

Oracle® Financial Services Application Studio Cloud Service

Oracle Financial Services Application Studio User Manual



Release 14.8.2.0.0

G55941-01

April 2026

ORACLE®

G55941-01

Copyright © 2026, Oracle and/or its affiliates.

This software and related documentation are provided under a license agreement containing restrictions on use and disclosure and are protected by intellectual property laws. Except as expressly permitted in your license agreement or allowed by law, you may not use, copy, reproduce, translate, broadcast, modify, license, transmit, distribute, exhibit, perform, publish, or display any part, in any form, or by any means. Reverse engineering, disassembly, or decompilation of this software, unless required by law for interoperability, is prohibited.

The information contained herein is subject to change without notice and is not warranted to be error-free. If you find any errors, please report them to us in writing.

If this is software, software documentation, data (as defined in the Federal Acquisition Regulation), or related documentation that is delivered to the U.S. Government or anyone licensing it on behalf of the U.S. Government, then the following notice is applicable:

U.S. GOVERNMENT END USERS: Oracle programs (including any operating system, integrated software, any programs embedded, installed, or activated on delivered hardware, and modifications of such programs) and Oracle computer documentation or other Oracle data delivered to or accessed by U.S. Government end users are "commercial computer software," "commercial computer software documentation," or "limited rights data" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, the use, reproduction, duplication, release, display, disclosure, modification, preparation of derivative works, and/or adaptation of i) Oracle programs (including any operating system, integrated software, any programs embedded, installed, or activated on delivered hardware, and modifications of such programs), ii) Oracle computer documentation and/or iii) other Oracle data, is subject to the rights and limitations specified in the license contained in the applicable contract. The terms governing the U.S. Government's use of Oracle cloud services are defined by the applicable contract for such services. No other rights are granted to the U.S. Government.

This software or hardware is developed for general use in a variety of information management applications. It is not developed or intended for use in any inherently dangerous applications, including applications that may create a risk of personal injury. If you use this software or hardware in dangerous applications, then you shall be responsible to take all appropriate fail-safe, backup, redundancy, and other measures to ensure its safe use. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software or hardware in dangerous applications.

Oracle®, Java, MySQL, and NetSuite are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

Intel and Intel Inside are trademarks or registered trademarks of Intel Corporation. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. AMD, Epyc, and the AMD logo are trademarks or registered trademarks of Advanced Micro Devices. UNIX is a registered trademark of The Open Group.

This software or hardware and documentation may provide access to or information about content, products, and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third-party content, products, and services unless otherwise set forth in an applicable agreement between you and Oracle. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third-party content, products, or services, except as set forth in an applicable agreement between you and Oracle.

Contents

1	Overview	
2	Service API Creation	
2.1	Service creation for Data Segment	2
2.2	Service for Individual Component	3
2.3	Service for Maintenance Summary Component	4
2.4	Import Service	5
2.5	Download Service	6
2.6	Edit Service	7
2.7	Delete Service	8
3	UI Designs – User Interface Designs	
3.1	Introduction	1
3.2	Page Controls	2
3.3	UI Design Patterns	3
4	UI Design Patterns Creation	
4.1	Datasegment Screen	1
4.2	Process Flow Datasegment	4
4.3	Standalone Screen	7
4.4	Maintenance Summary Screen	9
4.5	Import Screens	12
4.6	Edit Component	12
4.7	Publish Component	13
5	Audit Logs	
5.1	Tabular View	1
5.2	Visual Overview	2

6	Integrated AI Assistant	
6.1	Chat based UI design generation	1
6.2	Chat based UI and Service workflow configuration	2
7	OBMA App-Shell	
	Index	

Preface

- [Purpose](#)
- [Audience](#)
- [Module Pre-requisite](#)
- [Documentation Accessibility](#)
- [Diversity and Inclusion](#)
- [Related Resources](#)
- [Conventions](#)
- [Acronyms and Abbreviations](#)
- [Basic Actions](#)
- [Symbols and Icons](#)
- [Module Post-requisite](#)

Purpose

This user guide introduces Oracle Financial Studio (OFS) as a studio for designing, configuring, and governing financial-services solution artifacts in a controlled and reusable way. The guide outlines the steps to create, validate, version, and deploy OFS artifacts using current delivery practices, assuming all prerequisite platform and application setup is already complete.

