Contents

Preface

Audience vi
Documentation Accessibility vi
Related Resources vii
Conventions viii

1 Get Started with Integrations

Understand Integrations 1-1
Use the REST APIs, Sites SDK, and Content Delivery SDK 1-4
Enable Single Sign-On (SSO) 1-5

2 Use Apps and Services in Content and Experience Cloud

Understand Cross-Origin Resource Sharing (CORS) 2-1
Understand Application Integration Framework (AIF) 2-2
Manage Custom Applications 2-5

3 Use Content and Experience Cloud with Other Services

Embed the Web User Interface 3-1
Security for Content in Other Domains 3-2
Embed Content in Other Domains 3-2

4 Integrate with Oracle WebCenter Content

5 Integrate with JD Edwards

Understand JD Edwards Content and Conversation Integration 5-1
6 Integrate with Oracle Logistics Cloud

7 Integrate Oracle Eloqua Cloud Service

Understand Oracle Eloqua Cloud Service Integration 7-1
Create the Oracle Content and Experience Cloud App in Eloqua 7-2
Install the Eloqua Menu Service App 7-3
Configure Eloqua in Oracle Content and Experience Cloud 7-3
Create an Eloqua User for Oracle Content and Experience Cloud 7-3
Use the Oracle Content and Experience Cloud App in Eloqua 7-4

8 Integrate with Oracle Business Intelligence Publisher

9 Integrate with Oracle Process Cloud Service

Understand Oracle Process Cloud Service Integration 9-1
Benefits of Oracle Process Cloud Service 9-2
Configure Documents Settings in Oracle Process Cloud Service 9-3
Enable or Disable Documents and Conversations 9-4
Quick Tour of the Documents Page 9-5
Create a Document- or Folder-Initiated Process 9-7

10 Integrate with Oracle Developer Cloud Service

About Using Developer Cloud Service 10-1
Understand Component Development 10-1
Understand Site Template Development 10-3
About Oracle Developer Cloud Service Projects 10-4
Use a Template Project 10-4

11 Integrate with Oracle Visual Builder

12 Integrate with Oracle Policy Automation
Integrate with Oracle Cobrowse Cloud Service
Preface

This document describes how to configure Oracle Content and Experience Cloud and combine it with other services to create custom integrations.

Audience

This publication is intended for Oracle Content and Experience Cloud administrators and developers who want information about integrating Oracle Content and Experience Cloud with other services to provide custom applications.

Documentation Accessibility

For information about Oracle's commitment to accessibility, visit the Oracle Accessibility Program website at http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc.

Access to Oracle Support

Oracle customers that have purchased support have access to electronic support through My Oracle Support. For information, visit http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info or visit http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs if you are hearing impaired.

Related Resources

For more information, see these Oracle resources:

- Oracle Public Cloud:
  http://cloud.oracle.com
- Administering Oracle Content and Experience Cloud
- Creating Experiences with Oracle Content and Experience Cloud
- Getting Started with Oracle Cloud
- Managing Content with Oracle Content and Experience Cloud
- Developing for Oracle Content and Experience Cloud
- What's New for Oracle Content and Experience Cloud
- Known Issues with Oracle Content and Experience Cloud
## Conventions

The following text conventions are used in this document.

<table>
<thead>
<tr>
<th>Convention</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>boldface</strong></td>
<td>Boldface type indicates graphical user interface elements associated with an action, or terms defined in text or the glossary.</td>
</tr>
<tr>
<td><em>italic</em></td>
<td>Italic type indicates book titles, emphasis, or placeholder variables for which you supply particular values.</td>
</tr>
<tr>
<td><strong>monospace</strong></td>
<td>Monospace type indicates commands within a paragraph, URLs, code in examples, text that appears on the screen, or text that you enter.</td>
</tr>
</tbody>
</table>
Get Started with Integrations

Oracle Content and Experience Cloud’s integration features make it a key component in a number of Oracle offerings and also make it easy for you to leverage the service in your own applications.

- Understand Integrations
- Use the REST APIs, Sites SDK, and Content Delivery SDK
- Enable Single Sign-On (SSO)

Understand Integrations

Oracle Content and Experience Cloud provides multiple ways to leverage its functionality, whether you want to incorporate your processes or apps into Oracle Content and Experience Cloud, or whether you want to use Oracle Content and Experience Cloud in your enterprise application.

Oracle Content and Experience Cloud

Oracle Content and Experience Cloud provides rich content and experience management features, from folder and file viewing and sharing, to conversations, to websites that deliver your message and content securely.

- Integrations with JD Edwards, Oracle Business Intelligence, and other services show that Oracle Content and Experience Cloud is a key component in a number of Oracle integrations described in the following text.
- An embeddable version of the user interface and website components for interacting with folders, files, conversations, and processes provide ready-to-use integrations. See Use Content and Experience Cloud with Other Services.
- Application Programming Interfaces (APIs) and the Software Development Kit (SDK) let you access Oracle Content and Experience Cloud functionality to create your own integrations within the service or across services. See Use the REST APIs, Sites SDK, and Content Delivery SDK.
- Single sign-on (SSO) authentication provides a seamless user experience across services. See Enable Single Sign-On (SSO).

What integrations are there?

Oracle Content and Experience Cloud is a key component in a number of Oracle integrations. With some integrations, Oracle Content and Experience Cloud is provided “out of the box” as part of the service. For others, you must enable or configure the integration.
**Note:**

A number of the integrations described in this guide require that integrated services be in the same identity domain. For that reason, those integrations work only on traditional cloud accounts.

<table>
<thead>
<tr>
<th>Category</th>
<th>Integration</th>
</tr>
</thead>
<tbody>
<tr>
<td>Middleware</td>
<td><strong>Oracle WebCenter Content</strong>: Uses Oracle Content and Experience Cloud to provide a truly comprehensive hybrid enterprise content management (ECM) integration, with a unified ECM infrastructure and security from a single vendor. It combines anywhere access from the cloud with content retention and archiving from on-premises installations. See <a href="#">Integrate with Oracle WebCenter Content</a>.</td>
</tr>
<tr>
<td>Applications</td>
<td><strong>JD Edwards</strong>: Integrates with Oracle Content and Experience Cloud allowing you to attach managed documents to transactions and collaborate through conversations. See <a href="#">Understand JD Edwards Content and Conversation Integration</a>.</td>
</tr>
</tbody>
</table>
| Software as a Service (SaaS)  | • **Oracle Logistics Cloud**: Use Oracle Content and Experience Cloud to store and manage documents. See [Integrate with Oracle Logistics Cloud](#).  
• **Oracle Eloqua Cloud Service**: Use Oracle Content and Experience Cloud to store and manage documents to help plan and execute automated marketing campaigns. See [Integrate Oracle Eloqua Cloud Service](#). |

**Note:**

This integration is available with traditional cloud accounts.
## Understand Integrations

### Category

<table>
<thead>
<tr>
<th>Platform as a Service (PaaS)</th>
<th><strong>Integration</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td><strong>Business Intelligence Publisher:</strong> Integrates with Oracle Content and Experience Cloud and offers managed folders as a destination for generated reports. See <a href="#">Integrate with Oracle Business Intelligence Publisher</a>.</td>
</tr>
<tr>
<td></td>
<td><strong>Oracle Process Cloud Service:</strong> Automate business-driven, company-specific processes, such as employee on-boarding or IT service requests, and incorporate those processes into Oracle Content and Experience Cloud. See <a href="#">Integrate with Oracle Process Cloud Service</a>.</td>
</tr>
<tr>
<td></td>
<td><strong>Oracle Visual Builder:</strong> Rapidly create web and mobile applications with minimal to no coding using an open-source, standards-based integration to develop, collaborate on, and deploy applications within Oracle Content and Experience Cloud. See <a href="#">Integrate with Oracle Visual Builder</a>.</td>
</tr>
<tr>
<td></td>
<td><strong>Oracle Developer Cloud Service:</strong> Use project templates and tools to create, test, and package your own site templates, themes, and components for use in Oracle Content and Experience Cloud. See <a href="#">Integrate with Oracle Developer Cloud Service</a>.</td>
</tr>
</tbody>
</table>

**Note:**

This integration is available with traditional cloud accounts.

### Third-party Applications

Oracle Cloud Marketplace lists applications created by partners using the integration features provided with Oracle Content and Experience Cloud.

### Custom Applications

Use options such as REST APIs, Java services, and the Application Integration Framework (AIF), to create any number of applications.

### Use Apps and Services in Oracle Content and Experience Cloud

If you want to expand the service to include your own apps or to communicate with other services:

- The open architecture for site components means you can register and deliver hosted apps and create your own components using your preferred platform. For details about how to create your own components, see [Develop Components in Developing for Oracle Content and Experience Cloud](#).

