button, link, icon, or tab title</td>
<td>Page Composer</td>
</tr>
<tr>
<td>Manage help files and determine the help links to show on help windows</td>
<td>Help windows or Applications Help</td>
</tr>
<tr>
<td>Simultaneously replace multiple occurrences of a word or phrase that appear in the embedded help</td>
<td>User Interface Text</td>
</tr>
</tbody>
</table>

Some product families have additional tools available for application changes, for example, Application Composer and Business Process Composer.

Related Topics
- Overview of Flexfields

Personalization

*Personalization* refers to the changes that every user of the application can make to certain artifacts in the user interface (UI) at run time.

*Note:* Personalization changes remain for a user each time the user signs in to the application.

Personalization includes:
- Changes based on how you use the UI, such as changing the width of a column in a table
- Changes that you select to save, such as search parameters
- Changes you make to the springboard and infolets
Context Layers

Overview of Context Layers

You can use context layers to configure pages for specific sets of users. All you need to do is make sure that your sandbox or configuration tool is set to the correct context before you make your changes. If you have enabled the Unified Sandboxes feature, you set the context when you create your sandbox. If you haven’t, you can select the layer from the configuration tools you want to use. If a tool doesn’t have the option to set a context layer, then its changes are made to the Site layer.

Note: Changes made using the User Interface Text tool are made in all layers, and not just the site layer.

Available Layers

Different application families have different context layers. Every application has these layers:

- **Site**: Changes made in this layer affect all users of the application.
- **User**: Changes made in this layer affect just one specific user. But, you can’t use this layer to make changes for other users. Personalizations are stored in this layer, and users can make personalizations only for themselves. An administrator can’t make a personalization for another user.

You can configure different layers that are available for different application families:

- **Customer Relationship Management**
  - Site
  - External or Internal
  - Job Role
- **Human Capital Management**
  - Site
  - Country
  - Organization
  - Time Card Layout
- **Others**
  - Site

Layer Rules

Layers exist in a hierarchy. The lower a level is in the hierarchy, the more specific its context is. Layers at lower levels exist within the scope of the layers above them. This means that the value is set for a context layer within the scope of the context value for the layers above it. Let's say you’re configuring a page for the developer job role. When you select Job Role as the context level and set the value as Developer, you must also specify a value for the External or Internal layer.
An object or a page can have multiple configurations at the same time. But this is possible only when at least one of these conditions are met:

- Configurations are in different layers
- Configurations are in the same layer, but the layers have different context values

When a user requests an object or page, configurations at the lowest layer applicable to them are given precedence. Let's say you added three columns to the table. You then configured these columns for each context layer as follows:

1. At the Site layer, you added three columns to the Sales table, namely, Promotion Name, Sales Points and Partner Name.
2. For the external developer, the Sales Points column is hidden.
3. For the internal sales role, the Sales Points column is not hidden.

Liam, who's an internal salesperson, personalizes this page for himself and hides the Sales Points column.

This is how these columns appear for different users:

<table>
<thead>
<tr>
<th>Users</th>
<th>Columns</th>
</tr>
</thead>
<tbody>
<tr>
<td>Internal sales role</td>
<td>Sales Points is not hidden</td>
</tr>
<tr>
<td>External developer</td>
<td>Sales Points is hidden</td>
</tr>
<tr>
<td>Liam</td>
<td>Sales Points is hidden</td>
</tr>
</tbody>
</table>
Here’s a visual representation of these configurations in different context layers and values.

Examples of Working with Context Layers

Here are a couple of scenarios where you use context layers to make sure that the appropriate configurations and personalizations are available to the appropriate users. For instance, job role is a layer. You can configure a page to make certain changes visible to only a specific job role, such as a sales representative.
An Administrator Configures a Page for Sales Representatives

Let’s say you want to remove the Export button from the Sales home page. But you want to remove it only for sales representatives. Here’s what you do before you make your changes:

- Make sure you have the role you’re trying to make changes for. If you don’t, your security administrator can help you with that.
- Create and activate a sandbox.
  - If you have enabled the Unified Sandboxes feature, select Page Composer as a configuration tool when you create your sandbox, and set the context layer as Job Role with the value, Sales Representative.
  - If you haven’t enabled Unified Sandboxes, set the context layer as Job Role with the value, Sales Representative in Page Composer before you make your changes.

Once all these conditions are met, you can remove the Export button from your page. When you make this change, the configuration engine in Oracle Metadata Services generates an XML, and then stores it in the Oracle Metadata Services repository. So, the original file for the page remains untouched.

When a sales representative requests access to the Sales home page, the configuration engine checks the repository for XML files that satisfy these two conditions:

- Does the file match the requested artifact, in this case, the Sales home page?
- Does the file match the active context, in this case, the Sales Representative job role?

The configuration engine also looks for additional XML files with personalizations made by the specific sales representative who requested access to the Sales home page. But for now, let’s assume our sales representative hasn’t made any personalizations. Once the configuration engine finds an XML file that satisfies both these conditions, that XML file is then layered over the base artifact. In this scenario, the XML file removes the Export button from your sales representative’s page.
Here’s an image that shows you how this configuration is done.

A Recruiter Personalizes a Page
Let’s say you added three new page entries A, B, and C to the springboard on the home page. You added page entries A and B at the Site layer for all users and page entry C at the Job Role layer for just recruiters. But Liam, who is a recruiter, doesn’t use page entry B all that much, and wants to hide it from her springboard. Liam can do this by personalizing her springboard to show only page entries A and C. Personalizations are done in the user layer, and are applicable only to the user who made them.
To hide page entry B, Liam goes to the springboard and clicks the **Personalize Springboard** icon. She then deselects page entry B and clicks **OK**. When she does this, an XML file is generated for Liam’s user layer with this change.

The next time Liam accesses your home page, the configuration engine retrieves three XML files:

- The file for the changes you made at the **Site** layer
- The file for the changes you made at the **Job Role** layer
- The file for the personalization changes Liam made

The files are always applied in the order of their decreasing scope. The file for the context with the largest scope is applied first, and the file for the context with the most specific scope is applied last. In this scenario, this is what happens when each file is applied.

- The first file adds page entries A and B to the springboard for everyone.
- The second file adds page entry C since Liam is a recruiter.
- The third file removes B from the springboard since Liam chose to hide it.

When a different recruiter accesses this page, only the XML files for the Site and Job Role layer are applied. This user has A, B, and C on the springboard.
Here are two diagrams that show you how this personalization flow works. In the first image, you as an administrator add page entries A and B to the home page at the **Site** layer. You add page entry C at the **Job Role** layer for recruiters. The image also shows Liam hiding B.

The next image shows Liam and another recruiter accessing the same home page. The numbers and letters show the sequence of the steps.
Related Topics

- Autoprovisioning
- How do I provision roles to users
- How can I personalize the page entries on my home page
- Personalize Infolets
Example of Using Context Layers in Application Changes

While making application changes, when you use the dialog box to select the context layer, you can also include lower layers to view the application changes from those layers.

The following scenarios explain what happens based on your selected layers. For these examples, the available layers are Site, Country, and Job Role.

What You See While Making Application Changes

Suppose you choose to:

- Edit the Job Role layer and select Sales Representative as the value for that layer
- Include the Country layer and select France as the value

Note: The Site layer is automatically included because it applies to everyone.

While modifying pages in Page Composer, you see changes that apply to sales representatives in France, based on:

- What was defined for each layer
- Which is the highest layer with application changes for a specific artifact

What Your Application Changes Apply to

No matter what you see while making application changes, your changes apply only to the selected layer for your changes, that is, Job Role. For example, say a field is hidden in the Site layer, but displayed in the Country layer for France. No changes exist for the field in the Job Role layer for sales representative. Since Country is higher than Site, you see the field displayed while modifying pages in Page Composer. However, if you choose to hide the field as part of your changes, then that change applies to the Job Role layer for sales representatives. Users with other job roles in France may still see the field. However, Job Role is higher than Country. So, no sales representatives in any country can see the field, unless a layer higher than Job Role applies to any of these users and has the field displayed.

Business Process Models

The application is based on business process models that map out business flows. When you configure and extend your application, for example to add new pages, you can use these models to help you plan. For diagrams of business process models, see Oracle Fusion Business Process Models (1542019.1) on My Oracle Support at https://support.oracle.com.

Business Process Modeling Levels

The business flows are presented in a five-level hierarchy: industry (L0), business process area (L1), business process (L2), activity (L3), and tasks (L4).

- The hierarchy goes from a high-level, conceptual view to a low-level, application-specific view.
- L1 through L3 are business-driven and don’t depend on any specific implementation in the application.
- L4 aligns with specific features and functionality in the application.
Business Process Models In the Application

The application is organized around these hierarchy levels and flows, which puts focus on the activities and tasks that you must perform. Several aspects of the application are influenced by, if not directly based on, the business process modeling levels. For example, the navigation, user interface, and parts of security are influenced by the business process modeling levels.
2 Configuration Life Cycle

Overview of Configuration Life Cycle

Always make your configurations in a separate environment, called the test environment, and then migrate them to the production environment, which is the one you use for your everyday business. You must never make these changes directly in the application instance you use for your business needs. This is very important because any incomplete, flawed, or unvalidated configuration you make can disrupt your business.

You need to create and enter a sandbox before you can start using configuration tools such as Page Composer and Application Composer to modify your application. Changes you make with these tools are stored in the sandbox as XML files. These changes are then merged with the mainline metadata when you publish your sandbox.

Note: Changes you make in one sandbox aren't available in another one.

In essence, here’s what a typical configuration life cycle looks like:

1. An administrator configures the application in a sandbox in the test environment.
2. Administrator validates the changes in the sandbox.
3. The sandbox is published.
4. Quality Assurance validates the whole environment after all configurations are complete and published.
5. Configurations are migrated from the test environment to the production environment.

Configurations and Extensions

Configuration Workflow

While using Application Composer and Page Composer to make changes to your application, use sandboxes to save your changes in a segregated environment. For example, before making application changes, suppose you create a sandbox named MySandbox, and then make your changes in that sandbox. Now, if others want to see your changes, they can use MySandbox.
Note: If you have multiple users working on the same sandbox, then conflicts may arise within a sandbox. Hence, users must adhere to the prescribed guidelines to avoid such conflicts.

After you complete your application changes, others can review and approve your changes, and then publish to the test environment.

Note that a flexfield sandbox is for testing only and can’t be published. Instead, you can deploy a flexfield to the test environment using the flexfield UI. To test a flexfield configuration before deploying it to the test environment, deploy it to a flexfield sandbox. The changes that you deploy to a sandbox are isolated from the test environment. Users who make the flexfield sandbox active in their session can only see these changes. After you’re satisfied with the changes in the sandbox, you can deploy the changes to the test environment.

You can also use the Manage Configurations dialog box to:

• View others' configuration metadata files
• Download others' configuration metadata files for manually moving them to another environment or diagnosing any issues

Considerations for Viewing and Diagnosing Application Changes

Use the Manage Configurations dialog box to view and diagnose changes made to application pages. Application changes are role-dependent and by default, the Manage Configurations dialog box displays the changes that the signed-in user had performed.

Before you begin viewing application changes, ensure that you have administrative privileges to access the Manage Configurations dialog box. If you’re unable to display the page that contains the changes:

1. Click your user image or name in the global header, and select Manage Configurations from the Administration menu.
2. Use the Search text field on the Manage Configurations dialog box to search for the page, page fragment, or task flow.

You can view the application changes for a user in the Current Context column on the Manage Configurations dialog box. On this dialog box, you can change the page, page fragment, or task flow for which you’re viewing application changes using the Search field.

Developers too may be assigned to specific roles and can view only those application changes that are permitted for the specific roles. However, administrators can view all application changes made at the site level, and for any user, in the All Layers column on the Manage Configurations page. To view application changes made by more than one user, administrators can select multiple users.

Page-Level Changes

To diagnose issues pertaining to application changes, determine whether changes have been applied to a page. Use the Manage Configurations dialog box to determine if page-level changes exist. If a page modification causes problems, such as a user interface component disappears from a page, you can export the application changes and examine the document file.

Related Topics

• Overview of Context Layers
Sandboxes

Overview of Sandboxes

You use sandboxes to make application changes and test them without impacting other users in the environment. Wherever possible, make changes to the application in a sandbox rather than making direct changes in the mainline environment. Sandboxes set apart untested configuration changes from the mainline environment. So you can test your changes in the sandbox and then publish it. After publishing, your changes become available in the mainline metadata or other sandboxes after they're refreshed. So everyone can then see your changes in the environment. Mainline metadata is the primary branch of metadata a sandbox is published to.

Why You Need Sandboxes

Today’s business landscape is quite dynamic. Companies are expected to respond quickly to address both customer and market needs. So multiple teams need to make application changes at the same time while sharing the same data model and configuration starting point. But you may get conflicts between teams working that way. To avoid such conflicts, sandboxes come in handy.

Unified Sandboxes

You can either use the Unified Sandboxes UI, which is the default feature you get, or opt out of it using the Offerings work area to get the Classic Sandboxes feature.

With Unified Sandboxes, you can refresh your sandboxes to bring in the latest changes from the mainline metadata to your sandboxes, and do many other new and versatile sandbox activities. You get a consistent sandbox experience across all configuration tools and a more robust user interface with this feature.

With Unified Sandboxes, you can do these additional sandbox activities:

- Select the configuration tools to enable for your sandboxes while creating them.
- Enable all configuration tools in the same way using the Sandboxes UI. So you get a consistent sandbox experience across tools.
- Restrict access to various sandbox activities for users. For example, you can specify these access rights for your sandboxes:
  - Full access
  - Edit and preview access
  - View only access
- View just your application changes without having other context layers hide your content.
- Test your changes in a preview mode that shows you exactly how your application changes would appear in a published sandbox.
- Refresh and merge sandboxes with latest changes in mainline metadata from other published sandboxes. After merging all changes, you can publish your sandbox.
- After opting in to the Unified Sandboxes feature, if you register your target environment in your source environment, you can do these additional migration tasks using the Migration UI:
  - Migrate your changes from the test environment to the target environment without manually downloading and uploading the configuration set file.
Move only new changes from the source environment to the target environment.

**Sandbox Usage**

You typically use sandboxes for either of these purposes:

- **Test-Only**: You can make application changes using test-only sandboxes, which you don't want to publish to the mainline code.
- **Publish**: Once satisfied with the application changes made in the test-only sandbox, you can replicate these changes in a sandbox that you want to publish. And then publish your changes to the mainline code. This sandbox type is also known as the integration sandbox, because teams working in parallel use this sandbox as the final staging point before publication to the mainline code.

**Note**: Before each patch or upgrade, publish or delete your sandboxes. If you haven't yet completed your work, restart with a new sandbox.

**Unified Sandboxes**

**Enable or Disable the Unified Sandboxes Feature**

You get to use Unified Sandboxes by default. But if you want to use Classic Sandboxes instead of the default sandboxes, opt out of the Unified Sandboxes feature.

Before you start, consider these points:

- Make sure you totally understand what it means to opt in to or opt out of the feature and what impact that would have.
  - When you use Application Composer in your Unified Sandbox, an object can only be edited in any one sandbox in the environment at a time.
  - You can't deploy your flexfields to a sandbox after you opt in to Unified Sandboxes. You must deploy them directly to the mainline environment.
- You must publish or delete any sandboxes that are open right now. Use the Manage Sandboxes dialog box to do these tasks. You can open this dialog box by clicking your user image or name in the global header and selecting **Manage Sandboxes** under **Administration**. Once you opt in to Unified Sandboxes, you can no longer see the published Classic Sandboxes.

In the Offerings work area, enable or disable the Unified Sandboxes feature:

- **Offering**: Any with the Application Extensions functional area
- **Functional Area**: Application Extensions
- **Feature**: Unified Sandboxes
- **Opt In Task**: Click **Continue** if you're sure about enabling or disabling this feature.

**Related Topics**

- **Configure Offerings**

**Create and Activate Unified Sandboxes**

To make changes to the application, you must first store the changes in an active sandbox. You can either create a sandbox or select an existing one, and make it active. You must activate the configuration tools you want to use in your
sandbox. If you plan to use Page Composer in your sandbox and need to edit pages at any layer, not just Site, create a separate sandbox just for using this tool.

**Note:** You can create up to 20 sandboxes. But, you can increase this limit using the **Maximum Number of Sandboxes** profile option. In the Setup and Maintenance work area, use the **Manage Applications Core Administrator Profile Values** task in the Application Extensions functional area.

Create and Activate Sandboxes

Follow these steps to create and activate sandboxes for most configuration tools. For flexfields, use the **Manage Descriptive Flexfields** task or the **Manage Extensible Flexfields** task instead.

1. Click **Navigator > Configuration > Sandboxes**.
2. On the Sandboxes page, click **Create Sandbox**.
3. Enter a name and description for your sandbox.
4. In the **All Tools** section, select the tools you want to activate for this sandbox. The context layers for all selected tools are set as **Site** by default. So the changes you make using these tools affect all users.
5. If you select Page Composer, you can click the **Edit Sandbox Context** icon and change the context layer from **Site** to another layer, for example **External**. You can find the **Edit Sandbox Context** icon in the Support Context column.

**Note:** If you want to use other tools along with Page Composer in your sandbox, don't change the context layer for Page Composer, even though you can. That's because all tools except Page Composer support only a single context layer, **Site**. If you change the context layer for Page Composer from **Site** to any other layer, you can't activate those tools in the same sandbox.

6. Click **Create** to just create the sandbox, or **Create and Enter** to enter or activate the sandbox after creating it.

Here are a few things to know about activating tools in your sandbox.

- If you try to use a configuration tool in a sandbox without activating the tool in it, you get a message prompting you to activate the tool. You can add more tools to your sandbox later also.
- To create and manage saved searches and make UI adjustments (for example, change a table's column width) just for yourself, you must leave your sandbox before making these changes. But if you want to make these changes for others too, then make the changes with Page Composer open, in which case you also must be in a sandbox.

Activate Existing Sandboxes

You can activate only one sandbox at a time.

1. Click **Navigator > Configuration > Sandboxes**.
2. From the list of sandboxes, if available, find the one you want to activate, and click the **Enter Sandbox** icon for that sandbox. Your sandbox is activated, and you can see its name on the sandbox bar above the global header. You can use the options available on the sandbox bar to quickly do some activities, such as view sandbox details, publish the sandbox, or leave the sandbox.

**Related Topics**

- **Set Profile Option Values**
- **Considerations for Managing Flexfields**

**How to Resolve Conflicts in Unified Sandboxes**

When you're in a sandbox, if other users publish their sandboxes, you get refresh notifications on the sandbox bar above the global header. At this time, it's a good practice to refresh your sandbox. When you refresh, all changes published in the mainline environment are brought into your sandbox. You get sandbox merge conflicts in the merge
log when different users change a specific file using different sandboxes. If the changes are made to different files, they're automatically merged, and aren't even reported in the merge log.

**Note:** In some configuration tools, for example Application Composer, an object gets automatically locked when you create it or modify it in a sandbox. So in such cases, an object can only be edited in any one sandbox at a time. If the sandbox is published or deleted, the lock is removed.

You must resolve all conflicts flagged in the merge log so that you can publish your sandbox. To review the merge log, on the Sandbox Detail: <Sandbox Name> page, click the Merge Log tab. The log displays details about the sandbox merge statuses. Let’s understand what these statuses mean and how we can resolve the sandbox merge conflicts based on their statuses.

<table>
<thead>
<tr>
<th>Merge Status</th>
<th>Icon</th>
<th>What It Means</th>
<th>How I Can Resolve Merge Conflicts</th>
</tr>
</thead>
<tbody>
<tr>
<td>Automatically Merged Content</td>
<td>Auto Merge</td>
<td>Different users changed different attributes of the same file using different sandboxes.</td>
<td>These changes are merged automatically.</td>
</tr>
<tr>
<td>Resolvable Conflicts</td>
<td>Resolvable</td>
<td>Different users changed the same attribute of the same file using different sandboxes.</td>
<td>These changes can be merged to the mainline environment. On merging, the changes in the mainline environment overwrites your sandbox changes. So review the merge actions and accept or reject them.</td>
</tr>
<tr>
<td>Unresolvable Conflicts</td>
<td>Unresolvable</td>
<td>Different users changed files in different sandboxes, but the merge conflicts can’t be automatically resolved.</td>
<td>Do any of these tasks:</td>
</tr>
</tbody>
</table>

- If you accept, you can later redo any changes that were overwritten and then publish your sandbox.
- If you reject, your sandbox remains untouched, but you still need to accept a merge before you publish your sandbox.

- Undo your sandbox changes.
- In your sandbox, make the same change, which the other user made in the published sandbox, and try to resolve the conflict.
- Create another sandbox and make your changes in that one.
How the Refresh and Merge Processes Work in Unified Sandboxes

Sandbox changes are refreshed and merged when two different users make changes to the same file using two different sandboxes. Let’s look at an example. Suppose your manager creates a sandbox named Sandbox1 and you create another sandbox named Sandbox2. Your manager then makes a change to a file using Sandbox1 and publishes it to the mainline environment. Now if you make changes to the same file and refresh Sandbox2, all changes published in the mainline environment are merged into Sandbox2. What if both you and your manager entered different values to the same attribute of the file? In that case, the value that your manager entered using Sandbox1 persists because Sandbox1 is published to the mainline metadata. To bring back the changes that you made in Sandbox2, you need to reenter the sandbox and make the changes again.

Application Changes That Can or Can’t be Merged

Let’s again take the same example of Sandbox1 and Sandbox2. Here’s a list of application changes made in Sandbox2 that can merge with the mainline environment, when the same file is changed in both sandboxes.

<table>
<thead>
<tr>
<th>Application Changes</th>
<th>Tools Used to Make the Changes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Changes in business objects and their related fields</td>
<td>Application Composer and Configure Business Objects</td>
</tr>
<tr>
<td>Changes in data security</td>
<td>Data Security</td>
</tr>
<tr>
<td>Changes in lookups</td>
<td>Lookups</td>
</tr>
<tr>
<td>UI text changes</td>
<td>User Interface Text</td>
</tr>
<tr>
<td>Changes to messages, such as warning messages and information messages</td>
<td>Messages</td>
</tr>
</tbody>
</table>

**Note:** Flexfields and setup tasks, apart from the ones listed in the table, aren’t supported in Unified Sandboxes. So changes in these artifacts aren’t merged when Sandbox2 is refreshed.

And here’s a list of changes made in Sandbox2 that can’t merge with the mainline environment when the same file is changed in both sandboxes. To bring back those changes, you can create another sandbox, make the changes in it, and publish that sandbox.

<table>
<thead>
<tr>
<th>Application Changes</th>
<th>Tools Used to Make the Changes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Changes to the appearance of the application</td>
<td>Appearance</td>
</tr>
<tr>
<td>Changes in pricing configuration</td>
<td>Manage Service Mappings</td>
</tr>
<tr>
<td>Changes to the page content</td>
<td>Page Composer</td>
</tr>
</tbody>
</table>
**Application Changes** | **Tools Used to Make the Changes**
---|---
Changes to global page template | Page Template Composer
New pages created | Page Integration

**Related Topics**
- Considerations for Deploying a Flexfield to a Sandbox

**Options to Open Configuration Tools from Unified Sandboxes**
In Unified Sandboxes, after activating the configuration tools, you can also use shortcuts to quickly open some of these tools and make your changes. But you can't open all tools from there. In which case, you can get to those tools using regular navigation.

**Use Shortcuts in Sandboxes**
After you activate a sandbox, all tools activated in it are listed on the sandbox bar and the Sandbox Details page. You can open these tools from either of these locations:

- The Tools drop-down button on the sandbox bar above the global header
- The Active Tools section of the Sandbox Detail: <Sandbox Name> page

In this list, you may notice that some tools are available, while others, for example, Lookups and Messages are grayed out. You can click the available tools to open them directly from the Sandboxes UI.

**Use Regular Navigation**
This table lists the regular navigation options to open all tools that you can activate in your sandboxes. It also indicates whether you can open the tools from the Sandboxes UI.

<table>
<thead>
<tr>
<th>Tool Name</th>
<th>Can Open Using the Sandboxes UI?</th>
<th>Regular Navigation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Application Composer</td>
<td>Yes</td>
<td>Click <em>Navigator &gt; Application Composer.</em></td>
</tr>
<tr>
<td>Configure Business Objects</td>
<td>Yes</td>
<td>Click <em>Navigator &gt; Business Objects.</em></td>
</tr>
<tr>
<td>Appearance</td>
<td>Yes</td>
<td>Click <em>Navigator &gt; Appearance.</em></td>
</tr>
<tr>
<td>Manage Service Mappings</td>
<td>No</td>
<td>Click <em>Navigator &gt; Pricing Administration.</em> and then on the Tasks panel tab, click <em>Manage Service Mappings.</em></td>
</tr>
<tr>
<td>Page Integration</td>
<td>Yes</td>
<td>Click <em>Navigator &gt; Page Integration.</em></td>
</tr>
<tr>
<td>Structure</td>
<td>Yes</td>
<td>Click <em>Navigator &gt; Structure.</em></td>
</tr>
<tr>
<td>Tool Name</td>
<td>Can Open Using the Sandboxes UI?</td>
<td>Regular Navigation</td>
</tr>
<tr>
<td>--------------------</td>
<td>----------------------------------</td>
<td>-------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Lookups</td>
<td>No</td>
<td>In the Setup and Maintenance work area, use the lookup tasks, such as:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Manage Standard Lookups</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Manage Common Lookups</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Manage Set Enabled Lookups</td>
</tr>
<tr>
<td>User Interface Text</td>
<td>Yes</td>
<td>Click Navigator &gt; User Interface Text.</td>
</tr>
<tr>
<td>Messages</td>
<td>No</td>
<td>In the Setup and Maintenance work area, use the messages tasks, such as:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Manage Messages</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Manage Messages for General Ledger</td>
</tr>
<tr>
<td>Data Security</td>
<td>No</td>
<td>Click Navigator &gt; Security Console, and then click the Administrator tab, and click Manage Database Resources.</td>
</tr>
<tr>
<td>Page Composer</td>
<td>Yes</td>
<td>Click your user image or name in the global header and select <strong>Edit Pages</strong> under Administration.</td>
</tr>
<tr>
<td>Global Page Template</td>
<td>No</td>
<td>Click your user image or name in the global header and select <strong>Edit Global Page Template</strong> under Administration.</td>
</tr>
</tbody>
</table>

**Related Topics**
- Update Existing Setup Data

**Object Locking in Application Composer**

The automatic object locking ability in Application Composer avoids the risk of conflicts between multiple users working on objects in parallel and prevents any sandbox merge conflicts that may arise when different users change a specific object using different sandboxes.

Application Composer automatically places a lock on a business object when you create or modify it in a unified sandbox. The locked object displays a lock icon next to its object name in Application Composer’s object navigation tree. A gold lock indicates that the object is locked in the current sandbox. A gray lock indicates that the object is locked in a different sandbox. Hover over a locked object’s name in the navigation tree to display the name of the sandbox that holds the lock.

Application Composer displays and enforces object locks across all unified sandboxes. You can only edit a locked object in the sandbox that holds the lock. For example if the Service Request object is modified in Sandbox A, only Sandbox A holds the lock and anyone using Sandbox A can modify Service Request. Any other sandbox displays a gray lock on Service Request in Application Composer, and prevents users from modifying it.
The lock status of objects changes each time you publish or delete a sandbox, or when a new object lock is established. Collapse and then expand Application Composer's object tree to update the lock status of all objects. Any action that causes the object tree to be refreshed will update the lock status, including when you exit and reenter a sandbox, or refresh a sandbox. If an object is locked and that lock is not yet visible in the current sandbox, a new lock is not allowed to be placed on that object and an error message appears while saving the changes.

For more information on how object locking works in unified sandboxes, see the FAQs on Object Locking.

Best Practices for Preventing Object Locking
Here are some tips on how you can prevent or work around object locking:

- Plan your object model and user interface configurations, and divide the work to prevent two users requiring the same object or objects locked.
- Name sandboxes clearly with appropriate names or initials, so when a lock is in place, it's easy to identify who to contact to release the lock from the sandbox holding the lock.
- Perform configurations in smaller increments. Test and publish more frequently to prevent holding locks.
- Delete test sandboxes created to try out new configurations after they are no longer needed to prevent unnecessary locking.

Publish Unified Sandboxes
After you’re done making changes to the application, publish the sandbox to make your changes available to all users. You must have the Administer Sandbox (FND_ADMINISTER_SANDBOX_PRIV) privilege to publish sandboxes. Remember, you can’t make further changes in the sandbox once you publish it.

Before you start, do these tasks:

- Test or validate your changes in the sandbox in preview mode before actually publishing it. If you made changes using Page Composer, don’t forget to close it before testing. To preview your changes, click the Sandbox Mode drop-down button on the sandbox bar above the global header, and select Preview as if Published (Context: All).

  \[\text{Note: You can see the sandbox bar only when you’re in an active sandbox.}\]

- Resolve all conflicts flagged in the merge log of your sandbox.

To publish a sandbox:

1. Click Navigator > Configuration > Sandboxes.
2. On the Sandboxes page, click the name of the sandbox you want to publish.
3. Click Publish.

  \[\text{Note: The Publish button might be disabled for your sandbox because of various reasons. For example, you haven’t yet made any changes in your sandbox, or the Control Publish Sandbox Action in Production Environment profile option (FND_ALLOW_PUBLISH_SANDBOX) is set to No.}\]

4. Click Continue to Publish. The sandbox is published to the mainline metadata.
5. Click Done.
FAQs for Unified Sandboxes

Who can preview, edit, and publish sandboxes?
Users with the Administer Sandbox (FND_ADMINISTER_SANDBOX_PRIV) privilege can preview, edit, and publish sandboxes. This privilege is assigned by default to the administrators for product families. Your security administrator can define which users have job roles with this privilege.

**Note:** With other sandbox privileges, users can only do some tasks. Users having the Manage Sandbox (FND_MANAGE_SANDBOX_PRIV) privilege can edit but not publish sandboxes. While those having the View Sandbox (FND_VIEW_SANDBOX_PRIV) privilege can view sandboxes in read-only mode.

Why can't I create more sandboxes?
Probably, you have reached the limit for the maximum number of sandboxes. But don't worry, try any of these solutions:

- Increase the limit using the **Maximum Number of Sandboxes** profile option. In the Setup and Maintenance work area, use the **Manage Applications Core Administrator Profile Values** task in the Application Extensions functional area. The default value set for this profile option is 20.
- Delete an unused sandbox.
- Publish a sandbox.

**Related Topics**
- **Set Profile Option Values**

Why is the Publish button on my Sandbox Detail page disabled?
That could be because of one or more of these reasons:

- You haven't yet made any changes in your sandbox. The Publish button is enabled only after you make a configuration change in the sandbox.
- Another sandbox is currently being published in this environment. Try publishing your sandbox again later.
- Your mainline metadata has changes from other published sandboxes. You must refresh your sandbox and then publish it.
- You don't have the Administer Sandbox (FND_ADMINISTER_SANDBOX_PRIV) privilege, which is required for publishing sandboxes. To get this privilege, contact your security administrator.
- You just had a release update or upgrade, and all sandboxes that were available at the time now can't be published. In the future, publish your sandboxes before release updates or upgrades happen.
- A configuration set was just migrated into the environment, and now all sandboxes in the environment can't be published. In the future, publish or delete sandboxes in an environment before migrating configurations there.
- The **Control Publish Sandbox Action in Production Environment** profile option (FND_ALLOW_PUBLISH_SANDBOX) is set to No. To enable publishing, set this profile option to Yes. In the Setup and Maintenance work area, use the **Manage Applications Core Administrator Profile Values** task in the Application Extensions functional area.
- The application is in maintenance mode. Try publishing your sandbox again later.

**Related Topics**
- **Set Profile Option Values**
Why doesn’t the page I am editing with Page Composer work with the sandbox context?

Let’s see why you might get errors about the page not supporting the context layer of the active sandbox, and how you can resolve them.

- Maybe the context layer of your sandbox isn’t a layer that your page supports. Context layers vary based on the category you select for the sandbox. For example, the Customer Relationship Management category supports certain layers that apply to certain pages, and the Human Capital Management category supports other layers and pages. You can activate a sandbox with the appropriate layer and try editing the page again.

- You might not have a role that gives you access to what the sandbox layer covers. For example, if the Internal context layer requires a role with back-end access, you can edit pages at that layer only if you have the role. For more information, contact your security administrator.

Why are the lookup, message, and data security changes appearing in my Unified Sandbox even though I haven’t yet refreshed my sandbox?

Lookups, messages, and data security records are values stored in tables. These records aren’t copied to your sandbox until you make changes in these artifacts in your sandbox. But what if another user changes these artifacts in a different sandbox and publishes it before you make changes in your sandbox? In that case, the other user’s changes show up in your sandbox as soon as you make changes to these artifacts, even before you refresh your sandbox.

How can I deploy flexfields to a Unified Sandbox?

You can’t deploy your flexfields to a sandbox if you have opted in to Unified Sandboxes. You must deploy them directly to the mainline environment.

Related Topics

- Considerations for Managing Flexfields

How does object locking impact object workflows?

When you create an object workflow for an object that is not currently locked, a lock is created for that object. You can create a new object workflow for a locked object in the sandbox that holds the lock. However, an object locked in a different sandbox is disabled for selection in Application Composer’s list of available objects while creating an object workflow. Hover over the name of the disabled object in the list to identify which sandbox has a lock on that object.

When are object locks released?

An object lock exists until the sandbox that holds the lock is published or deleted. A lock is not removed when the change that caused the lock is deleted or reverted. For example, adding a new page layout creates a lock on an object if the object is not already locked, but deleting the layout will not release the lock.

Does importing and exporting configurations impact object locks?

Importing configurations can impact sandboxes. You must delete existing sandboxes before importing new configurations to release all locks and to prevent users from thinking that they can refresh and publish after an import.

How does object locking affect parent and child objects?

An object is locked only when it is directly modified. Modifying a child object will not result in a lock on the parent object. For example, consider a Custom Context subtab (a subtab that displays content of a different object), which is added to the Premium Accounts object. The Custom Context subtab displays the Accounts object, which is the parent object of
Premium Accounts. The object lock is placed on Premium Accounts and not Accounts, because Accounts (the parent object) was not modified.

How does object locking work for dynamic choice lists?  
When you create a custom dynamic choice list, a lock is placed on the object that the custom dynamic choice list is created for. The data source object (the object that the dynamic choice list pulls data from) is not locked.

Can I manually set or remove object locks?  
No, you can't manually set or remove object locks. Locks are automatically set when you create or modify objects in a unified sandbox, and automatically released when you publish or delete the sandbox that holds the locks.

Classic Sandboxes

How You Manage Configurations in Classic Sandboxes
You can apply different types of configurations to an application. For example, you can apply changes to an application's metadata stored in the metadata services repository or changes related to data security of the application. All such configurations are stored in sandboxes and are validated before applying them to an application.

Types of Configurations in Sandboxes
Sandboxes can contain the following types of configurations:

- Metadata changes - These changes (such as non-flexfield user interface (UI) page changes) are captured in a metadata sandbox.
- Data security changes - These changes are additionally captured in a data security enabled sandbox.
- Changes in the generated flexfields business components - These changes are captured in a flexfield that's deployed as a single flexfield sandbox.

Once you’re ready to make sandbox changes available in the mainline metadata, either publish the metadata or data security sandbox, or deploy the flexfield. You can download only metadata and data security sandboxes as a sandbox file for import to another application instance.

The following table lists the differences among the types of sandboxes.

<table>
<thead>
<tr>
<th>Type of Changes</th>
<th>Type of Sandbox</th>
<th>Method for Making Changes Available in Mainline Metadata</th>
<th>Downloadable?</th>
</tr>
</thead>
<tbody>
<tr>
<td>Metadata</td>
<td>Sandbox</td>
<td>Publish sandbox</td>
<td>Yes</td>
</tr>
<tr>
<td>Data security</td>
<td>Sandbox enabled for data security changes</td>
<td>Publish sandbox</td>
<td>Yes</td>
</tr>
<tr>
<td>Flexfield</td>
<td>Flexfield deployed as a flexfield-enabled sandbox</td>
<td>Deploy flexfield</td>
<td>No</td>
</tr>
</tbody>
</table>

Only one sandbox can be active at a time. All changes made in an active sandbox are captured in that sandbox.
Environment
To make application changes, you must first create a sandbox and then use tools, such as Page Composer to make the changes. These changes remain within the sandbox and don’t affect the mainline metadata. You must test and validate your changes in the sandbox. After testing, you can publish your sandbox to make the changes available in the production environment for other users.

Don’t make application changes directly in the mainline metadata. Make all application changes in the sandbox first. When you make changes to an application in a sandbox, you isolate the changes from the mainline metadata. After completing the changes in the sandbox, verify them. When you’re ready to save the changes, publish the metadata or security-enabled sandbox to the mainline metadata.

When you create a sandbox, you can see the information pertaining to only the existing application changes in the current mainline metadata. For example, suppose you make an application change in a sandbox and publish it. Then, on creating another sandbox for the next change, you will see the previous change in the new sandbox because that change exists in the current mainline metadata.

Flexfield sandboxes are for testing only and can’t be published. Make flexfield configurations that are stored in a database. Then, deploy those configurations to a sandbox to see the resulting deployment artifacts in a sandbox environment. Flexfields are deployed directly to the mainline metadata using the flexfield user interface.

Tools
You can use several configuration tools to make application changes. For example, you can modify objects and pages using Page Composer, which uses sandboxes. Oracle Business Process Composer and Oracle SOA Composer are also tools used for making application changes, but they don’t use sandboxes. They have their own mechanisms for handling application changes.

Managing a Flexfield Sandbox
To create a flexfield-enabled sandbox, deploy a flexfield to a sandbox using the Manage Flexfield task flow. The flexfield sandbox gets its name from the flexfield you deploy. You can’t test two flexfields in the same sandbox. After deploying a flexfield as a sandbox, sign out and sign in again to view how the sandbox reflects the flexfield changes, such as new segments. You can redeploy the same flexfield to the same sandbox repeatedly as you make incremental changes to the flexfield setup. A flexfield sandbox can’t be published. So, when the flexfield is deployed to the mainline metadata, any page changes or data security in the flexfield sandbox can’t reach the mainline metadata. If you’re entitled to do so, manage flexfield-enabled sandboxes in the Sandboxes UI.

Related Topics
- Considerations for Deploying a Flexfield to a Sandbox

Guidelines for Working with Classic Sandboxes
In the runtime configuration workflow, use sandboxes to isolate the changes from the mainline metadata for testing and validating. After you’re satisfied with the changes, you can publish the changes back to the mainline metadata.

The testing sandboxes are never published and therefore produce no concurrency conflicts between sandboxes. You can have several testing sandboxes at the same time.
Application changes in the sandboxes that are published are merged back to the mainline metadata. This figure illustrates the two types of sandboxes and their relationship to the mainline metadata.

**Working with a Single Sandbox**

Conflicts may arise when users edit a shared artifact, such as when a user performs an operation that adds or edits a translatable string. For example, suppose:

- One user edits a field’s display label or help text, or a validation rule’s error message. Whereas, another user performs an operation at the same time that similarly affects translatable strings.
- Two or more users are working in Navigator menus that are shared across applications. Whenever a conflict arises among users, the application displays concurrency warning messages.

**Working with Multiple Sandboxes**

Multiple sandboxes are used when configurations are stored in testing as well as production sandboxes. Suppose, after you create a sandbox, a concurrent change is made in the mainline metadata. Now, when you attempt to publish that sandbox, the application detects such conflicts at publication time, and you get error messages. For example, when you try to publish your sandbox, you may get a message showing a conflict on oracle/apps/menu/fnd/applcore/dataSecurity/dataSecurityService/mds/DSMO.xml. This message indicates that the security changes that you made in your sandbox conflict with other security changes in the mainline metadata. Delete the sandbox and recreate your changes in a new sandbox.
If multiple users are permitted to work in multiple sandboxes at the same time, follow these guidelines to avoid conflicts:

- Any number of test-only sandboxes can operate concurrently. That is, multiple users can use multiple sandboxes concurrently for testing if these sandboxes will never be published. Sandboxes that are used for testing only, and that aren’t published, cause no conflicts with each other. Be aware, however, that all modifications will be lost when the sandboxes are deleted.

- For sandboxes that aren’t for test-only and will be published, users can use multiple concurrent sandboxes only if they operate on mutually exclusive artifacts. For example, you can have:
  - One sandbox that contains a page that a user is modifying to add a task flow
  - Another sandbox that contains a different page from a different application

  However, some objects might still share underlying artifacts, and thus it’s not always obvious if two objects are truly mutually exclusive. Thus, proceed with caution when using multiple concurrent sandboxes that will be published. It’s still possible that a conflict could occur, which would require the deletion of one or more sandboxes.

- Suppose the users update an artifact in both, the mainline metadata and in one sandbox, or in two different sandboxes. Now, when you publish the sandbox, the application detects such conflicts and you get an error message. At this point, cancel publishing the sandbox to avoid overwriting previous changes.

  **Note:** For a sandbox that contains configurations pertaining to ADF Business Components, sign out and sign in again after switching in or out of this sandbox. This process ensures avoiding any inconsistencies between the runtime caches and the ADF Business Components definitions.

### Create and Activate Classic Sandboxes

To make changes to the application artifacts, you must first store them in an active sandbox. You can either create a sandbox or select an existing sandbox, and designate it as an active sandbox. The active sandbox holds the context for all the changes. The sandbox uses a database to store the actual changes. After testing your changes, you can publish the sandbox, or deploy the flexfield, and the changes are merged into the mainline metadata. Eventually, the sandbox is archived.

This procedure is for setting up non-flexfield sandboxes. For flexfields, use the Manage Descriptive Flexfields task or the Manage Extensible Flexfields task.

To create and activate a sandbox:

1. Click your user image or name in the **global header**, and select **Manage Sandboxes** from the **Administration** menu.
2. Use the Manage Sandboxes dialog box to create a sandbox.
3. Click **Save and Close**.
4. On the Manage Sandboxes dialog box, select the new sandbox or an existing one, and click **Set as Active**. The sandbox is designated as the active sandbox.
5. Close the Manage Sandboxes dialog box.

Only one sandbox can be active at a time. Once a sandbox is active for your session, the sandbox name is displayed on the sandbox bar above the global header.

### Publish Classic Sandboxes

After completing the application changes in the sandbox, publish them to make them available in the application for all others.
Before you start publishing your sandbox, test or validate your changes. If you made application changes using Page Composer, don't forget to close it while testing.

1. Click your user image or name in the *global header*, and select **Manage Sandboxes** from the **Administration** menu.
2. On the Manage Sandboxes dialog box, select the sandbox and click **Publish**. The Publish confirmation message box appears.
3. Click **Yes**. The sandbox is published to the mainline metadata.
4. Close the Manage Sandboxes dialog box.

### Migration

**Overview of Migration**

You can move your configurations from the test environment, which is the source, to the production environment, which is the target. You use the Migration page to create a migration set and export it from the source environment, and then import it into the target environment. A configuration set is a JAR file that contains all your configurations across all product families, such as those stored in Oracle Metadata Services repository, JEDI, CRM, and BI. Changes you export from the source environment can't be merged with any changes you manually make in the target environment. As such, it's very important that you never configure the target environment directly.

You can use either Classic Migration or Unified Migration to move your configurations. Classic Migration is the default migration you can do, and you can use Unified Migration after you opt in to the Unified Sandboxes feature. Both the appearance and behavior of your Migration page changes when you enable Unified Sandboxes.

**Note:** In Unified Migration, the configuration set is called migration set.

#### Classic Migration

Here's what you do in a classic migration:

- You always need to migrate all configurations in the source environment.
- You have to manually download the configuration set from the source environment, and manually upload it into the target.
- You can't preview configurations in the target environment before applying them.

#### Unified Migration

Here's what you can do in a unified migration:

- You can register the target environment in the source environment.
- You have the option to migrate only new changes if both environments are synchronized.
- Your migration set is automatically sent to the target environment for import, if the target is registered and available.
- Your migration set is imported into a sandbox instance before it's applied to the target environment. You can preview your configurations in this sandbox instance before applying them to the mainline.
Here’s a diagram that shows what you can do with classic and unified migrations.

**Contents of the Configuration Set**

You use the Migration page to create a set of configurations and extensions made to an environment, download the set, and then upload it into another environment. This configuration set includes configurations across all product families.