Audience

This document is intended for implementation teams, business analysts, and developers responsible for designing, configuring, and deploying Oracle Financial Studio (OFS) artifacts such as rules, mappings, templates, and other studio-managed configurations for solutions built on the Oracle Financial Services platform, including deployments aligned with modern microservices-based architectures.

Module Pre-requisite

Specify **User Id** and **Password**, and login to the **Home** screen.

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.

Diversity and Inclusion

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

Related Resources

For more information, see these related user guides:

- *Oracle Banking Common Core User Guide*
- *Oracle Banking Security Management System User Guide*
- *Oracle Banking Routing Hub Configuration User Guide*
- *Oracle Banking Task User Guide*

Conventions

The following text conventions are used in this document:

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

Acronyms and Abbreviations

The list of the acronyms and abbreviations that are used in this guide are as follows:

Table Acronyms and Abbreviations

Abbreviation	Description
SMS	Security Management System
CMC	Common Core
OFS	Oracle Financial Studio

Basic Actions

The basic actions performed in the screens are as follows:

Table Basic Actions

Actions	Description
New	Click New to add a new record. The system displays a new record to specify the required data. The fields marked with asterisk are mandatory. <ul style="list-style-type: none"> This button is displayed only for the records that are already created.
Save	Click Save to save the details entered or selected in the screen.
Unlock	Click Unlock to update the details of an existing record. The system displays an existing record in editable mode. <ul style="list-style-type: none"> This button is displayed only for the records that are already created.
Authorize	Click Authorize to authorize the record created. A maker of the screen is not allowed to authorize the same. Only a checker can authorize a record. <ul style="list-style-type: none"> This button is displayed only for the already created records. For more information on the process, refer Authorization Process.
Approve	Click Approve to approve the initiated record. <ul style="list-style-type: none"> This button is displayed once the user click Authorize.
Audit	Click Audit to view the maker details, checker details of the particular record. <ul style="list-style-type: none"> This button is displayed only for the records that are already created.
Close	Click Close to close a record. This action is available only when a record is created.
Confirm	Click Confirm to confirm the action performed.
Cancel	Click Cancel to cancel the action performed.
Compare	Click Compare to view the comparison through the field values of old record and the current record. <ul style="list-style-type: none"> This button is displayed in the widget once the user click Authorize.
View	Click View to view the details in a particular modification stage. <ul style="list-style-type: none"> This button is displayed in the widget once the user click Authorize.
View Difference only	Click View Difference only to view a comparison through the field element values of old record and the current record, which has undergone changes. <ul style="list-style-type: none"> This button is displayed once the user click Compare.
Expand All	Click Expand All to expand and view all the details in the sections. <ul style="list-style-type: none"> This button is displayed once the user click Compare.
Collapse All	Click Collapse All to hide the details in the sections. <ul style="list-style-type: none"> This button is displayed once the user click Compare.
OK	Click OK to confirm the details in the screen.

Symbols and Icons

The list of symbols and icons available on the screens are as follows:

Table Symbols and Icons - Common



















Symbol/Icon	Function
	Minimize
	Maximize
	Close
	Perform Search
	Open a list
	Date Range
	Add a new record
	Navigate to the first record
	Navigate to the last record
	Navigate to the previous record
	Navigate to the next record
	Grid view
	List view
	Refresh
	Click this icon to add a new row.
	Click this icon to delete a row, which is already added.
	Calendar
	Alerts

Table (Cont.) Symbols and Icons - Common











Symbol/Icon	Function
	Unlock Option
	View Option
	Reopen Option
 AI Map	AI Map Option

Table Symbols and Icons - Widget

Symbol/Icon	Function
	Open status
	Unauthorized status
	Rejected status
	Closed status
	Authorized status
	Modification Number

Module Post-requisite

After finishing all the requirements, log out from the **Home** screen.