- Cross-Origin Resource Sharing (CORS) allows a web page to make requests such as `XMLHttpRequest` to another domain. If you have a browser application that integrates with Oracle Content and Experience Cloud but is hosted in a different domain, add the browser application domain to Oracle Content and Experience Cloud’s CORS origins list. See [Understand Cross-Origin Resource Sharing (CORS)](#).

- If you use REST services that do not support Cross-Origin Resource Sharing (CORS) or that require service account credentials, Oracle Content and Experience Cloud includes a proxy service. See [Configure Proxy Service Settings in Administering Oracle Content and Experience Cloud](#).

- You can use Application Integration Framework (AIF) to create your own custom applications that define the actions that are exposed in the interface, respond to...
user selections, call third-party services, and specify how the results are presented to the user. The framework supports variables and expressions and provides multiple language support. See Understand Application Integration Framework (AIF).

- You can modify the user interface and menus to provide access to your applications and features. See Manage Custom Applications.

Use Oracle Content and Experience Cloud with Other Services

The Oracle Platform as a Service (PaaS) architecture means you can leverage the Oracle Content and Experience Cloud functionality where you need it:

- Provide direct interaction with Oracle Content and Experience Cloud in another web application with the embedded version of the user interface. See Embed the Web User Interface.
- Specify a list of domains where you allow content from Oracle Content and Experience Cloud to be displayed using either the embedded interface or REST calls. See Embed Content in Other Domains.

Use the REST APIs, Sites SDK, and Content Delivery SDK

REST Application Programming Interfaces (APIs) let you access Oracle Content and Experience Cloud functionality to create your own integrations within the service or across services. The Sites Software Development Kit (SDK) includes JavaScript calls that let applications interact with the underlying site architecture. The Content Delivery SDK lets applications interact with the Content Delivery REST API to retrieve structured content, digital assets, and content layouts.

Because the REST APIs are called from JavaScript code that runs in a browser, they rely on Cross-Origin Resource Sharing (CORS) to make HTTP calls to the Oracle Content and Experience Cloud server. An administrator must register the REST API domain in the Oracle Content and Experience Cloud CORS origins list. See Understand Cross-Origin Resource Sharing (CORS).

REST API for Content Delivery

With the REST API for Content Delivery, you can interact with the content stored in Content and Experience Cloud sites. See Overview of the REST API for Content Delivery in Developing for Oracle Content and Experience Cloud. For API reference information and examples, see REST API for Content Delivery.

REST API for Conversations

With the REST API for Conversations, you can create and manage conversations in your cloud resources that enable real-time collaboration between individuals and teams and connect your business processes, enterprise applications, and content. See Overview of the REST API for Conversations in Developing for Oracle Content and Experience Cloud. For API reference information and examples, see REST API for Conversations.

REST API for Documents

With the REST API for Documents, you can manage folders, files, shares, public links, application links, and metadata collections. You can also get user information, access items in a user's home folder, and get version and resource information about the API.
See Overview of the REST API for Documents in Developing for Oracle Content and Experience Cloud. For API reference information and examples, see REST API for Documents.

REST API for Users and Groups

With the REST API for Users and Groups, you can enable collaboration among users through conversations, manage users and one-on-one conversations, configure resources and services, and manage connections and access. For an overview, see Overview of the REST API for Users and Groups in Developing for Oracle Content and Experience Cloud. For API reference information and examples, see REST API for Users and Groups.

Content Delivery SDK

The Content Delivery SDK for Oracle Content and Experience Cloud is a light-weight JavaScript wrapper that interacts with the Content Delivery REST API. This is a read-only SDK for retrieving structured content, digital assets, and content layouts that are managed in Oracle Content and Experience Cloud.

The Content Delivery SDK consists of two main modules:

- **ContentSDK**: the main entry-point object. The ContentSDK object lets you create client objects to access content based on your requirements.
- **ContentDeliveryClient**: a client object that is set up to access published content items and digital assets.

See Content Delivery SDK.

Sites SDK

The Sites SDK helps application developers communicate across inline frame and domain boundaries, get and set properties, and handle site events. See Oracle Content and Experience Sites SDK Reference in Developing for Oracle Content and Experience Cloud.

Enable Single Sign-On (SSO)

If federated single sign-on (SSO) is currently available for your Oracle Content and Experience Cloud environment, you can enable it to customize sign-in procedures. When single sign-on (SSO) is enabled, users can sign in to one domain by using corporate security credentials and access another domain without signing in again. For example, an administrator for your organization, which has two Oracle Cloud services, must provision the services to your organization, roles, and users. Your company might also have on-premises applications and cloud services from other vendors. It's important that communication between these services and applications is done in a secure fashion. With SSO, users can sign in to all of them by using the same set of credentials that your identity domain system manages.

Shared Identity Management (SIM) is part of the Oracle Cloud security infrastructure. It is designed to let Oracle Cloud services set up user authentication, web single sign-on (SSO), outbound identity federation, and web services authorization. SIM brings users, services, and applications in the cloud together in a secure manner. It's accomplished using Oracle Identity Platform Service and OAuth as an authorizing mechanism.
OAuth can provide secure access to all services in Oracle Cloud. It provides an access token for communication between services. The token is valid for a limited time and contains the security credentials for a sign-in session. It identifies the user and the user’s groups.

**Overview of SSO Configuration**

Oracle Cloud uses the SAML 2.0 standard to enable secure cross-domain communication between Oracle Cloud and other SAML-enabled sites that are located on-premises or in a different cloud. The administrator must configure SAML 2.0 SSO between Oracle Cloud and the identity provider. When SSO is enabled, the identity provider performs authentication for Oracle Cloud.

The following steps provide an overview of SSO configuration:

1. Configure Oracle Cloud as the service provider by using the SSO Configuration page in My Services. You must have the necessary metadata from the identity provider to configure Oracle Cloud.

2. Configure the identity provider that users sign into. You can download metadata or copy URLs by using the SSO Configuration page in My Services. You can then use that data to configure the identity provider. You must configure your identity provider by using the identity provider’s interface.

3. Test the configuration to verify that it works. You can use a built-in test application on the SSO Configuration page in My Services. This application provides information to help you diagnose problems. Make sure to test the configuration before you enable it.

4. Enable SSO for your users by using the SSO Configuration page in My Services.

5. Update any metadata if necessary.
Use Apps and Services in Content and Experience Cloud

Oracle Content and Experience Cloud helps you expand the service to include your own apps or to communicate with other services.

- Understand Cross-Origin Resource Sharing (CORS)
- Understand Application Integration Framework (AIF)
- Manage Custom Applications

Understand Cross-Origin Resource Sharing (CORS)

Cross-Origin Resource Sharing (CORS) allows a web page to make requests such as XMLHttpRequest to another domain. If you have a browser application that integrates with Oracle Content and Experience Cloud but is hosted in a different domain, add the browser application domain to Oracle Content and Experience Cloud's Cors origins list.

The REST APIs use CORS because they're called from JavaScript code that runs in a browser and the REST APIs and Oracle Content and Experience Cloud are hosted in different domains.

Note:

If your browser application needs to use a REST endpoint that doesn't support CORS or that needs service account credentials, you can instead register and use the endpoint via Oracle Content and Experience Cloud's integrated proxy service. See Configure Proxy Service Settings.

In general, inline frames can host content if the protocol, domain, and port of the inline frame are identical to those for the content it displays. For example, by default, an inline frame on the page http://www.example.com:12345/home.html can host content only if the content's protocol is also http, the domain is www.example.com and the port is 12345.

However, if the application is in a different domain than Oracle Content and Experience Cloud, you need to need to add the application's host machine information to the list of front channel CORS origins, back channel CORS origins, or both.

- If the request is a cross-domain request (not originating from Oracle Content and Experience Cloud's domain) that will be served by Oracle Content and Experience Cloud, you need to add a front channel CORS origin. Front channel CORS is typically useful for custom application integration. For example, the REST APIs interact with the front channel.
If the request is directly from Oracle Content and Experience Cloud to a connected client in another domain, you need to add a back channel CORS origin. For example, Oracle Content and Experience Cloud can send back-channel messages (real-time updates) to an application.

If an application gets both front-channel and back-channel communication from Oracle Content and Experience Cloud, you need to add the domain to both the front and back channel CORS origins lists.

Note:
The CORS settings apply to all Oracle Content and Experience Cloud calls (documents, social, and content as a service).

See Enable Cross-Origin Resource Sharing (CORS) in Administering Oracle Content and Experience Cloud.

Understand Application Integration Framework (AIF)

Application Integration Framework (AIF) provides a simple and effective way to integrate third-party services and applications into the Oracle Content and Experience Cloud interface.