**Note:** In Unified Migration, the configuration set is called migration set.
What It Includes

The configuration set includes, but isn’t limited to these changes:

- Application changes done using Application Composer. However, not all changes made using Application Composer are migrated.
- Changes made to application artifacts using these tools:
  - Page Composer
  - Appearance
  - Structure
  - User Interface Text
  - Page Integration
- Changes in these artifacts of the Applications Core Setup application:
  - Messages
  - Lookups
  - Data security but not the ones created by the HCM Profiles UI
  - Descriptive, extensible, and key flexfields, and value sets
  - Attachment categories and metadata
  - Deep links
- Changes in Reports and Analytics, such as regeneration of SOAP services, including user-defined attributes
- Changes done using the Manage Oracle Social Network Objects task
- Changes in functional security settings made in Application Composer, including functional privileges that control access to custom objects. However, not all security changes are migrated

Note: If you migrate any functional security associated with roles in the source to a target instance, and the corresponding role doesn’t exist in the target instance, an error will occur on import. To avoid these errors, you can selectively create these roles in the target environment when you import your configuration set.

What It May Include

Depending on what modules you select during export, your configuration set may include these changes:

- Changes in CRM email templates created in Application Composer
- Changes in Enterprise Scheduler Service (ESS) module:
  - Job definitions
  - Job sets
  - Job schedules
  - Incompatibilities
  - Work shifts
  - Work assignments
  - Work assignment schedules
Job request parameters

- Changes in Service Oriented Architecture (SOA) artifacts, such as configurations done using SOA Composer
- Changes done using Oracle Business Intelligence Enterprise Edition, including but not limited to these features:
  - Oracle Business Intelligence Answers
  - Oracle Business Intelligence Delivers
  - Business Intelligence Composer
  - Dashboard Builder
  - Oracle Business Intelligence Publisher

Note: You can move configurations done using business intelligence tools only if the Business Intelligence in Configuration Set Migration Disabled profile option is set to No.

What It Doesn't Include

Your configuration set doesn't include these items:

- The following Application Composer changes:
  - Object artifacts that were generated from the Import and Export page in Application Composer to make extensions available for importing and exporting
  - User names and passwords for secured SOAP web service connections
  - The enabled attachment feature for custom objects

- Personalizations
- Unpublished configurations within a sandbox
- Deletions, for example, the set doesn't include the removal of a custom object. After you import a configuration set into the target environment, you must examine the environment for any deletions that you must make manually.

- Roles or role hierarchy changes
- Custom roles created outside of Application Composer
- Entitlements or privileges created outside of Application Composer
- Security Console changes, including these changes made directly in the security console:
  - Enterprise roles
  - New duty roles
  - Role hierarchy changes

You must manually update the target environment with any Security Console changes.

Related Topics

- Import and Export Custom Objects
Migrate Configurations Using Classic Migration

Using a configuration set, you can move your configurations from a test environment, which is your source, to a production environment, which is your target.

Before You Start

Before creating a configuration set, make sure you meet these conditions:

- The source and target environments are of the same release, with the same standard and one-off patches applied to both environments.
- All Page Composer configurations made in sandboxes are complete and published.
- All configurations and extensions made using the Structure page, the Manage Standard Lookups task, and Security Console, are complete.
- To move content created using Oracle Business Intelligence Enterprise Edition, set the Business Intelligence in Configuration Set Migration Disabled profile option to No in source and target environments. You can find this profile option in the Manage Administrator Profile Values task in the Setup and Maintenance work area.
- You have the following privileges to access the Migration page:
  - Manage Outgoing Configuration Set
  - Manage Incoming Configuration Set
  
  Contact your security administrator for details.
- You never make changes in the target or production environment while applying configurations.

Note: If you make changes to the production environment in emergency situations, you must make the same changes to the test environment. Making the changes to the test environment ensures that these changes are included in the next configuration migration.
- Don't make any changes in the source environment during the export process.
- You delete any temporary files on the server from previous migrations. If there are temporary files on the server, click the Delete button next to your previous import and export records.

Create Configuration Sets

1. In the source environment, click Navigator > Configuration > Migration .
2. On the Outgoing tab of the Migration page, click Create Configuration Set.
3. Provide a name for the configuration set.
4. Optionally, type a description of the configuration set.
5. Select the content you want to migrate. The choices available on this dialog box are filtered by the offerings you have opted in for.

Note: The Industry solution extensions module is for Oracle's internal use only and has no impact on migration.

6. Click Save and Close.
7. Click Refresh periodically to see the current status of the set creation because creating a configuration set can take a few minutes. You can click Log to review the process log, which provides more details about the configurations that are being compressed. If an error or exception occurs during this process, the log will give
you information about the configurations that failed to compress. The process runs asynchronously, so you can close the page and return to it later.

8. Eventually, the status changes to Ready for Download, which means that the configurations are ready for you to download. Before downloading the configurations, you can click **Content Read Me** to download the Readme file listing all the configurations you exported.

9. Click **Download** to download your configuration set. Ensure that the downloaded file is a JAR file.

10. After you download the file on your local file system, click **Delete** to remove the temporary files that were created on the server.

Apply Configuration Sets

After you apply configurations, end users must sign out and sign back in to see the changes. So apply configurations when fewer people are signed into the environment. To apply configurations to the target environment, follow these steps:

1. Open the Migration page in the target environment.
2. Click the Incoming tab.
3. Click **Browse**, specify the name and location of the configuration set file, and click **Open**.
4. Once the status for the configuration set on the Incoming page is Ready to Apply Configurations, review the roles in your configuration set that are missing in your target environment. Create any missing roles as required. To review and create these roles, follow these steps:
   a. Click **Details**.
   b. Select an application.
   c. Select the roles you want to create.
   d. Click **Create Role**, and click **Yes**.
   e. Click **OK** after the roles are created.
   f. Click **Save and Close**.
5. Click **Apply** to apply the configuration set to your target environment.
6. Periodically, click **Refresh** to view the current status of the Apply action. When the configuration set is successfully applied to the target environment, the status is displayed as Applied and Deleted. You can review the process log, if required. The process runs asynchronously, so you can close the page and return to it later. If problems occur during an Apply action, log a service request using My Oracle Support at https://support.oracle.com.

Post Migration Tasks

Do these tasks after you apply your configuration set to the target environment:

1. Access the target environment and examine the environment for any deletions that you must manually make.
2. Delete and recreate any web service connections in the target environment, using the target environment URL and credentials.
3. Deploy all flexfields that display a Patched status.
4. Do the following tasks to send the new and updated social network definitions to the social network server:
   a. In the Setup and Maintenance work area, open the **Manage Oracle Social Network Objects** task.
   b. As part of the applying configurations process, some objects are created or updated. If the **Enabled** value of such an object is anything other than No, trigger the process of sending its definition to the social network server. You can do this by disabling the object and enabling it again with its original status. For example, if the **Enabled** value is **Manual**, then you can do these tasks:
      i. Disable the object.
      ii. Enable the object, and select the value, **Manual**.
      iii. Click **OK** and save the changes.
On the Manage Oracle Social Network Objects page, click **Synchronize** to synchronize a selected object, or click **Synchronize All** to synchronize all objects at the same time.

5. Manually migrate all business processes created in the source environment to the target environment.

6. If a new theme was created and applied in the source environment, and you want to use that theme in the target environment, then go to the Appearance work area and manually apply that theme.

7. After applying configurations, do functional testing to verify the changes. Suppose testing exposes problems with the configurations, such as importing more than you intended, or the changes weren’t what you expected. In such cases, restore your environment to its pre-migration configuration.
   a. Open the configuration set in the Incoming tab or infotile of the Migration page.
   b. Click **Restore** to revert to the state before the configuration set was applied.

8. Finally, broadcast information to the users that they must sign out and sign in to view the most recent changes.

### Things to Know About Migration

- You can't restore your environment to an earlier configuration if the environment is upgraded after your previous migration. But if a new import is submitted in the upgraded instance, then the most recent import can be reversed.

- Lookup values for lookup fields that exist in both source and target aren't overwritten during the configuration import. The lookup values from source are added to the target and all the lookup values coexist for the same field. For example, the **Status** field in its source environment has values, **Open** and **Closed**. In the target environment, this field has values, **Yes** and **No**. After the import, the **Status** field in the target environment has values, **Open**, **Closed**, **Yes**, and **No**.

- During configuration import, the data security privileges aren't automatically revoked in the target environment. For example, say a specific privilege is granted in the target environment, but the corresponding privilege doesn't exist in the source environment. During import, the privilege in the target environment won’t be automatically revoked. To address this issue manually, add such a privilege to the source environment and revoke it. The revoke action is picked up as a configuration instance during the configuration import process and applied to the target environment.

- You can create reports directly in the target environment. However, make sure that you create the reports and reference them to subject areas that were created in the source environment. Don't create subject areas directly in the target environment.

- You can initiate configuration export and import tasks only from the mainline metadata. If you initiate migration from a sandbox, the process doesn't execute.

- While an upload or restore activity processes Presentation Services changes, these things can happen:
  - Reports that were submitted by Oracle Enterprise Scheduler to Oracle Business Intelligence Publisher and were scheduled to execute during the process, will fail.
  - The Reports and Analytics pane may not display.
  - Oracle Business Intelligence Publisher reports may not display on Oracle Business Intelligence Presentation Services analyses or dashboard pages.
  - Users may not be able to access these Oracle Business Intelligence Enterprise Edition features:
    - Oracle Business Intelligence Answers
    - Oracle Business Intelligence Delivers
    - Business Intelligence Composer
    - Oracle Business Intelligence Interactive Dashboards

- When you restore your environment to an earlier configuration, you lose all personalizations you made after the previous migration.
Related Topics

- Import and Export Custom Objects

Migrate Configurations Using Unified Migration

Use a migration set to move your configurations from the source environment to the target environment. With Unified Migration, you can import your configurations into a sandbox in the target environment before you apply them to the mainline. And if you register your target environment in your source environment, you can do these additional migration tasks:

- Migrate your changes from the test environment to the target environment without manually downloading and uploading the migration set file.
- Move only new changes from the source environment to the target environment. This type of migration is called delta migration. However, only sandbox aware modules support delta migration. All other modules move all changes every time they're migrated, even in delta migrations.

Before You Start

Before creating a migration set, make sure you meet these conditions:

- The source and target environments are of the same release, with the same standard and one-off patches applied to both environments.
- Both source and target environments have Unified Sandboxes feature enabled. You can't migrate your configurations if only one environment has this feature enabled in it.
- Delete or publish any sandboxes in the target environment that have Application Composer enabled, before you begin your migration.
- All Page Composer configurations made in sandboxes are complete and published.
- All configurations and extensions made using the Structure page, the Manage Standard Lookups task, and the Security Console, are complete.
- To move content created using Oracle Business Intelligence Enterprise Edition, set the Business Intelligence in Configuration Set Migration Disabled profile option to No in both the source and target environments. You can find this profile option in the Manage Administrator Profile Values task in the Setup and Maintenance work area.
- You have the following privileges to access the Migration page:
  - Manage Outgoing Configuration Set
  - Manage Incoming Configuration Set

  Contact your security administrator for details.

- Never make changes in the target or production environment while applying configurations.

  Note: If you make changes to the production environment in emergency situations, you must make the same changes to the test environment. Making the changes to the test environment ensures that these changes are included in the next configuration migration.

- Don't make any changes in the source environment during the export process.

- Delete any temporary files on the server from previous migrations. If there are temporary files on the server, click the Delete button next to your previous import and export records.
Register the Target Environment

1. In your source environment, open the Manage Configuration Set Migration Target Security Policy task in the Setup and Maintenance work area.
2. Enter the URL of your target environment. Use the full URL, including host and protocol information.
3. Enter your user name and password.
4. Click Save and Close.

Verify the Target Environment

1. In the source environment, click Navigator > Configuration > Migration.
2. Click the Environment Info infotile.
3. Verify that the page displays the same URL as the target instance.

If you don't see the correct target instance, try registering it again.

Create Migration Set

1. In the source environment, select Configuration > Migration from the Navigator menu.
2. Click Create Migration Set from the Outgoing infotile. If you want to migrate all configurations instead of just the new changes, click the Create Full Set link for a full migration.
3. Provide a name for the migration set.
4. Optionally, type a description of the set.
5. Select the content you want to migrate.

Note: The Industry solution extensions module is for Oracle's internal use only and has no impact on migration.
6. Click Save and Close.
7. Click the Refresh icon periodically to see the current status of the set creation. You can click the Log icon to review the process log, which provides more details about the configurations that are being compressed. If an error or exception occurs during this process, the log gives you information about the configurations that failed to compress. The process runs asynchronously, so you can close the page and return to it later.
8. Eventually, the status changes to Ready for Download, which means that the migration set is complete. You can click Content Read Me to download the Readme file listing all the configurations you exported.

If you have registered the target environment, you don't need to manually download the migration set from the source environment and upload it to the target environment. You can sign in to the target environment, go to the Incoming infotile on the Migration page, and skip to step 4 of the next subsection. However, if you didn't register the target environment, or if the target was unreachable during export, continue to step 9 of this section.
9. Click the Download icon to download your migration set. Ensure that the downloaded file is a JAR file.
10. After you download the file on your local file system, click Delete to remove the temporary files that were created on the server.

Apply Migration Set

After you apply configurations, end users must sign out and sign back in to see the changes. So, apply configurations when fewer people are signed in to the environment. To apply configurations to the target environment, follow these steps:

1. Open the Migration page in the target environment.
2. Click the Incoming infotile.
3. Click the Upload Migration Set link. Then, browse for your migration set file, and click OK.
4. Review the roles in your migration set that are missing in your target environment, and create them if required. To review and create these roles, follow these steps:
   a. Click the Details icon.
   b. Select an application.
   c. Select the roles you want to create.
   d. Click Create Role, and click Yes.
   e. Click OK after the roles are created.
   f. Click Save and Close.
5. Click Import to import your migration set into a sandbox instance in the target environment.
6. Wait for the status to change to Successfully Imported.
7. Optionally, click Preview to view your configurations in the sandbox preview mode.
8. Click Apply when you're ready to apply your configurations to the target environment.
9. Periodically, click the Refresh icon to view the current status of the apply action. You can review the process log, if required.
   The process runs asynchronously, so you can close the page and return to it. If problems occur during an Apply action, log a service request using My Oracle Support at https://support.oracle.com.
10. The migration set is successfully applied to the target environment when the status changes to Applied and Deleted.

**Post Migration Tasks**

Do these tasks after you apply your migration set to the target environment:

1. Access the target environment and examine the environment for any deletions that you must manually make.
2. Delete and recreate any web service connections in the target environment, using the target environment URL and credentials.
3. Deploy all flexfields that display a Patched status.
4. Do the following steps to send the new and updated social network definitions to the social network server:
   a. In the Setup and Maintenance work area, open the Manage Oracle Social Network Objects task.
   b. As part of the applying configurations process, some objects are created or updated. If the Enabled value of such an object is anything other than No, trigger the process of sending its definition to the social network server. You can do this by disabling the object and enabling it again with its original status. For example, if the Enabled value is Manual, then you can do this:
      i. Disable the object.
      ii. Enable the object, and select the value, Manual.
      iii. Click OK and save the changes.
   c. On the Manage Oracle Social Network Objects page, click Synchronize to synchronize a selected object, or click Synchronize All to synchronize all objects at the same time.
5. Manually migrate all business processes created in the source environment to the target environment.
6. If a new theme was created and applied in the source environment, and you want to use that theme in the target environment, then go to the Appearance work area and manually apply that theme.
7. After applying configurations, perform functional testing to verify the changes. Suppose testing exposes problems with the configurations, such as importing more than you intended, or the changes weren't what you expected. In such cases, restore your environment to its pre-migration configuration.
   a. Open the migration set in the Incoming infotile of the Migration page.
   b. Click Restore to revert to the state before the migration set was applied.
8. Finally, broadcast information to the users that they must sign out and sign in to view the most recent changes.
Things to Know About Migration

- You can’t restore your environment to an earlier configuration if the environment is upgraded after your previous migration. But if a new import is submitted in the upgraded instance, then the most recent import can be reversed.

- Lookup values for lookup fields that exist in both source and target aren’t overwritten during the configuration import. The lookup values from source are added to the target and all the lookup values coexist for the same field. For example, the `Status` field in its source environment has values, `Open` and `Closed`. In the target environment, this field has values, `Yes` and `No`. After the import, the `Status` field in the target environment has values, `Open`, `Closed`, `Yes`, and `No`.

- Lookups can be migrated with both configuration migration, and setup data migration. However, once you use one of these methods, you should use the same method for all subsequent lookup migrations. Don’t use both methods to migrate lookups between the same set of environments.

- During migration, data security privileges aren’t automatically revoked in the target environment. For example, say a specific privilege is granted in the target environment, but the corresponding privilege doesn’t exist in the source environment. During import, the privilege in the target environment won’t be automatically revoked. To address this issue manually, add such a privilege to the source environment and revoke it. The revoke action is picked up as a configuration instance during the configuration import process and applied to the target environment.

- You can create reports directly in the target environment. However, ensure that you create the reports and reference them to subject areas that were created in the source environment. Don’t create the subject areas directly in the target environment.

- You can’t initiate a migration if you’re in an active sandbox.

- While an upload or restore activity processes Presentation Services changes, these things can happen:
  - Reports that were submitted by Oracle Enterprise Scheduler to Oracle Business Intelligence Publisher and were scheduled to execute during the process, will fail.
  - The Reports and Analytics pane may not display.
  - Oracle Business Intelligence Publisher reports may not display on Oracle Business Intelligence Presentation Services analyses or dashboard pages.
  - Users may not be able to access these Oracle Business Intelligence Enterprise Edition features:
    - Oracle Business Intelligence Answers
    - Oracle Business Intelligence Delivers
    - Business Intelligence Composer
    - Oracle Business Intelligence Interactive Dashboards

- When you restore your environment to an earlier configuration, you lose all personalizations you made after the previous migration.

Related Topics
- Manage Themes

Preview Mode in Unified Migration

After you import your configurations into the target environment, you can preview them in a sandbox instance before you apply them. But you can’t preview all your configurations. You can only preview those configurations that are sandbox aware, and have no dependency on non-sandbox artifacts.
Note: When migrating configurations, make sure you have the appropriate administrator roles to view and test all your changes in preview mode.

What You Can See
The ADF module is the only module that's sandbox aware. As such, these MDS artifacts from ADF are available for preview:

- Changes to existing custom objects
- Pages
- Workflows
- Groovy scripts
- Triggers
- Lookups
- Attachments
- Data security
- Strings
- Web services

Note: Though visible, these web services won't be functional because their connections aren't migrated.

However, if any of these artifacts have any dependency on non-sandbox aware artifacts, they won't show up in your preview.

What You Can’t See
Artifacts that aren't part of the ADF module aren't visible in preview mode. Here are a few of those non-sandbox aware artifacts:

- BI reports
- BI custom subject areas
- Functional security
- New custom objects
- Email templates
- Workspace configurations
- Notification preferences
- ESS artifacts
- SOA artifacts
- Industry extensions
- Analytics
FAQs for Migration

What's the difference between a delta migration and full migration?
A delta migration moves only those changes that were made to an environment after the last migration was performed. On the other hand, a full migration moves all changes made to an environment.

For instance, let's say Alpha is your source environment, and Beta is your target environment. You create object X in Alpha environment and perform a migration. Then, you create object Y in Alpha environment and perform another migration. If the second migration is a delta migration, only the object Y is moved. However, if it's a full migration, both the objects X and Y are moved.

When can I do a delta migration?
You can do a delta migration if you meet these three conditions:
- You have registered your target environment in your source environment, without any issues.
- You have enabled Unified Sandboxes feature in both source and target environments.
- Both of your environments are still synchronized.

Why aren't my environments synchronized?
Your environments may not be synchronized if you did either of these things:
- Directly configured the target environment.
- Upgraded one of the environments or both of them.

How can I synchronize my environments?
If your source and target environments aren't synchronized, you need to do a full migration to synchronize them again.

Configuration Management

Considerations for Exporting and Moving Configurations

Configurations are stored in XML files. You can use these XML files to export configurations for the following reasons:
- To move configurations and extensions to another environment, such as the production environment.
- To diagnose issues noticed in the test environment.
- To send files to your help desk for further diagnosing.

The following table lists the tools to use to export and move configurations and extensions.

<table>
<thead>
<tr>
<th>Tasks</th>
<th>Tools to Use</th>
</tr>
</thead>
<tbody>
<tr>
<td>Move all configurations to another</td>
<td>Configuration Set Migration.</td>
</tr>
<tr>
<td>application environment.</td>
<td></td>
</tr>
</tbody>
</table>
## Tasks

<table>
<thead>
<tr>
<th>Tasks</th>
<th>Tools to Use</th>
</tr>
</thead>
<tbody>
<tr>
<td>Move only descriptive flexfield configurations to another application environment.</td>
<td>Setup and Maintenance work area. Moves configurations for a specified module. To move configurations for all modules, use Configuration Set Migration.</td>
</tr>
<tr>
<td>Move only extensible flexfield configurations to another application environment.</td>
<td>Setup and Maintenance work area. Moves configurations for a specified module. To move configurations for all modules, use Configuration Set Migration.</td>
</tr>
<tr>
<td>Move only value set configurations to another application environment.</td>
<td>Setup and Maintenance work area. Moves configurations for a specified module. To move configurations for all modules, use Configuration Set Migration.</td>
</tr>
<tr>
<td>Move only lookups to application environment.</td>
<td>Setup and Maintenance work area. Move application standard lookups, application common lookups, or both.</td>
</tr>
<tr>
<td>Move only data security policies to another application environment.</td>
<td>Setup and Maintenance work area. It doesn't move Oracle Fusion Human Capital Management roles.</td>
</tr>
<tr>
<td>Export configurations to a file to help diagnose an issue.</td>
<td>Manage Configurations dialog box.</td>
</tr>
</tbody>
</table>

### Downloading Configurations

You can download all configurations of a page for all layers using the **Download Configuration for All Layers** link in the Manage Configurations dialog box. To open Manage Configurations dialog box, click your user name in the global header, and select Manage Configurations. The file you download contains all the configuration XML files for the page. However, you can’t upload this file anywhere.

### Downloading Configuration Set Reports

After exporting configurations, you can view and download a configuration set report that contains a list of all configurations available in a configuration set. To do so, click **Content Read Me** from the Outgoing tab of the Configuration Set Migration page. To open the Configuration Set Migration page, select **Configuration > Migration** from the **Navigator** menu.

This report includes all new or updated:

- Objects
- Fields
- Pages
- Business intelligence (BI) changes

Business logic changes such as Groovy scripts and triggers aren’t included in the configuration set report.

**Related Topics**

- Guidelines for Moving Related Common Reference Objects
Download Configurations From a Page

Let's say the configurations you made to a page resulted in an error. You need to download the configurations you made to that page to diagnose what went wrong. You can use the Manage Configurations dialog box to download all the configuration XML files for that page. The downloaded files contain the changes made to all the layers of that page, and you can use them to diagnose configuration issues.

Download Configuration Files

Do you know the file path to the page you want to download configurations from? If you do, then follow these steps:

1. Click your user image or name in the global header, and on the Settings and Actions menu, select Manage Configurations.
2. In the Search field, enter the file path to your page and click the Search icon.
3. Click Download Configuration for All Layers.

If you don't know the file path, then follow these steps:

1. Create and activate a sandbox.
2. Go to the page you want to download configurations from.
3. Click your user image or name in the global header, and on the Settings and Actions menu, select Edit Pages.
4. Click the Structure tab.
5. Click any component on the page.
6. In the confirmation window, click Edit.
7. Go back to the Add Content tab.
8. Click Manage Configurations.
9. Scroll down and click Download Configuration for All Layers.
3 Page Modification

Overview of Page Modification

You can use Page Composer to change page content, layout, and more. Using other tools, you can create pages and change UI text, themes, infolets, and so on.

Let's look at some of the things that you can do using other tools:

- Change the UI text using tools, such as User Interface Text.
- Change the look and feel of the application using the Appearance page.
- Open an infolet page and modify it using Page Composer.
- Create and manage pages for hosting third-party applications using Page Integration page.

Note: You can't use Page Composer to configure your home page. Instead, you can use other tools such as, Structure and Appearance for home page configurations.

Related Topics
- Configure Home Page Navigation
- Overview of Configuring Home Page and Navigation
- Overview of Configuring Themes and Home Page Settings
- Define Home Page Appearance

Overview of Using Page Composer

Guidelines for Page Modification

Before modifying pages, you must do these tasks:

- Understand the typical workflows for working with application changes.
- Verify that the page can be modified. To do so, you can check if either the Edit Pages or the Edit <Page Name> Pages menu item is available in the Settings and Actions menu. If no, then that means the page can't be modified.
- Confirm that your privileges are sufficient for modifying the page.
- Activate a sandbox.

Related Topics
- Overview of Sandboxes
- Create and Activate Unified Sandboxes
- Configuration Workflow
Page Composer Views

You can use either Design view or Source view for viewing and changing page content and layout in Page Composer. To open a view option, select it from the View menu at the top left corner of the page. Although both views share many common page modification features, you can use some unique features in each view.

Design View

In Design view, you see one region that shows a WYSIWYG rendering of the page and its content. Work with components directly and access their properties from the toolbar in the chrome of the individual component.

Source View

In Source view, you see two regions:

- Selection pane, showing a WYSIWYG rendering of the page and its content
- Source pane, showing a hierarchical ordering of the page components, including some components that otherwise don’t appear on the page. You can select and configure such components in Source view.

Tip: Controls on individual components are inactive in Source view, but you can click an individual component to select it.

In Source view, you can:

- Click a component in the Selection pane to highlight the component in the hierarchical list. The cursor turns to a magnifier and a blue outline appears around the component selection. You can also traverse the hierarchy and select components directly.
- Click Edit on the view header to work with components indirectly and access their properties. You can also right click the object in the hierarchy, and click Edit.

Page Component Properties

All components have configurable properties that control, or express their appearance and functionality. Many properties are common to all component types, while some properties are unique to one component type. Use the Component Properties dialog box to view the properties of a component. You can open this dialog box by selecting the component that you want to edit and clicking Edit Component. You can see properties of similar functions in tabs that name the category of the properties.

Note: Properties and tabs can vary from component to component.

Component Property Tabs

This table describes the tabs that you may see in a component properties dialog box.

<table>
<thead>
<tr>
<th>Tab</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Parameters</td>
<td>Settings that control component aspects that are specific, or often unique to the component.</td>
</tr>
</tbody>
</table>
For example, on a page containing a map, a component may have a parameter that provides a choice between units of measurement.

Display Options

Includes these settings that affect the chrome of a component:

- Header, header text, and border
- Actions menu
- Edit, Remove, Expand, Collapse, and other icons
- Tooltips

For example, display options on image layout components specify the image source URL and its optional link target.

Child Components

The list of all components contained within and under the control of the parent component, including controls to order the child components.

Style

Content Style

Settings that affect the look and feel of the component chrome or the component contents. For example, font, color, and dimension. These settings override corresponding values from a parent object, such as a component, page, and application, providing an opportunity to fine-tune appearance.

Some style properties may be disabled at the component level if other page or application elements (such as the skin) don’t support modification to the property.

Events

Events and event handlers for all the components on the current page that the current component can consume.

For example, when you select a check box within the current context, it’s an event. The code that drives the result of an event, such as making another component visible is an event handler.

Considerations for Modifying Objects That Appear on Multiple Pages

Use Page Composer to modify objects that appear on multiple pages. Whether your application changes affect one or more pages depends on the way you include the object on the page and the applications that you use.

Consider these points when you modify objects that appear on multiple pages.

Appearance of Object Modifications Based on Shared and Non-Shared Task Flows

The following table briefly describes the task flow scenarios when object modifications appear on one or more pages.

<table>
<thead>
<tr>
<th>If the object is...</th>
<th>Then the modifications...</th>
</tr>
</thead>
<tbody>
<tr>
<td>Not part of a shared task flow</td>
<td>Don't appear on other pages</td>
</tr>
</tbody>
</table>
If the object is... | Then the modifications...
---|---
Part of a shared task flow | Do appear on all pages that include the shared task flow

**Caution:** When shared task flows include embedded logic that uses data from the page, the logic can override the application changes that you make in Page Composer.

---

## Modify Page Content Using Page Composer

### Modify Page Components Using Resource Catalog

Use the Resource Catalog to modify pages. This catalog provides a selection of task flows, portlets, and layout components. Open the Resource Catalog in Page Composer from either Design view or Source view.

Aspects of components pertinent to page modifications include:

- Opening the Resource Catalog
- Adding components
- Hiding components manually
- Hiding components programmatically

### Before You Start

Activate a *sandbox*.

### Open the Resource Catalog

In Design view:

1. Open the page that you want to modify in Page Composer.
2. From the existing components, select the one that you want to be the parent component.

   **Tip:** Alternatively, use the Add Box icons (Add Box Above, Add Box Below, Add Box Left, and Add Box Right) to insert a box component. Then select the component as the parent component.

3. Click the **Add Content** button associated with the parent component.

   The Resource Catalog appears.

### Add Components

In Design view:

1. Open the Resource Catalog.
2. In the Resource Catalog, find the component that you want to add.
3. Click the **Add** icon associated with the component.
4. Cut and paste, or drag and drop the component to place it.
In Source View:

1. Select the container component in the selection pane.
2. In the Source view toolbar, click Add Content.
3. In the Resource Catalog, find the component you want to add.
4. Click the Add icon associated with the component.

**Note:** If you added an HTML markup with references to an external CSS as your component, ensure that:

- The CSS has `Access-Control-Allow-Origin` header in its server.
- Your `link` tag has `crossorigin` attribute specified.

## Hide Components Manually

Use the **Show Component** property to specify whether the component appears to users at runtime. By default, all components are visible. To manually hide a component, deselect **Show Component** on the Component Properties dialog box.

If the component is a child component, then deselecting the **Show Component** property hides only the child component.

If the component is a parent component, then deselecting the **Show Component** property of the parent component hides the parent and all child components it contains. So, when you hide a parent component, you automatically hide all child components.

You can do any of the following:

- Hide a child component directly
- Hide a child component from within the parent component
- Hide a parent component and all child components

### To hide a child component directly:

1. Click the **Edit** icon in the header of the child component. This opens the Component Properties dialog box.
2. Click the **Display Options** tab.
3. Deselect **Show Component**.
4. Click **OK**.

### To hide a child component from within the parent component:

1. Click the **Edit** icon on the containing box's toolbar.
2. Click the **Child Components** tab.
3. Deselect the box next to the component you want to hide.
4. Click **OK**.

### To hide a parent component and all child components:

1. Click the **Edit** icon in the box header.
2. Click the **Display Options** tab.
3. Deselect **Show Component**.
4. Click **OK**.

## Hide Components Programmatically

You can add an Expression Language (EL) expression to a component that enables you to set a condition for hiding the component. For example, suppose you have two check boxes (1 and 2) on a page. You also have a button (B) that you
want to be visible only if check box 2 is selected. To step through the logic, ask yourself questions such as the ones in the following table.

<table>
<thead>
<tr>
<th>Question</th>
<th>Answer</th>
<th>Purpose of Question</th>
</tr>
</thead>
<tbody>
<tr>
<td>What’s the condition?</td>
<td>Check box 2 is selected</td>
<td>Determines what the occurrence, or event, is.</td>
</tr>
<tr>
<td>What action or event must happen?</td>
<td></td>
<td>Determines the component that triggers the event.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Determines what expression to write.</td>
</tr>
<tr>
<td>What happens when the condition is met?</td>
<td>Button “B” appears.</td>
<td>Determines the effect of the action.</td>
</tr>
<tr>
<td>What happens when the event happens?</td>
<td>Show the component: Button B</td>
<td></td>
</tr>
<tr>
<td></td>
<td>(The implication is that button B is hidden until the event occurs.)</td>
<td></td>
</tr>
<tr>
<td>What property determines whether a component is visible?</td>
<td>The Show Component property</td>
<td>Determines the property the code affects.</td>
</tr>
</tbody>
</table>

So the logic is: If 2 is checked, then the Show Component property of B is activated.

You place the expression on the component that receives the action.

Here’s a sample code that you may add to the component.

```
#{if checkbox2.selected = true}
```

After you think through the logic and find the correct expression, add it to the property. You can add an expression using the expression builder for the Show Component property only on dashboard pages; not on work area pages. Also, only administrators can perform this task.

To open the EL Editor and add an expression to a property for a dashboard page:

1. Click the Edit icon in the component header.
2. Click the Display Options tab.
3. Click the Edit icon next to the Show Component property, and select Expression Builder.
4. Add an expression to check for an event or condition, and set the property. Based on the result, turn the property on or off.

To hide a parent component and all child components programmatically for a dashboard page:

1. Click the Edit icon in the box header.
2. Click the Display Options tab.
3. Click the Edit icon next to the Show Component property, and select Expression Builder.
4. Add an expression to check for an event or condition, and set the property. Based on the result, turn the property on or off.

Related Topics
- Overview of Sandboxes
- Create and Activate Unified Sandboxes
Modify Pages

You can modify your user interface (UI) components by updating their properties. For example, you can change field labels, hide components, or make a check box required.

When you use Page composer, you always begin in Design view. You can use Design view to add content and make layout changes. But not in all pages. For these other pages, you must use Source view to make content and layout changes.

**Note:** Any changes you make apply only to the page that you're on. These changes also apply to all or specific groups of users. But that depends on the context layer you choose when you make your changes.

Before You Start

Activate a sandbox.

Modify UI Components

1. Click your user image or name in the global header and select **Edit Pages**.
2. Select the context layer in which you want to make your changes. For instance, you could make changes for only specific job roles.
   
   **Note:** When you modify a UI component for a specific **job role**, that job role must be assigned to you for you to test the application change in the sandbox. Your security administrator can either assign the job role to you directly, or make the job role self-requestable for you to add it yourself from the resource directory.
3. By default, you start in the Design view, which lets you navigate to the component you want to modify.
   
   To use Source view, you must select **Source** from the View menu. This menu isn't displayed by default. To display the View menu, set the **Source View for Page Composer Enabled** profile option (FND_PAGE_COMPOSER_SOURCE_VIEW) to Yes.
4. When you have found your UI component, click the **Select** button on top of the page.
5. Hover over the UI component until a border appears around it, and then click it.
6. Select **Edit Component**.
7. Update the component's properties to make your changes.

Each component has its own set of properties, which may include some of the properties in this table. In Design view, you only get the main properties. To get all properties, you have to use Source view.

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Description</td>
<td>Text used by screen readers, for information in addition to what is provided in the Short Desc property.</td>
</tr>
<tr>
<td>Label</td>
<td>Display text for the component, for example, the field prompt or the single prompt for a group of check boxes.</td>
</tr>
<tr>
<td>Read only</td>
<td>Whether users can edit the component, for example, if a check box can be selected or not.</td>
</tr>
</tbody>
</table>
### Related Topics

- Examples of Working with Context Layers
- Overview of Configuring Home Page and Navigation
- Role Provisioning and Deprovisioning
- Overview of Sandboxes
- Create and Activate Unified Sandboxes

 Modify Tabs on Pages

This example demonstrates how to modify tabs on pages using Page Composer. The following table summarizes the key decisions for this scenario.

<table>
<thead>
<tr>
<th>Decisions to Consider</th>
<th>In this Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>Who do you want to make the changes for?</td>
<td>All users</td>
</tr>
<tr>
<td>What changes are you going to make?</td>
<td>Hide the Incentive Management tab and the Projects tab from the Worklist: Notifications and Approvals page</td>
</tr>
</tbody>
</table>
Before You Start
Activate a sandbox.

Modify Tabs
Do the following:

1. From the Navigator menu, select Tools > Worklist.
2. Click your user image or name in the global header, and select Edit Pages in the Settings and Actions menu.
3. Select Site layer as the context layer, which affects all users.
4. In the View menu at the top of the page, make sure that the Design view is selected, which lets you navigate to the component you want to modify.
5. Click the Select tab.
6. Hover over any tab on the page, and click when the cursor turns to a magnifier and a blue outline appears around the tab.
7. Select Edit Parent Component.
8. In the Component Properties dialog box, click the Children tab.
9. Deselect Incentive Compensation and Projects to hide these tabs from the page.
10. Click OK.

Related Topics
- Overview of Sandboxes
- Create and Activate Unified Sandboxes

Change Page Layout
You can configure a page layout to define the number, placement, and orientation of content regions on your pages. Ordinarily, you set the layout style while creating a page. But for some layouts, you can change the layout style even after adding content to the page.

Note: You can't change the page layout for all pages.

Prerequisites
Activate a sandbox.

Change Page Layout

1. Open the page you want to edit.
2. Click your user name or image and select Edit Pages to open Page Composer.
3. From the View menu, select Design.
4. Click Change Layout.
5. Select the new layout.

Related Topics
- Overview of Sandboxes
- Create and Activate Unified Sandboxes
Modify Dialog Box Content

Use Page Composer and work in source view to modify the content in your dialog boxes.

Before You Start

Activate a sandbox.

Modify Dialog Boxes

To modify dialog box content:

1. Open the page where the dialog box appears, and then open Page Composer.
2. From the View menu, select Source. You must be in Source view to modify dialog box content.
3. Select the button that opens the dialog box.
4. Click Edit to open the Component Properties dialog box.
5. Click the Child Components tab.
6. Edit the dialog box content.
7. Click Apply to save your changes, then OK to save your changes and close the Component Properties dialog box.

Related Topics

• Overview of Sandboxes
• Create and Activate Unified Sandboxes

Make Application Changes Visible Based on User Roles

This example demonstrates how you can make application changes visible to a specific user role.

To control page components conditionally based on user role:

• Create security privileges
• Add an Expression Language expression to the component property that you want to control

The following table summarizes the key decisions for this scenario.

<table>
<thead>
<tr>
<th>Decisions to Consider</th>
<th>In This Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>What's the page object you're securing?</td>
<td>Reports link</td>
</tr>
<tr>
<td>To which user role you want to make the changes visible?</td>
<td>Hiring managers, Sherry Callaway and Terrance Whitaker</td>
</tr>
<tr>
<td></td>
<td>Note: Create a role just for testing application changes. Call it DEVCUST_TEST_ROLE. When you're sure that the application change works, change the security to the appropriate role.</td>
</tr>
</tbody>
</table>
Decisions to Consider | In This Example
--- | ---
Which expression to add for verifying whether a user has the appropriate privilege? | `#{securityContext.userGrantedPermission['MANAGERREPORTS LINK_PRIV']}`

As a prerequisite, activate a sandbox.

Make Application Changes Visible to a User Role

Follow these steps:

1. Create a privilege.
2. Add the 'MANAGER REPORTS_LINK_PRIV' object to 'DEVCUST_TEST_ROLE'.
3. Assign DEVCUST_TEST_ROLE to Terrance and Sherry.
   
   **Note:** Make sure that Terrance and Sherry have access to the page before you make any changes.

4. Open the page, having the Reports link that you want to modify, in Page Composer, and select the Reports link component.
5. Click the **Edit** icon. This opens the Component Properties dialog box.
6. Click the **Display Options** tab.
7. Click the **Edit** icon next to the Show Component property, and select **Expression Builder**....
8. Add an expression to verify whether the user has the appropriate privilege. Use this sample code:
   
   ```java
   #{securityContext.userGrantedPermission['MANAGER_REPORTS_LINK_PRIV']}
   ```
9. Save the property changes, and close Page Composer.

After you change the show component property, whenever users open this page, the application evaluates the expression. Since Sherry and Terrance have the privilege, the Show Component property evaluates to be selected. Hence, Sherry and Terrance can see the Reports link while all other users can't.

Related Topics

- Overview of Sandboxes
- Create and Activate Unified Sandboxes

FAQs for Page Content Configuration

**How can I move page components?**

To move page components using Page Composer:

- In **Design view**, drag and drop the component
- In **Source view**, do any of the following:
  - Cut and paste the component
  - Drag and drop the component
  - Open the Component Properties for the container component and rearrange the components on the Child Components tab
How can I delete components from a page?
Click the Delete icon in the component header in Page Composer.

Caution:
- Delete a component only if you’re certain that no other components or processes depend on that component. If you’re unsure whether any dependencies exist, then hide the component instead of deleting it.
- If you delete a parent component, you delete all the child components automatically.

What happens if my configurations make the page inaccessible?
If your configurations make a page inaccessible, log a service request using My Oracle Support at https://support.oracle.com.

How can I reset a page or task flow to a previously saved version?
To reset a page to a previously saved version or the original ready-to-use state, click Reset Page in Page Composer (Design or Source view).
To reset a task flow to a previously saved version or the original ready-to-use state, click Reset Task Flow in Page Composer (Source view only) while you have the task flow open.

Configure the Global Page Template

The global page template provides a common header area and the footer panel for all pages in your application. You can use Page Composer to configure the global page template. To open the global page template in Page Composer, click your user image or name in the global header, and select Edit Global Page Template from the Administration menu.

You can make the following changes to the global page template:
- Add components
- Edit components
  Example: Add expression language to hide the tags link
- Delete components
  Example: Remove the tags link

Tip: When you move your cursor over the global page template, the areas that you can edit display a blue outline.

Adding Components to the Global Page Template

To add components to the global page template:
1. Open the global page template in Page Composer.
2. Select the portion of the global header to which you want to add a component, and click Add Content.
3. In the component catalog, select Components to display the list of available components.
4. Click the Add button associated with the component you want to add.

The component appears in the global header.
5. Change component properties, as appropriate. For example, if you added the Text component, enter the text that you want to display.

   **Note:** If you added an HTML markup with references to an external CSS as your component, ensure that:
   - The CSS has `Access-Control-Allow-Origin` header in its server.
   - Your `link` tag has `crossorigin` attribute specified.

6. After completing your changes, click **Close**. When prompted, click **Save** to save your changes.

### Editing Components in the Global Page Template

To edit components in the global page template:

1. Open the global page template in Page Composer.
2. Select the component that you want to edit.
3. Click **Edit**.
4. Edit the component properties, and click **OK** to save your changes.
5. After completing your changes, click **Close**. When prompted, click **Save** to save your changes.

### Deleting Components from the Global Page Template

To delete components from the global page template:

1. Open the global page template in Page Composer.
2. Select the component that you want to delete.
3. When you move your cursor over the global page template, the areas that you can edit display a blue outline.
4. Click **Delete**. When prompted, click **Delete** to delete the component.
5. After completing your changes, click **Close**. When prompted, click **Save** to save your changes.

For information about making skin modifications, such as selecting a different color palette, see the Oracle Fusion Applications Extensibility Guide for Developers.

### Saved Search Management

#### Make Saved Searches Available to All Users

Use Page Composer at the site layer to create and edit saved searches, and make them available for all users. Create and edit saved searches using Page Composer at the **site layer**. Users can run these saved searches again later to use the same search criteria and other settings. You must create or edit saved searches only at the site layer to make them available for all users.

#### Create and Edit Saved Searches for All Users

Follow these steps:

1. Activate a **sandbox**.
2. On the search page that has a **Save** button, click your user image or name in the **global header**, and select **Edit <Page Name> Pages** in the **Settings and Actions** menu.
3. If prompted to select a context layer, select the site layer to open the search page in Page Composer.

4. From the View menu, select Design.

5. Create and edit saved searches.

   Note: The steps for creating and editing saved searches are the same regardless of whether you’re working on saved searches for yourself or for all users.

6. Save your changes and close Page Composer.

7. After testing your changes, publish the sandbox to make your changes available to all users.

Related Topics
- Overview of Sandboxes
- Create and Activate Unified Sandboxes
- What gets saved when I create a saved search for searches with multiple criteria

Save Searches with Multiple Criteria

On many pages, you can run a search with multiple search criteria to find specific business objects. Some of these searches have a Saved Search list, as well as a Save button after the search criteria. A saved search captures search criteria and other settings so that you can easily run the same search again later. Aside from using any predefined saved searches, you can create and edit them for your own use. If you have the appropriate roles, you can also create and edit saved searches for other users using Page Composer.

The following figure shows an example of a search with multiple search criteria fields and a Save button. For each field, you can select an operator and enter search terms. You can also select from the Saved Search list to use an existing saved search.

![Search](image)

Create Saved Searches

Follow these steps:

1. Go to a search that has a Save button.

2. Optionally add or reorder fields, if available.

3. Enter your search criteria values, and, click the Save button.

4. Name your saved search and define its settings:
   - **Set as Default:** The saved search is automatically selected whenever you open this page.
   - **Run Automatically (if available):** The saved search runs on this page as soon as you select it from the list of saved searches.

   If you select both options, then the saved search automatically runs whenever you open this page.
5. Close the dialog box.
Your saved search can be limited to the current page, or in some cases available in other searches for the same object.

### Change the Search Criteria in Saved Searches

Follow these steps:

1. Select the saved search if it's not selected already.
2. Set your search criteria, including any additional fields.
3. Click the **Save** button.
4. If the saved search is one that you created, save without changing the name.
   