1

Overview

This topic provides information on OBX and OFS.

This document provides a feature-by-feature comparison between OBX and OFS to clarify functional parity, highlight improvements, and document how OBX capabilities are being implemented in OFS. For each feature we outline the equivalent capability in OFS and the current implementation status along with any known limitations or planned enhancements.

Table 1-1 Enhancements or Limitations

Sr. No.	OBX	Oracle Financial Studio
1.	Standalone Component	Standalone components, widgets, virtual pages are supported. These components are generally used within other components which acts like screens or as standalone screens. For detailed explanation of how to create standalone components, refer to Section 4.3
2.	Maintenance Screens	OFS supports Maintenance screens to address specialized business use cases by building them as a combination of two web components: Summary screen and Maintenance screens of all steps. Standard framework features such as audit information and built-in menu actions (for example, New, Copy, Unlock, etc.) are available by default. In OFS, both Summary and Detail components can be created quickly using the guided component and UI generation flow, with straightforward endpoint mapping for data persistence, making Maintenance flows easy to develop and consistent with OBMA standards. Refer to Section 4.4
3.	Data Segments	Each step in a maintenance screen is a Data Segment. New Data Segment UI be created and attached to existing Maintenance screens. Refer to Section 4.1
4.	Standalone Service	In OFS, service creation is achieved through a UI-driven creation flow. Users can select Standalone Component – Service Creation and provide the required object and field details directly in the UI. OFS then generates the complete service artifacts following the OBMA architecture. Refer to Section 2.2
5.	Maintenance Service	Maintenance Services are used for Maintenance Screens with multiple Data Segments. Multiple business objects can be created based on the number of Data segments in the Maintenance. Each Data Segment will then have its own service. Refer to Section 2.3
6.	Data Segment Service	Data Segment services are used to plug new Data Segments into already existing Maintenance Screens. Data Persistence of that Data Segment is done through the mapped Service. Refer to Section 2.1
7.	Creation of Child Data-segment (Resource Segment) and Configuring in existing data-segment flow using screen-class maintenance	This feature enables the creation of a Data Segment to capture and persist data for a specific stage within a Maintenance screen flow. The Data Segment is implemented as an independent unit with its own data source and backing service, ensuring clear separation of concerns and easier maintenance. Once created, the Data Segment's UI and service are integrated and plugged into an existing Maintenance screen, allowing the Maintenance flow to be extended without requiring changes to the core screen structure. Refer to Section 7

Table 1-1 (Cont.) Enhancements or Limitations

Sr. No.	OBX	Oracle Financial Studio
8.	Changes in Generated Components	Users can edit components that were previously created using OFS. OFS supports updating key aspects of an existing component, including labels, data bindings, layout configuration, and other component-level properties, enabling teams to refine the UI and behaviour as requirements evolve. Refer to Section 4.6
9.	Batch Extensibility	Not Supported
10.	Validation/ Processing Logic Extensibility	Not Supported
11.	Extending the common UI infra methods using Pre post hooks	Not Supported
12.	Overriding base endpoint with custom endpoint	Not Supported
13.	Extending base labels	Not Supported
14.	Support for addition of custom columns and custom filters in Task based screens	Not Supported
15.	Dynamic data configure feature for LOV	Not Supported
16.	Process Workbench - For extending kernel conductor based workflows/ Creation of custom workflows	NA
17.	UDF - (Additional Attribute Services)	Not Supported