Using AIF, you can quickly define the actions that are exposed in the interface, respond to user selections, call third-party services, and specify how the results are presented to the user. The framework supports variables and expressions and provides multiple language support.

Note:
Custom AIF applications are not applied when you access them through an applink or public link.

The definition for one or more integrations is stored in a single file in JSON format. As an administrator, you can upload the configuration file and add it to a list of available applications. Also as an administrator, you can edit and validate the configuration file directly in the interface, enable or disable the app for general use, set administration-level preferences, or delete the app.

The definition for one or more integrations is stored in a single configuration file in JSON format. The configuration file defines and manages the interactions between the app, native objects and interface elements. The configuration file includes:

- App properties including tenant and user preferences
- Actions that are exposed in the interface and the service calls they make
- How the results are presented to the user
- Interface strings with support for multiple languages

The definition for one or more integrations is stored in a single configuration file in JSON format. The configuration file defines and manages the interactions between the app, native objects and interface elements. The configuration file includes:

- App properties including tenant and user preferences
- Actions that are exposed in the interface and the service calls they make
- How the results are presented to the user
- Interface strings with support for multiple languages
An administrator adds the configuration file, enables the integration, and provides tenant and account information to get started. Also as an administrator, you can edit and validate the configuration file directly in the interface, download the configuration file, or delete the app.

To manage apps created with Application Integration Framework, sign in as an administrator, open your user menu, choose Administration, and then choose Integrations. Under Custom Actions, click Add.
From the **Applications** page, an administrator can use the following options.

<table>
<thead>
<tr>
<th>Setting</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image" alt="On Off" /></td>
<td>Enable or disable the application for users. When you enable the application, you can specify preferences for the application from the user menu, by choosing <strong>Preferences</strong> and then <strong>Applications Settings</strong>. You specify the user preferences resource in the <code>userPrefs</code> element in the configuration file.</td>
</tr>
<tr>
<td><img src="image" alt="Add" /></td>
<td>Browse local folders and files to locate and upload an application configuration file.</td>
</tr>
<tr>
<td><img src="image" alt="Info" /></td>
<td>Display the information defined for the application and specified in the <code>info</code> element of the configuration file.</td>
</tr>
<tr>
<td><img src="image" alt="List" /></td>
<td>Display the preferences resource for administrators defined in the <code>tenantPrefs</code> element of the configuration file.</td>
</tr>
<tr>
<td>Setting</td>
<td>Description</td>
</tr>
<tr>
<td>---------</td>
<td>-------------</td>
</tr>
<tr>
<td>🔧</td>
<td>Open the configuration file in the integrated JSON editor. The editor validates the syntax of the file to ensure that the file contains valid JSON code. Changes you make to the configuration file are immediately available in the enabled application.</td>
</tr>
</tbody>
</table>

**Note:**
Changes you make to the configuration file are stored only in the server copy of the file. To back up your changes, use the Download icon to save the file locally.

<table>
<thead>
<tr>
<th>Download</th>
<th>Download the file from the server to a local destination.</th>
</tr>
</thead>
<tbody>
<tr>
<td>🗑️ Delete</td>
<td>Delete the configuration permanently.</td>
</tr>
</tbody>
</table>

**Note:**
When you delete a configuration file, the deletion is permanent. The file can't be restored from the trash.

**Related Topics**
- Application Integration Framework (AIF)

In *Developing for Oracle Content and Experience Cloud*.

**Manage Custom Applications**

You can create custom applications using the Application Integration Framework (AIF). With custom applications, you can change the menu options your users will see, add pop-up dialogs as needed, call third-party services, and specify how the results are presented to the user. For details about creating applications using AIF, see Application Integration Framework Overview in *Developing for Oracle Content and Experience Cloud*.

After creating an application, you can add it to Oracle Content and Experience Cloud and manage it from the Administration interface.

1. After you sign in to the Oracle Content and Experience Cloud web application as an administrator, click **Settings** in the Administration area of the navigation menu.

2. In the **Settings** menu, click **Integrations**.

3. To add an application, under **Custom Actions**, click **Add** and navigate to the location where the configuration file that contains the application information is stored and select it.
4. After adding the application, you can enable or disable it by selecting or clearing the check box.

After you add your application and enable it, you can manage it by clicking the appropriate icon for any of the following actions:

- View information about the application. What is shown is set by the application's info application property (for example, a popup window).
- Set the preferences for the application. What is shown is set by the application's tenantPrefs property.
- Edit the application. You can alter the application's code and click Load to test the application, save the application, or cancel your edit. If an action isn't allowed in the code, an error message is displayed, describing the error.
- Download the application. You can save the application or open it using an editor of your choice.
- Remove the application. When you delete an application, it is not moved to your trash. You will need to add the application file again if you want to use it.

When an application is disabled, you can only edit the application, download it, or delete it.

To view all enabled applications, open your user menu, click Preferences, and then, in the Preferences menu, select Applications. You can view the information for the application and the preferences. The Applications option is not shown in the Preferences menu unless at least one application is enabled.
Use Content and Experience Cloud with Other Services

The Oracle Platform as a Service (PaaS) architecture means you can leverage the Oracle Content and Experience Cloud functionality where you need it.

- Embed the Web User Interface
- Security for Content in Other Domains
- Embed Content in Other Domains

Embed the Web User Interface

Embed the web user interface for Oracle Content and Experience Cloud documents into your own web applications to get access to the folder and document management features of the service.

To embed the Oracle Content and Experience Cloud user interface for documents in an inline frame, add `/embed` to any member or public folder link immediately after the `/documents` element in the URL used to populate the inline frame. For example, the following URL calls the standard user interface for documents and shows the home folder for the current user:

https://www.example.com/documents/home/nameasc

To display the home folder in the embedded user interface, use the following form of the URL:

https://www.example.com/documents/embed/home/nameasc

To open a specified folder in the embedded interface, use the `folder` element in the URL and specify the globally unique identifier (GUID) of the folder. For example, the following link opens the specified folder in the embedded user interface.

https://www.example.com/documents/embed/folder/1713A5712BE73C37891915A0127B594F/nameasc

Note:
The embedded user interface adjusts the content to fit within windows as small as 320 pixels wide. Windows smaller than 320 pixels begin to hide content on the right edge of the window.

You can also embed member links and public links to folders and specify configuration parameters on the URL that control some aspects of the browser display. For information about configuring the appearance of the embedded user interface, see Browser Configuration Parameters.
Security for Content in Other Domains

When you embed the Oracle Content and Experience Cloud web user interface, it’s enclosed in an inline frame. Security policies are enforced to minimize the security risks of hosting external sites in inline frames.

In general, inline frames can host content if the protocol, domain, and port of the inline frame are identical to those for the content it displays. For example, by default, an inline frame on the page `http://www.example.com:12345/home.html` can host content only if the content's protocol is also `http`, the domain is `www.example.com` and the port is `12345`.

An administrator can explicitly identify domains outside of the host domain and allow content from those domains to display in the embedded web user interface.

Note:

Before implementing an interface integration that uses inline frames, be sure you understand the possible security risks associated with hosting external sites in inline frames. Security measures vary between different browsers and different browser versions. For more information, see [http://www.w3.org/TR/UISecurity/](http://www.w3.org/TR/UISecurity/).

Oracle Content and Experience Cloud displays content for authorized users in the embedded interface. Supported content includes folders, member links, and views of individual files.

All unsupported content, such as public links and content from unauthorized domains, opens in a new browser tab or window (depending on the browser settings).

**Embed Content in Other Domains**

You can display content from Oracle Content and Experience Cloud within other domains. For example, you might embed the Oracle Content and Experience Cloud web user interface into your own web applications to access folder and document management features inside your application.

To allow users to embed content, enable embedded content and add domains:

1. After you sign in to the Oracle Content and Experience Cloud web application as an administrator, click **Settings** in the Administration area of the navigation menu.
2. In the **Settings** menu, click **Security**.
3. Under **Embedded Content**, select **Enabled**.
4. In the **Allowed domains** box, enter a list of permitted domains, separated by commas. Domains must be in the form `www.example.com`.
   - To restrict the domain to a particular port, include the port in the specification. For example, `www.example.com:12345`. 
If you want to allow a domain that has multiple sub-domains, you can use the wildcard character. For example, `www.example.*` includes the domains `www.example.com`, `www.example.co.uk`, and so on.

To learn about embedding the Oracle Content and Experience Cloud web user interface, see Embed the Web User Interface in *Developing for Oracle Content and Experience Cloud*. 
Integrate with Oracle WebCenter Content

Integration Oracle Content and Experience Cloud with Oracle WebCenter Content makes it easy to manage content in either environment and to share content between the two.