   If it's predefined, then you can't overwrite it, so you create a new saved search with a unique name.

### Change Settings or Rename and Delete Saved Searches

Follow these steps:

1. Go to a search that has a **Save** button.
2. Select **Personalize** from the **Saved Search** list.
3. In the Personalize Saved Searches dialog box, select a saved search.
4. Change any of the settings, where available.
   
   - **Set as Default**: The saved search is automatically selected whenever you open this page.
   - **Run Automatically**: The saved search runs on this page as soon as you select it from the list of saved searches.
   - **Show in Search List**: The saved search is available for you to select and run on this page.
     
     - You can still find hidden saved searches in the Personalize Saved Searches dialog box.
     - You can't change this setting if the saved search is currently selected on your page.

   **Note**: Some settings can't be changed for predefined saved searches. What you do change applies only to you, unless you're editing the saved search within Page Composer.

5. If you selected a saved search that you created, then you can rename or delete it. You can't do so for predefined saved searches.
6. Save your changes and close the dialog box.

### Related Topics

- Create Watchlist Items
- What gets saved when I create a saved search for searches with multiple criteria

### Save Keyword Searches with Filters

One type of search you might find on the page is a keyword search with filters. Some of these searches have predefined saved searches, and you can also create and edit saved searches for your own use. A saved search captures entered search terms, filters, and other settings so that you can easily run the same search again later. If you have the appropriate roles, you can also create and edit saved searches for other users using Page Composer.

### Create and Edit Saved Searches

Follow these steps:

1. Enter search terms in the search field, and click **Search**, or select a saved search in the autosuggest.
2. Click the **Show Filters** link if filters are currently hidden.
3. Optionally organize filters as follows, depending on what's available to you:
   - Use the **Add** or **Reorder** buttons.
   - Click the **Personalize** icon to show, hide, or reorder filters, and click **OK**.
4. Set filter values or select filters, and click the **Save** button.
5. Name your saved search.
   - To create a saved search, enter a new, unique name.
   - To update an existing saved search that you created, save with the original name.

You can't overwrite predefined saved searches by using their names, but you can create a copy with a unique name.
6. Click **Set as Default** if you want the saved search to be automatically selected whenever you open this page.
7. Click **OK**.

Your saved search can be limited to the current page, or in some cases available in other searches for the same object.

### Change Settings or Rename and Delete Saved Searches

Follow these steps:

1. Click the **Show Filters** link if filters are currently hidden.
2. Select **Manage** from the **Saved Search** list.
3. Define settings for any saved search, predefined or user-defined, in the Manage Saved Searches dialog box:
   - **Default**: The saved search is automatically selected whenever you open this page.
   - **Show in Saved Search List**: The saved search is available for you to select and run on this page. You can still find hidden saved searches in the Manage Saved Searches dialog box.

   **Note**: Changes you make to predefined saved searches apply only to you, unless you're editing the saved search within Page Composer.

4. The **Active** setting identifies the saved search that's currently selected on the page. You can designate a different active saved search to have that saved search automatically selected as soon as you click **OK** in this dialog box.
5. For user-defined saved searches only, you can also rename or delete the saved search.
6. Click **OK**.

### Infolet Management

#### Overview of Configuring Infolets

You can configure **infolets** that aggregate key information for a specific area, for example, social networking or personal profile. Your users use the page control icons on the home page with the panel or banner layout to open the infolets. Or, if it's a home page with news feed layout, they use the infolets tabs in the Analytics section.
If your users don’t find infolet icons on the home page, you can enable them by clicking the Home Configuration tab in the Structure work area. Let’s look at some tasks you can do to configure infolets.

- Create infolets.
- Add content to infolets. For example, you can add a task flow or a performance tile report, and if an infolet contains a performance tile report, then you can add a link to a detailed report in the same infolet.
  
  **Note:** In the context of infolets, report can mean analysis.
- Edit infolets. For example, edit infolet content and add, change, or remove link to detailed report.
- Delete infolets.

**Related Topics**

- Personalize Infolets
- Configure Home Page Navigation

**Create Infolets**

Use infolet pages to create infolets. For some product-specific infolet pages, you can’t create infolets.

1. Activate a sandbox.
2. Open an infolet page.
3. Click your user image or name in the global header, and on the Settings and Actions menu, select Edit Pages.
4. Select a context layer. The infolet page opens in Page Composer.
5. Click the Infolet Repository icon, and select Create Infolet.

  **Note:** If you can’t find the Create Infolet menu item, it means you can't create infolets for this page.

6. Enter a title for the infolet and set its views.
   - Specify the dimensions for the front view.
   - Enable or disable the back view.
   - Enable or disable the expanded view, and specify its dimensions. The dimensions of the front and the back views must be the same, but the expanded view must be bigger because it displays more details.
7. Click Save and Close.
8. Add content to the infolet views and link a detailed report to the infolet.
9. Preview the infolet’s front view, drill down to the detailed report, and then preview the back and the expanded views.
10. After you made changes, click Close to leave Page Composer.
11. Test your changes and publish the sandbox to make the new infolet available to your users.

**Related Topics**

- Overview of Sandboxes
- Create and Activate Unified Sandboxes
Manage Infolets

On the infolet page, use the options available on each infolet to manage it. You can add and edit infolet content and delete infolets. You can’t add a business intelligence dashboard to an infolet because a dashboard report is generally bigger than an infolet.

Before You Start

You must open the infolet page in Page Composer.

1. Activate a sandbox.
2. Open an infolet page.
3. Click your user image or name in the global header, and on the Settings and Actions menu, select Edit Pages.
4. Select a context layer. The infolet page opens in Page Composer.

Note: After you’re done making changes, click Close to leave Page Composer, test your changes, and publish the sandbox.

Add Content to Infolets

You can add content to the infolet’s front, back, or expanded view.

1. Select the infolet’s view that you want to add content to.

   Note: To open infolet's back view, click the Back View icon on the bottom right corner of the infolet. And to open expanded view, click the Expanded View icon on the bottom right corner of the infolet’s back view.

2. Click Add Content on the infolet. You can find this button on any of the infolet’s views, but only if you don’t have any existing content in that view.

3. Search and select a performance tile or a task flow, and click Add. You can browse the business intelligence (BI) catalog to find the analytics and reports that you want to add.

4. Close the Add Content dialog box.

Edit Infolet Content

1. Click the Actions icon on the top right corner of the infolet, and select Edit Content.

   Note: To edit the infolet content of the back or expanded view, open the view and use the Actions icon on that view.

2. Click Add Content to replace the existing content of the infolet.

3. Search and select a performance tile or a task flow, and click Add. You can browse the business intelligence (BI) catalog to find the analytics and reports that you want to add.

4. Close the Add Content dialog box.

Edit Title and Views

To edit title and views of an infolet, click the Actions icon on the top right corner of the infolet, and select Edit Title and Views.
Link Detailed Reports to Infolets
To provide detailed information about a subject matter on an infolet, you can add a link to a detailed report. After you add the link, your users can click anywhere in the infolet area to drill down to that detailed report. The detailed report doesn't replace the existing infolet content.

Add Links to Detailed Reports
First, add a performance tile report to the infolet content, and then follow these steps:

1. Click the Actions icon on the top right corner of the infolet, and select Link Detailed Report.
3. Search and select a report, and click Add to add it to the infolet.
4. Close the Add Content dialog box.
5. Click Done.

Edit or Remove Detailed Report

1. Click the Actions icon on the top right corner of the infolet, and select Edit Detailed Report.
2. On the Detailed Report page, you can edit or remove the detailed report.
   - To edit, click Edit Report and make changes to the detailed report.
   - To remove, click Remove Report.
3. Click Done.

Delete Infolets
To delete an infolet, click the Actions icon on the top right corner of the infolet, and select Delete.

Caution: You can't easily retrieve an infolet once you delete it. But if you have accidentally deleted any infolet, don't worry. Contact My Oracle Support at https://support.oracle.com.

Edit Infolet Visibility
You can show or hide an infolet on the infolet page. To edit the visibility settings of an infolet, follow these steps:

1. Click the Actions icon on the top right corner of the infolet, and select Edit Visibility.
2. Select one of these options:
   - Yes: The infolet appears on the infolet page.
   - No: The infolet doesn't appear on the infolet page.
   - EL expression: The evaluation of the EL expression decides whether the infolet appears on the infolet page.

Related Topics
- Personalize Infolets
- Overview of Sandboxes
- Create and Activate Unified Sandboxes
FAQs for Infolet Management

Why is the icon for my infolet page not available in the page control on the home page?
The icon for your infolet page may be hidden. Change the visibility setting of the icon using the Home Configuration page of the Structure work area. To open this page, select Configuration > Structure from the Navigator menu, and then click the Home Configuration tab.

Related Topics
• Configure Home Page Navigation

How can I rename an icon for an infolet page in the page control on the home page?
You can rename an icon for an infolet page using the Home Configuration page of the Structure work area. To open this page, select Configuration > Structure from the Navigator menu, and then click the Home Configuration tab.

Related Topics
• Configure Home Page Navigation

What's the difference between a performance tile report and a detailed report added to the infolets content?
Performance tile report shows data in the small infolet format. When you add a performance tile report to an infolet, users can see only the summary information about the subject matter. But this report doesn't provide detailed information.
To provide detailed information about the subject matter on the same infolet, add a link to a detailed report. Users can click this link to gather more information.

Configure Infotiles on a Page

You can use Page Composer to decide how your infotiles appear on your page. You can show, hide, or reorder them any way you like.

1. Activate a sandbox.
2. Go to the page with your infotiles.
3. Click your user image or name in the global header, and select Edit Pages.
4. Click the Configure Layout icon.
5. Make changes to your page.
   • Select the infotiles you want to show on your page, and deselect the ones you want to hide.
   • Hover over the infotile you want to reorder, and click Move Up or Move Down.
6. Click OK.
7. Click Close.
8. Test your changes and publish your sandbox.

Related Topics
• Overview of Sandboxes
• Publish Unified Sandboxes

New Page Creation

Create Pages for Hosting Third Party Applications

Use the Page Integration pages to create pages for hosting third party applications to address needs specific to your organization. All these new pages are grouped in a single group on the Navigator menu and springboard.

Prerequisites

Activate a sandbox.

Creating the First Page

Follow these steps:

1. From the Navigator menu, select Configuration > Page Integration.
2. Click New Page.
3. On the Create Page page, enter a page name.
4. In the Category Name field, enter a name for the group to place your page in.
   - **Note:** You can move your new pages to other groups later using the Structure page.
5. Search and select an icon for the page.
6. From the list of application roles, select the role to whom you want to grant access to the page. This list contains only custom roles because you can't modify predefined roles.
7. In the Web Page field:
   - Enter the application URL that you want to host on this page.
   - Alternatively, you can create a secure web page URL:
     i. Click the Create Secure Web Page URL icon.
     ii. Select the name of the web application.
     iii. Enter the destination for the web application.
     iv. Enter a secure token name.
     v. Save and close the Create Secure Web Page URL dialog box.
   - **Note:** In a secure web page, the application validates the secure token and uses it to authenticate web services within the end-user context. Using this mode of modified access, a partner can directly perform an action or display information to the specific user without any additional authentication.
8. Click Save and Close.

New pages are secure. Your security administrator must assign the privileges shown on the Page Integration Overview page to users such that they can access these pages.

If you have only one page in a group, then that page icon appears at the top level (not in any group) on the springboard. However, such page icons appear in their respective categories on the Navigator menu.
Creating Subsequent Pages

After creating the first page, follow these steps to create more pages:

2. Enter a page name.
3. Follow steps 5 to 7 in the procedure for creating the first page.
4. Click **Save and Close**.

After you have created the first page, all subsequent pages are added in the same group as that of the first page, by default.

Related Topics
- Overview of Sandboxes
- Edit and Reorder Groups and Page Entries for Navigation
- Create and Activate Unified Sandboxes

Manage Pages Hosting Third Party Applications

After creating pages for hosting third party applications, manage them using the options available on the Page Integration Wizard: New Pages page and the Page Integration Wizard: Existing Pages page.

You can do the following actions:
- Edit page settings.
- Add tabs to new and existing pages.
- Edit page tabs.
- Rename Categories.
- Navigate to pages.

Prerequisites
Activate a **sandbox**.

Editing Page Settings

Follow these steps:

1. On the Page Integration Wizard: New Pages page, click the name link for the page that you want to edit.
2. On the Edit Page page, make the required changes.
3. Click **Save and Close**.

You can make the following changes to a page:
- Change the icon for the page.
- Change the web page URL that you want to host on this page.
- Add tabs to the page.
- Delete the page.

**Note:** If a page has additional tabs, apart from the one created by default with the page, then you can delete the page only after deleting its tabs.
• Edit the page tabs.

Adding Tabs to New and Existing Pages
When you create a page, a page tab is created by default. You can then add more tabs to your new and existing pages, as required.

To add tabs to new pages, follow these steps:

1. On the Page Integration Wizard: New Pages page, click the name link for the page containing the tab that you want to edit.
2. On the Edit Page page, click Add Tab.
3. On the Create Tab page, enter a tab name.
4. Search and select an icon for the page.
5. From the list of application roles, select the role to whom you want to grant access to the page. This list contains only custom roles because you can’t modify predefined roles.
6. In the Web Page field:
   - Enter the application URL that you want to host on this page.
   - Alternatively, you can create a secure web page URL:
     i. Click the Create Secure Web Page URL icon.
     ii. Select the name of the web application.
     iii. Enter the destination for the web application.
     iv. Enter a secure token name.
     v. Save and close the Create Secure Web Page URL dialog box.
7. Click Save and Close.

To add tabs to existing pages, follow these steps:

1. Click the Page Integration Wizard: Existing Pages icon on the left region of the Page Integration Wizard: New Pages page.
2. Click Add Tab to Existing Page.
3. In the Select Page dialog box, select a page to add a new tab to.
4. Follow steps 2 to 7 in the procedure for adding tabs to new pages.

Editing Page Tabs
To edit page tabs, follow these steps:

1. On the Page Integration Wizard: New Pages page, click the name link for the page containing the tab that you want to edit.
2. On the Edit Page page, click the name link for the page tab that you want to edit.
3. On the Edit Tab page, make the required changes.
4. Click Save and Close.

You can make the following changes to a page tab:

• Change the icon for the page.
• Change the web page URL that you want to host on this page tab.
• Delete the page tab.
Renaming Categories

All pages that you create using the Page Integration Wizard: New Pages page are grouped in a single category. To rename the category for all pages, click **Rename Category** on the Page Integration Wizard: New Pages page.

Navigating to Pages

On the Page Integration Wizard: New Pages page, click the icon for the page that you want to navigate to, and view its content.

Related Topics

- Overview of Sandboxes
- Configure Links for Page Entries
4 User Interface Text Modification

Considerations for Selecting a Tool to Change Text

You can modify and replace words or phrases that appear on pages, in messages, and other areas of the UI using several tools or tasks.

Following are the tools for making text changes:

- Application Composer
- User Interface Text
- Page Composer

Multiple factors influence the option you select. For example:

- The offering you use
- The extent and scope of your changes
- The components that you modify

This table presents the navigation and offering availability options associated with the tools you can use to modify user interface text.

<table>
<thead>
<tr>
<th>Tool or Task</th>
<th>Navigation</th>
<th>Offering Availability</th>
</tr>
</thead>
<tbody>
<tr>
<td>Application Composer</td>
<td>In the Navigator, select Configuration &gt; Application Composer.</td>
<td>Oracle Engagement Cloud</td>
</tr>
<tr>
<td>User Interface Text</td>
<td>In the Navigator, select Configuration &gt; User Interface Text.</td>
<td>All applications</td>
</tr>
<tr>
<td>Page Composer</td>
<td>Click your user image or name in the global header, and select Edit &lt;Page Name&gt; Pages in the Settings and Actions menu. If the Edit &lt;Page Name&gt; Pages option isn't available in the Settings and Actions menu, then select Edit Pages instead.</td>
<td>All applications</td>
</tr>
</tbody>
</table>

Text Modification Scenarios

The following table includes possible scenarios for modifying user interface text. Compare your situation to the scenario in the table to determine the most appropriate tool for modifying text in your application.
<table>
<thead>
<tr>
<th>Task</th>
<th>Scope</th>
<th>Tool or Task</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>Simultaneously replace multiple occurrences of a word or phrase that</td>
<td>Comprehensive.</td>
<td>User Interface Text</td>
<td>Change the word &quot;Employee&quot; to &quot;Associate&quot; on every page associated with</td>
</tr>
<tr>
<td>appear on multiple pages in multiple contexts</td>
<td>The changes affects multiple pages throughout your application. You can edit the embedded help (for example, hints) using this method.</td>
<td></td>
<td>Employee Self Service, Benefits, and Payroll.</td>
</tr>
<tr>
<td>Simultaneously replace multiple occurrences of a word or phrase that</td>
<td>Comprehensive.</td>
<td>User Interface Text</td>
<td>Change the word &quot;Employee&quot; to &quot;Associate&quot; in every message associated with Employee Self Service, Benefits, and Payroll.</td>
</tr>
<tr>
<td>appear in parts of messages in the message dictionary</td>
<td>The change affects multiple messages throughout your application.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Simultaneously replace multiple occurrences of the singular and plural</td>
<td>Comprehensive.</td>
<td>User Interface Text</td>
<td>Change the word &quot;Employee&quot; to &quot;Associate&quot; and &quot;Employees&quot; to &quot;Associates&quot;.</td>
</tr>
<tr>
<td>forms of a word or phrase that appear in messages and on pages</td>
<td>The change affects multiple pages and multiple messages throughout your application.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Replace a word or phrase that appears on a specific page</td>
<td>Targeted: A page</td>
<td>Page Composer</td>
<td>Change the word &quot;Customer&quot; to &quot;Account&quot; on two specific pages.</td>
</tr>
<tr>
<td>Replace a word or phrase that appears in a specific message in the</td>
<td>Targeted: A message</td>
<td>Manage Messages task</td>
<td>Change the word &quot;Recruit&quot; to &quot;Potential Employee&quot;, but only in two specific messages. All other messages continue to use the word &quot;Recruit&quot;.</td>
</tr>
<tr>
<td>message dictionary</td>
<td>The change affects part of a specific message in the message dictionary.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Simultaneously replace a word or phrase associated with a specific</td>
<td>Targeted: A business object</td>
<td>User Interface Text</td>
<td>Change the label of the opportunity business object, from &quot;Opportunity&quot; to &quot;Deal&quot;. You want the change to affect the business object wherever it appears.</td>
</tr>
<tr>
<td>object wherever the object appears</td>
<td>The change affects a specific component of a specific message in the message dictionary.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Replace words or phrases that appear in menus and menu items</td>
<td>Targeted: Navigator menu item text</td>
<td>User Interface Text</td>
<td>Change the menu item label from &quot;Total Compensation Statements&quot; to &quot;Compensation Package Statements&quot;.</td>
</tr>
</tbody>
</table>

Regardless of the tool you use to make changes, all application changes are written in a single override bundle. Hence, the latest application changes overwrites the previous ones.
Note: If you replace text using plain text as the input value, it supersedes any changes that use the override bundle. For example, if you enter a direct string in Page Composer, then Page Composer writes these changes in a file (not string resource) containing page modifications. Hence, such changes overwrite the changes in the override bundle.

Related Topics
- Guidelines for Page Modification
- Tools for Configurations and Extensions

Overview of User Interface Text Tool

You can use the User Interface Text tool to modify text in these application components:
- User Interface
- Messages
- Global Menu Label
- Enterprise Scheduler

You can use this tool to modify or translate your text at runtime. You can also export your strings to make your changes offline, if you opt in for User Interface Text Update.

User Interface Text Update

With the User Interface Text Update opt in, you get greater control of your text, and how it's edited.

Here are some things you can do:
- Export and import your text for offline translation and modification.
- Select and replace your text more easily.
- Edit your text in an improved UI.

How to Opt In

In the Offerings work area, enable the User Interface Text Update feature:
- Offering: Any with the Application Extensions functional area
- Functional Area: Application Extensions
- Feature: User Interface Text Update
- Opt In Task: Click Continue if you're sure about enabling this feature.

Note: If you want to disable this feature, you can opt out by repeating this same flow.

Related Topics
- Configure Offerings
Bulk Text Modification

Use the User Interface Text tool to simultaneously update multiple occurrences of entire words or phrases in the user interface (UI).

You can use this tool to do the following activities for bulk text modification:

- **Sandbox** integration
- Case-sensitive and whole word searches
- Singular and plural text replacement
- Contextual previews

⚠️ **Note:** You must activate a sandbox to use the User Interface Text tool.

To use the User Interface Text tool, on the Navigator, select **Configuration > User Interface Text**. Then, click **Search and Replace** to search and replace texts in bulk. The User Interface Text tool searches text on pages and in messages in the message dictionary. The search includes user assistance only if the user assistance text is in the message dictionary. The modification functionality for this tool doesn't extend to text in service oriented architecture (SOA) processes.

In the User Interface Text tool, you can:

- Search and replace
- Preview and adjust
- Save and publish

**Search and Replace**

After you activate a sandbox and click **Search and Replace**, enter the search text and the replacement text. You can enter the singular and plural forms of whole words or phrases. You can also use the following check boxes:

- **Match Case** - To perform case-sensitive searches.
- **Match Complete Word or Phrase** - To search for an exact match of your search text value.

⚠️ **Note:** You can't perform partial word searches, nor can you use wildcard characters as part of the search text.

The following table lists the sample values that you can use as a guide while entering search text.

<table>
<thead>
<tr>
<th>Search Text</th>
<th>Expected Match</th>
<th>Match?</th>
<th>Reason</th>
</tr>
</thead>
<tbody>
<tr>
<td>Flex</td>
<td>flex</td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Flex</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>The application searches for any occurrence of your search string without regard to its position in the strings it searches.</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Unless you select <strong>Match Case</strong>, all matches are considered exact.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Search Text</td>
<td>Expected Match</td>
<td>Match?</td>
<td>Reason</td>
</tr>
<tr>
<td>----------------</td>
<td>-----------------------------------------</td>
<td>--------</td>
<td>------------------------------------------------------------------------</td>
</tr>
<tr>
<td>flex</td>
<td>flexfields</td>
<td>No</td>
<td>The application treats your search text value as a whole word. The text flex isn't the same as the text flexfields.</td>
</tr>
<tr>
<td>^Flex$</td>
<td>flexfields</td>
<td>No</td>
<td>Use ^string$ in the search field to say this string must match the complete field.</td>
</tr>
<tr>
<td>^Flex$</td>
<td>flex</td>
<td>Yes</td>
<td>Use ^string$ in the search field to say this string must match the complete field.</td>
</tr>
<tr>
<td>flex credits</td>
<td>Flex Credits Configuration</td>
<td>Yes</td>
<td>The application searches for the exact spelling and sequence of words without regard to their position in the strings it searches.</td>
</tr>
<tr>
<td>flex credits</td>
<td>flex credit shell plan</td>
<td>No</td>
<td>The application searches for the exact spelling and sequence of words without regard to their position in the strings it searches.</td>
</tr>
</tbody>
</table>
Include one or more of the following match categories in your search:

- User Interface Text
- Enterprise Scheduler Text
- Global Menu Label Text
- Multipart Validation Message

After you enter the search text and replacement text, click **Preview Changes**. The tool looks for exact whole word matches.

**Preview and Adjust**

The preview sorts the search results and presents the matches on tabs based on match categories. Data grids on each tab present the matches in rows. You can adjust each row independently. The grids on each tab are similar, but not identical.

Caution: Some tab names on the Preview Text Changes page are followed by an asterisk (*) (for example, the Messages tab). For these tabs, once you save the text changes, these changes are applied to the mainline metadata. You can't undo your changes after you preview and save them, even though you're currently in a sandbox. The changes will still remain even if you delete the sandbox.

Each row on all tabs includes:

- A view of the existing text and the immediately surrounding text for context. You can't edit the existing view.
- A preview of the replacement text and the immediately surrounding text for context. You can edit the preview.
- An option to exclude the row and the specific match you see in the row from the change.

Each row on the Messages tab includes an indicator. This indicator identifies when the search text appears in a message subcomponent, not necessarily in the message body displayed in the row. You can expand the row and view subordinate rows that display the message subcomponent containing the match and the preview, in context as previously described.

For each row in the preview, you can:

- Do nothing to accept the change as you see it.
Select **Exclude** to eliminate the row from the batch update and maintain the existing text.

Modify the replacement text to fine-tune the change for the specific match in the row. The row remains part of the batch update, even if the actual update differs from the other rows.

In the Global Menu Label Text tab, you can't update a secure JSON Web Token (JWT).

If you have multiple languages in your application and you want to make similar text changes in those languages:

1. Change your language preference
2. Search and replace text
3. Validate your changes

### Save and Publish

After you review and adjust the matches:

1. Save your changes.
2. Thoroughly test the runtime pages to make sure that every occurrence of the text is replaced, as you wanted.
3. Publish the sandbox.

Note the following points:

- Don't publish a sandbox before you visually inspect and validate all pages and messages that contain text that you updated.
- Users can view:
  - Message and enterprise scheduler text modifications when you save them, even if you don't publish the sandbox.
  - Page text modifications when you publish the sandbox.

**Related Topics**

- **Overview of Sandboxes**
- **Create and Activate Unified Sandboxes**

### Modify Text with User Interface Text Update

You can use the User Interface Text Update feature to modify or translate multiple strings together at runtime.

**Before You Start**

- Activate a sandbox.
- Opt in for User Interface Text Update feature.

**Modify Text**

1. On the Navigator, select **Configuration > User Interface Text**.
2. In the **Find** field, enter the word or phrase you want to search for.
3. In the **Replace** field, enter your replacement word or phrase.
4. You can specify additional search parameters such as **Match Case** or **Match Complete Word or Phrase**.
5. Select the language in which you want to modify your text.
6. Select the application components you want to modify your text for.
7. Click Search. The search results are grouped into different tabs based on the application components you select.

**Caution:** Tabs marked with an asterisk (*) have live changes that are saved directly to the mainline metadata. You can't undo these changes once you save them. Make sure you review changes carefully before replacing your strings.

8. Preview and adjust changes in your search results.
   Here are a few things you can do:
   - Exclude specific strings you don't want to change by selecting Exclude for those particular rows.
     For results in the User Interface Text component, you can select options in the Exclude menu for advanced string selection. Your options are:
     - None: Don't exclude any result.
     - All: Exclude all results.
     - None (Page): Don't exclude any result on the page.
     - All (Page): Exclude all results on the page.
     You can also choose how many results you want to display on a page for this component.
   - Manually tune your replacement text for specific strings.
     For instance, let's say you have a string that says "Generate a bill". If you replace "bill" with "invoice", this string becomes "Generate a invoice", which is incorrect. You can tune your replacement text for this string to say "Generate an invoice" by manually editing the text in the replacement text box.
   - For the User Interface Text component, you can click Export All to export your search results to view in a .csv file.
   - Query your search results in the User Interface Text component to find specific strings.

9. Click Replace Strings. If you have live changes in your results, you will get a warning message. Click Yes if you have reviewed these changes carefully and want to proceed.

10. Test your changes in the sandbox.
11. Publish your changes to merge them with the mainline metadata.

**Related Topics**
- Overview of Sandboxes
- Publish Unified Sandboxes

**Overview of Translating Modified Text**

If you install and use multiple languages in your application and you modify text, then enter translations of the modified text for all languages. You can enter translations for existing and newly added strings manually at runtime. You can also translate your strings offline. But you need to opt in for the User Interface Text Update feature for offline translation.

You can use several configuration tools to update or add strings. For example, you can use lookups to add translations at runtime, or you can use the User Interface Text tool to update all strings and enter their translations.
Translate Existing Strings at Runtime

This example demonstrates how to translate existing strings manually at runtime.

The following table summarizes the key decisions for this scenario.

<table>
<thead>
<tr>
<th>Decisions to Consider</th>
<th>In this Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>What's the sandbox name that you want to use for translating existing strings?</td>
<td>Sandbox1</td>
</tr>
<tr>
<td>What's your base language?</td>
<td>English</td>
</tr>
<tr>
<td>What's the existing text that you want to modify?</td>
<td>Page</td>
</tr>
<tr>
<td>What's the replacement text that you want to replace the existing text with?</td>
<td>Work area</td>
</tr>
</tbody>
</table>

Enter Text Translations for Existing Strings

1. Activate Sandbox1.
2. On the Navigator, select Configuration > User Interface Text.
3. Click Search and Replace.
4. In the Search For field, enter the text, "page".
5. In the Replace With field, enter the text, "work area".
6. Click Preview Changes to preview and adjust the matches, as necessary.

**Caution:** Some tab names on the Preview Text Changes page are followed by an asterisk (*) (for example, the Messages tab). For these tabs, once you save the preview text changes, these changes are applied to the mainline metadata. You can’t undo your changes after you preview and save them, even though you're currently in a sandbox. The changes still remain, even if you delete the sandbox.

7. Save your text changes.

**Note:** Repeat steps 4 to 7 for any text changes required in other installed languages.

8. Test and verify all messages and pages affected by the text changes. Be sure to test across all applications.
   Your replacement text for the existing string is now available to all users.

Translate New Strings Added Using Configuration Tools

This example demonstrates how to translate new strings that were added using configuration tools. While creating strings using configuration tools, such as the Structure page, always use the same language, that is, your base language.

The following table summarizes key decisions for this scenario.

<table>
<thead>
<tr>
<th>Decisions to Consider</th>
<th>In this Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>What's the sandbox name that you want to use for translating newly added strings?</td>
<td>Sandbox2</td>
</tr>
<tr>
<td>What's the language you want to translate your newly added English string to?</td>
<td>French</td>
</tr>
<tr>
<td>What's the newly created English text that you want to translate in French?</td>
<td>Computer</td>
</tr>
<tr>
<td>What's the French replacement text that you want to replace the newly created English text with?</td>
<td>Ordinateur</td>
</tr>
</tbody>
</table>

Entering Text Translations for Newly Added Strings

1. Activate Sandbox2.
2. From the Navigator menu, select Configuration > User Interface Text.
3. Select French as the language you want to translate your new English string to.
4. In the Search For field, enter the newly created English string "computer". You must search in English because the French equivalent has not yet been created.
5. Enter the French string "ordinateur" as the replacement text.
6. Click Preview Changes to preview and adjust the matches, as necessary.

Caution: Some tab names on the Preview Text Changes page are followed by an asterisk (*) (for example, the Messages tab). For these tabs, once you save the preview text changes, these changes are applied to the mainline metadata. You can't undo your changes after you preview and save them, even though you're currently in a sandbox. The changes still remain, even if you delete the sandbox.

7. Save your text changes.
8. Test and verify all messages and pages affected by the text changes. While testing, you must sign in with French as the language and use sandbox2.

   Note: Repeat steps 3 to 8 for every active language.

   Your replacement text for the newly added string is now available to all users.

Translate Strings Offline

You can translate your modified strings offline if you opt in for the User Interface Text Update feature. To translate your strings offline, export them, add the necessary translations, and then import them back into the application.

   Note: You can export only modified strings.

Before you export your strings for translation, make sure you meet these conditions:

- Activate a sandbox.
- Opt in for User Interface Text Update.

Export Strings

1. Click Navigator > Configuration > User Interface Text.
2. Click the Export tab.
3. Select the source language for the text you want to export for translation.
4. Click Export, and download the file with all the modified strings that have been published as well as those that are in the current sandbox.

Translate Strings

1. Extract the .zip file you downloaded, and open the .xlf file inside it using a text editor of your choice.
2. Locate the strings you want to edit. You can find these strings inside <source> and </source>.
3. Replace the <target/> tag following the </source> tag with the translation you need. The translated string should be inside <target> and </target>.

   Let's look at an example of an English to Spanish translation.
   
   <source>All Receipts</source>
   <target>Todos los Recibos</target>

4. Save the file when you're done.
5. Compress the translated .xlf file into a .zip file.

Import Translated Strings

1. Click Navigator > Configuration > User Interface Text.
2. Click the Import tab.
3. Select the target language you want to import the translated strings to.
4. Click Browse, and select the .zip file you want to import.
5. Click Import.

Note: After importing your strings, test your text changes inside the sandbox. Publish the sandbox to apply your changes to the application.

Related Topics
  • Configure Offerings

FAQs for User Interface Text Modification

Can I undo text changes that I made using the User Interface Text tool if I haven't published the sandbox?

It depends on types of text changes in the sandbox. You can undo all text changes done in the user interface and global menu label by deleting the sandbox before publishing it. However, you can't undo the text changes done in messages, analyses and reports, and scheduled processes.

Can I get a report of all modified text if I want to analyze, troubleshoot, and diagnose the cause of unexpected action?

No, but you can use Configuration Setup Manager to export all your configurations to a .zip file. You can find the text changes in files ending in ".xlf.xml". These files list all text changes done in your application using browser-based tools, such as Application Composer, Page Composer, and User Interface Text. You can use the contents of these files for diagnosis and troubleshooting purposes. These files are read-only, so you can't edit their contents.

Why didn't text in my BI reports and SOA processes change when I used the User Interface Text tool to perform comprehensive text updates?

The bulk updates that you do using the User Interface Text tool affect only the text that appears on Application pages, in message diction messages, in Global Menu Label, and in Enterprise Scheduler.

Why can't I see some of my previous changes in the History tab of User Interface Text Update?

The History tab only shows the changes you made in your current sandbox. Previous changes which were made in or published from a different sandbox aren't recorded in the history for the current sandbox.


5 Theme Management

Overview of Configuring Themes and Home Page Settings

Use the Appearance work area to configure the general look and feel of your application, and the default layout and display settings of the home page. To open this work area, on the Navigator, select Configuration > Appearance. The Appearance work area has 2 tabs: Themes and Home Page Layout. Use the Themes page to configure the default home page layout, and to change the branding logo, background colors, icon styles, and so on. You can apply an existing theme to your application pages, or create your own theme and apply it. Use the Home Page Layout page to configure the display settings of the home page.

Themes and Default Home Page Layout

Use the Themes page of the Appearance work area to:

- Set the default home page layout as Panel, Banner, or News Feed. The home page with the:
  - Panel or banner layout contains a springboard with icons that you can use to open work areas. Also, based on setup, the home page shows either company announcements or social networking conversations. This information appears in a panel for the panel layout, and in a banner for the banner layout.
  - News feed layout mainly contains the Apps section and a series of updates with important information.
- Create and edit saved themes, which means themes that aren’t predefined. For example, you can change the following look and feel aspects of your application pages:
  - Logo
  - Background image
  - Panel image and style
  - Size and style of the icons on the springboard
  - Style of the cards, which appear on a page in a grid view. These cards display a summary of a single record, with attributes on the front side and optional back side. You can specify whether all cards should display with a dark-colored or a light-colored finish for users.
  - Shape of buttons, menus, and tabs
  - Colors for the background, global region, headings, page links, and buttons

Tip: While making changes on the Themes page, you can click Apply any time to preview your changes.
Display Settings of the Home Page

Use the Home Page Layout page of the Appearance work area to configure the display settings of the home page.

- For panel and banner layouts, you can specify the default content to be displayed on the home page panel and banner, and the display photo on the main panel of the home page.
- For the news feed layout, you can rename the home page sections, show or hide them, and reorder them.

Related Topics
- Define Home Page Appearance
- Overview of Sandboxes
- Create and Activate Unified Sandboxes

Create Themes

Use the Themes tab of the Appearance work area to create themes.

Before You Start
Activate a sandbox.

Create Themes

Follow these steps:

1. On the Navigator, select Configuration > Appearance.
2. From the Themes list, select your base theme.
3. From the Default Home Page Layout list, select Panel, Banner, or News Feed.
   - Panel or banner layout contains a springboard with icons that you can use to open work areas. Also, based on setup, the home page shows either company announcements or social networking conversations. This information appears in a panel for the panel layout, and in a banner for the banner layout.
   - News feed layout mainly contains the Apps section with icons to open work areas and a series of updates with important information.
4. Configure the various appearance settings for your application, as required. For example, select a branding logo, and specify color schemes.
5. On the Actions menu, select Save As.
6. Enter a theme name.
7. Optionally, deselect Apply this theme if you don’t want to apply the theme to the application immediately.
8. Click OK. If you have selected the Apply this theme check box, then your theme is saved and set as the current theme. If you haven’t selected the check box, your theme isn’t applied to the application. However, the theme is saved, and you can apply it to your application later.

Related Topics
- Overview of Sandboxes
- Create and Activate Unified Sandboxes
Manage Themes

Use the Themes page to manage your themes. To open the Themes page, on the Navigator, select **Configuration > Appearance**.

You can edit saved themes, apply themes to your application pages, and delete saved themes. You can’t edit or delete predefined themes.

Before You Start

Activate a **sandbox**.

Apply Themes

From the **Themes** list, select a theme, and click **Apply**. If the selected theme is a predefined one, then save it as a new theme, and then edit and apply the theme, as required.

Apply the Default Theme

On the Actions menu, select **Apply Default**. The default theme is applied to your application.

Edit Themes

Follow these steps:

1. On the Navigator, select **Configuration > Appearance**.
2. From the **Themes** list, select your base theme.
3. Set the default layout of the home page, and configure the various appearance settings for your application, as required. For example, select a branding logo and specify color schemes.
4. Click **Apply**. If the base theme you have selected is a:
   - Predefined theme, then enter a theme name, and click **OK** to create another theme with your modifications. This new theme is then applied to your application.
   - Saved theme, then your theme changes are directly applied to your application.

Delete Themes

From the **Themes** list, select a saved theme that you want to delete, and then on the Actions menu, select **Delete**.

Change the Logo and Background Image

Use the Themes page to define the:

- Branding logo, which appears above all application pages. You can use a logo of any size, but the recommended width is lesser than 200 px and height is lesser than 50 px.
- Watermark, which appears in the background of all application pages. Use an image that's as close to 1024 by 768 pixels as possible.
To select a logo and a background image, use one of the options:

- **File**: Browse and select a file from your local computer.
- **Predefined**: Select a file from the list of predefined images.
- **URL**: Enter a full URL for the logo or the watermark.

### Appearance Settings for Changing the Look and Feel of the Application

You configure the various appearance settings for your application using the Themes tab of the Appearance work area. These tables summarize the appearance settings that you can configure while creating and editing your themes and their default values.

**General Settings**

<table>
<thead>
<tr>
<th>Setting</th>
<th>Description</th>
<th>Values</th>
<th>Default Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Default Home Layout</td>
<td>The default layout of the home page.</td>
<td>• Panel</td>
<td>Panel</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Banner</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• News Feed</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>When you select a Panel or Banner layout, the home page contains a springboard with icons that you can use to open work areas. Whereas, when you select the News feed layout, your home page mainly contains the Apps section and a series of updates with important information.</td>
<td></td>
</tr>
<tr>
<td>Logo</td>
<td>The type of location where your logo for the global header is stored. You can browse and select your logo from that location.</td>
<td>• File</td>
<td>Predefined</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Predefined</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• URL</td>
<td></td>
</tr>
<tr>
<td>Configure bookmark icon for desktop</td>
<td>Selecting this check box displays the Desktop Icon list for you to choose a bookmark icon for your desktop.</td>
<td>• Checked</td>
<td>Unchecked</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Unchecked</td>
<td></td>
</tr>
<tr>
<td>Setting</td>
<td>Description</td>
<td>Values</td>
<td>Default Value</td>
</tr>
<tr>
<td>---------------------------------------</td>
<td>-----------------------------------------------------------------------------</td>
<td>-----------------</td>
<td>---------------</td>
</tr>
<tr>
<td>bookmark icon for desktop check box</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Configure shortcut icon for mobile</td>
<td>Selecting this check box displays the <strong>Mobile Icon</strong> list for you to choose a shortcut icon for your mobile device.</td>
<td>• Checked</td>
<td>Unchecked</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Unchecked</td>
<td></td>
</tr>
<tr>
<td>Mobile Icon</td>
<td>The shortcut icon that appears on your mobile device.</td>
<td>• URL</td>
<td>URL</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• File</td>
<td></td>
</tr>
<tr>
<td></td>
<td>This list is available only if you selected the <strong>Configure shortcut icon for mobile check box</strong>.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Background Image</td>
<td>The type of location where your background image of all application pages is stored. You can browse and select the image from that location.</td>
<td>• None</td>
<td>Predefined</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• File</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Predefined</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• URL</td>
<td></td>
</tr>
<tr>
<td>Repeat Image</td>
<td>Selecting this check box repeats the specified background image to display it both vertically and horizontally in the background of all application pages.</td>
<td>• Checked</td>
<td>Unchecked</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Unchecked</td>
<td></td>
</tr>
<tr>
<td>Background Image Position</td>
<td>The position of the background image for all application pages.</td>
<td>• Center</td>
<td>Center</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Start</td>
<td></td>
</tr>
<tr>
<td>Global Region</td>
<td>The background color of the global header, which is the uppermost region in the user interface.</td>
<td>Color palette</td>
<td>Transparent</td>
</tr>
<tr>
<td>Global Region Background Color</td>
<td>The background color of the global header, which is the uppermost region in the user interface.</td>
<td>Color palette</td>
<td>Transparent</td>
</tr>
<tr>
<td>Global Text and Icon Color</td>
<td>The color of the text that appear in general and with icons in the global header. This excludes the company logo.</td>
<td>Color palette</td>
<td>2E444C</td>
</tr>
<tr>
<td>Enable color gradient</td>
<td>Selecting this check box enables color gradient for</td>
<td>• Checked</td>
<td>Unchecked</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Unchecked</td>
<td></td>
</tr>
</tbody>
</table>
### Gradient Type

The gradient type for global text and icon colors. This excludes the company logo.

This field is available only if you enabled color gradient.

<table>
<thead>
<tr>
<th>Values</th>
<th>Default Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>- Top to bottom</td>
<td></td>
</tr>
<tr>
<td>- Left to right</td>
<td></td>
</tr>
<tr>
<td>- Top left to bottom right</td>
<td></td>
</tr>
<tr>
<td>- Top right to bottom left</td>
<td></td>
</tr>
<tr>
<td>Left to right</td>
<td></td>
</tr>
</tbody>
</table>

### Start Color and End Color

The start and end colors of the gradient for global text and icon colors.

This field is available only if you enabled color gradient.

<table>
<thead>
<tr>
<th>Values</th>
<th>Default Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Color palette</td>
<td>8CD3FF</td>
</tr>
</tbody>
</table>

### Cover Image

The type of location where your cover image, which displays in the background of the home page panel or banner, is stored. You can browse and select the image from that location.

<table>
<thead>
<tr>
<th>Values</th>
<th>Default Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>- None</td>
<td></td>
</tr>
<tr>
<td>- File</td>
<td></td>
</tr>
<tr>
<td>- Predefined</td>
<td></td>
</tr>
<tr>
<td>- URL</td>
<td></td>
</tr>
<tr>
<td>Predefined</td>
<td></td>
</tr>
</tbody>
</table>

### Panel Style

Specifies the style for the main panel or banner on the home page.

This field is available only if you selected the default home page layout as Panel or Banner.

<table>
<thead>
<tr>
<th>Values</th>
<th>Default Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>- Light: Overlays a portion of the home page panel or banner with a semi-transparent white layer and makes the text on top of it black.</td>
<td></td>
</tr>
<tr>
<td>- Dark: Overlays a portion of the home page panel or banner with a semi-transparent black layer and makes the text on top of it white.</td>
<td></td>
</tr>
<tr>
<td>Light</td>
<td></td>
</tr>
</tbody>
</table>

### Navigation Icons

The type of navigation icons on the home page.

This field is available only if you selected the default.

<table>
<thead>
<tr>
<th>Values</th>
<th>Default Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>- Solid</td>
<td></td>
</tr>
<tr>
<td>- Outlined</td>
<td></td>
</tr>
<tr>
<td>Solid</td>
<td></td>
</tr>
</tbody>
</table>

The size of the navigation icons on the home page.

This field is available only if you selected the default.