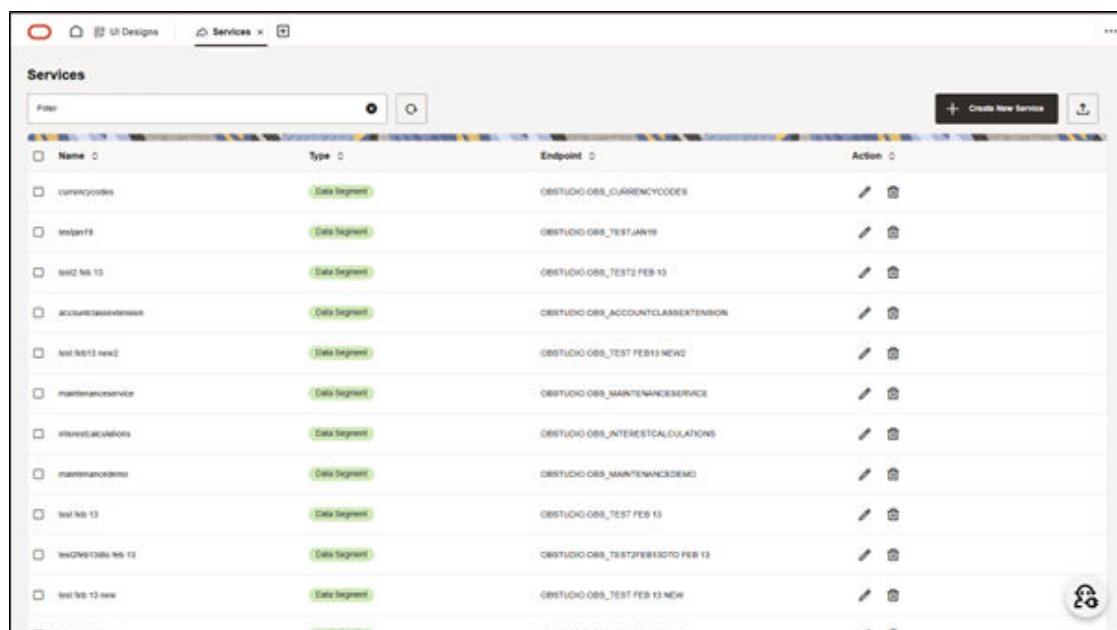
2

Service API Creation

Service API Creation is the process of designing and developing an interface that enables external systems to interact with a service's functionality.

Using OFS, we can create APIs for a wide range of component types through a standardized, end-to-end API creation process. This process provides the complete supporting infrastructure required to design, implement, and expose new services, including the foundational scaffolding and configuration needed for consistent development and deployment. Each generated service is aligned with the OBMA architecture, ensuring compliance with established design principles and promoting reusability, maintainability, and architectural consistency. As a result, these services can be seamlessly plugged into any UI component screen, enabling secure and reliable data persistence and supporting integration across different application modules.

Figure 2-1 Services



Name	Type	Endpoint	Action
currencycodes	Data Segment	OBSTUDIO OBS_CURRENCYCODES	[Edit] [Delete]
test19	Data Segment	OBSTUDIO OBS_TEST19	[Edit] [Delete]
test1913	Data Segment	OBSTUDIO OBS_TEST1913	[Edit] [Delete]
accountclassextension	Data Segment	OBSTUDIO OBS_ACCOUNTCLASSEXTENSION	[Edit] [Delete]
test1913new2	Data Segment	OBSTUDIO OBS_TEST1913NEW2	[Edit] [Delete]
maintenanceerror	Data Segment	OBSTUDIO OBS_MAINTENANCEERROR	[Edit] [Delete]
interestcalculations	Data Segment	OBSTUDIO OBS_INTERESTCALCULATIONS	[Edit] [Delete]
maintenancedemo	Data Segment	OBSTUDIO OBS_MAINTENANCEDEMO	[Edit] [Delete]
test1913	Data Segment	OBSTUDIO OBS_TEST1913	[Edit] [Delete]
test29131313131313	Data Segment	OBSTUDIO OBS_TEST29131313131313	[Edit] [Delete]
test1913new	Data Segment	OBSTUDIO OBS_TEST1913NEW	[Edit] [Delete]
test1913new2	Data Segment	OBSTUDIO OBS_TEST1913NEW2	[Edit] [Delete]