Need to collaborate with team members in the cloud? Users can collaborate on cloud content whether they are in the cloud, on their phone, or on-premises. This makes it possible for enterprises with investment in on-premises applications to leverage their investment while taking advantage of the rapidly expanding set of cloud offerings. Users can manage cloud content and move content between the cloud and on-premises installations. See Work with Oracle Documents Cloud Service in Oracle Fusion Middleware Using Oracle WebCenter Content, 12c (12.2.1.2.0)

When Oracle Content and Experience Cloud is enabled with Oracle WebCenter Content, you have a truly comprehensive hybrid enterprise content management (ECM) integration, with a unified ECM infrastructure and security from a single vendor. It combines anywhere access from the cloud with content retention and archiving from on-premises installations.

Setting up the integration in Oracle WebCenter Content is as easy as 1–2–3:

- Enable the component
- Configure the server connection
- Configure the host name verifier

See Configure Document Cloud Service Integration Settings in Oracle Fusion Middleware Administering Oracle WebCenter Content, 12c (12.2.1.2.0)
Integrate with JD Edwards

JD Edwards EnterpriseOne comes integrated with Oracle Content and Experience Cloud for folder and file content and conversations.

- Understand JD Edwards Content and Conversation Integration

Understand JD Edwards Content and Conversation Integration

JD Edwards EnterpriseOne integrates with Oracle Content and Experience Cloud allowing you to attach managed documents to transactions and collaborate through conversations.

With content stored in Oracle Content and Experience Cloud, you can:

- Access documents using web, desktop devices, and mobile devices
- View, search, and manage documents directly in the interface
- Collaborate through conversations, and conversations about specific transactions or documents
- Review document usage data

When you enable the integration, you can use the following from within JD Edwards EnterpriseOne:

- **User Conversations**: Start or participate in social conversations about a topic of interest within EnterpriseOne using the Conversation icon in the menu bar.

- **Contextual Conversations**: Start or participate in conversations about specific business records within EnterpriseOne. Create conversations within the context of a transaction using the Conversation icon in the header bar of the Attachment Manager.

- **Contextual Documents**: Add documents to the cloud and organize documents while still keeping them in the context of your business record by using the Document icon in the header bar of the Attachment Manager.
Integrate with Oracle Logistics Cloud

You can use Oracle Content and Experience Cloud to store and manage documents from within Oracle Logistics Cloud.

You simply configure the integration with URL, user name, and password to connect to the Oracle Content and Experience Cloud server. After authentication credentials are verified, you can upload, edit, and delete documents directly from Oracle Logistics Cloud. See https://docs.oracle.com/cloud/latest/otmcs_gs/OTMCI/OTMCI.pdf.

<table>
<thead>
<tr>
<th>Document</th>
<th>Review Status</th>
<th>Instructions</th>
</tr>
</thead>
<tbody>
<tr>
<td>BILL OF LOADING</td>
<td>IN REVIEW</td>
<td>OPTIONAL</td>
</tr>
</tbody>
</table>

Revisions
- SHP-TEST-001-BOL-01: DENIED
  Missing proper handling instructions
- SHP-TEST-001-BOL-02: PENDING

Status
- Type: GENERATE
  Value: GENERATE_GENERATED

Text Templates
<table>
<thead>
<tr>
<th>Text Template</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>HAZMAT</td>
<td>A test of hazardous material</td>
</tr>
<tr>
<td>TEST</td>
<td>A test of ship unit line text</td>
</tr>
</tbody>
</table>

Reviewers
<table>
<thead>
<tr>
<th>Subscriber</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>ESL</td>
<td>LOGISTICS</td>
</tr>
</tbody>
</table>

Subscribers
<table>
<thead>
<tr>
<th>Subscriber</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>ESL</td>
<td>LOGISTICS</td>
</tr>
</tbody>
</table>
Integrate Oracle Eloqua Cloud Service

Oracle Eloqua Cloud Service enables you to plan and execute automated marketing campaigns that include content stored and managed in Oracle Content and Experience Cloud.

- Understand Oracle Eloqua Cloud Service Integration
- Create the Oracle Content and Experience Cloud App in Eloqua
- Install the Eloqua Menu Service App
- Configure Eloqua in Oracle Content and Experience Cloud
- Create an Eloqua User for Oracle Content and Experience Cloud
- Use the Oracle Content and Experience Cloud App in Eloqua

Understand Oracle Eloqua Cloud Service Integration

Oracle Eloqua Cloud Service enables you to plan and execute automated marketing campaigns while delivering a personalized customer experience for your prospects.

You can allow Oracle Eloqua Cloud Service users to select content from Oracle Content and Experience Cloud to include in their marketing campaigns. After you enable integration in both Oracle Content and Experience Cloud and Oracle Eloqua Cloud Service, Eloqua users see Oracle Content and Experience Cloud in the Apps menu, allowing them to select any content they have access to.

For more information about Eloqua, see Oracle Eloqua Help Center.
Create the Oracle Content and Experience Cloud App in Eloqua

To create the Oracle Content and Experience Cloud app and menu service, perform the following steps in Oracle Eloqua Cloud Service:

1. Create the app:
   a. Click 🔧, and then select AppCloud Developer.
   b. Click Create App.
   c. Set the following values:
      i. Name: Enter a name for the app (for example, Oracle Content and Experience Cloud).
      ii. Description: Enter a description for the app so that users know what the app does.
      iii. Icon: Enter the URL to the icon image:
         https://<OracleContentAndExperienceCloud_Host>/documents/static/img/docs-app-icon.png
         Replace <OracleContentAndExperienceCloud_Host> with your company's Oracle Content and Experience Cloud host. For example:
         https://documents.xyzcompany.com/documents/static/img/docs-app-icon.png
      iv. Click Save.

2. Create the menu service:
   a. Under Services, click Add Services.
   b. Click Add Menu Service.
   c. Enter the same Name, Description, and Icon values you entered when creating the app.

3. Configure the visibility settings and style:
   a. User Access: Select where in Eloqua (for example, in Email or My Eloqua) you want the app to be visible in the Apps menu.
   b. Default View: Select whether the app should always be visible, or only when users are editing assets (Only visible for a selected asset).
   c. Action URL: Enter the following URL:
      https://<OracleContentAndExperienceCloud_Host>/documents/embed/folder/home/deloqua/nameasc/lyt=list/cfg=nvw,hbr,sdo
      Replace <OracleContentAndExperienceCloud_Host> with your company's Oracle Content and Experience Cloud host. For example:
d. **Content Display Layout:** Select Drawer (430x796px), otherwise the menu service page launches in a new web browser tab.

e. Click **Save**.
   
The app is installed in AppCloud developer.

## Install the Eloqua Menu Service App

1. In the breadcrumb, navigate up one level. This corresponds to the name of your new app.
2. Under **Lifecycle Setup**, click **Start App**.
3. Under **Install URL**, click the URL to start the installer for your app.
4. Click **Accept and Install**.
5. Sign out of Oracle Eloqua Cloud Service for the changes to take effect.
   
The Oracle Content and Experience Cloud app is now visible in **Apps** menu.

## Configure Eloqua in Oracle Content and Experience Cloud

1. After you sign in to the Oracle Content and Experience Cloud web application as an administrator, click **Settings** in the Administration area of the navigation menu.
2. In the **Settings** menu, click **Integrations**.
3. Under **Oracle Integrations**, select **Eloqua Cloud Service Integration** to enable the service.
4. Set the following values in the dialog box, and then click **Save**:
   
   **Service User:** Enter the name of the Eloqua user you created for Oracle Content and Experience Cloud. Use the following format:
   
   `<Company_Name>\<User_Name>`
   
   For example:
   
   XYZCompany\Cathy.Woods
   
   **Service Password:** Enter the password for the Eloqua user.
5. In the **Administration** menu, click **Security**.
6. In the **Embedded Content** section, select **Enabled**.
7. In the **Allowed domains** box, enter `*.eloqua.com`, and then click **Save**.

## Create an Eloqua User for Oracle Content and Experience Cloud

You must create an Eloqua user for Oracle Content and Experience Cloud actions. You’ll enter this user name when configuring Eloqua in Oracle Content and Experience Cloud, and all content copied from Oracle Content and Experience Cloud to Eloqua will be added by this user.
Use the Oracle Content and Experience Cloud App in Eloqua

1. While editing a marketing asset, such as an email template, click the Apps menu.
2. Click the Oracle Content and Experience Cloud app.

**Note:**
Your administrator might have named the app something else. If you can't find the app, contact your administrator.

3. Click Select Assets. You might be prompted for your Oracle login information.
4. In the Documents file picker, select one or more files you want to use in Eloqua, and then click Copy to Eloqua. Content that is classified as images is copied to the Images Components Library into a directory called Documents-Images. Content that is not classified as images is copied to the File Storage Components Library into a directory called Documents-Files.