<table>
<thead>
<tr>
<th>Values</th>
<th>Default Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>- Extra large: Sets the size of the icons to 96px.</td>
<td></td>
</tr>
<tr>
<td>- Large: Sets the size of the icons to 72px.</td>
<td></td>
</tr>
<tr>
<td>Extra large</td>
<td></td>
</tr>
<tr>
<td>Setting</td>
<td>Description</td>
</tr>
<tr>
<td>------------------------------</td>
<td>------------------------------------------------------------------------------</td>
</tr>
</tbody>
</table>
| home page layout as **Panel** or **Banner**. | • Medium: Sets the size of the icons to 64px.  
• Small: Sets the size of the icons to 48px. |                                |               |
| Image Color                  | The image color of the navigation icon on the home page.                     | Color palette                   | 2E444C        |
|                              | This field is available only if you selected the default home page layout as **Panel** or **Banner**. |                                |               |
| Group Indicator Icon         | The color of the three dots that appear on certain icons on the home page to indicate that these icons are group icons. | Color palette                   | FFFFFF        |
| Label Color                  | The color of the label text of the navigation icons.                         | Color palette                   | 2E444C        |
| Background Shape             | The shape that displays behind the navigation icons.                         | • None                          | Circle        |
|                              | • Circle                                                                     | • Square                        |               |
|                              | • Rounded square                                                             | • Transparent square            |               |
|                              | The Transparent square option is available only if you selected the default home page layout as News Feed. |                                |               |
| Background Card Color        | The background color of the navigation icons on the News Feed home page.     | • Light                         | Light         |
|                              | This field is available only if you selected:                               | • Dark                          |               |
|                              | • The default home page layout as **News Feed**.                             |                                |               |
|                              | • The background shape for navigation icons as **Transparent square**.       |                                |               |
| Background Card Opacity      | The background opacity of the navigation icons on the News Feed home page.   | Any value between 1 and 99      | 10            |
### Background Color

<table>
<thead>
<tr>
<th>Setting</th>
<th>Description</th>
<th>Values</th>
<th>Default Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Background Color</td>
<td>The color of the shape that displays behind the navigation icons.</td>
<td>Color palette</td>
<td>Multicolor</td>
</tr>
</tbody>
</table>

This field isn't displayed if you have selected None or Transparent square as the background shape for the navigation icons.

### Page Settings

<table>
<thead>
<tr>
<th>Setting</th>
<th>Description</th>
<th>Values</th>
<th>Default Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Heading color</td>
<td>The text color of the headers and subheaders of application pages.</td>
<td>Color palette</td>
<td>333333</td>
</tr>
<tr>
<td>Link Color</td>
<td>The color of the links that appear on the application pages.</td>
<td>Color palette</td>
<td>0A6DAA</td>
</tr>
<tr>
<td>Selection Color</td>
<td>The color of the highlighted or selected UI element on all work areas related to employees, for example Personal Information, Benefits, and Absences. These UI elements are boxes, and each box represents an option, for example a specific part of your personal profile to update. This color applies to the border of the box and the selected check box (if any) inside the box.</td>
<td>Color palette</td>
<td>047BC4</td>
</tr>
<tr>
<td>Highlight Color</td>
<td>The color used to highlight application items that need attention. For example, the part of the Notifications icon on the global header that</td>
<td>Color palette</td>
<td>C74200</td>
</tr>
</tbody>
</table>
### Setting

<table>
<thead>
<tr>
<th>Description</th>
<th>Values</th>
<th>Default Value</th>
</tr>
</thead>
<tbody>
<tr>
<td># displays the number of new notifications in your list.</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### Card Style

The style of cards displayed on the application pages having a grid view.

- **Dark**: Displays the cards with a white or light grey background color and dark text.
- **Light**: Displays the cards with a dark grey background color and light text.

- **Dark**

### Landing Page Icons for Employees

The icon color of the landing pages for all work areas related to employees, for example Personal Information, Benefits, and Absences.

- **Color palette**
- **309FDB**

### Button Settings

<table>
<thead>
<tr>
<th>Setting</th>
<th>Description</th>
<th>Values</th>
<th>Default Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Label</td>
<td>The color of the label text of the buttons on all application pages.</td>
<td>Color palette</td>
<td>FFFFFF</td>
</tr>
<tr>
<td>Border</td>
<td>The color of the button borders on all application pages.</td>
<td>Color palette</td>
<td>4F4F4F</td>
</tr>
<tr>
<td>Enable color gradient</td>
<td>Selecting this check box enables color gradient for buttons on all application pages.</td>
<td></td>
<td>Unchecked</td>
</tr>
<tr>
<td>Start Color and End Color</td>
<td>The start and end colors of the gradient for buttons on all application pages.</td>
<td>Color palette</td>
<td></td>
</tr>
<tr>
<td>Background</td>
<td>The background color for buttons on all application pages.</td>
<td>Color palette</td>
<td>4F4F4F</td>
</tr>
</tbody>
</table>
### Change the Logo and Color Schemes of the Application

This example demonstrates how to change a logo and the color schemes of an application using the Themes tab of the Appearance work area. Users see the logo in the *global header*.

**Note:** Changes made to the logo using Page Composer overwrite the changes done using the Appearance work area.

The following table summarizes the key decisions for this scenario.

<table>
<thead>
<tr>
<th>Decisions to Consider</th>
<th>In This Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>What’s the name of the new theme?</td>
<td>MyCompany</td>
</tr>
<tr>
<td>Which existing theme are you going to base this theme on?</td>
<td>Default</td>
</tr>
<tr>
<td>Which default home page layout are you going to use?</td>
<td>Banner</td>
</tr>
<tr>
<td>Which image are you going to use as the new logo?</td>
<td>MyCompany.png</td>
</tr>
</tbody>
</table>

**Note:** You can use a logo of any size, but the recommended width is lesser than 200 px and height is lesser than 50 px.

---

<table>
<thead>
<tr>
<th>Setting</th>
<th>Description</th>
<th>Values</th>
<th>Default Value</th>
</tr>
</thead>
</table>
| Corner Rounding | The extent to which the corners of the buttons on all application pages are rounded. | - None: Removes corner rounding from all buttons, which means applies a rectangular shape to all buttons.  
- Small: Applies a very small curve to the corners of all buttons.  
- Medium: Applies a moderate curve to the corners of all buttons.  
- Large: Applies a large curve to the corners of all buttons, thereby making a pill shape. | Small |
### Decisions to Consider

<table>
<thead>
<tr>
<th>Decisions to Consider</th>
<th>In This Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>Which image are you going to use as your application's bookmark icon for desktop?</td>
<td>MyDesktopIcon.png</td>
</tr>
<tr>
<td><strong>Note:</strong> For best results, use a square-shaped icon with each side measuring lesser than 16 px.</td>
<td></td>
</tr>
<tr>
<td>Which image are you going to use as your application's shortcut icon for mobile devices?</td>
<td>MyMobileIcon.png</td>
</tr>
<tr>
<td><strong>Note:</strong> For best results, use a square-shaped icon with each side measuring more than 57 px and lesser than 144 px.</td>
<td></td>
</tr>
<tr>
<td>Which background image are you going to use?</td>
<td>WatermarkOracle.png</td>
</tr>
</tbody>
</table>

### Before You Start

Activate a **sandbox**.

### Change the Logo and Color Schemes

You can specify the look and feel aspects of various UI elements, such as navigation icons and buttons, using the Themes page. The fields available for you to change these appearance settings may vary based on what you select for some fields. For example, if you select **News Feed** as the default home layout, the **Group Indicator Color** field isn’t displayed for selection because this field isn’t applicable for the news feed home page layout.

Follow these steps:

1. On the Navigator, select **Configuration > Appearance**. On the Appearance work area, click the **Themes** tab.
2. From the **Themes** list, select **Default**.
3. Select the default home layout as **Banner**.

**Note:** When you select a **Panel** or **Banner** layout, the home page contains a springboard with icons that you can use to open work areas. Also, based on setup, the home page shows either company announcements or social networking conversations. This information appears in a panel for the panel layout, and in a banner for the banner layout. When you select the **News feed** layout, your home page mainly contains the Apps section and a series of updates with important information.

4. From the **Logo** list, select **File** as the type of location where your logo is stored. Browse and select **MyCompany.png**.
5. Select the **Configure bookmark icon for desktop** check box, and from the **Desktop Icon** list, select **File**. Browse and select **MyDesktopIcon.png**.
6. Select the **Configure shortcut icon for mobile** check box, and from the **Mobile Icon** list, select **File**. Browse and select **MyMobileIcon.png**.
7. From the **Background Image** list, select **File** as the type of location where your background image is stored. Browse and select **WatermarkOracle.png**.

8. To specify the general appearance of the application, enter the values as shown in this table, or select the colors from the color palette.

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Global Region Background Color</td>
<td>A9A9A9 (Dark Gray)</td>
</tr>
<tr>
<td></td>
<td>This field specifies the background color of the global header, which is the uppermost region in the user interface.</td>
</tr>
<tr>
<td>Global Text and Icon Color</td>
<td>4169E1 (Royal Blue)</td>
</tr>
<tr>
<td></td>
<td>This field specifies the color of the text that appear in general and with icons in the global header.</td>
</tr>
<tr>
<td>Enable color gradient</td>
<td>Select the check box, specify the gradient type as Top to bottom, and start and end colors of the gradient as follows:</td>
</tr>
<tr>
<td></td>
<td>◦ Start Color: 00BFFF (Deep Sky Blue)</td>
</tr>
<tr>
<td></td>
<td>◦ End Color: 4169E1 (Royal Blue)</td>
</tr>
<tr>
<td>Cover Image</td>
<td>Select <strong>File</strong> as the type of location where your cover image is stored. Browse and select <strong>MyCover.png</strong>.</td>
</tr>
<tr>
<td></td>
<td>For the home page with the:</td>
</tr>
<tr>
<td></td>
<td>◦ Panel layout, this image appears on the main panel. The recommended image size is 344x622 px.</td>
</tr>
<tr>
<td></td>
<td>◦ Banner layout, this image appears on the banner. The recommended image size is 2600x290 px.</td>
</tr>
<tr>
<td></td>
<td>◦ News feed layout, this image appears in the Things to Finish section. The recommended image size is 2600x290 px.</td>
</tr>
<tr>
<td>Panel Style</td>
<td>Light</td>
</tr>
<tr>
<td></td>
<td>This field specifies the style of the main panel on the home page.</td>
</tr>
</tbody>
</table>

9. To specify the appearance of the navigation icons, enter the values as shown in this table, or select the colors from the color palette.

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Icon Type</td>
<td>Solid</td>
</tr>
<tr>
<td>Size</td>
<td>Small</td>
</tr>
<tr>
<td>Image Color</td>
<td>Multicolor</td>
</tr>
</tbody>
</table>
### Field Management

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Group Indicator Color</strong></td>
<td>D02090 (Violet Red)</td>
</tr>
<tr>
<td>This field specifies the color of the three dots that appear on certain icons on the springboard to indicate that these icons are group icons.</td>
<td></td>
</tr>
<tr>
<td><strong>Label Color</strong></td>
<td>000000 (Black)</td>
</tr>
<tr>
<td>This field specifies the color of the label text of the navigation icons.</td>
<td></td>
</tr>
<tr>
<td><strong>Background Shape</strong></td>
<td>Circle</td>
</tr>
<tr>
<td><strong>Background Color</strong></td>
<td>Multicolor</td>
</tr>
<tr>
<td>This field isn't displayed if you have selected None as the background shape.</td>
<td></td>
</tr>
</tbody>
</table>

10. To specify the appearance of the application pages, enter the values as shown in this table, or select the colors from the color palette.

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Heading Color</strong></td>
<td>8B008B (Dark Magenta)</td>
</tr>
<tr>
<td>This field specifies the color of the header text on the application pages.</td>
<td></td>
</tr>
<tr>
<td><strong>Link Color</strong></td>
<td>0000FF (Blue)</td>
</tr>
<tr>
<td>This field specifies the color of the links that appear on the application pages.</td>
<td></td>
</tr>
<tr>
<td><strong>Selection Color</strong></td>
<td>FFB6C1 (Light Pink)</td>
</tr>
<tr>
<td>This field specifies the color of a selected or highlighted UI element on a page.</td>
<td></td>
</tr>
<tr>
<td><strong>Highlight Color</strong></td>
<td>FFB6C1 (Yellow)</td>
</tr>
<tr>
<td><strong>Card Style</strong></td>
<td>Light</td>
</tr>
<tr>
<td>This field specifies the style of cards displayed on application pages having a grid view.</td>
<td></td>
</tr>
<tr>
<td><strong>Landing Page Icons for Employees</strong></td>
<td>Blue</td>
</tr>
<tr>
<td>This field specifies the icon color of the landing pages for all work areas related to employees, for example Personal Information, Benefits, and Absences.</td>
<td></td>
</tr>
</tbody>
</table>
11. To specify the appearance of the buttons, enter the values as shown in this table, or select the colors from the color palette.

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Label</td>
<td>000000 (Black)</td>
</tr>
<tr>
<td></td>
<td>This field specifies the color of the label text of the buttons.</td>
</tr>
<tr>
<td>Border</td>
<td>000000 (Black)</td>
</tr>
<tr>
<td></td>
<td>This field specifies the color of the button borders.</td>
</tr>
<tr>
<td>Enable color gradient</td>
<td>Select the check box and specify the start and end colors of the gradient as follows:</td>
</tr>
<tr>
<td></td>
<td>◦ Start Color: 4169E1 (Royal Blue)</td>
</tr>
<tr>
<td></td>
<td>◦ End Color: 00BFFF (Deep sky Blue)</td>
</tr>
<tr>
<td>Corner Rounding</td>
<td>Small</td>
</tr>
<tr>
<td></td>
<td>This field specifies the extent to which the button corners are rounded. For example, None indicates that the button corners aren't rounded and Small indicates that the button corners are slightly rounded.</td>
</tr>
</tbody>
</table>

**Tip:** While making changes on the Themes page, you can click Apply any time to preview your changes.

12. On the Actions menu, select Save As.
13. Enter the theme name as MyCompany.
14. Make sure that Apply this theme is selected.
15. Click OK.

**Related Topics**
- Overview of Sandboxes
- Create and Activate Unified Sandboxes
- Overview of Work Areas

**FAQs for Theme Management**
What happens to my theme if changes that affect themes are made using Page Composer?

Application changes made using Page Composer overwrite the changes made using the Appearance work area.

Caution: Before using the Appearance work area to change the look of your application, you must undo any changes affecting the theme done using Page Composer. Otherwise, the changes that you make using the Appearance work area may not show up in your application as you wanted.
6 Flexfields

Overview of Flexfields

A flexfield is a set of placeholder fields associated with business objects and placed on the application pages to contain additional data. You can use flexfields to modify the business objects and meet enterprise data management requirements without changing the data model or performing any database programming. Flexfields help you to capture different data on the same database table and provide a means to modify the applications features.

For example, an airline manufacturer may require specific attributes for its orders that aren't predefined. Using a flexfield for the order business object, you can create and configure the required attribute.

Types of Flexfields

Flexfields that you see on the application pages are predefined. However, you can configure the flexfields or modify their properties. Users see these flexfields as field or information attributes on the UI pages. To manage flexfields, use any of the following tasks in the Setup and Maintenance work area:

- **Manage Descriptive Flexfields**: Expand the forms on the application page to accommodate additional information that is important and unique to your business. You can use a descriptive flexfield to collect invoice details on a page displaying invoices.
- **Manage Extensible Flexfields**: Establish one-to-many data relationships and make application data context-sensitive. The flexfields appear only when the contextual data conditions are fulfilled. Thus, extensible flexfields provide more flexibility than the descriptive flexfields.
- **Manage Key Flexfields**: Store information combining several values, such as a number combination. The key flexfields represent objects such as accounting codes and asset categories.
- **Manage Value Sets**: Use a group of values to validate the data entered in the flexfields.

**Note**: You can manage value sets within the Manage Descriptive Flexfields or Manage Extensible Flexfields tasks.

Related Topics

- Modules in Application Taxonomy

Overview of Flexfield Configuration

Configuring a flexfield involves identifying the need for enhancing a business object with user-defined attributes and then integrating the attributes into deployment. In the case of key flexfields, configuring the flexfield involves identifying value set assignments and determining segment structures.
Overall Process for Configuring User-Defined Attributes

Before using flexfields to create attributes, familiarize yourself with the context layers and the configuration life cycle of the application. You can add attributes to a business object using a flexfield, if developers have registered that object to a flexfield. For descriptive and extensible flexfields, the overall configuration process involves the following:

1. Use the Highlight Flexfields feature from the Administration menu to find flexfields on pages associated with business objects.
2. Plan the flexfield configuration.
3. Plan flexfield validation.
4. Define the attributes by configuring the flexfield segments.
   a. Use the Manage Extensible Flexfields or Manage Descriptive Flexfields tasks, or use the **Configure Flexfield** icon button directly on the page where the flexfield is highlighted. For simple configurations, use the **Add Segment**, **Add Context Value**, and **Edit Segment** icon buttons directly on the page where the flexfield is highlighted.
   b. Optionally, validate the flexfield configuration.
   c. Optionally, deploy the flexfield to a sandbox for initial testing.
5. Deploy the flexfield to the **mainline metadata** to display the attributes on the application pages and to make them available for integration with other tools such as Oracle Business Intelligence.
6. Perform the necessary steps to integrate the attributes into the technology stack.

A simple configuration is limited to such actions as adding a format-only field or adding a field with a basic list of values.

Overall Process for Configuring User-Defined Keys

Using key flexfields, you can configure intelligent key codes containing meaningful parts according to your business practices. You configure the key flexfield to have one segment for each part that makes up your key code.

For key flexfields, the overall configuration process involves the following:

1. Use the Highlight Flexfields feature from the Administration menu to find flexfields on pages associated with business objects.
2. Plan the flexfield configuration.
3. Plan the flexfield validation.
4. Define the value sets before configuring the key flexfield segments by going to the Manage Value Sets task.
5. Define the key flexfield structures and their segments, and define structure instances for each structure.
   a. Use the Manage Key Flexfields task or the **Configure Flexfield** icon button directly on the page where the flexfield is highlighted.
   b. Optionally, validate the flexfield configuration.
   c. Optionally, deploy the flexfield to a sandbox for initial testing.
6. Deploy the flexfield to the mainline metadata to display it on the application pages and to make it available for integration with other tools such as Oracle Business Intelligence.
7. Perform the necessary steps to integrate the flexfield into the technology stack.

**Related Topics**

- **Overview of Context Layers**
Flexfield Components

A flexfield is made up of several data entities that store and render information pertaining to flexfield configuration. Flexfields are made up of the following components:

- Segments
- Value Sets
- Contexts
- Structures

Segments

A segment is a field within a flexfield and represents a single table column of your database. When configuring a flexfield, define the appearance and meaning of individual segments. Segments represent attributes of information. Segments can appear globally wherever the flexfield is implemented, or based on a structure or context. Each segment captures a single atomic value and represents an attribute of information.

The characteristics of a segment vary based on the type of flexfield in which it’s used.

- In key flexfields, a segment describes a characteristic of the entity. For example, a part number that contains details about the type, color, and size of an item.
- In a descriptive or extensible flexfield, a segment represents an information attribute on the application page. For example, details about a device containing components, some of which are global while the remaining are contextually dependent on the category of the device.

Value Sets

Users enter values into segments while using an application. A value set is a named group of values that validate the content of a flexfield segment. You configure a flexfield segment with a value set to enforce entries of only valid values for that segment.

The configuration involves the following tasks:

- Defining the values in a value set, including characteristics such as the length and format of the values.
- Specifying formatting rules or values from an application table or predefined list.

Multiple segments within a flexfield, or multiple flexfields, can share a single value set.

Contexts

Context-sensitive flexfield segments are available to an application based on a context value. You define contexts as part of configuring a flexfield. Users see global segments as well as any context-sensitive segments that apply to the selected context value.

In descriptive flexfields and extensible flexfields, you can reuse the context-sensitive segments that are based on the database columns, in multiple contexts.
Structures
Key flexfields have structures. Each key flexfield structure is a specific configuration of segments. Adding or removing segments, or rearranging their order, produces a different structure. You can reuse the segments that are based on the database columns, in multiple structures.

Note: You can translate all these flexfield components to the preferred languages without changing the language session of the application. To specify the translations in all the enabled language rows, use the Translation Editor option on the respective edit pages. Once the updates are made, users can view the translated text for the specific flexfield components at runtime.

Related Topics
• Enter or Edit Translated Text

Flexfields at Runtime
Business objects have an associated descriptive or extensible flexfield. Using these, you can create attributes for the business object at run time. Some business objects have an associated key flexfield for configuring flexible multiple part keys.

Finding Flexfields on a Page
At run time, the attributes you define as flexfield segments appear in the application page just like any other attribute. However, each type of flexfield appears in a different way.

The following characteristics help you determine the type of flexfield on the application page:
• Descriptive flexfield segments appear as label and field pairs or as a table of fields that correspond to the column headers. The fields represent the flexfield segments and accept values that derive from the segment's assigned value set.
• Extensible flexfield segments appear grouped within labeled regions, where each grouping is a context and the region labels are the context names.
• Key flexfields appear in the application page as a field with a key flexfield icon, where the field's value is a collection of segments.

To locate flexfields on a page, in the global header, select your user name and in the Settings and Actions menu, select Highlight Flexfields. The page renders in a special mode, displaying the location of flexfields, if any, on the page. Do the following:
• Hover over the Information icon to view flexfield details.
• Click the Configure Flexfield icon to manage the flexfield using the Manage Flexfields task.
• Click the Add Context Value, Add Segment, or Edit Segment icons to add a context value or edit a global or context-sensitive flexfield segment. This applies to both descriptive and extensible flexfields.

Note: You can't create attributes on all flexfields. For example, some flexfields are protected, and you either can't edit their configurations at all, or can do only limited changes to them. Consult the product-specific documentation to verify whether there are any restrictions on using the flexfield.

All segments of a single flexfield are grouped together by default. The layout and positions of the flexfield segments depend on where the application developer places the flexfield on the page. Flexfields may also be presented in a
separate section of the page, in a table, or on their own page or a dialog box. You can use Oracle Composer to edit the layout, position, or other display features of the flexfield segments.

When you no longer want to view the flexfields on a page, select **Unhighlight Flexfields** from the Administration menu.

### Flexfield Modification Using Page Composer

You can use Page Composer to modify flexfields specific to a page.

#### Extensible Flexfield Modification

In Page Composer, open the page with the flexfield you want to modify. Switch to Source view, and look for a region that is bound to an EffContextsPageContainer task flow. This is the container for the extensible flexfield attributes and contexts. To view the flexfield code and identifying information, open the properties panel for the region. To modify any component within the region, select the desired tag and click Edit.

#### Descriptive Flexfield Modification

In Page Composer, open the page with the flexfield you want to modify. Switch to Source view, and look for the `<descriptiveFlexfield>` element of that flexfield. Open the properties panel for this element to view the flexfield code and identifying information. Within the properties panel, you may modify properties for the global and context-sensitive segments or re-order the segments on the page.

**Note:** Flexfield segments can't be edited using their individual component elements. They can only be edited from the properties panel of the `<descriptiveFlexfield>` element they belong to. To find these components, open the properties panel of the descriptive flexfield, switch to the relevant tab, and search using the unique identifying information. For instance, if you need to find a context sensitive segment, open the properties panel for the flexfield, go to the Flexfield Context Segments tab, and search for the segment using its Context Value and Segment Code.

### How Flexfields Work with Oracle Application Cloud Architecture

To capture additional data, administrators or implementors configure **flexfield segments** that represent attributes of **business objects**. Business objects are enabled for both **descriptive flexfields** and **extensible flexfields**.

The following figure shows the layers involved in configuring a **flexfield**:

- The business entity table and metadata in the database.
- The ADF business component objects. These are derived from the metadata and stored in Oracle Metadata Services (MDS) repository.
- The user interface where fields defined by the flexfield segments are rendered.
The following figure illustrates that the flexfield definition consists of all the metadata defined during configuration and stored in the database.

Application developers create a flexfield and register it so that it's available for configuration. Administrators and implementation consultants configure segments and other properties of the available flexfields. This information is stored as additional flexfield metadata in the database. Deploying the flexfield generates ADF business components based on the flexfield metadata in the database.

The following aspects are important in understanding how flexfields and Oracle Applications Cloud architecture work together:

- Integration
- Deployment
• Import and export  
• Run time  
• Patching

Integration
The attributes that you add by configuring flexfields are available throughout the Oracle Fusion Middleware technology stack. You can use the flexfield segment's Application Programming Interface (API) to identify segments and integrate the flexfields in the following:
  • User interface pages  
  • Service-oriented Architecture (SOA) infrastructure  
  • Oracle Business Intelligence  
  • Extended Spread Sheet Database (ESSbase)  

Flexfield configurations are preserved across application updates.

Deployment
The metadata for the flexfield is stored in the application database as soon as you save your configuration changes. Deploying the flexfield generates the ADF business components so that the run time user interface reflects the latest flexfield definition in the metadata.

Importing and Exporting
Using the Setup and Maintenance work area, you can import and export flexfields across the implementation site. The deployment status must be either Deployed or Deployed to sandbox. Therefore, before you attempt migration, verify and ensure that a flexfield is successfully deployed.

Run Time
The latest definitions of a flexfield reflect on the user interface at run time only if the flexfield is deployed. When the user interface accesses a business object, the deployed flexfield definition identifies the attributes associated with the captured values. On a page, if you add display configurations for a flexfield using Oracle Composer, the same flexfield segments can appear differently on different pages.

Patching
Flexfield configurations are preserved during patching and upgrading.

Flexfield Management
Considerations for Managing Flexfields

Managing *flexfields* involves registering, planning, and configuring flexfields.

You plan and configure the registered flexfields provided in your applications by applications developers. How you configure *flexfield segments* determines how the flexfield segments appear to users. Optionally, you can modify the UI page to change how the flexfield segments appear to users on that page.

The following figure shows the processes involved in making flexfields available to users. The tasks in the Define Flexfields activity let administrators configure and deploy flexfields. After you configure and deploy a flexfield to a *sandbox*, deploy it again to the *mainline metadata* so that it's available to the users.
Consider the following aspects of managing flexfields:

- Registering flexfields
- Planning flexfields
- Configuring flexfields
- Enabling a flexfields segment for business intelligence
- Deploying flexfields
- Optionally changing a flexfield segment's appearance in a user interface page
- Identifying flexfields on a run time page and troubleshooting

### Registering Flexfields

A flexfield must be registered before it can be configured. Therefore, application development registers flexfields so that they are available to administrators and implementation consultants for configuration. The registration involves reserving columns of entity tables for use in flexfields. For more information about registering flexfields, see Oracle Fusion Applications Developer's Guide.

### Planning Flexfields

Before you begin planning flexfields, determine what type is appropriate to your needs, and which business objects are available for modifying flexfields. All flexfields consist of segments which represent attributes of an entity. The value a user enters for an attribute is stored in a column of the entity table. Carefully plan flexfields before configuring them. Before configuring new segments for your flexfields, be sure to plan their implementation carefully.

If you have determined that a business object supports flexfields, and those flexfields have been registered, you can begin planning their configuration. Note the code name of the flexfield you intend to configure so that you can find it easily in the Define Flexfield activity. In some cases you can determine and configure how the flexfield appears on the page. See Oracle Applications Cloud Help for specific products to determine any restrictions on using product-specific flexfields.

### Configuring Flexfields

Administrators or implementors configure flexfields so they meet the needs of the enterprise. Some flexfields require configuration to make an application operate correctly. You can configure flexfields using the following methods:

- Use the manage flexfield tasks in the Setup and Maintenance work area.
- Use the Highlight Flexfields command in the Administration menu while viewing a run time page.
  - Use the **Configure Flexfield** icon button to manage all aspects of a flexfield, such as change a segment’s sequence number or configure a flexfield segment’s business intelligence label.
  - Use the **Add Segment** and **Edit Segment** icon buttons to add and edit descriptive or extensible flexfield segments with simple configurations.
  - Use the **Add Context** icon button to add descriptive or extensible flexfield context values.

Configuring a flexfield includes the following:

- Defining value sets against which the values entered by users are validated
- Defining the structure or context of the segments in the flexfield
- Specifying the identifying information for each segment
- Specifying the display properties such as prompt, length and data type of each flexfield segment
- Specifying valid values for each segment, and the meaning of each value within the application
Tip: You can create value sets while creating descriptive and extensible flexfield segments. However, define value sets before configuring key flexfield segments that use them, because you assign existing value sets while configuring key flexfield segments.

When creating table-validated, independent, dependent, or subset value sets while creating descriptive and extensible flexfield segments, you can optionally specify to display the description of the selected value next to the segment at run time. You can assign sequence order numbers to global segments and to context-sensitive segments in each context. Segment display is always in a fixed order based on the segments’ sequence numbers. You cannot enter a number for one segment that is already in use for a different segment. Therefore, you may consider numbering the segments in multiples, such as 4, 5, or 10, to make it easy to insert new attributes.

A flexfield column is assigned to a new segment automatically, but you can change the assignment before saving the segment. If you must set a specific column assignment for a segment, create that segment first to ensure that the intended column isn’t automatically assigned to a different segment.

Enabling a Flexfield Segment for Business Intelligence
You can enable flexfield segments for business intelligence if the flexfield is registered in the database as an Oracle Business Intelligence-enabled flexfield. For more information about enabling segments for business intelligence, see points to consider when enabling descriptive, extensible, and key flexfield segments for business intelligence. For extensible flexfield segments, you can’t assign labels to equalize segments across contexts that are semantically equivalent.

Deploying Flexfields
Once you have configured a flexfield, you must deploy it to make the latest definition available to run time users. In the Define Flexfields tasks, you can deploy a flexfield using either of the following commands:

- The Deploy Flexfield command deploys a flexfield to the mainline metadata. This command is for general use in a test or production environment.
- The Deploy to Sandbox command deploys a flexfield to sandbox. This command is for confirming that the flexfield is correctly configured before deploying it to the mainline metadata.

In Highlight Flexfields mode, when using the:

- Add Context, Add Segment, and Edit Segment tools for extensible flexfields, use the Save command to save your changes. Then use the Deploy command to deploy the flexfield to the mainline metadata
- Add Segment and Edit Segment tools for descriptive flexfields, use the Save and Deploy command to save your changes. Then deploy the flexfield to the mainline metadata

Once deployed, the deployment status indicates the state of the currently configured flexfield relative to the last deployed definition.

Optionally Changing a Flexfield Segment Appearance
The flexfield attributes that you define integrate with the user interface pages where users access the attributes’ business object. Application development determines the UI pages where business objects appear and the display patterns used by default to render flexfield segments.

After a flexfield has been deployed to the mainline MDS repository so that it appears on application pages, you can modify it on a per-page basis using Page Composer. For example, you can hide a segment, change its prompt or other properties, or reorder the user-defined global attributes so that they are interspersed with the core attributes in the same parent layout. You can modify the appearance of descriptive and extensible flexfield segments in the UI page using Page Composer, once the flexfield is deployed to the mainline metadata.
If the applications are running in different locales, you can provide different translations for translatable text, such as prompts and descriptions. Enter translations using the locale that requires the translated text. In the global header, click your user name and from the Settings and Actions menu, select Set Preferences. Then change the text to the translated text for that locale.

**Identifying Flexfields on a Run Time Page**

The **Highlight Flexfields** command in the Administration menu of the Setup and Maintenance work area identifies the location of flexfields on the run time page by displaying an **Information** icon button for accessing details about each flexfield.

Even if a descriptive or extensible flexfield isn't yet deployed and no segments appear on the run time page in normal view, the flexfield appears in the Highlight Flexfield view for that page. For descriptive flexfields, the segments as of the last deployment appear. For extensible flexfields, any segments and contexts that have been saved but not yet deployed also appear as disabled.

**Highlight Flexfields** accesses the current flexfield metadata definition. Use the highlighted flexfield's **Configure Flexfield** icon button to manage flexfields directly. Alternatively, note a highlighted flexfield's name to search for it in the tasks for managing flexfields.

For more information about creating flexfields and adding them to a UI page, see the Oracle Fusion Applications Developer's Guide. For more information about modifying flexfield segment appearance with Page Composer, see guidance on modifying existing pages in the Oracle Applications Cloud Configuring and Extending Applications guide.

**Flexfield Segment Properties**

Independent of the **value set** assigned to a **segment**, segments may have properties that affect how they're displayed and how they function.

The following aspects are important in understanding

- Display properties
- Properties related to segment values
- Properties related to search
- Range validation segments
- Rule validation of segment values
- Naming conventions

**Display Properties**

The following table summarizes display properties.

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enabled</td>
<td>Whether the segment can be used.</td>
</tr>
<tr>
<td>Sequence</td>
<td>The order the segment appears in relation to the other configured segments.</td>
</tr>
<tr>
<td>Prompt</td>
<td>The string to be used for the segment's label in the user interface.</td>
</tr>
</tbody>
</table>
Properties Related to Search

Extensible flexfield segments can be marked as selectively required in search using the indexed property. The indexed property requires users to enter a value before conducting a search on the attribute represented by the indexed segment. A database administrator must create an index on the segment column representing the indexed attribute.

Range Validation of Segments

Range validation enables you to enforce an arithmetic inequality between two segments of a flexfield. For example, a product must be ordered before it can be shipped. Therefore, the order date must be on or before the ship date. Also, the order date segment value must be less than or equal to the ship date segment value. You can use range validation to ensure this relationship.

The conditions for range validation are as follows:

- Segments must be configured for range validation in pairs, one with the low value and one with the high value.
- Both segments must be of the same data type.
- Both segments must be parts of the same structure in a key flexfield or parts of the same context in a descriptive flexfield or extensible flexfield.
- The low value segment must have a sequence number that’s lesser than that of the high value segment.
- Non-range validated segments can exist between a range validated pair, but range validated pairs can’t overlap or be nested.

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Display type</td>
<td>The type of field in which to display the segment.</td>
</tr>
<tr>
<td>Selected and deselected values</td>
<td>If the display type is check box, the actual values to save. For example, Y and N or 0 and 1.</td>
</tr>
<tr>
<td>Display size</td>
<td>The character width of the field.</td>
</tr>
<tr>
<td>Display height</td>
<td>The height of the field as measured in visible number of lines when the display type is a text area.</td>
</tr>
<tr>
<td>Read only</td>
<td>Whether the field should display as read-only, not editable text.</td>
</tr>
<tr>
<td>Description help text</td>
<td>The field-level description help text to display for the field. Use description help text to display a field-level description that expands on or clarifies the prompt provided for the field. If description help text is specified, a Help icon button is displayed next to the field in the run time application. The description help text is displayed when the user hovers over the Help icon button.</td>
</tr>
<tr>
<td>Instruction help text</td>
<td>The field-level instruction help text to display for the field. Use instruction help text to provide directions on using the field. If instruction help text is specified, it’s appears in an in-field help note window when users move the cursor over the field.</td>
</tr>
</tbody>
</table>
You can configure as many range validated pairs as you want within the same flexfield. Your application automatically detects and applies range validation to the segment pairs that you define, in sequence order. It must detect a low value segment first, and the next range validated segment that it detects must be a high value segment. These two segments are assumed to be a matching pair. The low value and the high value can be equal.

**Rule Validation of Segment Values**
Validation rules on descriptive and extensible flexfield segments determine how an attribute is validated. The value entered for an attribute on a business object must match a specified format or be restricted to a list of values. You can use a value set or a Groovy validator to specify the validation rules.

Value set validation is required for global segments and context-sensitive segments, and optional for context segments. In the case of context segments, the application may validate a value instead of the value set validating the value against the context segment. However the application entered values must match exactly the valid context segment values. If the context segment values are a superset or subset of the input values, you must assign a table-validated value set or independent value set to validate context values.

You can also use Groovy validation to set additional restrictions or requirements for what values are allowed for certain attributes of business objects. This is useful when you need to use the same value set to validate multiple segments, but the exact validation requirement changes with each case. These validators can be defined at the global segment level, or at the context level, based on your business needs. They have a validator code, validation expression, error message, and description. After adding a new validator, click the **Groovy Expression Builder** icon to open the expression builder window where you define your validation expression. Groovy validation is done when a user tries to save their values to an attribute that has a Groovy validator. If the value for this attribute fails validation against the Groovy expression, the text defined in the **Error Message** column is displayed as an error message.

When you configure a descriptive flexfield segment, you can specify a constant to use for setting the initial value. The initial value can be an available parameter. For every planned segment, list the constant value or parameter, if any, to use for the initial value.

**Naming Conventions**
Enter a unique code, name, and description for the segment. These properties are for internal use and not displayed to end users. You can't change the code after the segment is created.

The Application Programming Interface (API) name is a name for the segment that isn't exposed to users. The API name is used to identify the segment in various integration points including web services, rules, and business intelligence. Use alphanumeric characters only with a leading character. For example, enter a code consisting of the characters A-Z, a-z, 0-9 with a non-numeric leading character. The use of spaces, underscores, multi-byte characters, and leading numeric characters isn't permitted. You can't change the API name after the segment has been created.

**Naming Conventions for Flexfield APIs**
Application Programming Interface (API) name is a name for your flexfield component, which isn't exposed to users. These names are used to identify flexfield components in various integration points, including web services, rules, and business intelligence.

**How to Frame API Names**
You must use only alphanumeric characters for API names. For example, enter a name with the characters A-Z, a-z, or 0-9, with a non-numeric leading character. Don't use spaces, underscores, multi-byte characters, or leading numeric characters in your API names.
**Caution:** Do not change API names after they're created. Doing so could break integration points.

### Words You Can't Use for API Names

You can't use certain words for API names when configuring flexfields because they're reserved words in ADF, SQL, PL/SQL, Java, Groovy, and so on.

This table lists some of the reserved words that can't be used as API names. This list isn't exhaustive.

<table>
<thead>
<tr>
<th>Letter</th>
<th>Reserved Words</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>ABORT, ABSTRACT, ACCEPT, ACCESS, ACTIONENABLED, ADD, ALL, ALLROWSINRANGE, ALTER, AND, ANY, ARRAY, ARRAYLEN, AS, ASC, ASSERT, ASSIGN, AT, ATTRIBUTEDEF, ATTRIBUTEDEFS, ATTRIBUTEVALUE, ATTRIBUTEVALUES, AUDIT, AUTHORIZATION, AVG</td>
</tr>
<tr>
<td>B</td>
<td>BEGIN, BETWEEN, BINDINGS, BODY, BOOLEAN, BREAK, BY, BYTE, BYVALUE</td>
</tr>
<tr>
<td>C</td>
<td>CASCADE, CASE, CAST, CATCH, CATEGORY, CHAR, CHECK, CHILDREN, CLASS, CLONE, CLOSE, CLUSTER, CLUSTERS, COALESCE, COLAUTH, COLUMN, COLUMNS, COMMENT, COMMIT, COMPRESS, CONNECT, CONST, CONSTANT, CONSTRAINT, CONTAINS, CONTINUE, COUNT, CRASH, CREATE, CURRENT, CURRENTROW, CURRVAL, CURSOR</td>
</tr>
<tr>
<td>D</td>
<td>DATABASE, DATACONTROL, DATE, DBA, DEBUGOFF, DEBUGON, DECIMAL, DECLARE, DEFAULT, DEFINITION, DELAY, DELETE, DELTA, DESC, DESCRIPTION, DIGITS, DISPLAY, DISPLAYDATA, DISPLAYHINT, DISPLAYHINTS, DISPOSE, DISTINCT, DIV, DO, DOUBLE, DROP</td>
</tr>
<tr>
<td>E</td>
<td>ELSE, ELSIF, EMPTY, ENABLED, ENABLEDSTRING, END, ENTRY, EQ, EQUALS, ERROR, ESTIMATEDROWCOUNT, EXCEPTION, EXCLUSIVE, EXISTS, EXIT, EXTRACTS</td>
</tr>
<tr>
<td>F</td>
<td>FALSE, FETCH, FILE, FINAL, FINALIZE, FINALLY, FINDMODE, FLOAT, FOR, FORM, FROM, FULLNAME, FUNCTION, FUTURE</td>
</tr>
<tr>
<td>G</td>
<td>GE, GENERIC, GETCLASS, GOTO, GRANT, GROUP, GROUPBY, GT</td>
</tr>
<tr>
<td>H</td>
<td>HASHCODE, HAVING, HINTS</td>
</tr>
<tr>
<td>I</td>
<td>IDENTIFIED, IF, IMMEDIATE, IMPLEMENTS, IMPORT, IN, INCREMENT, INDEX, INDEXES, INDICATOR, INITIAL, INNER, INPUTVALUE, INSERT, INSTANCEOF, INT, INTEGER, INTERFACE, INTERSECT, INTO, IS, ITERATORBINDING</td>
</tr>
<tr>
<td>J</td>
<td>JAVA</td>
</tr>
<tr>
<td>K</td>
<td>KEY</td>
</tr>
<tr>
<td>L</td>
<td>LABEL, LABELS, LABELSET, LE, LEVEL, LIKE, LIMITED, LOCK, LONG, LOOP, LT</td>
</tr>
</tbody>
</table>
### How Flexfields Segments are Rendered

Flexfield segments appear on pages as attributes of business objects.

### Settings That Affect Flexfield Segment Display

When you configure flexfield segments, the value you enter for the segment’s display type determines how the segment appears at run time.
How Display Type Values Appear

The following series of figures (A to K) represent how the display types render on the UI at run time. Each display type screenshot is assigned an alphabet that maps to the display type and its description in the table.

The following figure contains the representation of a check box, a drop-down list, a list of values, and a search enabled list of values.

A. Check Box

B. Drop-down List

C. List of Values

D. Search Enabled List of Values

The following figure contains the representation of a radio button group, text area, text box, date and time, and rich text editor.
This figure contains the representation of a color palette and a static URL field.
The following table describes each display type.

<table>
<thead>
<tr>
<th>Figure Reference</th>
<th>Display Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>Check Box</td>
<td>The field appears as a check box. If the user selects the check box, the checked value is used. Otherwise, the deselected value is used.</td>
</tr>
<tr>
<td>B</td>
<td>Drop-down List</td>
<td>The field appears as a list of values available to the user for selection.</td>
</tr>
<tr>
<td>C</td>
<td>List of Values</td>
<td>The field appears as a list of values available to the user for selection. The user can also click Search to find more values.</td>
</tr>
<tr>
<td>D</td>
<td>Search Enabled List of Values</td>
<td>The field appears as a text field with a Search icon button. The users can type a value in the text field or they can click</td>
</tr>
<tr>
<td>Figure Reference</td>
<td>Display Type</td>
<td>Description</td>
</tr>
<tr>
<td>------------------</td>
<td>--------------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td></td>
<td></td>
<td>the Search icon button to open another window for searching.</td>
</tr>
<tr>
<td>E</td>
<td>Radio Button Group</td>
<td>The field appears as a set of radio buttons. The user can select one button. Selecting a button deselects any previously selected button in the set.</td>
</tr>
<tr>
<td>F</td>
<td>Text Area</td>
<td>The field appears as a text area in which the user can type multiple lines of text. The display width and height specify the visible width and number of lines in the text area, respectively.</td>
</tr>
<tr>
<td>G</td>
<td>Text Box</td>
<td>The field appears as a text field in which the user can type a single line of text. The display width controls the width of the text box.</td>
</tr>
<tr>
<td>H</td>
<td>Date Time</td>
<td>The field enables the user to enter a date if the data type is Date, or a date and time if the data type is Date Time. The user can select the date in a calendar. If the data type is Date Time, the field also displays fields for specifying the hour, minutes, seconds, AM or PM, and time zone.</td>
</tr>
<tr>
<td>I</td>
<td>Rich Text Editor</td>
<td>The field appears as a text area in which the user can enter and edit multiple lines of formatted text. The display width and height specify the visible width and number of lines in the rich text editor, respectively. Note: This display type is available for extensible flexfields only.</td>
</tr>
<tr>
<td>J</td>
<td>Color</td>
<td>The field displays a color palette for the user to select a color at run time and assign it to the segment. During setup, this display type appears in the list for selection only if:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• You are working on an extensible flexfield segment.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• The value set for the segment is set to ORA_FND_COLOR_RGBGGBB.</td>
</tr>
<tr>
<td>K</td>
<td>Static URL</td>
<td>The field appears as a text field in which users can enter a fixed URL that opens the web page when clicked.</td>
</tr>
</tbody>
</table>
How Flexfields and Value Sets Work Together

*Value sets* are specific to your *enterprise*. When gathering information using *flexfields*, your enterprise’s value sets validate the values that your users enter based on how you defined the value set.

You can assign a value set to any number of *flexfield segments* in the same or different flexfields. Value set usage information indicates which flexfields use the value set.

The following aspects are important in understanding how flexfields and value sets work together:

- Defining value sets
- Shared value sets
- Deployment

### Defining Value Sets

As a key flexfield guideline, define value sets before configuring the flexfield, because you assign value sets to each segment as you configure a flexfield. With descriptive and extensible flexfields, you can define value sets when adding or editing a segment.

> Note: Ensure that changes to a shared value set are compatible with all flexfield segments that use the value set.

### Shared Value Sets

When you change a value in a shared value set, the change affects the value set for all flexfields that use that value set. The advantage of a shared value set is that a single change propagates to all usages. The drawback is that the change shared across usages may not be appropriate in every case.