The following data types are supported for DTO field types:

1. **String** – This is an inbuilt field type of OFS; it gets translated to string type in java files and input text field in UI component.
2. **Number** – This is an inbuilt field type of OFS; it gets translated to integer type in java files and input number field in UI component.
3. **Date** – This is an inbuilt field type of OFS; it gets translated to Date type in java files and input date field in UI component.

In addition, OFS supports API creation across multiple component categories, including Data Segment, Individual, and Maintenance components. This enables teams to generate services that align with the specific functional and data requirements of each component type while still following a consistent, standardized service-generation approach. As a result, the produced

APIs can be integrated more easily with corresponding UI screens, helping ensure uniform behavior for data capture, validation, and persistence across different parts of the application.

- [Service creation for Data Segment](#)
This topic provides information about the process of creating a service for a data segment within a system.
- [Service for Individual Component](#)
This topic provides information about the service used to manage and maintain an individual component within a system.
- [Service for Maintenance Summary Component](#)
This topic provides information about the service used to manage and maintain the summary component within a system.
- [Import Service](#)
This topic provides information about the process of uploading and integrating a service-related file, configuration, application package, or resource into a system.
- [Download Service](#)
This topic provides information about the process of retrieving and saving a service-related file, configuration, application package, or resource from a system to a local device.
- [Edit Service](#)
This topic provides information about the process of modifying or updating an existing service, application, or resource within a system.
- [Delete Service](#)
This topic provides information of the process of removing or deactivating an existing service, application, or resource within a system.

2.1 Service creation for Data Segment

This topic provides information about the process of creating a service for a data segment within a system.

1. Select **Data Segment** in the **Component Type** field and give appropriate name and description. The Base Path of the service is generated as “/extensions/<service_name>”.
2. Enable validations, if required by entering a validation code from the OBMA Rule Engine to enable the corresponding validation for the service being created.
3. Create the **Business Object** by adding the required fields and their data types required in the DTO. The DTO fields will be mapped to elements in the UI component.

Figure 2-2 Create a new Service - Data Degment

Create a new Service

Component Type: DataSegment

Business Object: Interest Calculations DTD

Field	Description	Data Type	Query Param	Action
Field 1	Field 1	[Dropdown Menu]		✓ X

No data to display.

- Click on **Save** after all required fields are configured to generate a new service. All required database entries are automatically added for each service.