**Note:**
- If the file you select has a file name with more than 100 characters, copying to Eloqua will fail. You must edit the file name in Oracle Content and Experience Cloud before you can copy the file to Eloqua.
- When you copy files from Oracle Content and Experience Cloud to Oracle Eloqua Cloud Service, they aren’t synchronized, so any changes made in Oracle Content and Experience Cloud won’t be copied to Oracle Eloqua Cloud Service.

If you want to upload a file to Oracle Content and Experience Cloud, click Upload, and then select a file from your computer. After uploading the file Oracle Content and Experience Cloud, you must copy it to Eloqua as described above.

5. When you’re done, click Close.

For information on additional Oracle Content and Experience Cloud features available in the file picker, see Manage Your Files and Folders in Managing Content with Oracle Content and Experience Cloud.
Integrate with Oracle Business Intelligence Publisher

When integrated with Oracle Content and Experience Cloud, Business Intelligence Publisher offers managed folders as a destination for generated reports.

For example, payroll reports can be sent to individual employee folders. Manage the folders and folder contents with the easy-to-use interface from mobile, web, or desktop devices. All of the familiar security, sharing, viewing, and collaboration features of Oracle Content and Experience Cloud are available to view and manage your reports.

See Setting Output Options in *Fusion Middleware User's Guide for Oracle Business Intelligence Publisher*.
Integrate with Oracle Process Cloud Service

When you integrate Oracle Content and Experience Cloud and Oracle Process Cloud Service, you can apply processes, such as routing and approval processes, to folders and files. Role-based processes, folder and file security, versioning, and collaboration tools such as conversations all come together seamlessly.

Note:
Integration between these services requires SSO sign-ons, so both services must be in the same identity domain.

- Understand Oracle Process Cloud Service Integration
- Benefits of Oracle Process Cloud Service
- Configure Documents Settings in Oracle Process Cloud Service
- Enable or Disable Documents and Conversations
- Quick Tour of the Documents Page
- Create a Document- or Folder-Initiated Process

Understand Oracle Process Cloud Service Integration

You can allow your users to access Oracle Process Cloud Service functionality, which lets users manage business processes in the cloud, such as document routing for approval or review.

Integrating Oracle Content and Experience Cloud with Oracle Process Cloud Service benefits document-intensive processes by organizing, managing, and restricting access to documents that must be submitted, reviewed, and approved or rejected by different roles and organizations during the business process. Conversations enable users to easily discuss things that come up during the process.

Oracle Content and Experience Cloud integrates documents and conversations with your process applications.

- Documents: Oracle Process Cloud Service provides simple file attachment functionality, but if you need something more robust to handle document-intensive processes, you can integrate Oracle Content and Experience Cloud. This service enables you to organize files into folders, manage access to each folder, and even start a process when you upload a document. For example, if you’re processing a home loan, you need to manage documents such as loan applications, employment histories, and house appraisals, making sure that the right users see the documents they need to submit, review, or approve, but they don’t get access to restricted information.
• Conversations: When you integrate conversations, users can easily discuss things that come up during the process. This provides a record of what happened, enabling you to quickly bring new stakeholders up to speed or refer back to things as necessary. Plus, the conversation tools work like the social media tools users regularly use, but with enterprise-wide security and controls. For example, if you’re working on a contract you might need to discuss some of the terms, while still making sure your discussion is confidential.

• Document- and Folder-Initiated Processes: You can automatically start a process when someone uploads a document (or folder of documents) to a chosen document folder.

You must configure settings in both Oracle Process Cloud Service and Oracle Content and Experience Cloud before users can take advantage of the integrated functionality. For an overview, see the video Integrate Oracle Documents with Oracle Process Cloud Service. For information about configuring Oracle Content and Experience Cloud, see Integrate with Oracle Process Cloud Service in Administering Oracle Content and Experience Cloud.

After both services have been configured for integration, Oracle Process Cloud Service users can take actions (such as approvals) on the files directly in Oracle Process Cloud Service. Oracle Content and Experience Cloud users can upload files into folders to initiate a workflow associated with the folder. Oracle Content and Experience Cloud site designers can create web pages with ready-to-use components that provide folder and file access, process selection and initiation, associated conversation display and interaction and much more. See Using Built-In Components in Creating Experiences with Oracle Content and Experience Cloud.

Developers can learn more about how Oracle Content and Experience Cloud integrates with Oracle Process Cloud Service in Process Cloud Service Integration with Documents and Folders in Developing for Oracle Content and Experience Cloud.

Benefits of Oracle Process Cloud Service

Although you can use email to discuss a task and send associated documents, it’s much easier to have all that content available while viewing the task and tracking a process instance. Oracle Content and Experience Cloud integrates documents and conversations with your process applications.

What are the benefits of integrating Oracle Content and Experience Cloud:

• **Documents**: Process provides simple file attachment functionality, but if you need something more robust to handle document-intensive processes, you can integrate Oracle Content and Experience Cloud. This service enables you to organize files into folders, manage access to each folder, and even start a process when you upload a document. For example, if you’re processing a home loan, you need to manage documents such as loan applications, employment histories, and house appraisals, making sure that the right users see the documents they need to submit, review, or approve, but they don’t get access to restricted information.

• **Conversations**: When you integrate conversations, users can easily discuss things that come up during the process. This provides a record of what happened, enabling you to quickly bring new stakeholders up to speed or refer back to things as necessary. Plus, the conversation tools work like the social media tools users regularly use, but with enterprise-wide security and controls. For example, if you’re working on a contract you might need to discuss some of the terms, while still making sure your discussion is confidential.
• **Document- and Folder-Initiated Processes**: You can automatically start a process when someone uploads a document (or folder of documents) to a chosen document folder.

After you integrate Oracle Content and Experience Cloud, document folders and, if enabled, conversations are automatically created for each new application. You can also enable document and conversation functionality in existing applications. You can create additional folders as necessary to further organize documents, setting access to restrict content to the appropriate users. Then your users can upload and view documents and participate in conversations while viewing the task (through icons on the Task Details page).

For a summary of the roles and responsibilities involved in configuring and using Oracle Content and Experience Cloud with Process, see Roles and Responsibilities. For general information about Oracle Content and Experience Cloud, see Oracle Content and Experience Cloud Online Documentation Library.

For a summary of the roles and responsibilities involved in configuring and using Oracle Content and Experience Cloud with Oracle Process Cloud Service, see Roles and Responsibilities in *Using Processes in Oracle Integration Cloud*.

### Configure Documents Settings in Oracle Process Cloud Service

As the administrator, you must also configure the connection between Oracle Integration Cloud and Oracle Content and Experience Cloud. You need to enter information such as the URL and sign-in credentials for your Oracle Content and Experience Cloud.

To configure the settings in Oracle Integration Cloud:

1. In the Integration Cloud navigation pane, click **My Tasks**, click **Workspace**, and then click **Administration**.
2. Click **Services**.
3. In the Oracle Content and Experience Cloud section, enter the following information:
   
   • **URL**: the web address of your Oracle Content and Experience Cloud. Your service administrator receives a Welcome to Oracle Cloud email when the service is ready to use. The email has the URL for your Oracle Content and Experience Cloud. For example, https://your_service_name.com/documents.
   
   • **Identity Domain**: the name of the identity domain that your Oracle Content and Experience Cloud belongs to. The identity domain is mandatory only if you want conversations as well as documents. You can successfully configure a connection to your Oracle Content and Experience Cloud (for the documents feature) without providing an identity domain.
   
   • **User** and **Password**: the account credentials for a user who has access to your Oracle Content and Experience Cloud. This user account is used to test the connection between the services. It's also used during runtime to connect to the services and perform all the runtime operations, such as creating folders and processing conversations.
Important:

If you want to collaborate using conversations, then the user you specify here must be assigned the following role:

```
SERVICE_INSTANCE_NAME OSN_SERVICE_INTEGRATOR_USER_ROLE
```

Use the Oracle Cloud My Services application to assign roles to your users.

4. Click Test.

Whenever you make any changes to the configuration settings, it’s a good idea to verify that the values you entered are correct. You want to confirm that a successful connection has been established with Oracle Content and Experience Cloud.

Review the test results, which may include messages, errors, and warnings.

5. Select one of the following options to continue:
   - If there are any errors or warnings, make the necessary changes and then click Test again to verify the new values. Repeat the test each time you change the settings.
   - If the connection test is successful, click Save to save the configuration settings.
   - If you want to cancel and return to the last-saved values, click Revert.

Enable or Disable Documents and Conversations

After an administrator establishes a connection between Oracle Content and Experience Cloud and Oracle Integration Cloud, you can use documents and conversations in your process applications. These features are enabled by default for all new applications you create.