### Value Set Values

To configure user-defined attributes to be captured on the value set values in the Manage Value Sets task, configure the *Value Set Values* descriptive flexfield. The object’s code is `FND_VS_VALUES_B`. This flexfield expects the context code to correspond to the value set code. For each value set, you can define a context whose code is the value set code, and whose context-sensitive segments are shown for the values of that value set. By default, the context segment is hidden since it maps to the value set code and is not expected to be changed.

You can also define global segments that are shown for all value sets. However, this would be quite unusual since it would mean that you want to capture that attribute for all values for all value sets.
Deployment

When you deploy a flexfield, the value sets assigned to the segments of the flexfield provide users with the valid values for the attributes represented by the segments.

Default Segment Values

To populate a flexfield segment with a default value when a row is created, specify a default type of constant, parameter, or Groovy, and a default value or expression.

To synchronize a segment's value with another field's value whenever it changes, specify the derivation value to be the flexfield parameter from which to derive the attribute's value. Whenever the parameter value changes, the attribute's value is changed to match. If you derive an attribute from a parameter, consider making the attribute read-only, as values entered by users are lost whenever the parameter value changes. When setting a default value or deriving a default value from a parameter, only those attributes designated by development as parameters are available for selection. Different combinations of making the segments read only or editable in combination with the default or derivation value or both, have different effects.

If your segment's default type is Groovy, you can set the Groovy expression you need using the expression builder. To open the expression builder, select Groovy Expression as your Default Type and click the Groovy Expression Builder icon. But you should know that Groovy defaulting doesn't support derivation when a dependent parameter changes. The expression is evaluated only at segment creation.

The following table maps these different combinations. Initial runtime action corresponds to the row for the attribute value being created in the entity table. If the default value is read only, it can't subsequently be changed through the user interface. If the default value isn't read only, users can modify it. However, if the segment value is a derived value, a user-modified segment value is overwritten when the derivation value changes.

<table>
<thead>
<tr>
<th>Default Type</th>
<th>Default value specified?</th>
<th>Derivation value specified?</th>
<th>Initial runtime action</th>
<th>Runtime action after parameter changes</th>
</tr>
</thead>
<tbody>
<tr>
<td>None</td>
<td>No</td>
<td>Yes</td>
<td>No initial segment value</td>
<td>The changed parameter derivation value updates segment value</td>
</tr>
<tr>
<td>Constant</td>
<td>Yes</td>
<td>No</td>
<td>Default segment value</td>
<td>N/A</td>
</tr>
<tr>
<td>Constant</td>
<td>Yes</td>
<td>Yes</td>
<td>Default segment value</td>
<td>The changed parameter derivation value updates segment value</td>
</tr>
<tr>
<td>Parameter</td>
<td>Yes</td>
<td>No</td>
<td>The default segment value is the parameter's default value</td>
<td>N/A</td>
</tr>
<tr>
<td>Parameter</td>
<td>Yes</td>
<td>Yes, and same as default value</td>
<td>The default segment value is the</td>
<td>The changed parameter derivation</td>
</tr>
<tr>
<td>Default Type</td>
<td>Default value specified?</td>
<td>Derivation value specified?</td>
<td>Initial runtime action</td>
<td>Runtime action after parameter changes</td>
</tr>
<tr>
<td>---------------</td>
<td>--------------------------</td>
<td>-----------------------------</td>
<td>------------------------</td>
<td>----------------------------------------</td>
</tr>
<tr>
<td>Parameter</td>
<td>Yes</td>
<td>Yes, and different from default value</td>
<td>The default segment value is the parameter’s default value</td>
<td>The changed parameter default value doesn’t update segment value. Only the changed derivation value updates the segment value.</td>
</tr>
<tr>
<td>Groovy Expression</td>
<td>Yes</td>
<td>N/A</td>
<td>The default value of the segment is determined by evaluating the groovy expression.</td>
<td>Groovy expressions are evaluated only at segment creation. They’re not evaluated when a dependent parameter is modified.</td>
</tr>
<tr>
<td>SQL</td>
<td>Yes</td>
<td>No</td>
<td>The default segment value is the value returned by executing SQL statement</td>
<td>N/A</td>
</tr>
<tr>
<td>SQL</td>
<td>Yes</td>
<td>Yes</td>
<td>The default segment value is the value returned by executing SQL statement</td>
<td>The changed parameter derivation value updates segment value</td>
</tr>
</tbody>
</table>

### Flexfield Usages

The flexfield usage specifies the table with which the flexfield and its segments are associated. A flexfield can have multiple usages. However, the first table registered for a flexfield indicates the master usage. Segments are based on the master usage. Other usages of the same table for the same flexfield use the same segment setup, though the column names may have a differentiating prefix.

On the Manage Descriptive Flexfields and Manage Extensible Flexfields pages, click the **Show Entity Usages** icon for a specific flexfield to view its entity usage. On the Manage Value Sets page, you can view the flexfield usages for a selected value set.

### Extensible Flexfields

For extensible flexfield contexts, you can configure a different usage. The use of an extensible flexfield context determines the scenarios or user interfaces in which the segments of a context appear to users. For example, the Supplier page displays an extensible flexfield’s supplier usage and the Buyer page for the same flexfield displays the buyer usage. Then, a context that is associated only with the supplier usage appears only on the Supplier page and not on the Buyer page.
Value Sets
The use of value sets specifies the flexfields having segments where the identified value set is assigned.

FAQs for Flexfield Management

How can I access predefined flexfields?
Search for predefined flexfields using the manage flexfields tasks.

1. In the Setup and Maintenance work area, go to any of the following tasks:
   - Manage Descriptive Flexfields
   - Manage Extensible Flexfields
   - Manage Key Flexfields
2. On the page for the type of flexfield you're looking for, enter any of the search parameters and click Search.
   - Tip: If you don't know the flexfield name or the code, use the Module field to filter search results.
3. Click a flexfield to view its details.

For configuration that's not available through the manage flexfields tasks and the UI, contact My Oracle Support at https://support.oracle.com.

Related Topics
- Update Existing Setup Data

Why can't I edit my flexfield or value set configuration?
Your flexfield or value set configuration may be protected. Application developers mark some configurations as protected, indicating that you can't edit them.

Some examples of configurations that may be protected are:
- Descriptive flexfields
- Extensible flexfield contexts
- Extensible flexfield pages
- Value sets

Why did my page not display any flexfield?
For a flexfield to be available on the page, it must be registered by developers and also deployed. The segments appear on the page only after you have successfully deployed the flexfield.

A flexfield's deployment status indicates whether the flexfield segments are available to users. The flexfield segments that users see at run time correspond to the flexfield definition last deployed successfully.

For information about registering flexfields, see the Oracle Fusion Applications Developer's Guide. Some business objects aren't designed to support flexfields. For information about how to enable business objects with flexfield capability, see Getting Started with Flexfields in the Oracle Fusion Applications Developer's Guide.

Note: Oracle Engagement Cloud doesn't support flexfields.
To add attributes to these applications, you may use Application Composer. For more information, see the product-specific documentation.

**Why did my flexfield changes not appear in the runtime UI?**

The ADF business components or artifacts of a flexfield, which are generated into an Oracle Metadata Services (MDS) Repository when the flexfield is deployed, are cached within a user session. You must sign out and sign back in again to view flexfield definition changes reflected in the runtime application user interface page.

**How can I enable flexfield segments for Oracle Social Network Cloud Service?**

When you manage Oracle Social Network Objects during setup and maintenance, search for the business object that includes descriptive flexfields. Select the attributes that are defined as flexfield segments and enable them.

### Flexfield Deployment

#### Overview of Flexfield Deployment

Deployment generates or refreshes the Application Development Framework (ADF) business component objects that render the flexfield in a user interface. The deployment process adds user-defined attributes to the Web Services Description Language (WSDL) schemas exposed by Oracle ADF services and used by SOA composites. Flexfields are deployed for the first time during the application provisioning process. After you configure or change a flexfield, you must deploy it to make the latest definition available to users.

If a descriptive flexfield is enabled for business intelligence, the deployment process redeployes the flexfield's business intelligence artifacts.

You can deploy a flexfield to a sandbox for testing or to the mainline metadata for use in a test or production run time environment. You can deploy extensible flexfields as a background process.

After deployment, the user-defined attributes are available for incorporating into the SOA infrastructure, such as business process and business rule integration. For example, you can now write business rules that depend on the user-defined attributes. You must sign out and sign back in to Oracle Applications Cloud to see the changes you deployed at run time.

The following aspects are important in understanding flexfield deployment:

- Deployment Status
- Initial Deployment Status
- Metadata Validations
- Metadata Synchronization
- Deployment as a Background Process
- Export of Artifacts from Flexfield MDS

#### Deployment Status

Every flexfield has a deployment status. Check the deployment status of your flexfield after patching. The following table lists the different deployment statuses a flexfield can have.
<table>
<thead>
<tr>
<th>Deployment Status</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>Edited</td>
<td>The flexfield metadata definition hasn't been deployed yet. Updates of the metadata definition aren't applied in the run time environment yet.</td>
</tr>
<tr>
<td>Patched</td>
<td>The flexfield metadata definition has been modified through a patch or a data migration action, but the flexfield hasn't yet been deployed. So, the updated definition isn't reflected in the run time environment.</td>
</tr>
<tr>
<td>Deployed to Sandbox</td>
<td>The current metadata for the flexfield is deployed in ADF artifacts and available as a flexfield-enabled sandbox. The status of the sandbox is managed by the Manage Sandboxes dialog box available in the Settings and Actions menu.</td>
</tr>
<tr>
<td>Deployed</td>
<td>The current metadata for the flexfield is deployed in ADF artifacts and available to users. No changes have been made to the flexfield after being deployed to the mainline metadata.</td>
</tr>
<tr>
<td>Error</td>
<td>The deployment attempt in the mainline metadata failed.</td>
</tr>
</tbody>
</table>

**Note:** Whenever a value set definition changes, the deployment status of a flexfield that uses that value set changes to edited. If the change results from a patch, the deployment status of the flexfield changes to patched.

**Initial Deployment Status of Flexfields**

The Oracle Applications Cloud implementation loads flexfield metadata into the database. This initial load sets the flexfield status to Edited. During installation, the application provisioning process deploys the flexfields of the provisioned applications, setting their status to Deployed if no errors occur.

In a provisioned application, deployed flexfields are ready to use. In some cases, flexfield availability at run time requires setup, such as defining key flexfields.

**Metadata Validation**

Use the Validate Metadata command to view possible metadata errors before attempting to deploy the flexfield. Metadata validation is the initial phase of all flexfield deployment commands. By successfully validating metadata before running the deployment commands, you can avoid failures in the metadata validation phase of a deployment attempt. The deployment process ends if an error occurs during the metadata validation phase. Metadata validation results don't affect the deployment status of a flexfield.

**Metadata Synchronization**

When an extensible or descriptive flexfield is deployed, the deployment process regenerates the XML schema definition (XSD). As a result, the user-defined attributes are available to web services and the SOA infrastructure.

After deploying a flexfield configuration, you must synchronize the updated XML schema definition (XSD) files in the MDS repositories for each SOA application.

**Note:** To synchronize the updated XSD files in the MDS repositories in Oracle Cloud implementations, log a service request using My Oracle Support at http://support.com/
Deployment as a Background Process
You can deploy extensible flexfields offline as a background process and continue working in the session without having to wait for the deployment to complete. You can queue up several extensible flexfields and deploy as a background process. The flexfields are deployed, one at a time, in the order that you deploy them to the queue. You must deploy extensible flexfields with more than 30 categories as a background process.

You can remove an extensible flexfield from the deployment queue with the Cancel Background Deployment command. When an extensible flexfield is deployed in a background process, its offline status indicates that the flexfield is in a background deployment process. A flexfield's offline status is cleared and its deployment status updated when the background deployment process has completed.

Export of Artifacts from Flexfield MDS
You can export business components from MDS for descriptive, extensible, or key flexfields, mainly for use in troubleshooting issues with flexfields. Use Download Flexfield Archive on the Manage Flexfields page to export MDS artifacts of the selected flexfield, and import them to an archive on your local computer. You can use these archived business components of flexfields for troubleshooting purposes.
Alternatively, export the deployed artifacts using exportMetadata WLST.

How Flexfield Deployment Status is Calculated
Flexfield deployment status indicates how the flexfield metadata definition in the Oracle Applications Cloud database relates to the Application Development Framework (ADF) business components residing in an Oracle Metadata Services (MDS) Repository.

The following aspects are important in understanding how flexfield deployment status is calculated:

- Settings that affect flexfield deployment status
- How deployment status is calculated

Settings That Affect Flexfield Deployment Status
If you have made a change to a flexfield and expect a changed deployment status, ensure that you have saved your changes. No settings affect flexfield deployment status.

How Deployment Status Is Calculated
If the flexfield definition has been edited through the Define Flexfields activity task flows, the status is Edited. The latest flexfield metadata definition diverges from the latest deployed flexfield definition. Any change, including if a value set used in a flexfield changes, changes the deployment status to Edited. If a flexfield has never been deployed, its status is Edited.

Note: When an application is provisioned, the provisioning framework attempts to deploy all flexfields in that application.

If you deploy the flexfield to a sandbox successfully, the status is Deployed to Sandbox. The latest flexfield metadata definition in the application matches with the metadata definition that generated ADF business components in a sandbox MDS Repository. Whether the sandbox is active or not doesn’t affect the deployment status. If the flexfield was deployed to a sandbox and hasn’t been edited or redeployed to the mainline metadata since then, the status remains Deployed to Sandbox independent of whether the sandbox is active, or who is viewing the status.
If you deploy the flexfield successfully to the mainline metadata, the status is Deployed. The latest flexfield metadata definition in the application matches the metadata definition that generated ADF business components in a mainline MDS Repository. Change notifications are sent when a flexfield is deployed successfully to the mainline metadata. If either type of deployment fails and that the current flexfield definition isn't deployed, the status is Error. The deployment error message gives details about the error. The latest flexfield metadata definition in the application likely diverges from the latest successfully deployed flexfield definition.

If the flexfield definition has been modified by a patch, the status is Patched. The latest flexfield metadata definition in the application diverges from the latest deployed flexfield definition. If the flexfield definition was Deployed before the patch and then a patch was applied, the status changes to Patched. If the flexfield definition was Edited before the patch and then a patch was applied, the status remains at Edited to reflect that there are still changes (outside of the patch) that aren't yet in effect.

When a deployment attempt fails, you can access the Deployment Error Message for details.

How Deploying a Flexfield-Enabled Sandbox Works with Mainline Metadata

The flexfield definition in a sandbox corresponds to the flexfield metadata definition in the Oracle Applications Cloud database at the time the flexfield was deployed to the sandbox. When the flexfield is ready for end users, the flexfield must be deployed to the mainline metadata.

A flexfield-enabled sandbox uses the following components.

- Flexfield metadata in the Oracle Applications Cloud database
- Flexfield business components in a sandbox Oracle Metadata Services (MDS) repository
- User interface modifications for the flexfield in the mainline MDS repository

The following figure shows the two types of deployment available in the Manage Flexfield tasks of the Define Flexfields activity. Deploying a flexfield to a sandbox creates a sandbox MDS Repository for the sole purpose of testing flexfield behavior. The sandbox is only accessible to the administrator who activates and accesses it, not to users generally. Deploying a flexfield to the mainline metadata applies the flexfield definition to the mainline MDS Repository where it is available to end users. After deploying the flexfield to the mainline metadata, modify the page where the flexfield
segments appear. Modifications done to the page in the sandbox MDS Repository cannot be published to the mainline MDS Repository.

---

**Sandbox Metadata Services Repository Data**

Deploying the flexfield to a sandbox generates the Application Development Framework (ADF) business components of a flexfield in a sandbox MDS Repository for testing in isolation.

**Caution:** Don’t modify flexfield segment display properties using Page Composer in a flexfield-enabled sandbox as these changes will be lost when deploying the flexfield to the mainline metadata.
Mainline Metadata Services Repository Data

The Oracle Fusion Applications database stores the single source of truth about a flexfield. When the flexfield is deployed, the ADF business component objects that implement the flexfield in the run time user interface are generated in the mainline MDS Repository from this source.

Related Topics
- How You Manage Configurations in Classic Sandboxes

Considerations for Deploying a Flexfield to a Sandbox

Deploying a flexfield to a sandbox creates a flexfield-enabled sandbox. Each flexfield-enabled sandbox contains only one flexfield.

You can test the run time behavior of a flexfield in the flexfield-enabled sandbox. If changes are needed, you return to the Define Flexfield tasks to change the flexfield definition.

When you deploy a flexfield to sandbox, the process reads the metadata about the segments from the database, generates flexfield Application Development Framework (ADF) business component artifacts based on that definition, and stores in the sandbox only the generated artifacts derived from the definition.

When you deploy a flexfield sandbox, the process generates the name of the flexfield sandbox, and that flexfield sandbox is set as your current active sandbox. When you next sign in to the application, you can see the updated flexfield configurations. The Oracle Applications Cloud global header displays your current session sandbox.

Note: Unlike a standalone sandbox created using the Manage Sandboxes dialog box, the sandbox deployed for a flexfield contains only the single flexfield. You can manage flexfield sandboxes, such as setting an existing flexfield sandbox as active or deleting it, using the Manage Sandboxes dialog box.

When you deploy a flexfield to the mainline metadata after having deployed it to the sandbox, the sandbox-enabled flexfield is automatically deleted.

Sandbox MDS Repository Data

The sandbox data lets you test the flexfield in isolation without first deploying it in the mainline metadata where it could be accessed by users.

Caution: Don't modify flexfield segment display properties using Page Composer in a flexfield-enabled sandbox as these changes will be lost when deploying the flexfield to the mainline metadata.

Managing a Flexfield-Enabled Sandbox

When you deploy a flexfield as a sandbox, that flexfield-enabled sandbox automatically gets activated in your user session. When you sign back in to see the changes, the sandbox is active in your session.

You can only deploy a flexfield to a sandbox using the Define Flexfields task flow pages.

You also can use the Manage Sandboxes dialog box in the Administrator menu of the Setup and Maintenance work area to activate and access a flexfield-enabled sandbox.

Note: Whether you use the Define Flexfields or Manage Sandboxes task flows to access a flexfield-enabled sandbox, you must sign out and sign back in before you can see the changes you deployed in the run time.
You cannot publish the flexfield from the sandbox to the mainline metadata. You must use the Define Flexfields task flow pages to deploy the flexfield for access by users of the mainline metadata because the flexfield configuration in the mainline metadata is the single source of truth.

**Related Topics**
- How You Manage Configurations in Classic Sandboxes

## Value Sets

### Overview of Value Sets

A **value set** is a group of valid values that you assign to a **flexfield segment** to control the values that are stored for business object attributes.

A user enters a value for an attribute of a business object while using the application. The **flexfield** validates the value against the set of valid values that you configured as a value set and assigned to the segment.

For example, you can define a required format, such as a five-digit number, or a list of valid values, such as green, red, and blue.

Flexfield segments are usually validated, and typically each segment in a given flexfield uses a different value set. You can assign a single value set to more than one segment, and you can share value sets among different flexfields.

**Note:** Ensure that changes to a shared value set are compatible with all flexfield segments using the value set.

The following aspects are important in understanding value sets:

- Managing value sets
- Validation
- Security
- Precision and scale
- Usage and deployment
- Protected value set data

### Managing Value Sets

To open the Manage Value Sets page, use the Manage Value Sets task. You can also use the Manage Descriptive Flexfields and Manage Extensible Flexfields tasks for configuring a segment, including its value set. To open the Manage Values page, select the value set from the Manage Value Sets page, and click **Manage Values**. Alternatively, click **Manage Values** from the Edit Value Set page.

### Validation

The following types of validation are available for value sets:

- Format only, where users enter data instead of selecting values from a list
- Independent, a list of values consisting of valid values you specify
- Dependent, a list of values where a valid value derives from the independent value of another segment
• Subset, where the list of values is a subset of the values in an existing independent value set
• Table, where the values derive from a column in an application table and the list of values is limited by a WHERE clause

A segment that uses a format only value set doesn't present a list of valid values to users. If required, you may add table validated value sets to the list of available value sets available for configuration.

**Note:** For the Accounting Key Flexfield value sets, you must use independent validation only. If you use other validations, you can’t use the full chart of accounts functionality, such as data security, reporting, and account hierarchy integration.

### Security

Value set security only works in conjunction with usage within flexfield segments. You can specify that data security be applied to the values in flexfield segments that use a value set. Based on the roles provisioned to users, data security policies determine which values of the flexfield segment users can view or modify.

The application of value set security has the following conditions:

• At the value set level: The value set is the resource secured by data security policies. If a value set is secured, every usage of it in any flexfield is secured. Disabling security for individual usages of the same value set isn’t possible.
• Applies to independent, dependent, or table-validated value sets.
• Applies mainly when data is being created or updated, and to key flexfield combinations tables for query purposes. Value set security doesn’t determine which descriptive flexfield data is shown upon querying.
• Security conditions defined on value sets always use table aliases. When filters are used, table aliases are always used by default. When predicates are defined for data security conditions, make sure that the predicates also use table aliases.

For **key flexfields**, the attributes in the view object corresponding to the account combination ID, structure instance number (SIN), and data set number (DSN) can't be transient. They must exist in the database table. For key flexfields, the SIN segment is the discriminator attribute, and the account combination segment is the common attribute.

### Precision and Scale

If the data type of a value set is Number, you can specify the precision (maximum number of digits user can enter) or scale (maximum number of digits following the decimal point).

### Usage and Deployment

The usage of a value set is the flexfields where that value set is used. The deployment status of flexfields in which the value set is used indicates the deployment status of the value set instance.
The following figure shows a value set used by a segment in a key flexfield and the context segment of a descriptive flexfield.

For most value sets, when you enter values into a flexfield segment, you can enter only values that already exist in the value set assigned to that segment.

Global and context-sensitive segment require a value set. You can assign a value set to a descriptive flexfield context segment. If you specify only context values, not value sets for contexts, the set of valid values is equal to the set of context values.

Protected Value Set Data
Application developers may mark some value sets as protected, indicating that you can't edit them.

You can edit only value sets that are not marked as protected. You can't edit or delete protected value sets. If the value set type supports values (such as independent, dependent or subset value sets), then you can't add, edit, or delete values.
Note: References to protected value sets aren't restricted. Value sets, protected or not, may be assigned to any flexfield segment. Likewise, other value sets may reference protected value sets; for example, an unprotected dependent value set may reference a protected independent value set.

Related Topics
- Chart of Accounts Components
- What's the difference between a lookup type and a value set

Validation Type Options for Value Sets
Validation and usage of value sets determine where and how users access valid values for attributes represented by flexfield segments.

Tip: As a flexfield guideline, define value sets before configuring the flexfield, because you can assign value sets to each segment as you configure a flexfield. With descriptive and extensible flexfield segments, you can create value sets when adding or editing a segment on the runtime page where the flexfield appears.

The following aspects are important in defining value sets:
- Value sets for context segments
- Format-only validation
- Interdependent value sets
- Table validation
- Range
- Security
- Testing and maintenance

Value Sets for Context Segments
When assigning a value set to a context segment, you can only use table-validated or independent value sets.

You can use only table and independent value sets to validate context values. The data type must be character and the maximum length of the values being stored must not be larger than the context's column length. If you use a table value set, the value set can't reference flexfield segments in the value set's WHERE clause, other than the flexfield segment to which the value set is assigned.

Format Only Validation
The format only validation type enables users to enter any value, as long as it meets your specified formatting rules. The value must not exceed the maximum length you define for your value set, and it must meet any format requirements for that value set.

For example, if the value set permits only numeric characters, users can enter the value 456 (for a value set with maximum length of three or more), but can't enter the value ABC. A format only value set doesn't otherwise restrict the range of different values that users can enter. For numeric values, you can also specify if a numeric value should be zero filled or how many digits should follow the radix separator.

Interdependent Value Sets
Use an independent value set to validate data against a list that isn't stored in an application table, and not dependent on a subset of another independent value set. You can't specify a dependent value set for a given segment without
having first defined an independent value set that you apply to another segment in the same flexfield. Use a dependent value set to limit the list of values for a given segment based on the value that the user has defined for a related independent segment. The available values in a dependent list and the meaning of a given value depend on which value was selected for the independently validated segment.

For example, you could define an independent value set of the states in the USA with values such as CA, NY, and so on. Then you define a dependent value set of cities in the USA with values such as San Francisco and Los Angeles that are valid for the independent value CA. Similarly, New York City and Albany are valid for the independent value NY. In the UI, only the valid cities can be selected for a given state.

Because you define a subset value set from an existing independent value set, you must define the independent value set first. Users don’t have to select a value for another segment first to have access to the subset value set.

Independent, dependent, and subset value sets require a user-defined list of valid values. Use the Manage Values page to create and manage a value set’s valid values and the order in which they appear.

**Tip:** You can configure the Manage Value Sets page to capture additional attributes for each valid value by adding context-sensitive segments in a new context for FND_VS_VALUES_B descriptive field.

**Table Validation**

Typically, you use a table-validated set when the values you want to use are already maintained in an application table, such as a table of supplier names. Specify the table column that contains the valid value. You can optionally specify the description and ID columns, a WHERE clause to limit the values to use for your set, and an ORDER BY clause.

If you specify an ID column, then the flexfield saves the ID value, instead of the value from the value column, in the associated flexfield segment. If the underlying table supports translations, you can enable the display of translated text by basing the value set’s value column on a translated attribute of the underlying table. You should also define an ID column that’s based on an attribute that isn’t language-dependent so that the value’s invariant ID (an ID that doesn’t change) is saved in the transaction table. The run time displays the corresponding translated text from the value column for the run time session’s locale.

Table validation lets you enable a segment to depend on multiple prior segments in the same context structure. You can’t reference other flexfield segments in the table-validated value set’s WHERE clause. Which means, the WHERE clause can’t reference SEGMENT.segment_code or VALUESET.value_set_code.

Table-validated value sets have unique values across the table, irrespective of bind variables. The WHERE clause fragment of the value set is considered if it doesn’t have bind variables. If it has bind variables, the assumption is that the values are unique in the value set. If you use table validated value sets for key flexfields, then you can’t use all integration options supported for key flexfields, such as:

- Data security
- Oracle Transactional Business Intelligence (OTBI)
- Extended Spread Sheet Database (ESSbase)
- Tree or hierarchy integration

To use these integration options for key flexfields, you must use independent value sets only.

**Range**

In the case of format, independent, or dependent value sets, you can specify a range to limit which values are valid. You can specify a range of values that are valid within a value set. You can also specify a range validated pair of segments where one segment represents the low end of the range and another segment represents the high end of the range.

For example, you might specify a range for a format-only value set with format type Number where the user can enter only values between 0 and 100.
Security
In the case of independent and dependent values, you can specify that data security be applied to the values in segments that use a value set. Based on the roles provisioned to users, data security policies determine which values of the flexfield segment users can view or modify.

To enable security on a value set, specify a database resource, typically the code value for the value set. Using the Manage Database Security Policies task, specify conditions, such as filters or SQL predicates, and policies that associate roles with conditions. You can use a filter for simple conditions. For more complex conditions, use a SQL predicate.

Value set data security policies and conditions differ from data security conditions and policies for business objects in the following ways:

- You can grant only read access to users. You can't specify any other action.
- When defining a condition that's based on a SQL predicate, use VALUE, VALUE_NUMBER, VALUE_DATE, VALUE_TIMESTAMP, or VALUE_ID to reference the value from a dependent, independent, or subset value set. For table value sets, use a table alias to define the table, such as &TABLE_ALIAS category=70.

When you enable security on table-validated value sets, the security rule that's defined is absolute and not contingent upon the bind variables (if any) that may be used by the WHERE clause of the value set. For example, suppose a table-validated value set has a bind variable to further filter the value list to x, y and z from a list of x, y, z, xx, yy, zz. The data security rule or filter written against the value set must not assume anything about the bind variables. Instead the whole list of values must be available and you write the rule, for example, to permit x, or to permit y and z. By default in data security, all values are denied and show only rows to which access has been provided.

Testing and Maintenance
You don't have to define or maintain values for a table-validated value set, as the values are managed as part of the referenced table or independent value set, respectively.

You can't manage value sets in a sandbox.

When you change an existing value set, the deployment status for all affected flexfields changes to Edited. You must redeploy all flexfields that use that value set to make the flexfields reflect the changes. In the UI pages for managing value sets, the value set's usages show which flexfields are affected by the value set changes.

If your application has more than one language installed, or there is any possibility that you might install one or more additional languages for your application in the future, select Translatable. This doesn't require you to provide translated values now, but you can't change this option if you decide to provide them later.

Considerations for Planning Value Sets
The value sets you create and configure depend on the valid values on the business object attributes that will use the value set. When creating value sets, you first give the value set a name and description, and then define the valid values of the set.

The following aspects are important in planning value sets:

- List of values
- Plain text
- Value ranges
- Value format specification
- Security
List of Values
You can use one of the following types of lists to specify the valid values for a segment:

- Table column
- User-defined list. Also include a sub list.
- Dependent user-defined list

If the valid values exist in a table column, use a table value set to specify the list of values. To limit the valid values to a subset of the values in the table, use a SQL WHERE clause. Table value sets also provide some advanced features, such as enabling validation depending on other segments in the same structure.

Use an independent value set to specify a user-defined set of valid values. For example, you can use an independent value set of Mon, Tue, Wed, and so forth to validate the day of the week. You can also specify a subset of an existing independent value set as the valid values for a segment. For example, if you have an independent value set for the days of the week, then a weekend subset can comprise entries for Saturday and Sunday.

Use a dependent value set when the available values in the list and the meaning of a given value depend on which independent value was selected for a previously selected segment value. For example, the valid holidays depend on which country you are in. A dependent value set is a collection of value subsets, with one subset for each value in a corresponding independent value set.

For lists of values type value sets, you can additionally limit the valid values that an end user can select or enter by specifying format, minimum value, and maximum value. For list of values type value sets, you can optionally implement value set data security. If the applications are running in different locales, you might need to provide different translations for the values and descriptions.

Plain Text
Use a format-only value set when you want to allow users to enter any value, as long as that value conforms to formatting rules. For example, if you specify a maximum length of 3 and numeric-only, then end users can enter 456, but not 4567 or 45A. You can also specify the minimum and maximum values, whether to align the text to either side, and whether to zero-fill. With a format-only value set, no other types of validation are applied.

Value Ranges
You can use either a format-only, independent, or dependent value set to specify a range of values. For example, you might create a format-only value set with Number as the format type where the end user can enter only the values between 0 and 100. Or, you might create a format-only value set with Date as the format type where the end user can enter only dates for a specific year, such as a range of 01-JAN-93 to 31-DEC-93. Because the minimum and maximum values enforce these limits, you need not define a value set that contains each of these individual numbers or dates.

Value Format
Flexfield segments commonly require some kind of format specification, regardless of validation type. Before creating a value set, consider how you will specify the required format.

The following table shows options for validation type and value data type.

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Value data type</td>
<td>Character, Number, Date, Date Time.</td>
</tr>
<tr>
<td>Value subtype</td>
<td>Text, Translated text, Numeric digits only, Time (20:08), Time (20:08:08).</td>
</tr>
</tbody>
</table>
Option | Description
--- | ---
An additional data type specification for the Character data type for the Dependent, Independent, and Format validation types.
Maximum length | Maximum number of characters or digits for Character data type.
Precision | Maximum number of digits the user can enter.
Scale | Maximum number of digits that can follow the decimal point.
Uppercase only | Lowercase characters automatically changed to uppercase.
Zero fill | Automatic text alignment and zero-filling of entered numbers (affects values that include only the digits 0-9).

Note: You cannot change the text value data type to a translated text value subtype after creating a value set. If there is any chance you may need to translate displayed values into other languages, choose Translated text. Selecting the Translated text subtype doesn’t require you to provide translated values.

Value Sets for Context Segments
You can use only table and independent value sets to validate context values. The data type must be character and the maximum length of the values being stored must not be larger than the context's column length. If you use a table value set, the value set cannot reference flexfield segments in the value set's WHERE clause other than the flexfield segment to which the value set is assigned.

Security
When enabling security on a value set, the data security resource name is an existing value set or one that you want to create. The name typically matches the code value for the value set. You cannot edit the data security resource name after you save your changes.

Related Topics
• What's the difference between a lookup type and a value set

Considerations for Bind Variables in Table-Validated Value Sets
After you assign a value set to a flexfield, you can use bind variables in the WHERE clause.

These bind variables refer to flexfield elements:
• :{(SEGMENT..<segment_code>)}
• :{(CONTEXT.<context_code>;SEGMENT.<segment_code>)}
• :{(VALUESET.<value_set_code>)}
• :{(FLEXFIELD.<internal_code>)}
• :{(PARAMETER.<parameter_code>)}
Segment Code

: {SEGMENT.<segment_code>}

This bind variable refers to the ID or value of a segment where <segment_code> identifies the segment. Where referring to the ID, the value set is ID-validated. Where referring to the value, the value set isn't ID-validated. The data type of the bind value is the same as the data type of the segment's column.

For both descriptive and extensible flexfields, the segment must be in the same context as the source segment. The source segment contains the WHERE clause. For descriptive flexfields, if the segment is global, then the source segment must be global.

The segment must have a sequence number that's less than the sequence number of the target segment with this bind variable. A matching segment must exist in the current flexfield context.

This bind variable is useful when the set of valid values depends on the value in another segment. For example, the values to select from a CITIES table might depend upon the selected country. If SEGMENT1 contains the country value, then the WHERE clause for the CITIES table might be <country_code> = :{SEGMENT.SEGMENT1}.

Context Code

: {CONTEXT.<context_code>; SEGMENT.<segment_code>}

This bind variable, which is valid only for extensible flexfields, refers to the ID (if the value set is ID-validated) or value (if not ID-validated) of a segment that's in a different context than the target segment (the segment with the WHERE clause).

- The <context_code> identifies the context and must be in the same category or in an ancestor category. It can't be a multiple-row context.
- The <segment_code> identifies the segment. The data type of the bind value is the same as the data type of the segment's column.

**Note:** The target segment should appear in the UI after the source segment to ensure the source segment has a value. If the target segment's context is a single-row context, the source and target segments must be on separate pages and the target page must follow the source page.

The framework of extensible flexfields doesn't perform any additional validation related to mismatched values for segments defined with cross context bind parameters. Administrators must populate the correct pair of segment values.

This bind variable is useful when the set of valid values depends on the value of a segment in another context. For example, the values to select from a CERTIFICATION table for a segment in the Compliance and Certification context might depend on the value of the country segment in the Manufacturing context.

Value Set Code

: {VALUESET.<value_set_code>}

This bind variable refers to the ID (if the value set is ID-validated) or value (if not ID-validated) of the segment that's assigned to the value set that's identified by the <value_set_code>. The data type of the bind value is the same as the data type of the segment's column.

The segment must have a sequence number that's less than the sequence number of the segment with this bind variable. If more than one segment is assigned to the value set, the closest prior matching segment will be used to resolve the bind expression. A matching segment must exist in the current flexfield context.

This bind variable is useful when the set of valid values depends on the value in another segment and that segment code can vary, such as when the value set is used for more than one context or flexfield. For example, the values
to select from a CITIES table might depend upon the selected country. If the value set for the segment that contains the country value is COUNTRIES, then the WHERE clause for the CITIES table might be `<country_code> = : (VALUESET.COUNTRIES)`. 

**Flexfield Internal Code**

:`{FLEXFIELD.<internal_code>}`

This bind variable refers to an internal code of the flexfield in which the value set is used, or to a validation date. The `internal_code` must be one of the following:

- **APPLICATION_ID** - the application ID of the flexfield in which this value set is used. The data type of APPLICATION_ID and its resulting bind value is NUMBER.
- **DESCRIPTIVE_FLEXFIELD_CODE** - the identifying code of the flexfield in which this value set is used. The data type of DESCRIPTIVE_FLEXFIELD_CODE and its resulting bind value is VARCHAR2. Note that you use this string for both descriptive and extensible flexfields.
- **CONTEXT_CODE** - the context code of the flexfield context in which this value set is used. The data type of CONTEXT_CODE and its resulting bind value is VARCHAR2.
- **SEGMENT_CODE** - the identifying code of the flexfield segment in which this value set is used. The data type of SEGMENT_CODE and its resulting bind value is VARCHAR2.
- **VALIDATION_DATE** - the current database date. The data type of VALIDATION_DATE and its resulting bind value is DATE.

**Flexfield Parameters**

:`{PARAMETER.<parameter_code>}`

This bind variable refers to the value of a flexfield parameter where parameter_code identifies the parameter. The data type of the resulting bind value is the same as the parameter's data type.

**Note:** You can't assign a table value set to a context segment if the WHERE clause uses VALUESET.value_set_code or SEGMENT.segment_code bind variables.

**Create Table-Validated Value Sets Based on Lookups**

In an application user interface, you want to display a list of values that customers use to enter satisfaction scores. The value column name is 1, 2, 3, 4, 5 and the value column description is Extremely Satisfied, Satisfied, and so on. Users can select the appropriate value or description which stores the corresponding name so the name value can be used in a calculation expression.

In this case, you can use the FND_LOOKUPS table as the basis for a table-validated value set. The lookup meaning corresponds to the Value Column Name and the lookup description corresponds to the Description Column Name. The following table lists the properties of the value set.

<table>
<thead>
<tr>
<th>Property</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>FROM clause</td>
<td>FND_LOOKUPS</td>
</tr>
<tr>
<td>WHERE clause</td>
<td><code>lookup_type = 'CN_XX_CUST_SATISFACT_SCORE'</code></td>
</tr>
</tbody>
</table>
Creating a Value Set Based on a Lookup

1. From the Setup and Maintenance work area, open the Manage Value Sets task and click the Go to Task icon button.
2. On the Manage Value Sets page, click the Create icon button.
3. On the Create Value Set page, enter the following values:
   a. In the Value Set Code field, enter \texttt{CN\_XX\_CUSTOMER\_SATISFACTION\_SCORES}
   b. In the Description field, enter Customer satisfaction score.
   c. In the Module field, select Search
   d. In the Search and Select: Module subwindow, enter \texttt{Incent} in the User Module Name field
   e. Select Incentive Compensation.
   f. Click \texttt{OK}.
4. On the Create Value Set page, enter the following values:
   a. In the Validation Type field, select Table.
   b. In the Value Data Type field, select Character.
   c. In the Definition section FROM Clause field, enter \texttt{FND\_LOOKUPS}.
   d. In the Value Column Name field, enter \texttt{DESCRIPTION}.
   e. In the Description Column Name field, enter \texttt{MEANING}.
   f. In the ID Column Name field, enter \texttt{LOOKUP\_CODE}.
   g. In the Enabled Column Name field, enter ‘Y’.
   h. In the Start Date Column Name field, enter \texttt{START\_DATE\_ACTIVE}.
   i. In the End Date Column Name field, enter \texttt{END\_DATE\_ACTIVE}.
   j. In the WHERE Clause field, enter \texttt{LOOKUP\_TYPE = 'CN\_XX\_CUST\_SATISFACT\_SCORE'}.
5. Click \texttt{Save and Close}.
6. In the Manage Value Sets page, click Done.
Add Attributes to the Manage Value Sets Page

You can add attributes to independent, dependent, and subset value sets. The attributes appear on the Manage Value Sets page where you can store additional information about each valid value. To display attributes on an application page, you must programmatically modify the application.

To add attributes and subsequently view them on the Manage Value Sets page, perform the following steps:

1. Using the Manage Descriptive Flexfields task, find the FND_VS_VALUES_B flexfield and open it for editing.
2. Click Manage Contexts.
3. Create a new context and use the value set code for the context code.
4. Add new attributes as context-sensitive segments and save the changes.
5. Deploy FND_VS_VALUES_B to run time.
6. Sign out and sign back in.
7. Open the Manage Value Sets page to view the new attributes.

Import Value Set Values

You can import a file containing values that you want to edit or add to a given independent or dependent value set. For example, uploading a hundred values may be more efficient than creating them individually using the Manage Value Sets task. However, for just a few values, it may be quicker to perform the relevant tasks.

Importing Value Set Values

To import value set values:

1. Create a flat file containing the values in the value set that you want to add or update.

   **Note:**
   - When creating the file, you must specify an existing value set code to which you want to add values or edit existing values. If the value set does not exist, add the value set using the appropriate Manage Value Sets setup task in the Setup and Maintenance work area.
   - The file that you create must adhere to the formatting and content requirements for creating flat files containing value set values.

2. Upload the flat file to the content repository using the File Import and Export page.
3. Import the file using the appropriate Manage Value Sets setup task in the Setup and Maintenance work area. To import the file:
   a. Click Actions > Import in the Manage Value Sets page.
   b. In the File Name field, enter the name of the flat file you uploaded using the File Import and Export page.
   c. In the Account field, select the user account containing the flat file.
   d. Click Upload.

   **Note:** Alternatively, you can import the file using either of the following methods:
   - Run the Upload Value Set Values scheduled process.
   - Use the Applications Core Metadata Import web service. For more information on the Applications Core Metadata Import web service, see the SOAP Web Services guide for your cloud services.
Related Topics

- Overview of Files for Import and Export

Requirements for Flat Files to Upload Value Set Values

You can import large volumes of *value set* value data from the content repository. To upload value set values to the content repository, create a flat file containing the values in the value set that you want to add or update. You upload these flat files to the content repository using the File Import and Export page.

General Requirements

The first line of the flat file must contain the column names for the value set value data, including all mandatory columns, and separated by the '|' (pipe) character. Each subsequent line should contain a row of data specified in the same order as the column names, also separated by the '|' character.

The requirements for creating flat files vary with the type of value sets:

- Independent value sets
- Dependent value sets

Independent Value Set

A flat file for uploading values for independent value sets must contain the mandatory columns. The following table lists the three mandatory columns and their data types.

<table>
<thead>
<tr>
<th>Column Name</th>
<th>Data Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>ValueSetCode</td>
<td>VARCHAR2(60)</td>
</tr>
<tr>
<td>Value</td>
<td>VARCHAR2(150)</td>
</tr>
<tr>
<td>Enabled Flag</td>
<td>VARCHAR2(1), Y or N</td>
</tr>
</tbody>
</table>

**Note:** You can also specify optional columns.

Examples:

- To upload values to a COLORS independent value set with the minimum columns, you can use the following flat file:
  ```
  ValueSetCode | Value | EnabledFlag
  COLORS | Red | Y
  COLORS | Orange | Y
  COLORS | Yellow | Y
  ```

- To upload values to a STATES independent value set with more (optional) columns, you can use the following flat file:
  ```
  ValueSetCode | Value | Description | EnabledFlag
  ```
Dependent Value Sets

A flat file for uploading values for dependent value sets must contain the mandatory columns. The following table lists the four mandatory columns and their data types.

<table>
<thead>
<tr>
<th>Column Name</th>
<th>Data Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>Value Set Code</td>
<td>VARCHAR2(60)</td>
</tr>
<tr>
<td>Independent Value</td>
<td>VARCHAR2(150)</td>
</tr>
<tr>
<td>Value</td>
<td>VARCHAR2(150)</td>
</tr>
<tr>
<td>Enabled Flag</td>
<td>VARCHAR2(1), Y or N</td>
</tr>
</tbody>
</table>

Note: You can also specify optional columns.

Example:

To upload values to a CITIES dependent value set (dependent on the STATES independent value set), you can use the following flat file:

```
ValueSetCode | IndependentValue | Value | EnabledFlag
-------------|------------------|-------|-------------
CITIES | AK | Juneau | Y
CITIES | AK | Anchorage | Y
CITIES | CA | San Francisco | Y
CITIES | CA | Sacramento | Y
CITIES | CA | Los Angeles | Y
CITIES | CA | Oakland | Y
```

Additional Optional Columns

In addition to the mandatory columns, you can add optional columns. The following table lists the optional columns for both dependent and independent value sets.

<table>
<thead>
<tr>
<th>Column Name</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>Translated Value</td>
<td>VARCHAR2(150), for use in value sets that are translatable</td>
</tr>
<tr>
<td>Description</td>
<td>VARCHAR2(240)</td>
</tr>
<tr>
<td>Start Date Active</td>
<td>DATE, formatted as YYYY-MM-DD</td>
</tr>
</tbody>
</table>
### Upload Value Set Values Process

This process uploads a flat file containing *value set* values for *flexfields*. You can use the *scheduled process* to upload a file containing values you want to edit or add to an existing independent or dependent value set. This process is useful for adding or updating large volumes of value set value data in an automated or recurring fashion. For example, you can upload a hundred values on a recurring basis when scheduled as a recurring process. This method could be more efficient than using the Import action in the Manage Value Sets tasks in the Setup and Maintenance work area. However, for a task of uploading a hundred values, it may be quicker to use the Import action in the relevant tasks.