Figure 2-3 Summary Component - Data Segment_1

ID	RESOURCE_NAME	BASE_PATH	PLATO_ID_ENDPOINT_KEY	SCREEN_CLASS_CODE	API_VERSION	IN_PROGRESS	SUMMARY_ZCA_NAME	SUMMARY_ID_ACTIVITY_CODE	MAINTENANCE_ZCA_NAME
1	7a275a2-07f2-47d0-9d8b-07f8a8888888	InterestCalculations	INTEREST_CALC_ENDPOINT_KEY	INTEREST_CALC_SCREEN	4.0	N	InterestCalculations	InterestCalculations	InterestCalculations
2	8a275a2-07f2-47d0-9d8b-07f8a8888888	InterestCalculations	INTEREST_CALC_ENDPOINT_KEY	INTEREST_CALC_SCREEN	4.0	N	InterestCalculations	InterestCalculations	InterestCalculations
3	9a275a2-07f2-47d0-9d8b-07f8a8888888	InterestCalculations	INTEREST_CALC_ENDPOINT_KEY	INTEREST_CALC_SCREEN	4.0	N	InterestCalculations	InterestCalculations	InterestCalculations
4	0a275a2-07f2-47d0-9d8b-07f8a8888888	InterestCalculations	INTEREST_CALC_ENDPOINT_KEY	INTEREST_CALC_SCREEN	4.0	N	InterestCalculations	InterestCalculations	InterestCalculations
5	1a275a2-07f2-47d0-9d8b-07f8a8888888	InterestCalculations	INTEREST_CALC_ENDPOINT_KEY	INTEREST_CALC_SCREEN	4.0	N	InterestCalculations	InterestCalculations	InterestCalculations
6	2a275a2-07f2-47d0-9d8b-07f8a8888888	InterestCalculations	INTEREST_CALC_ENDPOINT_KEY	INTEREST_CALC_SCREEN	4.0	N	InterestCalculations	InterestCalculations	InterestCalculations
7	3a275a2-07f2-47d0-9d8b-07f8a8888888	InterestCalculations	INTEREST_CALC_ENDPOINT_KEY	INTEREST_CALC_SCREEN	4.0	N	InterestCalculations	InterestCalculations	InterestCalculations
8	4a275a2-07f2-47d0-9d8b-07f8a8888888	InterestCalculations	INTEREST_CALC_ENDPOINT_KEY	INTEREST_CALC_SCREEN	4.0	N	InterestCalculations	InterestCalculations	InterestCalculations
9	5a275a2-07f2-47d0-9d8b-07f8a8888888	InterestCalculations	INTEREST_CALC_ENDPOINT_KEY	INTEREST_CALC_SCREEN	4.0	N	InterestCalculations	InterestCalculations	InterestCalculations
10	6a275a2-07f2-47d0-9d8b-07f8a8888888	InterestCalculations	INTEREST_CALC_ENDPOINT_KEY	INTEREST_CALC_SCREEN	4.0	N	InterestCalculations	InterestCalculations	InterestCalculations
11	7a275a2-07f2-47d0-9d8b-07f8a8888888	InterestCalculations	INTEREST_CALC_ENDPOINT_KEY	INTEREST_CALC_SCREEN	4.0	N	InterestCalculations	InterestCalculations	InterestCalculations
12	8a275a2-07f2-47d0-9d8b-07f8a8888888	InterestCalculations	INTEREST_CALC_ENDPOINT_KEY	INTEREST_CALC_SCREEN	4.0	N	InterestCalculations	InterestCalculations	InterestCalculations
13	9a275a2-07f2-47d0-9d8b-07f8a8888888	InterestCalculations	INTEREST_CALC_ENDPOINT_KEY	INTEREST_CALC_SCREEN	4.0	N	InterestCalculations	InterestCalculations	InterestCalculations
14	0a275a2-07f2-47d0-9d8b-07f8a8888888	InterestCalculations	INTEREST_CALC_ENDPOINT_KEY	INTEREST_CALC_SCREEN	4.0	N	InterestCalculations	InterestCalculations	InterestCalculations
15	1a275a2-07f2-47d0-9d8b-07f8a8888888	InterestCalculations	INTEREST_CALC_ENDPOINT_KEY	INTEREST_CALC_SCREEN	4.0	N	InterestCalculations	InterestCalculations	InterestCalculations
16	2a275a2-07f2-47d0-9d8b-07f8a8888888	InterestCalculations	INTEREST_CALC_ENDPOINT_KEY	INTEREST_CALC_SCREEN	4.0	N	InterestCalculations	InterestCalculations	InterestCalculations
17	3a275a2-07f2-47d0-9d8b-07f8a8888888	InterestCalculations	INTEREST_CALC_ENDPOINT_KEY	INTEREST_CALC_SCREEN	4.0	N	InterestCalculations	InterestCalculations	InterestCalculations