For your process applications that were created before the connection with Oracle Content and Experience Cloud was configured, the documents and conversations features are disabled by default. If you want these features in existing process applications, then you need to manually enable the features.

There are many benefits to having documents and conversations available when you’re modeling a process, viewing tasks, and tracking a process instance. However, for some particular process applications, you might not want the documents or conversations feature to be available. In such cases, you can manually disable one or both features.

To enable or disable the documents and conversations features for a particular process application:

1. Open the application so you can see it on the Application Home tab.
   - If you're viewing a QuickStart App or a QuickStart Master, then you'll need to click Switch to Advanced View.

2. Click More details to expand the Information pane.
The check boxes next to the Documents icon show you the current status of the documents and conversations features for this application.

3. Use the check boxes to enable or disable each feature for this application.

4. Activate a new revision of the process application. On the Activate Options tab, select the **Force Default** option.
   - Once you can activate Content and Experience Cloud for a process application already in use, new instances will use documents and existing instances can continue to use native attachments.
   - For any given process application, documents and attachments are mutually exclusive. For applications where **Documents Integration** is enabled, then those applications can use documents and only documents. When you disable **Documents Integration** for an application, then that application can use attachments and only attachments. By disabling documents, you enable attachments.

---

**Quick Tour of the Documents Page**

The Documents page provides the tools required to quickly update the settings and permissions for folders and documents that are stored in Oracle Content and Experience Cloud and used in Oracle Integration Cloud at runtime.

**Note:**

The Documents page is available only if an administrator for Oracle Integration Cloud created a connection to Oracle Content and Experience Cloud.

In addition, for process applications that were created *before* a connection to Oracle Content and Experience Cloud was configured, the Documents feature and the Conversations feature are disabled by default. For these cases, you must manually enable the Documents and Conversations features.

**Configured Folders and Documents**

The Documents page displays information about the folders and documents that have been configured for your process applications.

For each process application that has the Documents feature enabled, Oracle Integration Cloud automatically creates the following default folders:

- **Application Folder**: This main root folder contains all the process instance folders.
• **Instance Folder**: One root folder is created for each process instance. It contains the set of managed folders that have been defined in the design-time environment.

  **Note:**
  
  The incoming documents and incoming folders that have been defined in design-time environment aren't stored in the Instance Folder.

• **Managed Folders**: These folders are defined to organize the uploaded files and to set the access level. One managed folder is automatically defined and marked as the Startup folder. The **Startup folder** is the only folder that’s shown for users to upload documents when starting an application. You can define additional managed folders. All managed folders are created for each process instance.

  Select **Properties** to change the name of the root folders or to specify a different folder as the startup folder.

---

**Information Displayed for Each Folder or Document**

Each property card includes an icon that indicates whether the definition is for a managed folder ( ), an incoming folder ( ), or an incoming document ( ).

Each property card also displays the following information:

• The name of the folder or document. You can click the name to edit the basic properties.

• A description of the folder or document. If no description was provided, then the card displays the definition type: **Managed Folder**, **Incoming Folder**, or **Incoming Document**.
• The access type displays the default permission level required to access the folder or document. You can override the default access type at the task level. The available access types are Contributor (default), Downloader, Viewer, or None.

Note that Startup ✓ indicates which folder has been selected as the Startup folder. Select Properties to select a different folder as the startup folder. The Startup folder is the only folder that is shown when a process is started.

Summary of Tools for Managing Folders and Documents

Use the following tools to view, configure, and manage your folders and documents:

• Click Views to alternate between viewing the page in either a grid format or a list format.

• Click New Folder or Document to configure folders and documents.

• Click Properties to edit the name of the root folders or to change the startup folder.

• Click the folder name to edit the basic properties of the folder or document.

• Click Delete to delete a folder or document.

Create a Document- or Folder-Initiated Process

Use the Document Start event to model a process that can be initiated by a document. Use the Folder Start event to model a process that can be initiated by a folder.

You must have an Oracle Content and Experience Cloud, and you must configure a connection between that service and your Oracle Integration Cloud before you can create a document-initiated or a folder-initiated process.

By enabling the Oracle Content and Experience Cloud integration, you can define folders that will be created automatically on Oracle Content and Experience Cloud for every process instance, providing a predefined organization of the documents involved. You can also override the access type at the task-level to define the right permissions for that folder or document for a particular task based on your business needs. For example, you might want to prevent users from viewing a classified document or folder associated with a task.

Note these access requirements for a successful integration:

• The Oracle Content and Experience Cloud user configured in Oracle Integration Cloud must have full access to the folder (that is, the folder of the incoming document) configured in Oracle Content and Experience Cloud to be able to kick off a process.

• For a folder-initiated process (that is, a process with a Folder Start event), the Oracle Content and Experience Cloud user configured in Oracle Integration Cloud must have manager access to that folder in order to access the folder in Oracle Integration Cloud. For example, manager access is required to see the folder on the Task Details page and the Process Tracking page, access the folder when embedding Process UI components in an external application, or see the folder in the Process Mobile application.

• The Oracle Integration Cloud user configured in Oracle Content and Experience Cloud must be granted the process initiator role in order to see the process in the
process list for a folder and to be able to initiate the process instance upon arrival of a new document.

To design a process that can be started by a document or folder:

- Define the **Incoming Document** or the **Incoming Folder** in Oracle Integration Cloud.
- Model a process that has a document start event or a folder start event.
- Customize its implementation to map the start event with the corresponding incoming document or folder you created. Implementation options allow you to define the way in which the document or folder is exposed to users.
- Modify what role can access the document or folder at the task level (optional).
- Configure the folder on Oracle Content and Experience Cloud to initiate a process on document arrival.

Alternatively, you can use REST APIs to instantiate a process instance and provide all the input values.

**Define the Incoming Document or Folder**

To define the incoming document or folder, open the process application you are modeling and click **Documents**. The Documents page lists the incoming documents and incoming folders that have already been defined.

To create a new incoming document or folder, click **New**, select the appropriate type, enter a name, and select the default access (permission).

**Model Processes with Document or Folder Start Events**

Next, build your process that has a document start or folder start event. In the current release, there are some limitations for how you can create the process. Failure to follow the required procedure will invalidate the process.

⚠️ **Caution:**

You must add the document start or folder start event from the Elements palette. Only a single, and only the first, start event that you add to a process will be supported. Multiple start events aren’t supported in this release. Also, don’t change a start event using the Change Type option, and don’t delete the first start event. Both actions will invalidate the process for use.

To model a process that has a document start or folder start event:

1. On the Application Home tab, click **Processes**.
2. Click **New process** to open the Create Process dialog.
3. Select **None**.

✏️ **Note:**

Be sure to select **None**. You will change this empty start event later.
4. Enter a name for the process, confirm the Open Immediately check box is selected, and then click Create.

The canvas displays the start and end events in the process.

5. Click Events in the Elements Palette.

6. Drag either the Document Start event or the Folder Start event to the canvas.

7. Add a sequence flow from your new document or folder start event to the end event.

8. Delete the empty start event.

9. Continue to modify and define the process. Be sure to test and deploy the process.

Allow All Users to Start a Document- or Folder-Initiated Process

To allow any valid user to upload a document to start a process:

1. Open the process.

2. In the swimlane with the document or folder start event, click the role name and then click Edit.

3. In the Role field, select Automatic Handler.

Customize Your Document- or Folder-Initiated Process

After you model the process, you can add a start document or folder in a process and customize it.

In your process, click Document Start or Folder Start event in the process diagram.

From the icons that display, click Menu and then select Open Properties. The Properties pane expands into view below your process diagram.
Define how you want to handle the incoming document or folder:

- **In Place**: Selecting this value keeps the location of the document as it is wherever it is. Optionally, you can map the incoming document or folder with one of the predefined documents or folders (open the drop-down list) for its management. If there isn’t a predefined incoming document or folder suitable for this particular process, then click **New** to create one.

- **Unmanaged**: The document or folder is ignored by the current process. Oracle Integration Cloud won’t show an unmanaged document or folder in runtime. It’s up to the process modeler to handle the incoming document or folder. For example, if you want to move the incoming folder or document to another location, then you can use an XPath expression to get properties, such as `id` or `type`, and pass the property information to a REST service.

### Customize Access at the Task Level

In your process, click a user task in the flow diagram. From the icons that display, click **Menu** and then select **Open Properties**. The Properties pane expands into view below your process diagram.

Click **Override** to customize the current permission of any element defined.

### Configure the Folder

To have a process automatically start when a document is uploaded to a folder, you need to configure the folder on Oracle Content and Experience Cloud to initiate a process on document arrival:

1. Sign in to Oracle Content and Experience Cloud.
2. Select the folder.
3. Select **Properties** from the menu bar.
4. Enable the **Initiate process on document arrival** setting.
5. Select the process from the list.
6. Click **Save**.