Run this process from the Scheduled Processes Overview page. You can run it on a recurring basis whenever the flat file in the content repository account is updated.

You must create the flat file containing the values data, and upload the flat file to the content repository using the File Import and Export page.

### Parameters

**Flat File Name**

Enter the name of the flat file you uploaded using the File Import and Export page.

**Account**

Select the user account containing the flat file in the content repository to upload.

### Related Topics

- Overview of Files for Import and Export
- Overview of Scheduled Processes
Translation of Flexfield and Value Set Configurations

When you first configure a flexfield or segment, the translatable text that you enter, such as prompts and descriptions, is stored as the text for all installed locales. You may then provide a translation for a particular locale. If you don't provide a translation for a given locale, then the value that was first entered is used for that locale.

To translate the text for a particular locale, sign in with that locale, or in the global header, select Settings and Actions > Personalization > Set Preferences and specify the locale. Then, update the translatable text in the flexfield using the Manage Descriptive Flexfields task, Manage Key Flexfields task, or Manage Extensible Flexfields task. Your modifications change the translated values only for the current session's locale.

After you complete the translations, deploy the flexfield.

You can define translations for a dependent value set or an independent value set, if the value set is of type Character with a subtype Translated text. You define the translations by setting the current session to the locale for which you want to define the translation. Then use the Manage Value Sets task to enter the translated values and descriptions for that locale.

You can define translated values for a table value set for which multiple languages are supported and that the value set's value column is based on a translated attribute of the underlying table. For more information about using multilanguage support features, see the Oracle Fusion Applications Developer's Guide.

FAQs for Value Sets

What happens if a value set is security enabled?

Value set security is a feature that enables you to secure access to value set values based on the role of the user in the application.

As an example, suppose you have a value set of US state names. When this value set is used to validate a flexfield segment, and users can select a value for the segment, you can use value set security to restrict them to selecting only a certain state or subset of states based on their assigned roles in the application.

For example, Western-region employees may choose only California, Nevada, Oregon, and so on as valid values. They cannot select non-Western-region states. Eastern-region employees may choose only New York, New Jersey, Virginia, and so on as valid values, but cannot select non-Eastern-region states. Value set security is implemented using Oracle Applications Cloud data security.

How can I set a default value for a flexfield segment?

When you define or edit a flexfield segment, you pick a value from the assigned value set and set it as default.

You can set the default value for a descriptive flexfield segment to be a parameter. The mapped entity object attribute provides the initial default value for the segment.

You can set the default value to be a constant, if appropriate to the data type of the value set assigned to the segment.

In addition to an initial default value, you can set a derivation value for updating the attribute's value every time the parameter value changes. The parameter you select identifies the entity object source attribute. Any changes in the value of the source attribute during run time are reflected in the value of the segment.
If the display type of the segment is a check box, you can set whether the default value of the segment is checked or unchecked.

## Descriptive Flexfields

### Overview of Descriptive Flexfields

Use **descriptive flexfields** to add attributes to business object entities, and define validation for them. All the business object entities that you can use in the application are enabled for descriptive flexfields. However, configuring descriptive flexfields is an optional task.

### Context

A descriptive flexfield can have only one context segment to provide context sensitivity. The same underlying database column can be used by different segments in different contexts.

For example, you can define a Dimensions context that uses the following attributes:

- ATTRIBUTE1 column for height
- ATTRIBUTE2 column for width
- ATTRIBUTE3 column for depth

You can also define a Measurements context that uses the same columns for other attributes:

- ATTRIBUTE1 column for weight
- ATTRIBUTE2 column for volume
- ATTRIBUTE3 column for density

### Segments and Contexts

The following table lists the different types of descriptive flexfield **segments**.

<table>
<thead>
<tr>
<th>Segment Type</th>
<th>Run Time Appearance</th>
</tr>
</thead>
<tbody>
<tr>
<td>Global segment</td>
<td>Always available</td>
</tr>
<tr>
<td>Context segment</td>
<td>Determines which context-sensitive segments are displayed</td>
</tr>
<tr>
<td>Context-sensitive segment</td>
<td>Displayed depending on the value of the context segment</td>
</tr>
</tbody>
</table>
The following figure displays a descriptive flexfield having one context segment called Category for which there are three values: Resistor, Battery, and Capacitor. Additionally, the descriptive flexfield comprises two global segments that appear in each context, and three context-sensitive segments that only appear in the specific context.

Application development determines the number of segments available for configuring. During implementation, configure the flexfield by determining the following:

- Attributes to add using the available segments
- Context values
- The combination of attributes in each context

**Value Sets**

For each global and context-sensitive segment, you configure the values permitted for the segment. Based on it, the values that end users enter are validated, including interdependent validation among the segments.

**Protected Descriptive Flexfield Data**

Application developers may mark some data configurations in a descriptive flexfield as protected, indicating that you can't edit them.
Considerations for Planning Descriptive Flexfields

Once you have identified a flexfield to configure, plan the configuration in advance. Compile a list of the UI pages and other artifacts in your deployment that are affected by the configuration. Verify that you are provisioned with the roles needed to view and configure the flexfield. View the flexfield using the Highlight Flexfields command in the Administration menu while viewing the run time page where the flexfield appears. Plan how you will deploy the flexfield for test and production users. Review the tools and tasks available for managing flexfields for adding and editing flexfield segments.

Planning a descriptive flexfield can involve the following tasks:

1. Identify existing parameters.
2. Identify existing context values and whether the context value is derived.
3. Identify user-defined attributes and plan the descriptive flexfield segments, segment properties, and structure.
5. Plan initial values.
6. Plan attribute mapping to Oracle Business Intelligence objects.

Identify Existing Descriptive Flexfield Parameters

Some descriptive flexfields provide parameters that can be used to specify the initial value of a descriptive flexfield segment. The parameter is external reference data, such as a column value or a session variable. For example, if a flexfield has a user email parameter, you can configure the initial value for a customer email attribute to be derived from that parameter.

Review the list of available parameters in the Derivation Value field in the Create Segment page for a descriptive flexfield. If you decide to use one of the parameters to set an initial value, select that parameter from the Derivation Value drop-down list when you add the descriptive flexfield segment.

Evaluate Whether the Context Value Is Derived

The context value for a descriptive flexfield might have been preconfigured to be derived from an external reference. For example, if the context is Marriage Status, then the value might be derived from an attribute in the employee business object. When the context value is derived, you might need to take the derived values and their source into consideration in your plan.

To determine whether the context value is derived, access the Edit Descriptive Flexfield task to view the list of configured context values for the flexfield. The Derivation Value field in the Context Segment region displays a list of available parameters. If context values have been preconfigured, see Oracle Applications Cloud Help for product-specific information about the use of those values.

Plan the Segments, Segment Properties, and Structure

Identify the user-defined attributes you need for a business object to determine the segments of the descriptive flexfield. Determine the segment properties such as the prompt, display type, or initial value.

The structure of the descriptive flexfield is determined by its global, context, and context-sensitive segments. Plan a global segment that captures an attribute for every instance of the business object. Plan a context for segments that depend on a condition of situation applying to a particular instance of the business object. Plan context-sensitive segments to capture attributes that are relevant in the context.

There is only one context segment available for descriptive flexfields. If you have more than one group of user-defined attributes where you could use the context segment, you will have to pick one group over the others, based on your company’s needs and priorities, and add the other user-defined attributes as global segments.
Plan Validation Rules
Define each segment’s validation rules and check if value sets exist for those rules or you must create new ones. If you must create a value set, you can create it either before configuring the flexfield or while creating or editing a segment.

When determining a segment’s validation rules, consider the following questions:

- What is the data type - character, date, date and time, or number?
- Does the segment require any validation beyond data type and maximum length?
- Should a character type value be restricted to digits, or are alphabetic characters allowed?
- Should alphabetic characters automatically be changed to uppercase?
- Should numeric values be zero-filled?
- How many digits can follow the radix separator of a numeric value? In base ten numeric systems, the radix separator is decimal point.
- Does the value need to fall within a range?
- Should the value be selected from a list of valid values? If so, consider the following questions:
  - Can you use an existing application table from which to obtain the list of valid values, or do you need to create a list?
  - If you are using an existing table, do you need to limit the list of values using a WHERE clause?
  - Does the list of valid values depend on the value in another flexfield segment?
  - Is the list of valid values a subset of another flexfield segment’s list of values?

Plan Initial Values
For every segment, list the constant value or SQL statement, if any, to use for the initial value of the user-defined attribute.

Plan How Segments Map to Oracle Business Intelligence Objects
You can extend descriptive flexfields into Oracle Transactional Business Intelligence (OTBI) for ad hoc reporting purposes. Determine the descriptive flexfield segments to be made available for reporting, and select the **BI Enabled** check box accordingly on the Manage Descriptive Flexfields page. You must run a process to extend the BI enabled segments into OTBI. For more information about extending the BI enabled segments into OTBI, see the Flexfields chapter in the Oracle Applications Cloud Creating and Administering Analytics and Reports guide.

Depending on the reporting needs, you may map similar context-sensitive attributes from different contexts to the same attribute in OTBI. For example, there may be a segment tracking the Product Color attribute in different contexts of a context sensitive descriptive flexfield. You can use segment labels to map these context-sensitive attributes together by defining a segment label and updating the BI Label list accordingly.

Related Topics
- Overview of Transactional Business Intelligence Configuration of Descriptive Flexfields

Considerations for Managing Descriptive Flexfields
Configuring **descriptive flexfields** involves managing the available flexfields registered with your Oracle Applications Cloud database and configuring their flexfield-level properties, defining and managing descriptive flexfield **contexts**, and configuring global and **context-sensitive segments**.
Every descriptive flexfield is registered to include a context segment, which you may choose to use or not.

In general, configuring descriptive flexfields involves:

1. Creating segment labels for business intelligence enabled flexfields.
2. Configuring global segments by providing identity information, the initial default value, and the display properties.
3. Configuring the context segment by specifying the prompt, whether the context segment should be displayed, and whether a value is required.
4. Configuring contexts by specifying a context code, description, and name for each context value, and adding its context-sensitive segments, each of which is configured to include identifying information, the column assignment, the initial default value, and the display properties.

The following aspects are important in understanding descriptive flexfield management:

- Segments
- Adding segments to highlighted descriptive flexfields
- Usages
- Parameters
- Delimiters
- Initial Values
- Business Intelligence

Segments

You can assign sequence order numbers to global segments and to context-sensitive segments in each context. Segment display is always in a fixed order. You can't enter a number for a segment if that number is already in use for a different segment.

Value sets are optional for context segments and follow specific guidelines:

- The value set that you specify for a context segment consists of a set of context codes.
- Each context code corresponds to a context that's appropriate for the descriptive flexfield.
- The value set must be independent or table-validated.
- If table-validated, the WHERE clause must not use the VALUESET.value_set_code or SEGMENT.segment_code bind variables.
- The value set must be of data type Character with the maximum length of values being stored no larger than the context's column length.
- If you don't specify a value set for a context segment, the valid values for that context segment are derived from the context codes. The definition of each context segment specifies the set of context-sensitive segments that can be presented when that context code is selected by the end user.
- For reasons of data integrity, you can't delete an existing context. Instead, you can disable the associated context value in its own value set by setting its end date to a date in the past.
- You can configure the individual global segments and context-sensitive segments in a descriptive flexfield. These segment types are differentiated by their usage, but they're configured on application pages that use most of the same properties.

Adding Segments to Highlighted Descriptive Flexfields

When you highlight flexfields on a run time page and use an Add Segment icon button to create a segment, the segment code, name, description, table column, and sequence number are set automatically. If you use an Add Segment icon button to configure descriptive flexfield segments, you can't use an existing value set. Value sets are
created automatically when you add the segments. You can enter the valid values, their descriptions, and the default value or specify the formatting constraints for the value set, such as minimum and maximum values.

Depending on display type, the value set you create using the **Add Segment** icon button is either an independent value set or a format-only value set. The following table shows which type of value set is created depending on the segment display component you select.

<table>
<thead>
<tr>
<th>Display Component</th>
<th>Value Set Created Using Add Segment</th>
</tr>
</thead>
<tbody>
<tr>
<td>Check Box</td>
<td>Independent</td>
</tr>
<tr>
<td>Drop-down List</td>
<td>Independent</td>
</tr>
<tr>
<td>List of Values</td>
<td>Independent</td>
</tr>
<tr>
<td>Radio Button Group</td>
<td>Independent</td>
</tr>
<tr>
<td>Text Field With Search</td>
<td>Independent</td>
</tr>
<tr>
<td>Text box</td>
<td>Format Only</td>
</tr>
<tr>
<td>Text area</td>
<td>Format Only</td>
</tr>
<tr>
<td>Date/Time</td>
<td>Format Only</td>
</tr>
</tbody>
</table>

**Tip:** After you add a context value, refresh the page to see the new value.

**Usages**

Descriptive flexfield usages allow for the same definition to be applied to multiple entities or application tables, such as a USER table and a USER_HISTORY table. Descriptive flexfield tables define the placeholder entity where the flexfield segment values are stored once you have configured the descriptive flexfield. When you configure a flexfield, the configuration applies to all its usages.

**Parameters**

Some descriptive flexfields provide parameters, which are attributes of the same or related entity objects. Parameters are public arguments to a descriptive flexfield. Parameters provide outside values in descriptive flexfield validation. You use parameters to set the initial value or derivation value of an attribute from external reference data, such as a column value or a session variable, rather than from user input. Parameters can be referenced by the logic that derives the default segment value, and by table-validated value set WHERE clauses.

**Delimiters**

A segment delimiter or separator visually separates segment values when the flexfield is displayed as a string of concatenated segments.
Initial Values
The SQL statement defining an initial value must be a valid statement that returns only one row and a value of the correct type.

You can use two types of SQL statements:

- SQL statement with no binding. For example, select MIN(SALARY) from EMPLOYEES.
- SQL statement with bind variables. You can use these bind variables in the WHERE clause of the SQL statement.

- \{SEGMENT.<segment_code>\}: Identifies a segment in the same context.
- \{PARAMETER.<parameter_code>\}: Identifies a parameter.
- \{CONTEXT.<context_code>;SEGMENT.<segment_code>\}: Identifies a segment in a different context. The context must be in the same category or in an ancestor category, and it can't be a multiple-row context.
- \{VALUESET.<value_set_code>\}: Identifies the closest prior segment in the same context that's assigned to the specified value set.
- \{FLEXFIELD.<internal_code>\}: Identifies a flexfield.

Business Intelligence
Selecting a global, context, or context-sensitive segment's BI Enabled check box specifies that the segment is available for use in Oracle Business Intelligence.

When the flexfield is imported into Oracle Business Intelligence, the label you selected from the BI Label drop-down list equalizes the segment with segments in other contexts, and maps the segment to the logical object represented by the label.

Considerations for Enabling Descriptive Flexfield Segments for Business Intelligence

A descriptive flexfield that is registered in the database as enabled for Oracle Business Intelligence (BI) includes a BI Enabled setting for each of its segments. When a global, context, or context-sensitive segment is BI-enabled, it is available for use in Oracle Business Intelligence.

The following aspects are important in understanding BI-enabled flexfield segments:

- Flattening business components to use BI-enabled segments in Oracle BI
- Equalizing segments to prevent duplication and complexity in the flattened component
- Mapping attributes of flattened business components to logical objects in Oracle BI
- Managing the labels that map segments to logical objects in Oracle BI

After you deploy a business intelligence-enabled flexfield, use the Import Oracle Fusion Data Extensions for Transactional Business Intelligence process to import the flexfield changes into the Oracle Business Intelligence repository. Users can make use of the newly-generated attributes in business intelligence applications. For example, a user can generate a report that includes attributes added by the descriptive flexfield. For additional information about logical objects and import, refer to the Oracle Transactional Business Intelligence Administrator’s Guide.
Flattening

When you deploy a business intelligence-enabled descriptive flexfield, the deployment process generates an additional set of flattened Application Development Framework (ADF) business components in addition to the usual ADF business components and ADF faces run time artifacts that are generated during deployment. The flattened business components include attributes for business intelligence-enabled segments only. Flattening means each user-defined column in each context shows up as an attribute in an Oracle Business Intelligence folder.

Flattened components include one attribute for the BI-enabled context-segment, and one attribute for each business intelligence-enabled global segment. For BI-enabled context-sensitive segments, consider the following:

- If you assigned a label to the segment, the flattened components include an additional single attribute representing segments with that label.
- If you didn’t assign a label, the flattened components include a discrete attribute for each BI-enabled context-sensitive segment in each context.

Mapping to Logical Objects in Business Intelligence

You can simplify reporting by representing similar segments as a single logical object in Business Intelligence.

If you assign a label to any set of context-sensitive segments that serve the same purpose in different contexts, you can consolidate or equalize the segments into a single attribute. This prevents duplication and the extra workload and complexity that result from the flattening process. For example, a United States context might have a Passport segment and a Canada context might have Visa segment. If you assign the NationalID segment label to both the Passport and Visa segments, they are equalized into the same NationalID attribute in the flattened business component.

Non-labeled context-sensitive segments aren’t equalized across context values, so the flattened components include a separate attribute for each context-sensitive segment for each context value. It may not be possible to equalize similarly labeled segments if they have incompatible data types or value set types.

Assign a label to a global segment, context segment, or context-sensitive segment to map the corresponding attribute in the flattened components to a logical object in Oracle Business Intelligence. Using labels to map segments to BI logical objects minimizes the steps for importing the flexfield into Oracle Business Intelligence.

Note: Assigning a label to a context-sensitive segment serves to equalize the attribute across contexts, as well as map the equalized attribute to business intelligence.

Managing Labels

You may assign a predefined label (if available) to segments or create new labels for assignment, as needed. Specify a code, name, and description to identify each label. In the BI Object Name field, enter the name of the logical object in Oracle Business Intelligence to which the segment label should map during import. Specifying the BI logical object minimizes the steps for importing the flexfield into Oracle Business Intelligence and helps to equalize context-sensitive segments across contexts.

If no labels are assigned to a BI-enabled segment, or the BI Object Name on the assigned label doesn’t exist in business intelligence, you must manually map the segment to the desired logical object when importing into Oracle Business Intelligence.

In addition, context-sensitive segments without labels cannot be equalized across context values. The flattened components include a separate attribute for each non-labeled context-sensitive segment in each context.

Importing to Oracle Business Intelligence Repository

After you deploy a business intelligence-enabled flexfield, import the flexfield changes into the Oracle Business Intelligence repository to make use of the newly flattened business components in business intelligence and then
propagate the flexfield object changes. When you import the metadata into the Oracle Business Intelligence repository, you must do so as the FUSION_APPS_BI_APPID user.

To import flexfield changes into the Oracle Business Intelligence repository in Oracle Cloud implementations, run the Import Oracle Fusion Data Extensions for Transactional Business Intelligence process. For additional information about import, refer to the Oracle Transactional Business Intelligence Administrator's Guide.

Note: When you import a flexfield into the Oracle Business Intelligence repository, you see both <name> and <name>_c attributes for each segment, along with some other optional attributes. The <name> attribute contains the value. The <name>_c attribute contains the code of the value set that the value comes from, and is used for linking to the value dimension. You must import both attributes.

Extensible Flexfields

Overview of Extensible Flexfields

Extensible flexfields are like descriptive flexfields, with some additional features.

- You can add as many context-sensitive segments to the flexfield as you need. You aren't restricted by the number of columns predefined and registered for the flexfield.
- You can configure a one-to-many relationship between the entity and its extended attribute rows.
  - A row of data can have multiple contexts associated with it.
  - A row of data can have multiple occurrences of the same context.
- You can configure attributes in groups to form a context so that the attributes in the context always appear together in the user interface.
- You can use existing hierarchical categories so that entities inherit the contexts that are configured for their parents. Contexts are reusable throughout categories.
- Application development has registered some extensible flexfields to support view and edit privileges. For such flexfields, you can specify view and edit privileges at the context level to control who sees the attributes and who can change the attributes' values.

When you configure a context for multiple rows per entity, the segments are displayed as a table.

Unlike descriptive flexfields, the extension columns corresponding to extensible flexfields segments are part of extension tables, separate from the base application table. Unlike descriptive flexfield contexts, the set of attributes in an extensible flexfield context remains constant and doesn't differ by context value. An extensible flexfield describes an application entity, with the run time ability to expand the database that implementation consultants can use to define the data structure that appears in the application. Extensible flexfields support one-to-many relationships between the entity and the extended attribute rows. To get a list of predefined extensible flexfields, use the Manage Extensible Flexfields task in the Setup and Maintenance work area.

The following aspects are important in understanding extensible flexfields:

- Usages
- Categories
- Pages
- Security
- Protected Extensible Flexfield Data

**Usages**

Similar to the descriptive flexfields, you can define multiple usages for an extensible flexfield, which enables several application tables to share the same flexfield.

For example, a flexfield for shipping options can be used by both a Supplier table and a Buyer table. In addition, you can associate a context with one, some, or all of the flexfield's usages. Thus, with the shipping information example, you can associate a warehouse context with the Supplier usage, a delivery location context with the Buyer usage, and a ship-via context with all usages.

Usages include security information for applying no security to user access or enforcing view and edit privileges. Some product-specific extensible flexfields have specialized usage fields beyond those for security.

**Categories**

You can configure multiple extensible flexfield contexts and group the contexts into categories. All extensible flexfields have at least one category. For some extensible flexfields, you can configure a hierarchy of categories. A child category in the hierarchy can inherit contexts from its parent category.

You can define categories for extensible flexfields, and you can associate any combination of contexts with a given category.

For example, the Electronics and Computers category hierarchy might include a Home Entertainment category, which in turn might include an Audio category and a TV category, and so on. The Home Entertainment product might have contexts that specify voltage, dimensions, inputs and outputs. Contexts are reusable within a given extensible flexfield. For example, the dimensions context could be assigned to any category that needs to include dimensional information.

**Pages**

Extensible flexfields let you combine contexts into groups known as pages, which serve to connect the contexts so they will always be presented together in the application user interface.

Each application page corresponds to one extensible flexfield category, with a separate region of the page for each associated context.

**Security**

When you configure a flexfield, you set the privileges for a context at the usage level by selecting actions for the view and edit privileges of a context usage.

When an end user performs a search, the user interface displays only the attribute values of the contexts for which the user has view privileges. The user can perform a search using all attributes for all contexts, regardless of view privileges.

If end users access a context through a web service, an exception is thrown if they perform an action for which they don't have privileges.

All extensible flexfields have a base data security resource. Some data security resources for extensible flexfields are preconfigured with actions that you can use to specify access privileges. If no action is preconfigured, a security administrator can create actions and policies to support access control on the extensible flexfield attributes.

Some extensible flexfields have a translatable option; these flexfields also have a translation data security resource.
Protected Extensible Flexfield Data

Application developers may mark some data configurations in an extensible flexfield as protected, indicating that you can't edit them.

If an extensible flexfield is partially protected, then you can't edit the protected portions of the flexfield's configuration. For example:

- If an extensible flexfield context is protected, you can't edit its:
  - Context details
  - Context segments
  - Context usages

- If an extensible flexfield page is protected, you can't:
  - Edit the page details or delete the page
  - Edit the contexts associated with the page

**Note:**

- There is no restriction on page references to protected contexts. The pages you create may contain any context, whether protected or not.
- There is a restriction on category references to protected contexts. If a context is protected, you can't add it to or delete it from any category.

Related Topics

- Update Existing Setup Data

Considerations for Planning Extensible Flexfields

Once you have identified a flexfield, plan its configuration in advance. Compile a list of the UI pages and other artifacts in your deployment that are affected by the configuration. Verify that you are provisioned with the roles required to view and configure the flexfield. View the flexfield using the Highlight Flexfields option in the Administration menu while viewing the run time page where the flexfield appears. Plan how to deploy the flexfield for test and production users. Review the tools and tasks available for managing flexfields for adding and editing flexfield segments.

Planning an extensible flexfield involves:

1. Identifying the following:
   - A hierarchical structure of categories
   - Existing context values
   - User-defined attributes, the relevant extensible flexfield segments, segment properties, and the structure

2. Planning the following:
   - Validation rules
   - Initial values
   - Security
   - Attribute mapping to Oracle Business Intelligence objects.
Category Hierarchy Structure
Existing category hierarchy structures provide the framework for planning what segments to add to an extensible flexfield as user-defined attributes of an entity. Some applications provide user interfaces to create and manage an extensible flexfield's category hierarchy.

Contexts and Existing Context Values
If related attributes can be grouped together, plan adding the attributes as a context of segments, and plan the order in which the attributes should appear. Some extensible flexfields have preconfigured context values. Region headers displayed in the user interface page or pages that contain the flexfield segments identify existing contexts. Using the Manage Extensible Flexfields task, find and open the flexfield for editing to view the list of configured context values.

See product-specific information for guidance in using preconfigured context values.

Plan the Segments and Segment Properties
List all the user-defined attributes that you want to add as extensible flexfield segments. For each segment, define properties, including the indexed property.

Plan Validation Rules
Define each segment's validation rules and check if value sets exist for those rules or you must create. If you must create a value set, you can create it either before you configure the flexfield or at the same time that you create or edit a segment.

When determining a segment's validation rules, consider the following questions:

- What is the data type: character, date, date and time, or number?
- Does the segment require any validation beyond data type and maximum length?
- Should a character type value be restricted to digits, or are alphabetic characters permitted?
- Should alphabetic characters automatically be changed to uppercase?
- Should numeric values be zero-filled?
- How many digits can follow the radix separator of a numeric value? In base ten numeric systems, the radix separator is a decimal point.
- Should the value be within a range?
- Should the value be selected from a list of valid values? If yes, consider the following questions:
  - Can you use an existing application table from which to obtain the list of valid values, or do you have to create a list?
  - If you are using an existing table, do you have to limit the list of values using a WHERE clause?
  - Does the list of valid values depend on the value in another flexfield segment?
  - Is the list of valid values a subset of another flexfield segment's list of values?

Plan Initial Values
For every segment, list the constant value or SQL statement, if any, to use for the initial value of the user-defined attribute.
Plan Security

Determine what privileges to set for view and edit access to context attributes, such as providing all users with view access but only managers with edit access.

If your security restrictions apply to several contexts, you can create generic actions. At a minimum, create the generic actions for the base data security resource. If the flexfield has a translatable option and you plan to use translatable contexts, then also create the generic actions for the translation data security resource. For example, the Item flexfield supports the translatable option and has a data security resource ITEM_EFF_VL in addition to the base data security resource ITEM_EFF_B. Then, create actions for both data security resources, such as EDIT_NONTRANS_ATTRS for ITEM_EFF_B and EDIT_TRANS_ATTRS for ITEM_EFF_VL.

If your security restrictions are more fine-grained, such as having to secure each context with a different privilege, then you can create more fine-grained actions.

Plan Which Segments Map to Oracle Business Intelligence Objects

If an extensible flexfield has been enabled for Oracle Business Intelligence, you can make the attributes available for use in Oracle Business Intelligence Applications.

Considerations for Managing Extensible Flexfields

Configuring extensible flexfields involves managing the available flexfields registered with your application database.

The following sequence describes how to configure extensible flexfields:

1. Configuring contexts by creating each context segment and the context-sensitive segments for each context segment, and providing the following for each segments:
   a. Identifying information
   b. Column assignment
   c. Initial default value
   d. Display properties
2. Configuring context usages and usage security by selecting actions to which users should have access:
   o View
   o Edit
   o None, if no special privileges should be enforced.
3. Configuring categories and category details.
4. Associating contexts with a category.
5. Creating logical pages for a category.

The following aspects are important in understanding extensible flexfield management:

- Contexts and pages
- Categories
- Initial values
- Adding segments to highlighted extensible flexfields
- Indexed segments
- Security
- Deployment
Contexts and Pages
Each context is displayed to end users as a region containing its context-sensitive segments. You can specify instruction help text to display instructions that explain how to use the region and its attributes to end users. Instruction help text is displayed at the beginning of the context region. A context can be defined as single row or multi-row. Single row contexts are the same as descriptive flexfields contexts. A single row context has only one set of context-sensitive segments. A multi-row context enables you to associate multiple sets of values with the same object instance.

For example, for a BOOK table, you could create a multi-row context named chapters that contains a segment for chapter and a segment for number of pages. Multiple chapters can then be associated with each book in the BOOK table.

For contexts that store multiple rows, you can uniquely identify each row by having the values in each row form a unique key.

If flexfield has a category hierarchy, then you can leverage the hierarchy to reuse contexts for similar entities, such as similar items in a product catalog.

Set the context to translatable so that free-form text entered by end users is stored in the language of the user's locale, and different translations of that text can be stored in other languages. Segments in the translated contexts should use format-only value sets for storing free-form, user-entered text.

Set the context security to give an end user view or edit access to a context. The context's task flow and region appear in the user interface only for users with view access. With edit access, an end user can edit the context's attribute values. With no action specified for a usage, no special privileges are enforced through the context's configuration.

Define logical pages to group contexts together in the user interface. For a given category, you may create one or more logical pages. You may add one or more of the category's associated contexts to each of the category's logical pages.

You can specify:

- The sequence of the contexts within each page.
- The sequence in which the logical pages appear.
- Instruction help text to display instructions that explain how to use the page to end users. Instruction help text is displayed at the beginning of the logical page, preceding all of its context regions.

Categories
A category is a grouping of related data items that can be considered to belong together. You can associate any combination of contexts with a given category. Extensible flexfields with more than 30 categories must be deployed as a background process.

A category hierarchy logically organizes a set of categories. For example, the Electronics and Computers category hierarchy might include a Computer category and a Home Entertainment category, which in turn might include an Audio category and a TV category, and so on.

A category can be a child or sibling of an existing category. The hierarchy can be as simple or as complex as desired, with any combination of zero or more sibling categories and zero or more child categories. If no category is defined, the data items are grouped in a single predefined default category.

Each category has associated contexts that store relevant information about a data item in that category. For example, a Home Entertainment product has contexts that specify Voltage, Dimensions, Inputs and Outputs. Contexts are reusable within a given extensible flexfield. Then, the Dimensions context could be assigned to any category that needs to include dimensional information.
If a hierarchy includes child categories, each child category inherits the contexts from its parent category; for example, the Home Entertainment category inherits Voltage and Dimensions from the Electronics and Computers category.

Each extensible flexfield is associated with a particular category hierarchy. Consider category hierarchies to be defining framework for extensible flexfields and their contexts. A category hierarchy specifies which contexts are valid for each category.

An extensible flexfield can include multiple contexts which you define to support a given category. These contexts can be suitable for various purposes, but within a particular category, some contexts might be considered to be related to, or dependent on, each other. You can combine these contexts into groups known as logical pages, and determine the sequence in which the pages appear. This serves to connect the contexts so they will always be presented together and in a particular order in the application user interface.

For example, the Home Entertainment category might have an Electrical Specifications page that contains the Voltage, Inputs and Outputs contexts, and a Physical Specifications page that contains the Dimensions and Form Factor contexts.

Initial Values
The SQL statement defining an initial value must be a valid statement that returns only one row and a value of the correct type.

You can use two types of SQL statements:

- SQL statement with no binding. For example, select MIN(SALARY) from EMPLOYEES.
- SQL statement with bind variables. You can use these bind variables in the WHERE clause of the SQL statement.
  
  o :{SEGMENT.<segment_code>}: Identifies a segment in the same context.
  
  o :{PARAMETER.<parameter_code>}: Identifies a parameter.
  
  o :{CONTEXT.<context_code>;SEGMENT.<segment_code>}: Identifies a segment in a different context. The context must be in the same category or in an ancestor category, and it can't be a multiple-row context.
  
  o :{VALUESET.<value_set_code>}: Identifies the closest prior segment in the same context that's assigned to the specified value set.
  
  o :{FLEXFIELD.<internal_code>}: Identifies a flexfield.

Adding Segments to Highlighted Extensible Flexfields
When you highlight flexfields on a run time page and use an Add Segment icon button to create a segment, the segment code, name, description, table column, and sequence number are set automatically. If you use an Add Segment icon button to configure extensible flexfield segments, you can't use an existing value set. Value sets are created automatically when you add segments. You can enter the valid values, their descriptions, and the default value or specify the formatting constraints for the value set, such as minimum and maximum values.

Depending on display type, the value set you create with the Add Segment icon button is either an independent value set or a format-only value set. The following table shows which type of value set is created depending on the segment display component you select.

<table>
<thead>
<tr>
<th>Display Component</th>
<th>Value Set Created Using Add Segment</th>
</tr>
</thead>
<tbody>
<tr>
<td>Check Box</td>
<td>Independent</td>
</tr>
<tr>
<td>Display Component</td>
<td>Value Set Created Using Add Segment</td>
</tr>
<tr>
<td>------------------------</td>
<td>-------------------------------------</td>
</tr>
<tr>
<td>Drop-down List</td>
<td>Independent</td>
</tr>
<tr>
<td>List of Values</td>
<td>Independent</td>
</tr>
<tr>
<td>Radio Button Group</td>
<td>Independent</td>
</tr>
<tr>
<td>Text Field With Search</td>
<td>Independent</td>
</tr>
<tr>
<td>Text box</td>
<td>Format Only</td>
</tr>
<tr>
<td>Text area</td>
<td>Format Only</td>
</tr>
<tr>
<td>Rich Text Editor</td>
<td>Format Only</td>
</tr>
<tr>
<td>Date/Time</td>
<td>Format Only</td>
</tr>
</tbody>
</table>

**Tip:** After you add a context value, refresh the page to see the new value.

### Indexed Segments

You can designate an extensible flexfield segment as indexed so that it's one of the selectively required attributes a user can use in an attribute search. If you indicate in the Manage Extensible Flexfield UI page that a segment should be indexed, the column representing the segment must be added to the database index. Commonly, a database administrator (DBA) adds columns to the database index.

When an extensible flexfield with indexed segments is deployed, search task flows are generated along with the other flexfield artifacts and specify the indexed attributes as selectively required. In the deployed extensible flexfield's search task flow, an end user must specify at least one of the indexed attributes in the search criteria. This prevents non-selective searches, which could cause performance issues.

For example, if you index the memory and processor attributes and ensure that the corresponding columns in the database are indexed, a user can search an item catalog for computers by entering processor or memory or both as a search criteria. No search is performed if an end user enters an attribute that isn't indexed as a search criterion.

### Security

An extensible flexfield's base data security resource typically has a name with an _B suffix. The translation data security resource is a view of a translation table that typically has a name with an _VL suffix.

If a flexfield supports the translatable option and has a translation data security resource, make sure that you create the action for the appropriate data security resource.

- If you create a context-specific action for a nontranslatable context, add it to the base data security resource.
- If you create a context-specific action for a translatable context, add it to the translation data security resource.
Deployment

You can only deploy extensible flexfields using the Manage Extensible Flexfields task. You can deploy extensible flexfields offline as a background process and continue working in the session without having to wait for the deployment to complete. You can queue up several extensible flexfields and deploy as a background process. The flexfields are deployed, one at a time, in the order that you deploy them to the queue. You must deploy extensible flexfields with more than 30 categories as a background process.

You can remove an extensible flexfield from the deployment queue with the Cancel Background Deployment command. When an extensible flexfield is deployed in a background process, its offline status indicates that the flexfield is in a background deployment process. A flexfield's offline status is cleared and it's deployment status updated when the background deployment process has completed.

Note: The Offline Status column refreshes when you perform a new search in the Manage Extensible Flexfields task.

Considerations for Enabling Extensible Flexfield Segments for Business Intelligence

An extensible flexfield that is registered in the database as enabled for Oracle Business Intelligence (BI) includes a BI Enabled setting for each of its segment instances. When a segment instance is BI-enabled, it's available for use in Oracle Business Intelligence.

The following aspects are important in understanding BI-enabled extensible flexfield segments.

- Flattening business components to use BI-enabled segments in Oracle BI
- Mapping attributes of flattened business components to logical objects in Oracle BI

After you deploy a business intelligence-enabled flexfield, use the Import Oracle Fusion Data Extensions for Transactional Business Intelligence process to import the flexfield changes into the Oracle Business Intelligence repository. Users can make use of the newly-generated attributes in business intelligence applications. For additional information about logical objects and import, refer to the Oracle Transactional Business Intelligence Administrator's Guide.

Flattening

When you deploy a business intelligence-enabled extensible flexfield, the deployment process generates an additional set of flattened business components for use in business intelligence. The flattened business components include attributes for business intelligence-enabled segment instances only.

If you assigned a label to a segment, the flattened components include a single attribute representing all segment instances with that label. If you didn't assign a label, the flattened components include a discrete attribute for each BI-enabled segment instance in each structure.

Importing to Oracle Business Intelligence Repository

After you deploy a business intelligence-enabled flexfield, import the flexfield changes into the Oracle Business Intelligence repository to make use of the newly flattened business components in business intelligence and then propagate the flexfield object changes. When you import the metadata into the Oracle Business Intelligence repository, you must do so as the FUSION_APPS_BI_APPID user. To import flexfield changes into the Oracle Business Intelligence repository in Oracle Cloud implementations, run the Import Oracle Fusion Data Extensions for Transactional Business Intelligence
Intelligence process. For additional information about import, refer to the Oracle Transactional Business Intelligence Administrator's Guide.

**Tip:** When you import a flexfield into the Oracle Business Intelligence repository, you see both `<name>_` and `<name>_c` attributes for each segment, along with some other optional attributes. The `<name>_` attribute contains the value. The `<name>_c` attribute contains the code of the value set that the value comes from, and is used for linking to the value dimension. You must import both attributes.

### Considerations for Managing Extensible Flexfield Categories

Categories are a way of extending the number of context-sensitive segments for a flexfield beyond the columns reserved for flexfield segments.

For example, an Items extensible flexfield has a category for each item and each category can have one or more contexts. The laptop item belongs to the Computers category. Since extensible flexfields are mapped to separate extension tables, not just to columns as with descriptive flexfields, the thirty reserved columns on the extensible flexfield table let you define up to thirty context-sensitive segments for each context.

If you add a Dimensions context to the Computers category, thirty segments are available. But if you need to add more than thirty attributes, create another context and associate it to the same category. You could now add an Electronics Attributes context to the same Computers category in which you create another thirty segments. You can continue creating more contexts and adding them to the Computers category. In this way your laptop computer item can be extended with as many attributes as you need, because it is mapped to a category and you can keep adding contexts to that category.

A descriptive flexfield on an items table with thirty columns reserved for segments can only have a single context. Once you configure the columns for that one context, you cannot create any more segments.

### Predefined and Preconfigured Categories

How you structure the flexfield configuration depends on how categories are defined for the flexfield. If the extensible flexfield is preconfigured with one category, associate all your contexts and pages with that category. If a product-specific extensible flexfield is preconfigured with several categories, associate your contexts and pages with those categories. If the extensible flexfields provide user interfaces for configuring multiple categories, associate a context with more than one category using inheritance.

Some products provide an activity or task for creating and maintaining categories for an extensible flexfield. See product-specific information to determine if you can create categories for the flexfield.

You can view a flexfield's category hierarchies by using either the Highlight Flexfields feature or the Manage Extensible Flexfields task to find and open the flexfield for editing.

### Disabling Categories

While configuring an extensible flexfield, you can disable a category. The Enabled column in the Category table of the Edit Extensible Flexfield page, indicates which categories are enabled.

**Note:** When you deploy an extensible flexfield that has a disabled category, that category and its descendant categories aren't deployed. Contexts and their segments are deployed only if they belong to at least one enabled category.

### Contexts

Group similar attributes into contexts. The group is displayed together in a region. The region's header is the context value.
If a category hierarchy exists for the flexfield, then you can leverage the hierarchy to reuse contexts for similar entities, such as similar items in a product catalog.

The following figure shows the Item Extended Attributes flexfield, which uses the category hierarchy feature to reuse contexts. The flexfield’s Electronics and Computers category contains contexts for compliance and certification, voltage, and materials and substances. The TV and Video subcategory and the Computer Products subcategory inherit the Electronics and Computer contexts in addition to having their own contexts. The Materials and Substances context belongs to both the Electronics and Computer Products category and the Tools, Auto, and Industrial Products category.

The following table shows an example of category hierarchy for an extensible flexfield. To store voltage information for all electronic and computer items, associate a Voltage context with the Electronics and Computers category. Both the TV and Video subcategory and the Computers subcategory then inherit the Voltage context from the parent Electronics and Computers category.

<table>
<thead>
<tr>
<th>Display Name</th>
<th>Code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Electronics and Computers</td>
<td>PROD_ELECTRONICS</td>
<td>Electronics and Computers</td>
</tr>
<tr>
<td>• TV and Video</td>
<td>PROD_TV_VIDEO</td>
<td>Television and Video</td>
</tr>
<tr>
<td>• Computers</td>
<td>PROD_COMPUTERS</td>
<td>Computers</td>
</tr>
<tr>
<td>Office Products and Supplies</td>
<td>PROD_OFFICE_PRODUCTS_SUPPLIES</td>
<td>Office Products and Supplies</td>
</tr>
<tr>
<td>Tools, Auto, and Industrial</td>
<td>PROD_TOOLS_AUTO_INDUSTRIAL</td>
<td>Tools, Automotive, and Industrial</td>
</tr>
</tbody>
</table>
### Example of Configuring an Item Extended Attributes Flexfield

The Item Extended Attributes flexfield provides segments for extending the Item business object. In the Manage Extensible Flexfields task, configure your product business object to include a technical specifications logical page on the user interface for the Electronics and Computers category of items.

In this example, your configuration of this flexfield groups attributes into the following contexts:

- Materials and Substances
- Compliance and Certification
- Voltage

### Scenario

The following list shows an example plan for computer attributes for the Item Extended Attributes flexfield. In this example, the Electronics Information page is inherited from the parent Electronics and Computers category.

- **Page: Electronics Information**
  - **Context: Compliance and Certification, single row**
    - ISO 14001 (International Organization for Standardization for an Environmental Management System)
    - ENERGY STAR (energy efficiency guidelines)
    - ROHS (Restriction of the use of certain hazardous substances in electrical and electronic equipment)
  - **Context: Voltage, single row**
    - Minimum voltage
    - Maximum voltage
    - Current type
  - **Context: Materials and Substances, multiple rows**
    - Material
    - Contain recyclate
    - Percent unit mass

- **Page: Computer Information**
  - **Context: Processor Specifications, single row**
    - Manufacturer
    - CPU type
    - Processor interface
    - Processor class

---

<table>
<thead>
<tr>
<th>Display Name</th>
<th>Code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sports and Outdoors</td>
<td>PROD_SPORTS_OUTDOORS</td>
<td>Sports and Outdoors</td>
</tr>
</tbody>
</table>

---

**NOTE:**

- Display Name: The field name as it appears in the user interface.
- Code: The unique identifier for the flexfield.
- Description: The description of the flexfield's purpose.
The following table summarizes key decisions for this scenario.

<table>
<thead>
<tr>
<th>Decisions to Consider</th>
<th>In This Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>Which extensible flexfield is available for configuring a hierarchy of categories?</td>
<td>Item Extended Attributes flexfield</td>
</tr>
</tbody>
</table>

Collecting Technical Specifications

Your product inventory pages for electronics and computers require a technical specifications page. Your product inventory pages for furniture require a furniture specifications page and an assembly instructions page. Items in both the electronics and computer category, and in the furniture category, share attributes for specifying materials and substances.

The following figure shows a technical specifications logical page in the user interface for the Electronics and Computers category. It contains attributes in the context of Recovery and Recycling, Compliance and Certification, Operating Conditions, and Materials and Substances. The Materials and Substances context is configured for multiple rows. Your users can select all the materials and substances required to make a single product.

Analysis

Use logical pages to determine how the contexts appear on the user interface. Use a context to store all the materials and substances required to make a single product. You can configure a context to store multiple rows per entity. The multiple rows are displayed in a table, like the Materials and Substances context.
The Technical Specifications logical page contains the attributes for the four contexts.

- Recovery and Recycling
- Compliance and Certification
- Operating Conditions
- Materials and Substances

The following figure is an example, where the Furniture category is configured to include a Furniture Specifications logical page and an Assembly Instructions logical page. The two categories (Electronics and Computers and Furniture) share the Materials and Substances context.

Configure Security for the Item Flexfield Configuration

The following table shows an example of data security policies for the Item flexfield.

<table>
<thead>
<tr>
<th>Data Security Resource</th>
<th>Policy</th>
<th>Role</th>
<th>Action</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>ITEM_EFF_B</td>
<td>A</td>
<td>VOLTAGE_SPEC</td>
<td>edit_nontrans_voltage_ctx</td>
<td>All values</td>
</tr>
<tr>
<td>ITEM_EFF_VL</td>
<td>B</td>
<td>COMPLIANCE_SPEC</td>
<td>edit_trans_compliance_ctx</td>
<td>All values</td>
</tr>
<tr>
<td>ITEM_EFF_VL</td>
<td>C</td>
<td>COMPUTER_SPEC</td>
<td>edit_trans_attrs</td>
<td>ComputerCategoryFilter</td>
</tr>
</tbody>
</table>
The following table shows the privileges for the three flexfield contexts.

<table>
<thead>
<tr>
<th>Context</th>
<th>Edit Privilege</th>
<th>View Privilege</th>
</tr>
</thead>
<tbody>
<tr>
<td>Voltage</td>
<td>edit_nontrans_voltage_ctx</td>
<td>NONE</td>
</tr>
<tr>
<td>Compliance and Certification</td>
<td>edit_trans_compliance_ctx</td>
<td>NONE</td>
</tr>
<tr>
<td>Materials and Substances</td>
<td>edit_trans_attrs</td>
<td>NONE</td>
</tr>
</tbody>
</table>

In this example, anyone can view the contexts' attributes, but the edit privileges are restricted as follows:

- Voltage: Only voltage specialists can edit this value.
- Compliance and Certification: Only compliance specialists can edit this value.
- Materials and Substances: Only computer specialists can edit these attributes for items in the computer category. Only television specialists can edit these attributes for items in the TV category.

To sum up, in this entire example, the Materials and Substances context is secured by a generic action with a condition applied to restrict access by category. Voltage and Compliance and Certification are secured by actions specific to each context.

### File Format for Importing Extensible Flexfields

To import extensible flexfields into an application, you create separate text files for the value sets, contexts, and context segments. Then, you upload them to the Oracle WebCenter Content document repository. Each file must follow a specific format, as described here. After the files are placed in the document repository, you can import the values sets, contexts, and context segments into the application.

While creating the file, adhere to the following guidelines:

- Use a vertical bar or pipe ( | ) as a delimiter between fields for both header and value rows.
- Set the file encoding to UTF-8 without the Byte Order Mark (BOM) as per the Oracle WebCenter Content specification.

The following sections contain specific details about each file format.

**Prerequisite**

You must have worked with flexfields in Oracle Cloud applications.
## Value Sets

To create a file containing value sets, include the headers listed in the following table:

<table>
<thead>
<tr>
<th>Header</th>
<th>Data Type</th>
<th>Data Size</th>
<th>Description</th>
<th>Required or Optional</th>
</tr>
</thead>
<tbody>
<tr>
<td>ValueSetCode</td>
<td>String</td>
<td>60</td>
<td>This value identifies your value set across components.</td>
<td>Required</td>
</tr>
<tr>
<td>ModuleType</td>
<td>String</td>
<td>60</td>
<td>This value determines the module type of your value set.</td>
<td>Required</td>
</tr>
<tr>
<td>ModuleKey</td>
<td>String</td>
<td>60</td>
<td>This value determines the specific module of your value set.</td>
<td>Required</td>
</tr>
<tr>
<td>ValidationType</td>
<td>String</td>
<td>30</td>
<td>This value determines your value set type. The values you can use are DEP, INDEP, FORMAT, SUBSET, and RELATED.</td>
<td>Required</td>
</tr>
<tr>
<td>ValueDataType</td>
<td>String</td>
<td>30</td>
<td>This value determines the data type that your value set uses. Supported data types are VARCHAR2, NUMBER, DATE, and TIMESTAMP.</td>
<td>Required</td>
</tr>
<tr>
<td>ValueSubType</td>
<td>String</td>
<td>30</td>
<td>This value determines the data subtype your value set uses. Supported data subtypes are TEXT, TIME_HM, NUMERIC, TIME_HMS, and TRANSLATED</td>
<td>Required for ValueDataType VARCHAR2</td>
</tr>
<tr>
<td>MaximumLength</td>
<td>Integer</td>
<td>-</td>
<td>This value determines the maximum length of values in your value set.</td>
<td>Required for ValueDataType VARCHAR2</td>
</tr>
<tr>
<td>Description</td>
<td>String</td>
<td>240</td>
<td>This value gives your value set a description.</td>
<td>Optional</td>
</tr>
<tr>
<td>Precision</td>
<td>Integer</td>
<td>-</td>
<td>This value determines the number of digits in a value.</td>
<td>Optional</td>
</tr>
<tr>
<td>Header</td>
<td>Data Type</td>
<td>Data Size</td>
<td>Description</td>
<td>Required or Optional</td>
</tr>
<tr>
<td>-------------------------</td>
<td>-----------</td>
<td>-----------</td>
<td>------------------------------------------------------------------------------</td>
<td>----------------------</td>
</tr>
<tr>
<td>Header</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Scale</td>
<td>Integer</td>
<td>-</td>
<td>This value determines the number of digits to the right of the decimal point in the number data you add in your value set.</td>
<td>Optional</td>
</tr>
<tr>
<td>UppercaseOnlyFlag</td>
<td>String</td>
<td>1</td>
<td>If this value is set to Y, then your value set will accept only uppercase characters. If it’s N, then both uppercase and lowercase characters are supported.</td>
<td>Optional</td>
</tr>
<tr>
<td>ZeroFillFlag</td>
<td>String</td>
<td>1</td>
<td>If this value is set to Y, then zeroes are added to the left of the text you add in your value set, till the text length matches your MaximumLength. If the value is N, no zeroes are added.</td>
<td>Optional</td>
</tr>
<tr>
<td>SecurityEnabledFlag</td>
<td>String</td>
<td>1</td>
<td>If this value is set to Y, you can add a data security resource to your value set. If the value is N, no data security resource is added.</td>
<td>Optional</td>
</tr>
<tr>
<td>DataSecurityObjectName</td>
<td>String</td>
<td>60</td>
<td>This value selects the data security resource you want to add to your value set. This value must match a data security resource in the application.</td>
<td>Required if SecurityEnabledFlag is set to Y.</td>
</tr>
<tr>
<td>MinimumValue</td>
<td>String</td>
<td>150</td>
<td>This value specifies the minimum value for the values in your value set.</td>
<td>Optional</td>
</tr>
<tr>
<td>MaximumValue</td>
<td>String</td>
<td>150</td>
<td>This value specifies the maximum value for the values in your value set.</td>
<td>Optional</td>
</tr>
</tbody>
</table>
Here's a sample file that contains the header values at the beginning of the file, followed by line entries of the three value sets that are to be imported. For importing several value sets, add more line entries in a similar format.

```
ValueSetCode|ModuleType|ModuleKey|ValidationType|ValueDataType|ValueSubtype|MaximumLength|Description
VS_TEST_91|APPLICATION|FND|INDEP|VARCHAR2|TEXT|2|desc1
VS_TEST_92|APPLICATION|FND|INDEP|VARCHAR2|TEXT|3|
VS_TEST_93|APPLICATION|FND|INDEP|VARCHAR2|TEXT|3|desc3
```

**Context**

To create a file containing the contexts, include the headers in the following table:

<table>
<thead>
<tr>
<th>Header</th>
<th>Data Type</th>
<th>Data Size</th>
<th>Description</th>
<th>Required or Optional</th>
</tr>
</thead>
<tbody>
<tr>
<td>ApplicationId</td>
<td>Long</td>
<td>-</td>
<td>This value determines the Application to which your flexfield and context belongs to.</td>
<td>Required</td>
</tr>
<tr>
<td>EffCode</td>
<td>String</td>
<td>40</td>
<td>This value is the code of the extensible flexfield to which you're adding your context. This value should match the code of an extensible flexfield that's already in the application.</td>
<td>Required</td>
</tr>
<tr>
<td>ContextCode</td>
<td>String</td>
<td>80</td>
<td>This value is the code for your context.</td>
<td>Required</td>
</tr>
<tr>
<td>Name</td>
<td>String</td>
<td>80</td>
<td>This value determines the display name of your context.</td>
<td>Required</td>
</tr>
<tr>
<td>MultirowFlag</td>
<td>String</td>
<td>1</td>
<td>If this value is set to Y, then your context will support multi-row</td>
<td>Required</td>
</tr>
<tr>
<td>Header</td>
<td>Data Type</td>
<td>Data Size</td>
<td>Description</td>
<td>Required or Optional</td>
</tr>
<tr>
<td>-----------------------</td>
<td>------------</td>
<td>-----------</td>
<td>-----------------------------------------------------------------------------</td>
<td>----------------------</td>
</tr>
<tr>
<td>EnabledFlag</td>
<td>String</td>
<td>1</td>
<td>If this value is set to Y, then your context is enabled. If the value is N, then your context is disabled.</td>
<td>Required</td>
</tr>
<tr>
<td>Description</td>
<td>String</td>
<td>240</td>
<td>This value gives your context a description.</td>
<td>Optional</td>
</tr>
<tr>
<td>TranslatableFlag</td>
<td>String</td>
<td>1</td>
<td>If this value is set to Y, then the segments in your context are translatable. If the value is N, they're not.</td>
<td>Optional</td>
</tr>
<tr>
<td>ContextIdentifier</td>
<td>String</td>
<td>30</td>
<td>This value determines the API name for your context. There are a set of conventions to be followed when naming APIs.</td>
<td>Optional</td>
</tr>
<tr>
<td>InstructionHelpText</td>
<td>String</td>
<td>400</td>
<td>This text gives the user instructions about how to use the context region.</td>
<td>Optional</td>
</tr>
<tr>
<td>FlexfieldUsageCode</td>
<td>String</td>
<td>30</td>
<td>This value determines the usage code for your context. The value should match a usage code in the application.</td>
<td>Required</td>
</tr>
<tr>
<td>ViewPrivilegeName</td>
<td>String</td>
<td>400</td>
<td>This value determines which privileges can view this context usage. The value should match a privilege in the application.</td>
<td>Optional</td>
</tr>
<tr>
<td>EditPrivilegeName</td>
<td>String</td>
<td>400</td>
<td>This value determines which privileges can edit this context usage. The value should match a privilege in the application.</td>
<td>Optional</td>
</tr>
</tbody>
</table>
### Header Data Type Data Size Description Required or Optional

<table>
<thead>
<tr>
<th>Header</th>
<th>Data Type</th>
<th>Data Size</th>
<th>Description</th>
<th>Required or Optional</th>
</tr>
</thead>
<tbody>
<tr>
<td>CExtAttribute1</td>
<td>String</td>
<td>150</td>
<td>This header adds a user defined attribute to a context usage. You can use up to 5 headers of this type, CExtAttribute1 to CExtAttribute5 to add your user defined attributes.</td>
<td>Optional</td>
</tr>
<tr>
<td>CONTEXT_ CExtAttribute1</td>
<td>String</td>
<td>150</td>
<td>This header adds a user defined attribute to a context. You can use up to 5 headers of this type, CONTEXT_ CExtAttribute1 to CONTEXT_ CExtAttribute5 to add your user defined attributes.</td>
<td>Optional</td>
</tr>
</tbody>
</table>

Here's a sample file that contains the header values at the beginning and lists three contexts to be imported. For importing several contexts, add more entries in the same format.

```
ApplicationId|EFFCode |ContextCode|Name|EnabledFlag|MultirowFlag|Description|FlexfieldUsageCode|ViewPrivilegeName|CExtAttribute1|CExtAttribute2|CExtAttribute3|CExtAttribute4|CExtAttribute5
-----------------------------------------------
0|FLEX_SN_EFF1|OBJ_TEST_4|Object test 44|Y|N|desc 44 3363|FLEX_SN_EFF1_USAGE2|flex_sn_sitems_view|flex_sn_view|flex_sn_sitems_view|
0|FLEX_SN_EFF1|OBJ_TEST_3|Object test 33|Y|N|flex_sn_sitems_view|FLEX_SN_EFF1_USAGE1|flex_sn_view|flex_sn_sitems_view|flex_sn_sitems_view|NEW_TEST11|
0|FLEX_SN_EFF1|OBJ_TEST_3|Object test 33|Y|N|new desc aug 14|FLEX_SN_EFF1_USAGE2|flex_sn_sitems_view|flex_sn_sitems_view|flex_sn_sitems_view|CE1_TESTupd_aug15|
```

### Context Segment

To create a file containing context segments, include the headers in the following table:

<table>
<thead>
<tr>
<th>Header</th>
<th>Data Type</th>
<th>Data Size</th>
<th>Description</th>
<th>Required or Optional</th>
</tr>
</thead>
<tbody>
<tr>
<td>ApplicationId</td>
<td>Long</td>
<td>-</td>
<td>This value determines the Application to which your flexfield, context, and segment belongs to.</td>
<td>Required</td>
</tr>
<tr>
<td>EFFCode</td>
<td>String</td>
<td>40</td>
<td>This value is the code of the extensible flexfield to which you’re adding your segment. This value should match the code of an extensible flexfield</td>
<td>Required</td>
</tr>
<tr>
<td>Header</td>
<td>Data Type</td>
<td>Data Size</td>
<td>Description</td>
<td>Required or Optional</td>
</tr>
<tr>
<td>-----------------</td>
<td>-----------</td>
<td>-----------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
<td>----------------------</td>
</tr>
<tr>
<td>ContextCode</td>
<td>String</td>
<td>80</td>
<td>This value is the code for the context to which you're adding your segment. This value should match a context code that's in the application.</td>
<td>Required</td>
</tr>
<tr>
<td>SegmentCode</td>
<td>String</td>
<td>30</td>
<td>This value is the code for your segment.</td>
<td>Required</td>
</tr>
<tr>
<td>Name</td>
<td>String</td>
<td>60</td>
<td>This is the back end name for your segment.</td>
<td>Required</td>
</tr>
<tr>
<td>ColumnName</td>
<td>String</td>
<td>30</td>
<td>This value determines the table column your segment uses to save data. This value must match a column that's in the application.</td>
<td>Required</td>
</tr>
<tr>
<td>ValueSetCode</td>
<td>String</td>
<td>60</td>
<td>This value is the code for the value set you want to use in your segment. This value must match a value set code in the application.</td>
<td>Required</td>
</tr>
<tr>
<td>DisplayType</td>
<td>String</td>
<td>30</td>
<td>This value determines the display type of the segment. Valid values for this attribute are TEXT_BOX, TEXT_AREA, RICH_TEXT_EDITOR, HIDDEN, LOV, POP_UP_LIST, DROP_DOWN_LIST, RADIO_BUTTON_GROUP, STATIC_URL, CHECKBOX, COLOR, DATE_TIME</td>
<td>Required</td>
</tr>
<tr>
<td>Prompt</td>
<td>String</td>
<td>80</td>
<td>This value determines the display name of the segment.</td>
<td>Required</td>
</tr>
<tr>
<td>Header</td>
<td>Data Type</td>
<td>Data Size</td>
<td>Description</td>
<td>Required or Optional</td>
</tr>
<tr>
<td>-------------------------</td>
<td>-----------</td>
<td>-----------</td>
<td>------------------------------------------------------------------------------</td>
<td>----------------------</td>
</tr>
<tr>
<td>ShortPrompt</td>
<td>String</td>
<td>80</td>
<td>This value determines the shortened display name of the segment.</td>
<td>Required</td>
</tr>
<tr>
<td>EnabledFlag</td>
<td>String</td>
<td>1</td>
<td>If this value is set to Y, then your segment is enabled. If the value is N, then it's disabled.</td>
<td>Required</td>
</tr>
<tr>
<td>RequiredFlag</td>
<td>String</td>
<td>1</td>
<td>If this value is set to Y, user input for this segment is mandatory during run time. If the value is N, user input is optional.</td>
<td>Required</td>
</tr>
<tr>
<td>ReadOnlyFlag</td>
<td>String</td>
<td>1</td>
<td>If this value is set to Y, then your segment is read-only. If the value is N, it's not.</td>
<td>Required</td>
</tr>
<tr>
<td>Description</td>
<td>String</td>
<td>240</td>
<td>This value gives your segment a description.</td>
<td>Optional</td>
</tr>
<tr>
<td>UOMclass</td>
<td>String</td>
<td>20</td>
<td>This value determines the unit for the data you add in your segment.</td>
<td>Optional</td>
</tr>
<tr>
<td>TerminologyHelpText</td>
<td>String</td>
<td>80</td>
<td>This text provides a description for the segment.</td>
<td>Optional</td>
</tr>
<tr>
<td>InFieldHelpText</td>
<td>String</td>
<td>160</td>
<td>This text provides instructions on how to use the segment.</td>
<td>Optional</td>
</tr>
<tr>
<td>SequenceNumber</td>
<td>Integer</td>
<td>-</td>
<td>This value determines the order in which your segments are displayed in your context.</td>
<td>Optional</td>
</tr>
<tr>
<td>DefaultType</td>
<td>String</td>
<td>30</td>
<td>This value determines the type of default value your segment takes if the user doesn't enter any data in your segment. The types are CONSTANT, SQL, and GROOVY_EXPRESSION</td>
<td>Optional</td>
</tr>
<tr>
<td>Header</td>
<td>Data Type</td>
<td>Data Size</td>
<td>Description</td>
<td>Required or Optional</td>
</tr>
<tr>
<td>-----------------------------</td>
<td>-----------</td>
<td>-----------</td>
<td>-----------------------------------------------------------------------------</td>
<td>----------------------</td>
</tr>
<tr>
<td>DefaultValue</td>
<td>String</td>
<td>4000</td>
<td>This value determines the default value your segment takes if the user doesn’t enter any data.</td>
<td>Optional</td>
</tr>
<tr>
<td>DisplayWidth</td>
<td>Integer</td>
<td>-</td>
<td>This value determines the display width of your segment, in terms of maximum number of characters displayed in a line.</td>
<td>Optional</td>
</tr>
<tr>
<td>DisplayHeight</td>
<td>Integer</td>
<td>-</td>
<td>This value determines the display height of your segment, in terms of maximum number of lines displayed in the segment.</td>
<td>Optional</td>
</tr>
<tr>
<td>CheckboxCheckedValue</td>
<td>String</td>
<td>30</td>
<td>This value determines the value of a checked check box in your segment.</td>
<td>Required for display type CHECKBOX.</td>
</tr>
<tr>
<td>CheckboxUncheckedValue</td>
<td>String</td>
<td>30</td>
<td>This value determines the value of an unchecked check box in your segment.</td>
<td>Required for display type CHECKBOX.</td>
</tr>
<tr>
<td>RangeType</td>
<td>String</td>
<td>30</td>
<td>This value sets two fields as two ends of a range of values. Set value as LOW or HIGH for your low end and high end fields respectively.</td>
<td>Optional</td>
</tr>
<tr>
<td>BIEnabledFlag</td>
<td>String</td>
<td>1</td>
<td>This value determines whether your segment is BI enabled. Y means it’s, N means it’s not.</td>
<td>Optional</td>
</tr>
<tr>
<td>MultirowUniqueKeyFlag</td>
<td>String</td>
<td>1</td>
<td>If this value is set to Y, the segment is marked as the unique key for a multirow context. Default value for this header is N. At least 1 segment in a multirow context</td>
<td>Required for multi-row contexts</td>
</tr>
<tr>
<td>Header</td>
<td>Data Type</td>
<td>Data Size</td>
<td>Description</td>
<td>Required or Optional</td>
</tr>
<tr>
<td>-----------------------</td>
<td>-----------</td>
<td>-----------</td>
<td>-----------------------------------------------------------------------------</td>
<td>----------------------</td>
</tr>
<tr>
<td>ShowValueDescription</td>
<td>String</td>
<td>1</td>
<td>If this value is set to Y, the value set description for the value set associated to the segment is displayed. If the value is N, it's not displayed.</td>
<td>Optional</td>
</tr>
<tr>
<td>SegmentIdentifier</td>
<td>String</td>
<td>30</td>
<td>This value determines the API name for your segment. There are a set of conventions to be followed when naming APIs.</td>
<td>Optional</td>
</tr>
</tbody>
</table>

Here's a sample file that contains the header values at the beginning and lists a context segment to be imported. For importing several context segments, add more entries in the same format.

```
ApplicationId|EFFCode|ContextCode|SegmentCode|Name|ColumnName|ValueSetCode|DisplayType|Prompt|ShortPrompt|EnabledFlag|RequiredFlag|ReadOnlyFlag|Description|UomClass
0|FLEX_SN_EFF1|ColdSourceTargetContext|Cold_TEST_SEG1|cold seg 1|ATTRIBUTE_CHAR3|EFF_BASIC_FMT_CHR|TEXT_BOX|Prompt|Short Prompt|Y|Y|N|TEST desc aug 14_3PM|
```

**Related Topics**
- Overview of Files for Import and Export
- Upload Files to WebCenter Content Server

**Import Extensible Flexfields**

Use the Import option on the Manage Extensible Flexfield page to bulk import the flexfield data that includes value set, context, and context segment details.

**Prerequisite**
The files containing the flexfield details are available in the document repository of Oracle WebCenter Content.

**Importing Flexfields**
To import flexfields:

1. Sign in to the application as an implementation consultant or an administrator.
2. In the Setup and Maintenance work area, open the Manage Extensible Flexfield task or a similar task for importing flexfields.
3. In Search Results, from the Actions menu, select Import.
4. On the Upload Flexfield Data dialog box, select the WebCenter Content account to which the files were uploaded.
5. Enter the names of the separate files containing the value set, context, and context segment information. The names here must match with the names of the files uploaded to the selected account.
6. Click **Upload**. The flexfield details are imported.

| Note: If the import fails, click the link to the log file on the confirmation dialog box and examine the cause of failure.

**Related Topics**
- Import Flexfields, Lookups, or Profile Values Using Web Services
- Import Value Set Values Using Web Services

**FAQs for Extensible Flexfields**

**Why did the extensible flexfield context not appear at run time?**
If a deployed extensible flexfield context doesn't appear in the user interface, verify that the context is associated with one of the category's pages defined for the extensible flexfield.

**Key Flexfields**

**Overview of Key Flexfields**

*Key flexfields* provide a means to capture a key such as a part number, a job code, or an account code. A key flexfield consists of one or more *segments*, where each segment can have a meaning.

For example, a part number 10-PEN-BLA-450 might correspond to a black pen from supplier #450 sold by division #10 (office supplies). Behind the scenes, the application uses a unique number, 13452, for this part, but the user always sees the 10-PEN-BLA-450 part number.

The following aspects are important to understanding key flexfields:

- Architecture
- Segments and segment labels
  - *Structures*
  - *Segment* and *structure instances*
- Combinations
- Dynamic combination creation
- Security

Key flexfields aren't optional. You must configure key flexfields to ensure that your applications operate correctly. You configure and maintain key flexfield definitions with the Manage Key Flexfields task. To get a list of predefined key flexfields, use the Manage Key Flexfields task in the Setup and Maintenance work area. For information about specific key flexfields, see the help for the product where the associated business component is implemented.
Architecture
Flexfield metadata is stored in the flexfield metadata tables. When you configure a key flexfield, you define metadata about the key flexfield covering aspects such as:

- Segments are in a structure
- Structures in the flexfield
- Value sets in each segment

Based on the flexfield metadata, actual part numbers are captured at run time as a combination of segment values and stored in a combinations table. A combinations table contains all the segment columns for a flexfield, a unique ID column, and a structure instance number column. The structure instance number column differentiates multiple arrangements of the segment columns. For example, a part number containing multiple segments can be represented by a key flexfield. A part number key flexfield has a corresponding combinations table. In that table, the flexfield stores a list of the complete codes, with each segment of the code in a column, with the corresponding unique ID and structure instance number for the code. When users define a new part number or maintain existing part numbers in the parts catalog, they directly maintain rows in the combinations table.

The foreign key table contains a different business entity than the combinations table. For example, the business entity in the foreign key table is order lines or invoice lines that contain foreign key references to parts for ordering. Any number of foreign key tables can reference a particular entity represented by a key flexfield.

Segments and Segment Labels
A key flexfield contains segments and a segment label identifies a particular segment within a key flexfield. Segment labels are defined and made available by the product development. A segment contains the following details:

- A prompt
- A short prompt
- Display width
- The sequential position of the segment within the key flexfield structure
- The range type
- Column name of the attribute being stored by the segment
- A default value set
- A label for the segment

Applications identify a particular segment for some purpose such as security or computations. Segment name or segment order cannot reliably identify a segment because key flexfield segments can be configured to appear in any order with any prompts. A segment label functions as a tag for a segment.

For example, the requirement is to identify which segment in the accounting flexfield contains balancing information and which segment contains natural account information. A segment label determines which segment you are using for natural account information. When you define your accounting flexfield, you must specify which segment labels apply to which segments. Some labels must be unique, and cannot be applied to more than one segment in each structure. Other labels are required, and must be applied to at least one segment in each structure.

A segment label helps a user searching for segments, such as the Cost Center label for all segments across key flexfields that store a value for the cost center.
Structures

A key flexfield structure definition includes the number of segments and their order.

In some applications, different users like to see different segment structures for the same flexfield. A key flexfield can have multiple structures if registered to support more than one structure.

The flexfield can display different fields for different users based on a data condition in your application data, such as the value of another field entered by the user or the user’s role. For example, the correctly formatted local postal address for customer service inquiries differs based on locale. A postal address key flexfield could display different segments and prompts for different users based on a location condition in your application data, such as the user’s role or a value entered by the user.

Each structure can have one or more segments. Thus a segment is a child of a structure. To store a particular segment, such as Cost Center, in two different structures, you must define the segment separately in each structure. Each structure may have one or more structure instances. Each instance of a structure shares the same number and order of segments, but differs in the values or value sets used in validating the segments.

Structure and Segment Instances

You can define multiple configurations of a key flexfield structure. These structure instances have the same segment structure, in the same sequence order. They differ primarily in how each segment is validated. You define a structure instance for each key flexfield and each key flexfield structure instance.

The segments in a key flexfield structure instance are segment instances. A segment instance is a segment with a specific value set assigned to it. If a key flexfield is registered with a tree structure, you can specify a tree code for a segment instance.

Combinations

A combination is a complete code, or combination of segment values that makes up the code, that uniquely identifies an object.

For example, each part number is a single combination, such as PAD-YEL-11x14 or 01-COM-876-7BG-LTN. In these combinations, the hyphen is the segment separator. If you have ten parts, define ten combinations. A valid combination is an existing or new combination that can be used because it's currently active and doesn't violate cross-validation or security rules. A combination has different segments depending on the flexfield structure being used for that combination. Any combination is associated with only one particular flexfield structure.

Many applications refer to a key flexfield combination by using the name of the entity or the key flexfield itself. For example, Assets uses the asset key flexfield and refers to one of its combinations as an asset key or asset key flexfield. In another example, Oracle Fusion General Ledger refers to combinations of the accounting flexfield as account or GL account.

Each key flexfield has one corresponding table, known as the combinations table, where the flexfield stores a list of the complete codes, with one column for each segment of the code, together with the corresponding unique ID number (an account combination ID) for that code. Then, other tables in the application have a column that stores just the unique ID for the code. For example, you may have a part number code, such as PAD-YEL-11x14. The Parts combinations table stores that code along with its ID, 57494. If your application lets you take orders for parts, you might then have an Orders table that stores orders for parts. That Orders table would contain a single column that contains the part ID, 57494, instead of several columns for the complete code PAD-YEL-11x14. Typically, one combinations page maintains the key flexfield, where the key flexfield is the representation of an entity in your application. Maintain individual combinations, such as part numbers in the combinations page.
Dynamic Combination Creation
Dynamic combination creation is the insertion of a new valid combination into a combinations table from a page other than the combinations page. The following table lists the levels at which dynamic combination creation may be enabled.

<table>
<thead>
<tr>
<th>Level Of Dynamic Combination Creation</th>
<th>Controlled By:</th>
</tr>
</thead>
<tbody>
<tr>
<td>Flexfield</td>
<td>Application development</td>
</tr>
<tr>
<td>Each usage or reference to the key flexfield</td>
<td>Application development</td>
</tr>
<tr>
<td>Structure instance</td>
<td>Administrators and implementation consultants</td>
</tr>
<tr>
<td>Other</td>
<td>Administrators and implementation consultants</td>
</tr>
</tbody>
</table>

If your key flexfield or certain usages or references of the key flexfield don't permit dynamic combination creation, you may control whether dynamic combination creation is enabled for each structure instance. If enabled, a user can enter a new combination of segment values using the flexfield window from a foreign key page. For example, when entering a transaction, a GL user can enter a new expense account combination for an account that doesn't yet exist. Your application creates the new account by inserting the new combination into the combinations table behind the scenes. Assuming that the new combination satisfies any existing cross-validation rules, the flexfield inserts the new combination into the combinations table, even though the combinations table isn't the underlying table for the foreign key page.

Related Topics
- Update Existing Setup Data

Considerations for Planning Key Flexfields
Your first step in planning your key flexfields is to determine which key flexfields your application requires. Your plan should include:

- The purpose of the key flexfield
- The number and length of its available segment columns
- Whether your key flexfield permits more than one structure
- Whether more than one structure must be defined
- The number, order and length of your segments for each structure

Before You Begin
Once you have identified a flexfield, plan its configuration in advance. Compile a list of the UI pages and other artifacts in your deployment that are affected by the configuration. Verify that you are provisioned with the roles required to view and configure the flexfield. Use the Highlight Flexfields command in the Administration menu to view the run time
page where the flexfield appears. Plan how you deploy the flexfield for test and production users and review the tools and tasks available for managing flexfields.

If you plan to use value sets, create them before configuring the key flexfield. You cannot create value sets for key flexfields at the time that you add and configure key flexfield segments.

Access to Flexfield-Related Tasks
To configure flexfields and value sets, you must have access to the tasks for managing flexfields. Contact your security administrator for details. For information about product-specific flexfield tasks, such as Manage Fixed Assets Key Flexfields, refer to the product-specific documentation.

Restrictions
If you plan to use value sets, create them before configuring the flexfield. Plan your key flexfield configuration to scale to your enterprise needs. For example, if you expect to disable old cost centers and enable new ones frequently, plan a larger maximum size for your cost center value set so that you can have more available values. A 3-character value set with one thousand available values provides more room for changes than a 2-character value set with 100 available values.

Note the code name of the flexfield you intend to configure so that you find it easily in the tasks for managing key flexfields. In some cases you can configure how the flexfield appears on the page. See product-specific documentation to determine any restrictions on using product-specific key flexfields.

Reporting
To report on your data by certain criteria or sub-entities, such as account number or project or region, consider making that sub-entity a distinct segment, rather than combining it with another sub-entity. You can categorize and report on smaller discrete units of information.

Considerations for Managing Key Flexfields
Consider the plans for a key flexfield, security, and resulting run time pages when configuring key flexfields.

Planning
Plan structures carefully and enable them for future needs. Don't change the number, order, and maximum length of segments once you have acquired flexfield data.

Structure Delimiters
A delimiter separates the segments when they appear to users. The delimiter value of a structure specifies the character used to visually separate segment values when the key flexfield is displayed as a string of concatenated segments in the UI.

Identify the delimiter value of your key flexfield carefully so that it doesn't conflict with the flexfield data. For example, if your data frequently contains periods, such as in monetary or numeric values, don't use a period as your segment separator. Any character you expect to appear frequently in your segment values or descriptions isn't a good choice for the delimiter. If you change the configuration of a key flexfield, such as the delimiter, the change affects the previously stored key flexfields with that structure.

Security
Oracle Fusion data security enforces value set security.
Within key flexfields, value set security applies to the selection of the individual segment values in the segment list of values. When selecting a key flexfield segment value from the combinations table, data security permits display of only the combinations whose segment values you have access to. Applications development controls whether or not value set security rules propagate to the foreign key table. By default they do.

Run Time Pages
Application development determines the user interface (UI) pages used to render flexfields. The types of key flexfield UI pages are as follows:

- Combinations pages where the underlying entity objects use the combinations table itself
- Foreign key pages where the underlying entity objects contain a foreign key reference to the combinations table
- Partial usage pages where some or all of the key flexfield's segment columns are in a product table

The same key flexfield can be used in different ways on different pages.

A page with a foreign key reference has a base table or view that contains a foreign key reference to a combinations table with the actual flexfield segment columns. This lets you manipulate rows containing account combination IDs (account combination).

A page with partial usage of a key flexfield presents segments that are defined on a product's transactional table in addition to being defined on a combinations table. In the case of a partial usage page, only a part of the configuration is likely to be visible. This enables the key flexfield to act more like a descriptive flexfield.

An account combination maintenance page or combinations page presents the combinations table. This enables directly creating and maintaining account combinations. The combinations table contains all key flexfield segment columns and a unique ID column.

A typical application has only one combinations page. An application might not have a combinations page if it doesn't support maintenance by administrators.

A page containing a search region enables users to select which attributes of the key flexfield view object to use as criteria to search for flexfield metadata.

For example, you can configure seven segments for the Account key flexfield. In a foreign key reference page, users see the typical key flexfield picker with all seven segments where they can search for combinations. In a partial usage page using the same key flexfield, users potentially could see only a single segment such as the Cost Center labeled segment, or they might see multiple segments but displayed as individual segments rather than options for selecting combinations.

For more information about key flexfield pages, see the Oracle Fusion Applications Developer's Guide.

Key Flexfield Structures
A key flexfield structure arranges the segments of a key so that you can reuse a single key flexfield in multiple combinations of the same segments or a subset of those segments. Multiple instances of a single structure can accommodate differences in the value sets assigned to the structure's segments.
The structure determines the following aspects of a key flexfield:

- The segments to include
- The order of the segments
- Segment labels on the included segments
- Properties for each segment applied to the instances of the segments in an instance of the structure

Managing Key Flexfield Structures

All the segments defined for a key flexfield are available to be included in a key flexfield structure. You can define as many segments as there are defined segment columns in your key flexfield combinations table. Ensure that you add segments in the order that your key requires. Once deployed, the order cannot be changed.

Enable segments to indicate that they are in use. A flexfield doesn't display disabled segments in run time. To protect the integrity of your data, disable a segment if you have already used it to enter data.

Key Flexfield Structure Instances and Segment Instances

A key flexfield structure can have one or more alternate structure instances. The instances of a key flexfield structure share the following aspects of the structure:

- The same set of segments
- The same arrangement of segments
- The same properties at the segment and structure levels

The differences among structure instances include whether dynamic combination creation is permitted. Likewise, at the structure instance level, differences among segment instances are based on the following:

- Value set
- Default type and default value
- Tree code
- Whether the segment is any of the following:
  - Required
  - Displayed
  - Enabled for business intelligence
  - Optional or required as a query criterion

For example, you can use one group of value sets for the US and another for France.
The following figure shows two structures instances for a part number structure.

The structures differ in the number of segments and the segment separators used. The structure instances share all the properties defined for that structure. However, the structure instances may vary if the properties are defined at the structure instance or segment instance level. For example, the value set assigned to the segment instances.

**Query Required Segment Instances**

You can designate a key flexfield segment instance as a query for making it a selectively required attribute. A user can use it as a key flexfield combination search. On the Manage Key Flexfields UI page, if you indicate that a segment instance requires indexing, add the column representing the segment to the database index. Commonly, a database administrator (DBA) adds columns to the database index.
Following deployment, the combination picker of the key flexfield displays the query required attributes as selectively required. A user must specify at least one of the query required attributes in the search criteria. This prevents unnecessary searches that could cause performance issues.

For example, you mark the cost center and account attributes as query required and ensure that the corresponding columns in the database are indexed. A user can search for combinations by entering cost center or account or both as search criteria. No search is performed if a user doesn't enter at least one query required attribute as search criteria.

**Tip:** Index the Structure Instance Number column on your combinations table to improve run time performance.

### Dynamic Combinations

If a *key flexfield* supports dynamic combination creation, you can select to enable this feature by selecting **Dynamic Combination Creation Allowed**. As a result, users enter values at run time that produce new account combinations for the flexfield. If **Dynamic Combination Creation Allowed** isn’t enabled, new valid combinations can only be entered using the combinations table for the flexfield.

### Trees

You may define a tree code for the value set assigned to the segment instance. When you assign the tree code to the segment instance, tree hierarchy search operations are available on the segment values.

For a segment instance to be based on a tree, the following must be true.

- Application development registered the key flexfield with a tree structure. The tree structure may be fixed across all segments in the flexfield, or may vary across segments.
- A tree code for that *tree structure* exists.
- The tree code includes tree versions containing the values of the value set assigned to the segment instance.
- You assign the required tree code directly to the segment instance.

If these conditions are satisfied, you can assign the same or different tree codes to the different segment instances that use the same value set.

### Cross-Validation Rules

You can control the creation of new key flexfield code combinations by defining cross-validation rules. A cross-validation rule defines validation across segments and enforces whether a value of a particular segment can be combined with specific values of other segments to form a new combination.

The following table compares segment validation to cross-segment validation:

<table>
<thead>
<tr>
<th>Type of validation</th>
<th>Type of control</th>
</tr>
</thead>
<tbody>
<tr>
<td>Segment validation</td>
<td>Controls the values you can enter for a particular segment</td>
</tr>
<tr>
<td>Cross-segment validation</td>
<td>Controls the combinations of values that administrators and end users can create for key flexfields</td>
</tr>
</tbody>
</table>
Cross-validation rules prevent the creation of combinations with values that can't coexist in the same combination. For example, your company requires that all revenue accounts must have a specific department. Therefore, account combinations that have revenue account values, such as all values between 4000 and 5999, must have a corresponding department value other than 000, which indicates no department is specified. You can define cross-validation rules that disallow creation of combinations with incompatible segments, such as 4100-000 or 5000-000.

Alternatively, suppose your accounting key flexfield has an Organization segment with two possible values, 01 and 02. You also have a Natural Account segment with many possible values, but company policy requires that Organization 01 uses the natural account values 001 to 499 and Organization 02 uses the natural account values 500 to 999. You can create cross-validation rules to ensure that users cannot create a general ledger account with combinations of values such as 02-342 or 01-750.

The following aspects are important to understanding cross-validation rules:

- Rule Definitions
- Enforcement
- Timing

**Rule Definitions**

The following table contains definitions used in cross-validation rules:

<table>
<thead>
<tr>
<th>Rule Definition</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Uniquely identifies cross-validation rules in a deployment.</td>
</tr>
<tr>
<td>Description</td>
<td>Helps administrators identify the purpose of the rule.</td>
</tr>
<tr>
<td>Error message</td>
<td>Explains why the attempted combination violates the rule.</td>
</tr>
<tr>
<td>Start Date, End Date</td>
<td>Indicates the period of time when the rule is in effect.</td>
</tr>
<tr>
<td>Enabled</td>
<td>Determines whether the rule is enforced.</td>
</tr>
<tr>
<td>Condition filter</td>
<td>Determines the conditions in which an enabled cross-validation rule should be evaluated.</td>
</tr>
<tr>
<td>Validation filter</td>
<td>Determines the validation that the rule enforces when that condition is met.</td>
</tr>
</tbody>
</table>

When the event specified in the condition filter is applicable, the validation filter condition must be satisfied before the combination can be created. If the event specified in the condition filter isn't applicable, then the combination is considered to pass the rule and the rule won't be evaluated even if it is enabled.

**Note:** If you don't specify any statement in the condition filter, then the condition is always true and the rule is always evaluated.
Enforcement

Cross-validation prevents creation of invalid combinations by administrators using maintenance pages and end users using dynamic insertion in foreign key pages.

Enabled rules are enforced when there is an attempt to create a new combination of segment values. Disabled rules are ignored. Deleting the rule has the same effect, but you can re-enable a disabled rule.

Timing

When users attempt to create a new combination, the key flexfield evaluates any cross-validation rules that are enabled and in effect.

Note: Cross-validation rules have no effect on combinations that already exist. The flexfield treats any existing invalid combinations that pre-date the rule as valid.

If you want to prevent users from using previously existing combinations that are no longer valid according to your cross-validation rules, manually disable those combinations using the combinations page for that key flexfield.

When defining a cross-validation rule, specify a start and end date to limit the time when the rule is in effect. The rule is valid for the time including the From and To dates.

Considerations for Cross-Validation Rules

To validate the key flexfield combinations of segment values across segments, optimize your cross-validation rules to improve the experience of administrators and users.

Consider the following when defining cross-validation rules:

- Filters
- Rule Complexity
- Maintenance

Filters

A cross-validation rule includes a condition filter and a validation filter. The rule is evaluated using the following logical order: If the condition filter is satisfied, then apply the validation filter.

The condition filter describes the event when the rule is evaluated. If the event specified in the condition filter isn't applicable, then the rule isn't evaluated, even if enabled. When the event specified in the condition filter is applicable, the validation filter condition must be satisfied before the combination can be created.

For example, your organization has determined that a certain company value called Operations can't use a specific cost center called Marketing. You can define a cross-validation rule to validate your combinations.

1. The rule evaluates the company condition filter.
2. When company is equal to Operations, the rule evaluates the cost center validation filter.
3. When cost center is equal to Marketing, the rule prevents a combination from being created.
4. The error message you defined for the rule displays to inform the user that the attempted combination violates the rule.

Such a rule doesn't affect the creation of combinations with Marketing cost center and company values other than Operations.
Rule Complexity
For optimal performance and ease of understanding, define several simple validation rules instead of using one complex rule. Simple validation rules let you provide a more specific error message and are easier to maintain over time. Avoid rules that control validation across more than two segments, where possible. While you can define cross-validation rules that span two or more segments, it becomes difficult to interpret cross-validation error messages and rectify invalid key flexfield combinations.

Maintenance
To maintain consistent validation, review existing key flexfields when you update your cross-validation rules. Regardless of your current validation rules, you can use an existing key flexfield combination if it's enabled. Therefore, to ensure accurate validation, you must review your existing combinations and disable any combinations that don't match the criteria of your new rules.

Tip: To keep this type of key flexfield maintenance to a minimum, decide upon your cross-validation rules when you first set up your key flexfield structure. Define cross-validation rules before creating combinations and before combinations are used in transactions.

To prevent users from using existing combinations that are no longer valid according to your cross-validation rules, disable them using the combinations page.

Edit a Cross-Validation Rule
Cross-validation rules prevent specific combinations of segment values in account combinations. You can use the Manage Cross-Validation Rules task to edit existing rules or to create one-off rules.

Scenario
Your organization has a cross-validation rule called Companies 131 and 151, which restricts account combinations for those companies to department 40 and product 211. Account combinations for both companies should now include department 30. To edit the cross-validation rule, perform these steps.

1. In the Setup and Maintenance work area, go to the following:
   - Offering: Financials
   - Functional Area: Financial Reporting Structures
   - Task: Manage Cross-Validation Rules
2. Select the chart of accounts for your organization and select the Companies 131 and 151 cross-validation rule.
The following figure shows the section of the Edit Cross-Validation Rules page with the condition and validation filter details for companies 131 and 151. A condition is defined for company values equal to 131 or 151, and the validation specifies the department value equals 40 and the product value equals 211.

3. Click the Validation Filter icon.
4. Click Add Fields and select the Department segment.
5. Accept the default operator, which is Equals, and select department 30.

The following figure shows the Validation Filter window with three validations: department equals 40, department equals 30, and product equals 211.

6. Click OK.
7. Click Save.
The following figure shows the details for the updated validation on the Edit Cross-Validation Rules page. The validation specifies departments equal to 30 or 40, and the product equal to 211.

![Validation Filter 3 Filter Conditions Defined](image)

( Department Equal to 30
OR Department Equal to 40 )
AND
Product Equal to 211
used with department 40 and product 211.

8. To update the error message, search for and select the Manage Messages for General Ledger task. Query the error message name for the cross-validation rule and edit the message to include department 30.

Related Topics
- Update Existing Setup Data

Considerations for Enabling Key Flexfield Segments for Business Intelligence

A key flexfield registered in the database as enabled for Oracle Business Intelligence (BI) includes a BI Enabled setting for each of its segment instances. When a segment instance is BI-enabled, it's available for use in Oracle Business Intelligence.

The following aspects are important in understanding BI-enabled key flexfield segments.

- Flattening business components to use BI-enabled segments in Oracle BI
- Equalizing segments to prevent duplication and complexity in the flattened component
- Mapping attributes of flattened business components to logical objects in Oracle BI
- Managing the labels that map segments to logical objects in Oracle BI

After you deploy a business intelligence-enabled flexfield, use the Import Oracle Fusion Data Extensions for Transactional Business Intelligence process to import the flexfield changes into the Oracle Business Intelligence repository. Users can make use of the newly-generated attributes in business intelligence applications. For additional information about logical objects and import, refer to the Oracle Transactional Business Intelligence Administrator's Guide.

Flattening

When you deploy a business intelligence-enabled key flexfield, the deployment process generates an additional set of flattened business components for use in business intelligence. The flattened business components include attributes for business intelligence-enabled segment instances only.
If you assigned a label to a segment, the flattened components include a single attribute representing all segment instances with that label. If you didn’t assign a label, the flattened components include a discrete attribute for each BI-enabled segment instance in each structure.