The folder and its subfolders are now available for use within the Oracle Integration Cloud interface. Any change to a file in the folder or any new file uploaded to the folder triggers the process associated with the folder. You can override the inherited process for a subfolder, but you can't disable the association with a process.

When a file is uploaded from Oracle Content and Experience Cloud into a folder configured for use with Oracle Integration Cloud, the file is used for the task associated with that folder. Users in Oracle Integration Cloud can take any actions on the files there, such as approvals. When a task step is completed, the file can be moved or managed according to the defined process.

When Oracle Content and Experience Cloud starts a process, the payload sent to launch the process includes this information:

- Document ID
- Document name
- ID of the user who started the process
• Type
• Role (indicates the role that should be used to generate subsequent application links)
• Version

The following example uses only the document ID and document name for display in the form in Oracle Integration Cloud. In addition, the document ID is used for making REST API calls to move or copy the file in Oracle Content and Experience Cloud into the task folder.

```
{
    "processDefId": "testing~UserFileApproval!1.0~FormApprovalProcess",
    "operation": "startEvent",
    "params": {
        "id": "D2806600E495B744E66BF3981212FF6185DE89BE6812",
        "type": "d",
        "name": "document-name",
        "startedBy": "user-id",
        "role": "role that should be used to generate subsequent applinks",
        "version": "version"
    }
}
```

**Note:**

The first operation in the process must be a start event. Otherwise, the process application will return a 500 error to Oracle Content and Experience Cloud.

As a developer, you must be aware of the following requirements for the process you develop:

• It needs to be a process that uses an Oracle Content and Experience Cloud Start event.
• When deploying the process, you need to share it with the user specified for enabling the integration so that user has the rights to trigger the process.
• For the user who uploaded the file to show up as the user who started the task, the process must use the value passed in the `startedby` field as the display name for the initiator.
• If you enable the process integration for a folder, you need to share this folder with the Oracle Content and Experience Cloud user that was used to enable the integration in Oracle Integration Cloud.
Integrate with Oracle Developer Cloud Service

Oracle Developer Cloud Service is a cloud-based software development platform that provides an open source, standards-based integration to develop, test, and deploy applications into other cloud services such as Oracle Content and Experience Cloud.

**Note:**

This integration works only on traditional cloud accounts because it requires that the service be in the same identity domain as Oracle Content and Experience Cloud.

- About Using Developer Cloud Service
- Understand Site Template Development
- Understand Component Development
- About Oracle Developer Cloud Service Projects
- Use a Template Project

**About Using Developer Cloud Service**

Oracle Developer Cloud Service is a cloud-based software development Platform as a Service (PaaS) and a hosted environment for your application development infrastructure. It provides an open source standards-based integration to develop, collaborate, and deploy applications within Oracle Cloud.

Developer Cloud Service is a collection of software and services hosted on Oracle Cloud to help you manage the application development life cycle effectively through integration with Git, Maven, issues, and wikis. Using Oracle Developer Cloud Service, you can commit your application source code to the Git repository on Oracle Cloud, track assigned issues and defects online, share information using wiki pages, peer review the source code, and monitor project builds. After successful testing, you can deploy the project to Oracle Content and Experience Cloud.

**Understand Component Development**

Oracle Developer Cloud Service helps you develop custom components for Oracle Content and Experience Cloud.

The Oracle Developer Cloud Service integration with Oracle Content and Experience Cloud provides a template with sample components and unit tests. The integration includes a git repository, scripts that show how to develop components, and a local test harness for quick, iterative development of custom components.
Oracle Developer Cloud Service can help you do the following tasks:

- Explore the sample components and use them to start developing custom components
- Create local components
- Write unit tests
- Optimize components
- Deploy your components to Oracle Content and Experience Cloud

The template provides the following components:

<table>
<thead>
<tr>
<th>Component</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>local</td>
<td>This sample shows a simple local (custom) component. Local components have access to the same features and capabilities as those provided by Oracle Content and Experience Cloud.</td>
</tr>
<tr>
<td>local-iframe</td>
<td>This sample shows a simple local (custom) component enclosed in an inline frame (iframe).</td>
</tr>
<tr>
<td>Sample-Documents-Manager</td>
<td>This sample shows how to use the embedded version of the Oracle Content and Experience Cloud documents interface to display and optionally manage folder contents.</td>
</tr>
<tr>
<td>Sample-Documents-Manager</td>
<td>This sample shows how to use the embedded version of the Oracle Content and Experience Cloud documents interface to display and optionally manage folder contents.</td>
</tr>
<tr>
<td>Sample-Facebook-Share</td>
<td>This sample shows how to integrate with external systems using a Facebook Share button to share a URL and include a personalized message. Component settings specify layout and button size and whether to share the current page or a specific URL.</td>
</tr>
<tr>
<td>Sample-File-List</td>
<td>This sample shows how to display and optionally interact with the contents of a folder in Oracle Content and Experience Cloud presented as a simple list.</td>
</tr>
<tr>
<td>Sample-Folder-List</td>
<td>This sample shows how to display and optionally interact with a list of folders in Oracle Content and Experience Cloud presented as a simple list.</td>
</tr>
<tr>
<td>Sample-News-API</td>
<td>This sample shows how to make REST API calls to external systems by rendering a list of top news headlines retrieved from an external news API. The news source is specified as a setting and is passed as a parameter to the external API. It also provides settings to show or hide the image or description portions of the news articles.</td>
</tr>
<tr>
<td>Sample-Process-Task-Details</td>
<td>This sample shows how to use a task detail form in conjunction with a process task list to display details for a selected task. If you have a process task list and a process detail form on the page, the process task detail component automatically displays the detail for a task selected from the list.</td>
</tr>
<tr>
<td>Sample-Process-Task-Form</td>
<td>This sample shows how to use a process start form to initiate a process defined with Oracle Process Cloud Service.</td>
</tr>
<tr>
<td>Sample-Process-Task-List</td>
<td>This sample shows how to use a process task list to selectively list processes defined with Oracle Process Cloud Service.</td>
</tr>
<tr>
<td>Sample-Stocks-Embedded</td>
<td>This sample shows how to embed external pages by embedding an external URL in an iFrame to show a stocks chart, watch list or ticker display. The display type is specified as a setting and is passed in the URL.</td>
</tr>
<tr>
<td>Component</td>
<td>Description</td>
</tr>
<tr>
<td>-------------------------------</td>
<td>-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Sample-Text-With-Image</td>
<td>This sample shows how to render interface components such as an image, title and paragraph. It provides settings to change image width and title.</td>
</tr>
<tr>
<td>Sample-To-Do</td>
<td>This sample shows how to render a To Do list that is stored in localStorage. The user can enter a task, mark a task completed, delete a task, and delete all completed tasks. It provides a testing framework using Mocha, Chai and PhantomJS.</td>
</tr>
<tr>
<td>Sample-Weather-Embedded</td>
<td>This sample shows how to embed external pages by embedding an external URL in an iFrame to show the current weather for a given postal code. The postal code is specified as a setting and is passed in the URL.</td>
</tr>
</tbody>
</table>

See Develop Custom Components with Developer Cloud Service in *Developing for Oracle Content and Experience Cloud*.

---

**Understand Site Template Development**

Oracle Developer Cloud Service helps you develop custom site templates and themes for Oracle Content and Experience Cloud.

The Oracle Developer Cloud Service integration with Oracle Content and Experience Cloud provides the following templates that you can use to create your own site templates.

<table>
<thead>
<tr>
<th>Component</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>StarterTemplate</td>
<td>This template provides a simple, fully functional example that you can explore and expand with components and interactions. The Starter Template includes the site code framework, a default site with sample pages and content, a theme with styling, resources such as images, and a custom component with trigger and action functionality. The sample pages include information about creating templates with links to resources providing more detailed information.</td>
</tr>
<tr>
<td>NewProductLaunch</td>
<td>This template provides a menu system and page layouts for case studies, details on features, pricing, your company, and contact information. The home page of the New Product Launch template has a rotating banner image with text.</td>
</tr>
<tr>
<td>JETStarterTemplate</td>
<td>This version of the Starter Template provides the ability to render a custom Oracle JET component.</td>
</tr>
<tr>
<td>CafeSupremoLite</td>
<td>This template provides a fully functional marketing demonstration site with different standard and custom components. It shows how to override styles for fonts and colors defined in the cascading style sheet (css), and provides responsive site behaviors that allow it to be displayed on everything from mobile devices to high-resolution desktop monitors.</td>
</tr>
</tbody>
</table>

See Develop Templates with Developer Cloud Service in *Developing for Oracle Content and Experience Cloud*. 
About Oracle Developer Cloud Service Projects

An Oracle Developer Cloud Service project is a collection of Git repositories, branch merge requests, wikis, issues, deployment configurations, and builds.