**Mapping to Logical Objects in Business Intelligence**

You can simplify reporting by representing similar segments as a single logical object in Business Intelligence. If you assign a label to segments that serve the same purpose in different structures, you can consolidate the segments into a single attribute. This prevents duplication and the extra workload and complexity that result from the flattening process. For example, an organization may have more than one definition of its key accounting flexfield to support different requirements for accounting reporting. A US Accounting Flexfield structure may have a segment called Subaccount to track project expenditures. The same type of information may be tracked in a UK accounting flexfield structure with a segment called Project. Equalize these two segments to create a single list of values for reporting.

Non-labeled segments aren’t equalized across context values, so the flattened components include a separate attribute for each segment for each structure. It may not be possible to equalize similarly labeled segments if they have incompatible data types or value set types.

Assign a label to a segment to map the corresponding attribute in the flattened components to a logical object in Oracle Business Intelligence. Using labels to map segments to BI logical objects minimizes the steps for importing the flexfield into Oracle Business Intelligence. Assigning a label to a segment serves to equalize the attribute across structures, as well as map the equalized attribute to business intelligence.

**Managing Labels**

You may assign a predefined label (if available) to segments or create labels for assignment, as needed. Specify a code, name, and description to identify each label. In the BI Object Name field, enter the name of the logical object in Oracle Business Intelligence to which the segment label should map during import. Specifying the BI logical object minimizes the steps for importing the flexfield into Oracle Business Intelligence and helps to equalize context-sensitive segments across structures.

If no labels are assigned to a BI-enabled segment, or the BI Object Name on the assigned label doesn’t exist in business intelligence, you must manually map the segment to the required logical object when importing into Oracle Business Intelligence. In addition, segments without labels cannot be equalized across structures. The flattened components include a separate attribute for each non-labeled segment in each structure.

**Importing to Oracle Business Intelligence Repository**

After you deploy a business intelligence-enabled flexfield, import the flexfield changes into the Oracle Business Intelligence repository to make use of the newly flattened business components in business intelligence. Then propagate the flexfield object changes. When you import the metadata into the Oracle Business Intelligence repository, you must do so as the FUSION_APPS_BI_APPID user.

To import flexfield changes into the Oracle Business Intelligence repository in Oracle Cloud implementations, run the Import Oracle Fusion Data Extensions for Transactional Business Intelligence process. For additional information about import, refer to the Oracle Transactional Business Intelligence Administrator’s Guide.

**Note:** When you import a flexfield into the Oracle Business Intelligence repository, you see both `<name>` and `<name>_c` attributes for each segment, along with some other optional attributes. The `<name>` attribute contains the value. The `<name>_c` attribute contains the code of the value set that the value comes from, and is used for linking to the value dimension. You must import both attributes.
Example of a Key Flexfields

A key flexfield can capture expense account information.

Scenario
When entering details for each expense, the user specifies an account to which the expense is charged.

Entering Expense Accounts
A user interface for entering expenses helps the user select an expense account that identifies the cost center and other details needed for processing the expense.

Analysis
The expense account field is a foreign key reference to an account combination (EXPENSE_LINES.EXPENSE_ACCOUNT = ACCOUNT.COMBINATION).

Account combinations Table for Entering Accounts and Employees
The account combinations table supports entering account information, such as for expense accounts.
The following figure shows the origin in the account combinations table of the account specified by the user. The account combination ID record stores the information of the key flexfield segments used to assemble the expense account based on the key flexfield configuration.

The combinations page, which is the maintenance page for the key flexfield, is for managing rows in the combinations table. In this example, managing the combinations means adding or editing account numbers that adhere to the key flexfield metadata rules.

The following figure shows the account combination details for the example expense account reflected in the flexfield configuration and the account combinations table.
If dynamic combination creation isn't enabled, then when entering an expense line, the user can only select an account that already exists in the ACCOUNTS (combinations) table. If they require an account that doesn't exist, they must consult with the appropriate application administrator who can add the account to the combinations table.

If dynamic combination creation is enabled, then when entering an expense line, the user can either select a preexisting account, or type in a new account that is created dynamically on the fly in the ACCOUNTS (combinations) table. Once the new combination is created, the same user can refer to it on the expense line.

When managing employee information, the user specifies the cost center that the employee belongs to. The cost center field corresponds to a single, labeled segment of the Account Key Flexfield and has metadata defined such as the allowable value set for that segment.

In the following figure, instead of specifying a cost center ID reference to an account, only the cost center segment is used and the value is stored directly on the employee table.
Related Topics

- Example of One Chart of Accounts Structure with Many Instances
7 Configuration of Home Page and Navigation

Overview of Configuring Home Page and Navigation

You can configure the Navigator and springboard, as well as define settings for the home page and springboard using the Structure work area. To open this work area, from the Navigator menu, select Configuration > Structure. This work area has 2 tabs: Navigation Configuration and Home Configuration. Use the Navigation Configuration page to configure the Navigator and springboard. Use the Home Configuration page to configure the home page.

Some of the key configuration tasks that you can do using the Structure work area are:

- Create, edit, and rename the Navigator menu items and springboard icons, which appear on the home page.
- Create a duplicate version of an existing predefined page entry and edit the duplicate page entry to meet your specific requirements.
- Change the visibility settings of the Navigator menu items and springboard icons, and reorder them.
- Change the visibility settings of the icons for infolet pages in the page control, which appear on the home page.
- Rename and reorder the icons for infolet pages in the page control.

Groups and Page Entries

To address needs specific to your organization, you can create or edit groups and page entries for the Navigator and springboard. For example, you may want to link page entries to web pages or external applications.

Groups and page entries are available on the springboard and navigator menu. A page entry is the navigator link or springboard icon that opens a page. A page is a single screen to perform related tasks. A few page entries may be categorized in a group. Depending on the number of page entries that you have access to, the page entries can appear at the top level (not in any group folder) on the springboard. If you have only one page entry in a group, then that page entry icon appears at the top level on the springboard. However, such page entry icons appear in their respective groups on the Navigator menu.

While creating or editing a page entry or group, you can use the Springboard field to specify whether to display them on the springboard. So, not all page entries and groups may appear on the springboard. The Navigator menu may have more page entries and groups than the springboard. If a page entry appears on both the Navigator menu and springboard, then you can use either of them to open the page. For page entries that don't appear on the springboard, use the Navigator menu to open those pages.

Navigator and Springboard Configuration

Use the Navigation Configuration page to configure the Navigator and springboard. You can do the following tasks on groups and page entries:

- Create
- Edit
- Show or hide
• Duplicate predefined page entries
• Reorder

Home Page Configuration

Use the Home Configuration page to configure the icons for infolet pages in the page control on the home page. You can rename these icons, change their visibility settings, and reorder them.

Related Topics
• Sign In and Get Started
• Set Profile Option Values
• Overview of Sandboxes

Configure Navigation

Create Groups and Page Entries for Navigation

Watch video

Use the Navigation Configuration page to create groups and page entries for configuring the Navigator and springboard.

You can do either of the following:
• Create a group and then create a page entry in that group.
• Create a page entry in an existing group or at the top level (not in any group).

Before You Start

Following are the prerequisites:

1. From the Navigator menu, select Configuration > Structure.
2. Activate a sandbox. If you're not in an active sandbox, click Edit in the Structure work area. You're prompted to activate a sandbox.
   Tip: If you're already in an active sandbox, then the Edit button doesn't appear in the Structure work area.
3. If prompted, select a context layer to determine the scope of users that your changes affect.

After you complete your changes, you can preview and test the changes, and then publish the sandbox to make your changes available to users.

Create Groups and Page Entries

To create a group or a page entry:

1. On the Navigation Configuration page, click Create, and select Create Group or Create Page Entry.
2. Enter a name for the group or page entry.
3. Search and select an icon for the group or page entry.
4. If you're creating a page entry, select the group in which you want to place the new page entry.
5. Select Yes, No, or EL Expression in the Show on Navigator field:
   - Yes: The group or page entry appears on the Navigator. It can also appear on the springboard, depending on what you select in step 6.
   - No: The group or page entry doesn't appear on the Navigator and springboard.
   - EL expression: The evaluation of the EL expression decides whether the group or page entry will appear on the Navigator and springboard.
6. If you have selected EL Expression for the Show on Navigator field, click the Edit icon next to the Show on Navigator list, and enter a value or expression. Don’t include spaces or double quotes in the EL expression.
7. For a page entry, if you have set the Show on Navigator field to Yes or EL Expression, set the Show on Springboard field. The value of this field is evaluated to determine if the page entry will actually appear on the springboard.
   - Yes: The page entry appears on the springboard.
   - No: The page entry doesn't appear on the springboard.
   - EL expression: The evaluation of the EL expression decides whether the page entry will appear on the springboard.
8. If you're creating a group, then click Save and Close. If you're creating a page entry, then perform steps 8 to 10 instead of this step.
9. Select Yes or No in the Mobile Enabled field to specify whether the page entry will be available for mobile devices when the home page layout is set to News Feed. This setting isn't applicable for the panel or banner layouts.
10. Select any of the following link types for the page entry:
    - An application page.
    - A dynamic URL of an external page (outside your application) where the host, port, or context root might change.
    - A Static URL of an external page (outside your application) where the host, port, or context root doesn't change.
11. Based on the link type, specify the required details to configure the link.
12. Click Save and Close.
13. Verify that the new group or page entry that you created appears on the Navigator menu in the sandbox, and then publish the sandbox to make your changes available to users.

Related Topics
- Configuring the Navigator and Springboard

Configure Links for Page Entries

While creating a page entry or editing a page entry that was already created for the Navigator and springboard, you can determine what the page entry links to.

Use the Create Page Entry or Edit Page Entry page to link a page entry to any of the following link types:
- Your application page.
• A dynamic URL of an external page (outside your application) where the host, port, or context root might change.

You can determine the host and port details, which a dynamic URL starts with, from a lookup based on the application name.

• A static URL of an external page (outside your application) where the host, port, or context root doesn't change. Static URLs don't require lookups.

• A secure token URL of a partner application, to which secure tokens are added. Tokens contain identity and security information about users, and can be used to authenticate them without additional authentication requirements, for example, user name and password.

Link to Application Pages
To link a page entry to one of your application pages:

1. Select the Application Page link type.
2. Enter the focus view ID of the target page.
3. Select the name of the web application.

You had entered this application name while creating this third party application using the Setup and Maintenance work area.

4. For secure access to the target application page from the page entry, provide the secured resource name and the name of the policy store's application stripe. An example of a secured resource name is oracle.apps.view.pageDefs.CaseList_Form_Attach_UIShellPagePageDef. When a user clicks the link, the application checks the secured resource and the Lightweight Directory Access Protocol (LDAP) policy store. Then, the application determines whether the user has the privilege to view the page.

You can get the application stripe from the jps.policystore.applicationid parameter in the application's weblogic-application.xml file. Examples of application stripes are crm, fscm, and hcm.

5. If the page takes parameters, then you can enter a semicolon-delimited string of name=value pairs (for example, org=m1;context=s1) in the Page Parameters List field.

You can use expression language (EL) to specify the parameters. If the EL evaluates to an object, the toString value of that object is passed as the value of the parameter. An application page may display or act differently based on the parameters that are passed in. For example, if you're opening a page from one group on the springboard or Navigator, the parameter might be set to status=Open. Whereas, if you're opening the page from another group, the parameter might be set to status=Closed.

Tip: If there is another page entry that links to the same application page, then you can enter the same details for all fields, except parameters. On the Navigation Configuration page, open the existing page to view the details, such as Focus View ID, Web Application, and Secured Resource Name. Then, fill in the fields for the new page entry on the Create Page Entry or Edit Page Entry page.

Link to Dynamic URLs
You can link a page entry to an external website or application that has a frequently changing host, port, or context root. Instead of updating the link to each application, you can update the details of the web application registration. This change affects all page entries that contain dynamic links pointing to that web application.

For example, suppose you want to link to a test version of an application. So, you use the dynamic URL link type. When you move the application from test to production environment, just change the host and port details of the web application registration. This change affects all page entries that contain dynamic links pointing to the web application.
As a prerequisite, use the Manage Third-Party Applications task to register the web application. The application details that you enter using this setup task will be used while configuring dynamic URL links for page entries.

To link a page entry to a dynamic URL:

1. Select the Dynamic URL link type.
2. Specify the name of the web application and the destination for web application.

For example, suppose you want to link to a complete URL: http://example:9011/myApp/faces/Page1.

Then you can do the following:

- From the Web Application list, select myApp. This list displays the application names that were added using the Manage Third-Party Applications task. Once you select the application name, the full URL that was added for this application in the setup task (for example, http://example:9011/myApp/) is automatically used as the start of the dynamic URL.
- Enter the destination for the web application, for example, /faces/Page1. This value is appended to the full URL.

After linking a page entry to a dynamic URL, when you click the page entry, the target page opens in a new browser window or tab.

Link to Static URLs

You can link a page entry to an external website or application that has a constant host, port, or context root.

To link a page entry to a static URL:

1. Select the Static URL link type.
2. Enter the URL destination. The URL must start with http:// or https://.

For example, you can use a static URL to link to http://www.oracle.com.

Link to Static URLs with Secure Destinations

To link a page entry to a secure token URL of a partner application (that is, outside your application):

1. Select the Static URL link type.
2. Select Secure Destination.
3. Select the name of the web application.
4. Enter the destination for the web application. An HTTPS protocol is required to access the application.
5. Enter the name of the secure token. All secure tokens have a predefined lifetime, and they expire after that duration. So, users must refresh the page to regenerate the tokens.

The application validates the secure token and uses it to authenticate web services within the end user context. Using this mode of modified access, a partner can directly perform an action or display information to the specific user without any additional authentication.

Related Topics

- Create Pages for Hosting Third Party Applications
- Register a Third-Party Application
Edit and Reorder Groups and Page Entries for Navigation

Use the Navigation Configuration page to edit and reorder the existing groups and page entries.

Before You Start

Following are the prerequisites:

1. From the Navigator menu, select Configuration > Structure.
2. Activate a sandbox. If you’re not in an active sandbox, click Edit in the Structure work area. You’re prompted to activate a sandbox.
   Tip: If you’re already in an active sandbox, then the Edit button doesn’t appear in the Structure work area.
3. If prompted, select a context layer to determine the scope of users that your changes affect.

After you complete your changes, you can preview and test the changes, and then publish the sandbox to make your changes available to users.

Edit Groups and Page Entries

Follow these steps:

1. On the Navigation Configuration page, click the name link for the group or page entry.
   Tip: You can use the search panel on the Navigation Configuration page to find the group or page entry you want to edit.
2. On the Edit Page Entry page or the Edit Group page, make the required changes.
3. Click Save and Close.

You can make the following changes to a group or page entry:

- Rename a group or page entry.
  Note: If a group or page entry was created using a different tool, then you can’t change its name using the Navigation Configuration page.
- Change the icon for a group or page entry.
  ◦ If a page entry was created using a different tool, then you can’t change its icon using the Navigation Configuration page.
- For a page entry, change the group in which the page entry is placed.
- Change the Show on Navigator property for the group or page entry.
- Change the Show on Springboard property for the page entry.
- Change the Mobile Enabled property for the page entry.
- For an administrator-defined page entry, change the settings for link configuration.
- Delete page entries that were created using the Navigation Configuration page.
- For a predefined page entry, use Create Duplicate on the Edit Page Entry page to create a duplicate page entry. You can then edit the duplicate page entry as you want, for example, you can place the duplicate page entry in a different group or at the top level.
For groups with associated quick actions, use the Quick Actions tab to create more quick actions or make changes to the existing ones, such as rename, show or hide, and reorder them.

**Edit Page Entries with Tabs**

Some pages (for example, Security Console) have tabs. Each tab is a task flow. To edit tabs, click the Tabs tab on the Edit Page Entry page. You can make the following changes:

- Click the tab name to rename it.
- Click the tab icon to search and select another icon for the tab.
- Click the Visible field for the tab, and change the option to show or hide the tab
- Use the Move Up and Move Down icons to adjust the relative position of the tabs within the page

**Edit Page Entries with Panel Tabs**

Some pages have panel tabs. To edit panel tabs, click the Panel Tabs tab on the Edit Page Entry page. You can make the following changes:

- Click the panel tab name to rename it.
- Click the panel tab icon to search and select another icon for the panel tab.
- Click the Visible field for the panel tab, and change the option to show or hide the tab.
- Use the Move Up and Move Down icons to adjust the relative position of the panel tabs within the page

**Reorder Groups and Page Entries**

Use the Move Up and Move Down icons on the Navigation Configuration page to reorder groups and page entries. For page entries, you can use the Move To icon to move page entries to different groups or to the top level.

After completing your changes, verify and test the changes in the sandbox, and then publish the sandbox to make your changes available to users.

**Related Topics**

- Configuring the Navigator and Springboard
- Quick Actions
- Configure Quick Actions

**Examples of EL Expressions for Configuring Navigation**

You can use EL expressions to configure navigation in the application such as, to show or hide the navigator menu items, and the icons for infolet pages in the page control on the home page. The following scenario shows how you might use EL expressions.

**EL Expressions for Configuring the Navigator Menu Items**

Use EL expressions to specify whether a group or page entry should appear on the Navigator and springboard. The evaluation of the EL expression decides whether the menu items are displayed for a user.

The following table provides examples of how you can use EL Expressions to show or hide groups and page entries from the Navigator menu or springboard for specific users:
### Who can see the group or page entry

<table>
<thead>
<tr>
<th>EL Expression and Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>Only users having any of the specific roles</td>
</tr>
<tr>
<td><code>#{securityContext.userInRole['&lt;RoleName&gt;']}</code></td>
</tr>
<tr>
<td><code>#{securityContext.userInRole['ORAFND_APPLICATION_ADMINISTRATOR_JOBORA_PER_EMPLOYEE_ABSTRACT']}</code></td>
</tr>
<tr>
<td>Only users not having any of the specific roles</td>
</tr>
<tr>
<td><code>#{!(securityContext.userInRole['&lt;RoleName&gt;'])}</code></td>
</tr>
<tr>
<td><code>#{!(securityContext.userInRole['ORAFND_APPLICATION_ADMINISTRATOR_JOBORA_PER_EMPLOYEE_ABSTRACT'])}</code></td>
</tr>
<tr>
<td>Only users having all of the specific roles</td>
</tr>
<tr>
<td><code>#{securityContext.userInAllRoles['&lt;RoleName&gt;']}</code></td>
</tr>
<tr>
<td><code>#{securityContext.userInAllRoles['ORAFND_APPLICATION_ADMINISTRATOR_JOBORA_PER_EMPLOYEE_ABSTRACT']}</code></td>
</tr>
<tr>
<td>Only users not having all of the specific roles</td>
</tr>
<tr>
<td><code>#{!(securityContext.userInAllRoles['&lt;RoleName&gt;'])}</code></td>
</tr>
<tr>
<td><code>#{!(securityContext.userInAllRoles['ORAFND_APPLICATION_ADMINISTRATOR_JOBORA_PER_EMPLOYEE_ABSTRACT'])}</code></td>
</tr>
<tr>
<td>Only users having access to specific resources, for example, the Social work area</td>
</tr>
<tr>
<td><code>#{securityContext.userGrantedResource['resourceTypeFNDResourceType;resourceName FNDLaunch Social_Menu;action= launch']} </code></td>
</tr>
</tbody>
</table>

Don’t include spaces or double quotes in EL expressions.

**Note:** Use an EL expression that can be evaluated from any page because the Navigator menu is used on all pages. So, don’t use product-specific EL expressions.

### Set the Availability Duration of Your Third-Party Application

Let’s say you integrate a registered third-party application in your Oracle Cloud Application. You want to secure this third-party application and make it available for users only for a specific duration. For that, you can set the validity duration using the Token Relay Validity Duration profile option and secure your application using JSON Web Token (JWT). Based on the value of this profile option, your third-party application will be available to users for a specific duration. After this duration, users must refresh the page to view it again. Remember that the minimum availability duration is six minutes. So for example, even if you set the availability duration as two minutes (120 seconds), your third-party application will be actually available to users for six minutes.

Before you start, make sure that you have activated a sandbox.

**Set the Validity Duration**

1. In the Setup and Maintenance work area, go to the Manage Administrator Profile Values task in the Application Extensions functional area.
2. Search for the **Token Relay Validity Duration (FND_TOKEN_VALIDITY_DURATION)** profile option.
3. In the FND_TOKEN_VALIDITY_DURATION: Profile Values section, click the **New** icon.
4. From the Profile Level list, select Site.
5. In the Profile Value field, enter the time in seconds.
6. Click Save and Close.

Secure Your Third-Party Application

1. Click Navigator > Configuration > Structure.
2. Select the page entry that links to the third-party application you want to secure.
3. From the Link Type list, select Static URL.
4. Select the Secure Destination check box.
5. Select the name of your application.
6. Enter the destination for the application. An HTTPS protocol is required to access the application.
7. In the Secure Token Name field, enter the name of the parameter (for example, jwt) that supports using the secure token.
8. Click Save and Close.

Related Topics
- Manage Pages Hosting Third Party Applications
- Register a Third-Party Application

FAQs for Configuring Navigation

Why can't I edit the Structure page entry or the Tools group?
While configuring the Navigator and springboard, you can't:
- Override the Navigator setting for the Structure page entry and Tools group. The default setting is Yes, so the Structure page entry and Tools group always appear on the Navigator and springboard.
- Move the Structure page entry to a different group or to the top level. The default group is Configuration, so the Structure page entry always appear in the Configuration group.

How can I display the missing springboard icons and Navigator menu items for the users?
Try these solutions:
- If the user can see the group or page entry in the Navigator, but not on the springboard, then enable the offering associated with the group or page entry.
- If the user can't see the group or page entry in both the springboard and the Navigator, then assign the required security privileges to the user.
- Check whether the group or page entry is hidden from the springboard or Navigator using the Structure work area. Use this work area to view the visibility settings of the group or page entry, and if it is hidden, change these settings to display it.

Related Topics
- Configure Offerings
- Overview of Configuring Offerings
Configure Home Page

Configure Home Page Navigation

Use the Home Configuration page to configure the icons for infolet pages or other configurable pages in the page control on the home page.

Before You Start

Following are the prerequisites:

1. From the Navigator menu, select Configuration > Structure.
2. Click the Home Configuration tab.
3. Activate a sandbox. If you're not in an active sandbox, click Edit in the Structure work area. You're prompted to activate a sandbox.

   Tip: If you're already in an active sandbox, then the Edit button doesn't appear in the Structure work area.

If prompted, select a context layer to determine the scope of users that your changes affect. After you complete your changes, you can preview and test the changes, and then publish the sandbox to make your changes available to users.

Define Settings

You can rename icons for infolet pages and other configurable pages in the page control, change their visibility settings, and reorder them. On the Home Configuration page, you can:

- Click the infolet name or any other configurable page name to rename it.
- Click the Visible field for an infolet or any other configurable page to change its visibility setting. You can show or hide the icon for these pages in the page control on the home page. You can select one of the following options:
  - Yes: The icon appears in the page control.
  - No: The icon doesn't appear in the page control.
  - EL expression: The evaluation of the EL expression decides whether the icon appears in the page control.
- Click the Default View field for an available configurable page to specify whether the page should be set as the default home view. You can select one of the following options:
  - Yes: The page is set as the default home view.
  - No: The page isn't set as the default home view.
  - EL expression: The evaluation of the EL expression decides whether the page is set as the default home view.

Note: Only specific configurable pages, such as Quick Actions, are available for you to set as the default home view. When you click the Default View field for such pages, you get the options to select Yes, No, or EL Expression. These options aren't available for other pages that you can't set as the default home view.
• Use the **Move Up** and **Move Down** icons to adjust the relative positions of the icons for the infolet pages or other configurable pages in the page control on the home page.

You can use profile options to define settings for the *filmstrip*, which you can find above all pages:

• To enable users to use the filmstrip, set the **Springboard Strip Enabled** profile option (**FND_USE_FILMSTRIP**) to *Yes*.

• If the **FND_USE_FILMSTRIP** profile option is set to *Yes*, then you can display the filmstrip as expanded by default. To do so, set the **Springboard Strip Expanded** profile option (**FND_EXPAND_FILMSTRIP**) to *Yes*. A user can still collapse or expand the strip on any page, and when done, this profile option is set by default for subsequent sessions of that user.

**Related Topics**

• Set Profile Option Values
• Overview of Sandboxes
• Create and Activate Unified Sandboxes

**Define Home Page Appearance**

Use the Appearance work area to first select your home page layout and then define its display settings. Use the Themes tab to select your default home page layout as panel, banner, or news feed. And then use the Home Page Display tab to define the display settings of your home page. Based on the home page layout you set using the Themes tab, the options available on the Home Page Display page may vary.

Let’s see how to define the home page appearance. But before you start, activate a *sandbox*.

**Define Home Page Layout**

1. Click **Navigator > Configuration > Appearance**.
2. Click the Themes tab.
3. From the **Themes** list, select a predefined or saved theme.
4. From the **Default Home Page Layout** list, select **Panel**, **Banner**, or **News Feed**.
5. Click **Apply**.

If you selected a predefined theme, enter a theme name, and click **OK** to create another theme with your changes. If you selected a saved theme, your theme changes are directly applied to your application.

**Define Display Settings for Home Page with Panel or Banner Layout**

If you selected the panel or banner layout on the Themes page, follow these steps to configure the display of information in the various sections of your home page:

1. Click the Home Page Display tab.
2. Select one of these options to display on the home page:
   - **Social**: Displays social networking content, such as the number of followers.
   - **Announcements**: Displays employee announcements.
   - **Cover image**: Displays the image for the main panel or banner, which you specify on the Themes page.
   - **None**
3. Specify whether to display the photo in the main panel or banner of the home page from the social network profile or from HCM.
4. Click Apply.

Define Display Settings for Home Page with News Feed Layout
If you selected the news feed layout on the Themes page, follow these steps to configure the display of information in the various sections of your home page:

1. Click the Home Page Display tab.
2. In the Name column of the table, click any section name to rename it.
3. Click the Visible field for a section to change its visibility setting. You can show or hide the section on the home page:
   - Yes: The section appears on the home page.
   - No: The section doesn't appear on the home page.
   - EL expression: The evaluation of the EL expression decides whether the section appears on the home page.
4. In the Order column, use the Move Up and Move Down icons to adjust the relative positions of the sections on the home page.
5. Click Apply.

Related Topics
- Manage Themes
- Overview of Configuring Themes and Home Page Settings
- Create Themes
- Overview of Sandboxes

Create and Enable an Announcement
You can create, edit, or delete company announcements. After you create an announcement, you can enable it to display on your home page.

Create an Announcement

1. From the Navigator, select Tools > Announcements.
2. Click Create.
3. Specify the details, such as subject, start date, and end date.
4. Select a category. If you select User-Defined, a text box appears, where you can provide additional details.
5. Select any of these options:
   - Predefined: Select an image from the list.
   - File: Browse and select a file from your local computer.
   - URL: Enter a full URL for the image

   Note: Make sure your image size is 776x437 px or larger to avoid image distortion.
6. Add the content in the text box. You can format your text using the formatting options.
7. Click Save and Close.
Your changes on the Announcements page apply immediately to all users, even if you saved your changes while a sandbox is active and not yet published.

Edit or Delete an Announcement

1. From the Navigator, select **Tools > Announcements**.
2. Select the announcement that you want to edit or delete.
3. Edit the announcement details or click **Delete**.

Enable an Announcement on the Home Page

After you create or edit an announcement, you can determine whether to display the announcement on your home page.

1. From the Navigator, select **Configuration > Appearance**.
2. Click the **Home Page Display** tab.
3. Based on your home page layout, use any of these options:
   - **Panel** or **Banner**: Select Announcements from the home panel options, and then click **Apply**.
   - **News feed**: Select **Yes** for News and Announcements, and then click **Apply**.

Your default home page layout also determines how the home page displays the announcement.

- **Panel** or **Banner**: The home page displays only the announcement content, not the subject or image.
- **News feed**: The home page displays the entire announcement along with the subject and image in the News and Announcements section.

Related Topics

- Overview of Configuring Themes and Home Page Settings
- Why can't I see announcements on the Home page

FAQs for Home Page Configuration

Why can't my users see social networking content on the panel of the Home page?

That could be because of these settings in the Appearance and Structure work areas.

- Home page display setting in the Appearance work area: Make sure you selected **Social** as the home page display option.
- Visibility setting in the Structure work area: Make sure you set the **Show on Navigator** field for the Social page entry to **Yes** or add a valid EL expression. The evaluation of the EL expression determines whether social content is displayed for a user. To check that, validate the EL expression. If your validation result is **True**, the social content will display for your user.

Deep Links

You can use deep links to open pages without navigating through the menu structure. Deep links come in handy if you have corporate internal portals and you want to enable direct navigation from the portals into the Oracle Fusion...
Applications. For example, you can enable direct navigation to the My Team page. Use the Deep Links work area to view a complete list of the available deep links.

You can simply copy a URL from the Deep Links page and paste it in your external portals as is. You don't even need to know the URL format.

**Login and Security**

Users with the View Administration Link (FND_VIEW_ADMIN_LINK_PRIV) privilege can view the Deep Link menu item in the Navigator. If you click a deep link URL from an external application and have not yet signed in, you're automatically redirected to the Sign In page.
8 Help Content Management

How You Manage Different Types of Help

You can edit the help content that comes with your application, and you can add your own help to the help windows. For example, you might want to add links to company policies or best practices. Let's look at the different types of help that you can manage.

<table>
<thead>
<tr>
<th>Help Type</th>
<th>How You See This Help</th>
<th>How You Edit or Add to It</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fields and other UI elements</td>
<td>Help text appears when you hover over or click certain UI elements on the page.</td>
<td>You can edit this text or add your own text to other UI elements using the Page Composer or User Interface Text tool.</td>
</tr>
<tr>
<td>Help windows</td>
<td>Many pages have help icons. Click these icons to open help windows that contain help text, links, or both.</td>
<td>You can use the User Interface Text tool to edit the help text. You can add or remove links by clicking the Manage Help Content link in the help window. In some cases, the same help window appears on more than one page. So whatever you do to the help window on one page applies to the same help window in other places.</td>
</tr>
<tr>
<td>Getting Started work area</td>
<td>Select Getting Started from the Navigator to open this work area and review the videos and other information for new users.</td>
<td>You can edit or add to this content by clicking the Edit Getting Started link on these pages.</td>
</tr>
</tbody>
</table>

What Else You Can Do

Here's what you can do with all the help that you or other people added to help windows and the Getting Started work area:

- If you add a lot of links in help windows, you might want to manage your help in a central place. You can use the Manage Help Content task in the Setup and Maintenance work area to view and edit all of the help that anyone added.
- If you create help in a test environment, you can migrate all added help by exporting and then importing a configuration package.
- If you want people to search all added help, make sure that the Help search category is enabled for global search.
Manage Help for Fields and Other UI Elements

Help text might appear when you hover over or click certain UI elements on the page. For example, you might see a help note when you put your cursor in a specific field, or hover over an icon.

How You Go About It

- Use Page Composer to edit, create, or delete hint text that appears when you hover over or click certain UI elements, for example buttons, icons, fields, and check boxes. Open the properties of the UI element to define the help text in the shortDesc field. Not all types of help on UI elements can be created or changed using Page Composer.
- Use the User Interface Text tool to edit the text for any type of help for UI elements, including informational text in help windows. You usually use this tool to make bulk changes, for example to change a phrase wherever it appears in any UI label, help for UI elements, messages, and so on.

Why can't I see the Manage Help Content or Edit Getting Started link?

You see the Manage Help Content link in help windows if the Help Content Management feature is enabled for any offering and if you have the Manage Help Content (ATK_CUSTOMIZE_HELP_TOPICS_PRIV) privilege. You need this same setup to see the Edit Getting Started link in the Getting Started work area too.

Help Windows

Add Your Content to Help Windows
The easiest way to add help is to start in the help window where you want the help link to appear. You can add links to web pages, upload files, or create help pages in a text editor.

1. Go to the page where you want to add help and click the help icon. If you don't see any help icons, click your user image or name in the global header and select Show Help Icons.
2. In the help window, click the Manage Help Content link.
3. In the Manage Help Content dialog box, click Create.
4. Select a help type:
   - Text: Use a rich text or HTML editor to enter your help content. You can include images, videos, and links.
   - File: Upload a file of any type. Your file opens when users click the link in the help window.
   - URL: Enter the full URL to a website or a file of any type, for example a YouTube video.
5. Select a help security group if you need to restrict access to your help content. Only users with roles in the security group will be able to see the help. The predefined Secured group includes all internal employees and contingent workers.
6. Set the status as Active if you want people to see your help in the help window. Or, you can inactivate your help so that it doesn't appear until you activate it later.
7. Enter the title, which is the text of the link in the help window.
8. If your help type is File, enter a description which users might see after they click the link on the help window, depending on their browser setting.
9. Select another language if your content isn't in American English. In help windows, users see added help in the language they're using for the application.
10. Select a country if your help is targeted at a specific country. Based on what users set in the Territory field for their regional preferences, they see generic added help plus added help tagged for their country. There's no impact on predefined help.
11. If your help type is Text, enter your content in the text editor. Or you can click the Source Code Editing Mode icon to switch to an HTML editor.
12. Save your work. Be aware that you won't be able to see the new link in the Manage Help Content dialog box or the help window if you set it up for a language, country, or security group that doesn't apply to you.
13. To check your work, click your link in the help window.

Tip: Help that you add using the Text or File type has a unique URL. So, you can link to it from other help or web pages. Updates to the application won't affect the URL or your help content.

Related Topics
- Create Groups to Limit Access to Added Help

How can I add YouTube videos to help windows?

Create help using the URL for the YouTube video.

1. Find the video in YouTube.
2. Click Share.
3. Click Embed.
4. Copy the URL within the embed code, for example https://www.youtube.com/embed/<unique ID>.

Note: Make sure that your copied URL starts with https.
5. Open the help window and click **Manage Help Content**.
6. Click **Create**.
7. Select **URL** as the help type.
8. Paste your copied URL in the **URL** field.
9. Enter other information, and click **Save and Close**.

Add Existing Help to Another Help Window

You can use the same added help in multiple help windows. Let’s say you already created some help content for a Create page, and now you want to add it to the corresponding Edit page too.

1. Open the Edit page and click the help icon.
2. In the help window, click the **Manage Help Content** link.
3. Click the **Select and Add** icon in the Added Help section.
4. Find the help you want to link to and select it in the search results.
5. Click **Add**.

   **Note:** When you use the same help in multiple windows, be aware that what you do in one window affects the other windows too. For example, if you change the content or status, the changes apply to all help windows where you added the help. Similarly, if you delete the help in one window, it’s removed from the other windows too.

Determine Which Links Appear and in What Order

Click the **Reorder** button on the Manage Help Content dialog box to set the order for links to the help that you added. Your links always show up first in the help window, before any predefined help. When you create help or edit the help that you added, you set the status so that the link is either shown or hidden in the help window.

To show or hide predefined help:

1. In the Manage Help Content dialog box, select a title in the Predefined Help section.
2. Click the **Change Status** button.
3. In the Change Status dialog box, select **Active** or **Inactive**.
4. Click **Save and Close**.

If multiple help windows link to the same predefined help, the status you set here affects all those windows.

Getting Started Work Area

Manage Getting Started Content

The Getting Started **work area** provides pages of information to introduce new users to the application. You can edit the predefined pages or add your own, so that people get content specific to your organization.

1. Click **Navigator > Getting Started**.
2. Click a link to open the set of pages you want to change.
3. Click the **Edit Getting Started** link.
Here's what you can do:

- Create and edit pages using a rich text or HTML source code editor.
- Reorder the pages.
- Activate or inactivate any page.

**Tip:** To hide the link to a set of Getting Started pages, inactivate all pages within that set. Users with access to edit Getting Started pages can still see the link, but everyone else can't. If you inactivate all sets except one, then users land on the first page of the active set when they open the Getting Started work area.

- Delete added pages (not predefined ones).

### How can I add YouTube videos to Getting Started pages?

Create or edit a page in the Getting Started work area, and include a piece of code from YouTube.

1. Find the video in YouTube.
2. Click the **Share** button.
3. Click the **Embed** button.
4. Copy everything in the text box.
5. Back in the application, open the Getting Started work area.
6. Open the set of pages you want to add the video to.
7. Click the **Edit Getting Started** link.
8. Open an existing Getting Started page or create a new one.
9. Click **Source Code Editing Mode** in the toolbar.
10. Paste in the code you copied from YouTube.
11. Click **Save and Close**.

### Manage All Added Help Content

Use the **Manage Help Content** task to review all the help that's been added across all help windows and in the Getting Started work area. You can search, for example, by title, status, or who last updated the content. From the search results, you can edit the content, delete it, or change its status.

You can also create content on this page. But then you should put the help on a help window so that people can find it where they need it most. The easiest way to do that is to go to the page where you want the help to appear, open the help window, and select and add your new topic. Or, you can search for the help window here using the **Page or Section** list, then select and add the new help.

### Search for and Manage Help

Here's what you do:

1. In the Setup and Maintenance work area, go to the **Manage Help Content** task in the Application Extensions functional area.
2. On the Manage Help Content page, search for added help.
3. Select search results and do what you need, for example, to change status or delete.
Select and Add Help to a Help Window

You can put added help on any help window.

1. On the Manage Help Content page, select a value for the **Page or Section** list in the Search section. The page or section value represents a help window, in this case, the one where you want to add a link.
2. In the Search Results section, click the **Select and Add to Help Location** icon.
3. On the Select and Add page, find the help you want to link to.
4. Select the title and click **Add**.

**Related Topics**

- Update Existing Setup Data

Page or Section Values

The page or section value tells you where people can click a help icon to open a help window. Use this value on the Manage Help Content page to search for help that was added to a specific help window, or to add more help to that window. Let’s see how we can figure out where a help window is, based on its page or section value.

Where You Find Help Windows

First, you need to know where you might find help windows. In most cases, they’re next to page or section titles, but you might also find them in other places:

- On tabs, dialog boxes (windows), **panel tabs**, **dashboards**, dashboard regions, and even desktop-integrated Excel workbooks
- In the Setup and Maintenance work area, for example within the table of tasks on the Setup page
- In the Offerings work area, for example within the table on the Edit Features page
- In the Learn More column in the New Features work area

How to Read the Values

The page or section value reflects how you navigate to the help window. For example, **Process Details window, Output tab** doesn’t mean that the help window is in two different places. It’s in the Output tab within the Process Details dialog box.

For the Setup and Maintenance, Offerings, and New Features work areas, the page or section value tells you the object that the help window is for. These values end with offering, functional area, feature, task, or task list. So for example, **Set Help Options task** refers to the help window that you see for the Set Help Options task, on the Setup page.

When the Same Help Window Is on Multiple Pages

In some cases, a page or section value does represent a single help window that appears on multiple pages. Here are a couple of examples:

- **Create and Edit Message pages** means that same help window is on both the Create Message and Edit Message pages. Note the plural **pages**.
• **Process Monitor section** means that this help window is in a section that appears on multiple pages. You can usually tell if you see something like a section or tab name without a page or window name in front of it.

**Related Topics**
- How Functional Setup Manager Components Work Together
- Guidelines for Using Desktop Integrated Excel Workbooks

## Copy Added Help for Migration

If you need to make a copy of all the help that's been added, say for testing, migration, or other purposes, then you're in luck. In the Setup and Maintenance work area, you can create a configuration package then use the export and import.

Your configuration package must include one of these:

- The Application Extensions functional area
- A source implementation project that contains the Define Help Configuration task list, with these objects selected to export:
  - Help Configuration
  - Help Topic

**Related Topics**
- Export Setup Data Using Offering or Functional Area
- Export Setup Data Using Implementation Project
Glossary

accounting flexfield
The structure that determines the chart of accounts, including the number and order of the individual segments, as well as assigning the value sets to the segments.

action
The kind of access, such as view or edit, named in a security policy.

ADF
Application Developer Framework. A set of programming principles and rules for developing software applications.

analysis
A selection of data displayed in one or more views, such as a table or chart, to provide answers to business questions.

analytics
Business intelligence objects such as analyses and dashboards that provide meaningful data to help with decision making.

business object
A resource in an enterprise database, such as an invoice or purchase order.

chrome
The set of visual elements (for example, header; expand and edit icons) around the perimeter of a component or task flow that enables users to act directly on the object.

configuration
A change to the predefined artifacts of the application. Configurations impact multiple users.

color
A grouping of flexfield segments to store related information.

color layer
A level that represents the scope of users impacted by configurations. For example, all users or only those who meet specific criteria.

color segment
The flexfield segment used to store the context value. Each context value can be associated with a different set of context-sensitive segments.
context-sensitive segment
A flexfield segment that may or may not appear depending upon a context. Context-sensitive segments are attributes that apply to certain entity rows based on the value of the context segment.

dashboard
A page that gives quick access to key tasks and summary information for a business process or object.

dashboard
A collection of analyses and other content that gives in-depth insight to help with business decisions.

data security
The control of access and action a user can take against which data.

descriptive flexfield
Expandable fields used for capturing additional descriptive information or attributes about an entity, such as a customer case. You may configure information collection and storage based on the context.

detailed report
A comprehensive report that provides detailed information about the subject matter. When you link a detailed report to an infolet, users can click anywhere in the infolet area to drill down to that detailed report.

enterprise
An organization having common control over one or more legal entities.

extensible flexfield
Expandable fields that you can use to capture multiple sets of information in a context or in multiple contexts. Some extensible flexfields let you group contexts into categories.

extension
A new artifact in addition to what's predefined in the application, for example a new business object or page.

feature
Business practices or methods applicable to the functional areas that enable the fine-tuning of business functionality.

filmstrip
The single strip of icons that you can use to open other pages. The strip appears between the global header and the page title.

flexfield
A flexible data field that you can configure such that it contains one or more segments or stores additional information. Each segment has a value and a meaning.
flexfield segment
An extensible data field that represents an attribute and captures a value corresponding to a predefined, single extension column in the database. A segment appears globally or based on a context of other captured information.

global header
The uppermost region in the user interface that remains the same no matter which page you're on.

infolet
A small, interactive widget on the home page that provides key information and actions for a specific area, for example social networking or your personal profile. Each infolet can have multiple views.

job role
A role, such as an accounts payable manager or application implementation consultant, that usually identifies and aggregates the duties or responsibilities that make up the job.

key flexfield
Configurable flexfield comprising multiple parts or segments, each of which has a meaning either individually or in combination with other segments. Examples of key flexfields are part numbers, asset category, and accounts in the chart of accounts.

key flexfield segment instance
A single occurrence of a key flexfield segment in a key flexfield structure instance.

key flexfield structure
The arrangement of segments in a key flexfield. In some cases, you can define multiple structures for a single key flexfield.

key flexfield structure instance
An occurrence of a key flexfield structure that shares the same order of segments as other instances of the key flexfield structure. However, each instance uses different value sets to validate the segments.

mainline metadata
The primary branch of metadata that a sandbox is published to. Once published, changes made in the sandbox become available to all users.

Navigator
The menu in the global header that you can use to open the work areas and dashboards that you have access to.

offering
A comprehensive grouping of business functions, such as Sales or Product Management, that is delivered as a unit to support one or more business processes.
**panel tab**
A tab that provides supplemental information or functionality for the page. Each panel tab is on the right side of the page, has an icon as the tab label, and slides out when you open the tab.

**performance tile**
A single, summary measure value displayed prominently with, in some cases, a trend line.

**performance tile report**
A summary report that shows data in the small infolet format. When you add a performance tile report to an infolet, users can see summary information about the subject matter.

**personalization**
A change that users make to control the look or behavior of the application. Personalizations impact only the user making the change.

**privilege**
A grant of access to functions and data; a single, real world action on a single business object.

**report**
An output of select data in a predefined format that's optimized for printing.

**role**
Controls access to application functions and data.

**run time**
The type of activities that users perform while they are in a running application.

**sandbox**
A testing environment that isolates untested code changes from the mainline environment so that these changes don't affect the mainline metadata or other sandboxes.

**scheduled process**
A program that you run to process data and, in some cases, generate output as a report.

**segment**
A segment is a single field within a flexfield and maps to a single table column in your database. When configuring a flexfield, you define the appearance and meaning of individual segments.
site layer
Application changes made in this layer affect all users.

springboard
The grid of icons on the home page that you can use to open pages.

task flow infolet
An infolet that displays summary information about a task.

tree
Information or data organized into a hierarchy with one or more root nodes connected to branches of nodes. A tree must have a structure where each node corresponds to data from one or more data sources.

tree structure
A set of guidelines or a framework applied to create a tree, include data, version a tree, or access a tree.

value set
A predefined set to validate the values that a user enters in the application. The set may be hierarchical.

work area
A set of pages containing the tasks, searches, and other content you need to accomplish a business goal.