The following table describes the pages of an Oracle Developer Cloud Service project.

<table>
<thead>
<tr>
<th>Page</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Project</td>
<td>Shows the navigation bar, Git and Maven repository URLs, news feed, and the recent activities feed of the project.</td>
</tr>
<tr>
<td>Code</td>
<td>Shows the Git source code repositories and enables you to view files and commits, manage branches and tags, and compare the source code of files.</td>
</tr>
<tr>
<td>Maven</td>
<td>Shows the project Maven repository and enables you to view, upload, and search artifacts.</td>
</tr>
<tr>
<td>Releases</td>
<td>Shows the various Release configurations created for the project.</td>
</tr>
<tr>
<td>Snippets</td>
<td>Shows the snippets created by you or shared by project members.</td>
</tr>
<tr>
<td>Merge Requests</td>
<td>Enables you to create and manage branch merge requests, and review the code.</td>
</tr>
<tr>
<td>Issues</td>
<td>Shows all issues of the project and enables you to create and manage tasks, defects, and new features.</td>
</tr>
<tr>
<td>Agile</td>
<td>Enables you to manage project issues using the Agile methodology.</td>
</tr>
<tr>
<td>Build</td>
<td>Enables you to create and manage project builds and jobs.</td>
</tr>
<tr>
<td>Deploy</td>
<td>Enables you to create deployment configurations and deploy project artifacts to Oracle Content and Experience Cloud.</td>
</tr>
<tr>
<td>Docker Registry</td>
<td>Shows the linked Docker registries of the project that project members can browse.</td>
</tr>
<tr>
<td>Wiki</td>
<td>Enables you to create and manage wiki pages.</td>
</tr>
<tr>
<td>Administration</td>
<td>Enables you to configure the project properties. The Administration page is visible to project owners only.</td>
</tr>
</tbody>
</table>

Click the Tutorial link to learn about the pages of Oracle Developer Cloud Service.

See also Work with Developer Cloud Service Projects in Using Oracle Developer Cloud Service.

Use a Template Project

If you chose a template project while creating a project, some artifacts from the template project are copied or added to the new project.

The following artifacts are cloned from the template project:

- Git repositories
  All Git repositories of the template project are cloned to your project. The Git repositories contain a copy of the template project's source code. You can clone the template project repositories to your local machine, modify them, add or
remove branches, and then push them back to the same repository or another repository of your choice. In the navigation bar, click **Code** to view the cloned Git repositories and their content.

- **Build jobs**
  All existing build jobs of the template project are copied to your project. In the navigation bar, click **Builds** to see all copied jobs. You can modify or copy these jobs.

- **Deployment configurations**
  All existing deployment configurations of the template project are copied to your project. In the navigation bar, click **Deploy** to see all copied deployment configurations.

- **Wiki pages**
  All existing wiki pages of the template project are copied to your project. In the navigation bar, click **Wiki** to see all copied wiki pages.

- **Announcements**
  All active project announcements of the template project are cloned to your project. In the navigation bar, click **Project** to see all copied announcements. The announcements are read-only and cannot be edited, but they can be activated or deactivated.

---

**Note:**

After you create a project based on a template, any updates made to the template project will not be reflected in the created project.

See also Use a Template Project in *Using Oracle Developer Cloud Service.*
Integrate with Oracle Visual Builder

You can allow your users to access Oracle Visual Builder Cloud Service functionality. Oracle Visual Builder Cloud Service (VBCS) is a hosted environment for your application development infrastructure. It provides an open-source standards-based integration to develop, collaborate on, and deploy applications within Oracle Cloud. This enables users to rapidly create web and mobile applications with minimal to no coding.

**Note:**
- Integration between these services requires SSO, so both services must be in the same identity domain.
- Only administrators with the *enterprise user* role can enable integration with VBCS. If you aren’t an enterprise user, the **Visual Builder Cloud Service Integration** option is grayed out.

On the VBCS side, the following must be done before this feature can be used with Oracle Content and Experience Cloud:

- Cross-Origin Resource Sharing (CORS) must be enabled on the VBCS site. See Allow Other Domains Access to Services in *[Administering Visual Builder in Oracle Integration Cloud]*.
- Apps must be created and made available for embedding. See Creating a New Web Application in *[Using Visual Builder in Oracle Integration Cloud]*.
- The apps must be configured for use with Oracle Content and Experience Cloud. See Embedding the Application in *[Using Visual Builder in Oracle Integration Cloud]*.
- Web applications must be created and made available for embedding in an iframe.
- The Sites SDK must be imported.
- The Sites SDK must be referenced in the web applications.
- A page URL parameter called “id” must be added to the web applications.

On the Oracle Content and Experience Cloud side, you need to configure integration with VBCS:

1. After you sign in to the Oracle Content and Experience Cloud web application as an administrator, click **Settings** in the Administration area of the navigation menu.
2. In the **Settings** menu, click **Integrations**.
3. Under **Oracle Integrations**, select **Visual Builder Cloud Service Integration** to enable the service.
4. Enter the **Service URL** of the Oracle Visual Builder Cloud Service.
Note:

If you have Universal Credits subscription, you must include `ic/builder` in your Service URL. For example, `https://vbcsserver.company.com/ic/builder`.

After both services have been configured for integration, Oracle Content and Experience Cloud users can create components for your VBCS apps and add them to site pages. See Oracle Visual Builder Cloud Service in Creating Experiences with Oracle Content and Experience Cloud.
Integrate with Oracle Policy Automation

You can allow your users to access Oracle Policy Automation functionality. Oracle Policy Automation (OPA) is used to implement online “interview” scenarios, such as feedback for troubleshooting or eligibility assessments for services. It delivers advice across channels by capturing rules in natural language Microsoft Word and Excel documents, then building interactive customer service experiences called interviews around those rules.

Note:

Only administrators with the enterprise user role can enable integration with OPA. If you aren’t an enterprise user, the Oracle Policy Automation Cloud Service Integration option is grayed out.

On the OPA side, interviews must be created and stored on the host site. In addition, the OPA administrator must add the Oracle Content and Experience Cloud domains (*.documents.* and *.sites.*) to the list of hosts authorized to embed interviews. See the Oracle Policy Automation documentation.

On the Oracle Content and Experience Cloud side, you need to configure integration with OPA:

1. After you sign in to the Oracle Content and Experience Cloud web application as an administrator, open your user menu and click Administration.
2. In the Administration menu, click Integrations.
3. Under Oracle Integrations, select Oracle Policy Automation Cloud Service Integration to enable the service, and then set these values:
   • Service URL: Enter the URL of the Oracle Policy Automation Cloud Service.
   • Service User: Enter the name of the OPA user. This user must be an Integration user and must have the Deploy Admin role for the OPA collections.
   • Service Password: Enter the user password.

After both services have been configured for integration, Oracle Content and Experience Cloud users can add an OPA component to site pages. See Oracle Policy Automation in Creating Experiences with Oracle Content and Experience Cloud.
Integrate with Oracle Cobrowse Cloud Service

You can allow your users to access Oracle Cobrowse Cloud Service functionality, which lets users share screens or initiate a cobrowsing session with another person. For example, you might want to use cobrowse on a sales site so that a sales representative can help a customer select appropriate products or services on the site.

Note:

Only administrators with the enterprise user role can enable integration with Oracle Cobrowse Cloud Service. If you aren't an enterprise user, the Oracle Cobrowse Cloud Service Integration option is grayed out.

To integrate with Oracle Cobrowse Cloud Service:

1. After you sign in to the Oracle Content and Experience Cloud web application as an administrator, open your user menu and click Administration.
2. In the Administration menu, click Integrations.
3. Under Oracle Integrations, select Oracle Cobrowse Cloud Service Integration to enable the service, and then set these values:
   - **Service URL**: Enter the URL for the Cobrowse service. See Log in to the Agent Console in the Standalone Cobrowse User Guide for the link (for example, https://www.livelook.com).
   - **Service User**: Enter the Oracle Cobrowse Cloud Service administrator user name.
   - **Service Password**: Enter the user’s password.

After you've configured the integration, Oracle Content and Experience Cloud users can enable cobrowse to work with a site and add the Cobrowse Launcher component to a site page. See Enable Cobrowse Integration and Use Cobrowse on a Page in Creating Experiences with Oracle Content and Experience Cloud.

Note:

If you later decide to disable cobrowse, you must disable the option on the Integrations page and in the site settings for any sites that use cobrowse. If you disable only the option on the Integrations page, any sites that use cobrowse will continue to do so, but users won't be able to add new cobrowse functionality.