Figure 2-4 Summary Component - Data Segment_2

ID	FIELD_NAME	FIELD_DATA_TYPE	FIELD_DESCRIPTION	IS_REQUIRED	IS_QUERY_PARAM	OBJECT_ID
1	aa1384f1-62e4-4a7f-8b0e-0890f74624c5	DATE	firstLiquidation	N	N	7a89b8ec-e089-473c-8b3c-e060c4374000
2	a776f3b7-8562-4308-8458-663a357ac2c6	NUMBER	accuracy	N	N	7a89b8ec-e089-473c-8b3c-e060c4374000
3	c38773a7-8c0f-4238-87ac-c0f72a869ba	STRING	compoundingFrequency	N	N	7a89b8ec-e089-473c-8b3c-e060c4374000
4	664539b0-808b-47d5-af5e-7c78a29acdae	NUMBER	days	N	N	7a89b8ec-e089-473c-8b3c-e060c4374000
5	62996196-5b6d-4a05-8a3e-d775263a812	STRING	frequency	N	N	7a89b8ec-e089-473c-8b3c-e060c4374000
6	3c1988a1-a727-4785-84ce-85aeb224415	NUMBER	deferBeforeMonthEndDays	N	N	7a89b8ec-e089-473c-8b3c-e060c4374000
7	8135a89-7a25-4a55-84fc-2a588b0b4fd	STRING	cycle months	N	N	7a89b8ec-e089-473c-8b3c-e060c4374000
8	9670c6cf-7e48-42f6-8822-738988888888	STRING	backwardsCalculation	N	N	7a89b8ec-e089-473c-8b3c-e060c4374000
9	853e278f-c04e-45ad-8175-588e9f73cc43	NUMBER	year	N	N	7a89b8ec-e089-473c-8b3c-e060c4374000
10	4c78af9f-4747-4473-a826-cb44ae95381	DATE	firstAccrualDate	N	N	7a89b8ec-e089-473c-8b3c-e060c4374000
11	a145528f-572c-4e1b-8584-5bade4e48eb	NUMBER	months	N	N	7a89b8ec-e089-473c-8b3c-e060c4374000

2.2 Service for Individual Component

This topic provides information about the service used to manage and maintain an individual component within a system.

- Select **Individual** as **Component Type** for which API is being created and give appropriate name and description. The Base Path of the service is generated as `/extensions/<service_name>`.
- Enable validations, as required by entering a validation code from the OBMA Rule Engine to enable the corresponding validation for the service being created

3. Create the **Business Object** by adding the required fields and their data types required in the DTO.

Figure 2-5 Create a new Service - Individual

4. Click on **Save** after all required fields are configured to generate a new service with the required database entries.

2.3 Service for Maintenance Summary Component

This topic provides information about the service used to manage and maintain the summary component within a system.

1. Select **Maintenance** as **Component Type** for which API is being created and give appropriate name and description. The Base Path of the generated service is saved as “/
<service_name>”.
2. Enable validations, if required by entering a validation code from the OBMA Rule Engine to enable the corresponding validation for the service being created
3. Create the **Business Object** by adding the required fields and their data types required in the DTO.
4. Each new step in the maintenance screen will have a different DTO. Click on **Add New Business Object** to create different DTOs for each new screen. You can also delete a DTO if not required.

Figure 2-6 Create a new Service - Maintenance

- Click on **Save** after all required fields are configured to generate a new service for maintenance components.

All required database entries are automatically added for each service. Entries like Summary CCA Name, Maintenance CCA Name are automatically added to the database.

Figure 2-7 Summary Component

ID	RESOURCE_NAME	BASE_PATH	PLATOOL_RESPONSE_KEY	RESOURCE_CODE	API_VERSION	PL_PROGRESS	SUMMARY_CCA_NAME	SUMMARY_ACTIVITY_CODE	MAINTENANCE_CCA_NAME
1	OracleFinancialServicesApplicationStudioUserManual	OracleFinancialServicesApplicationStudioUserManual	OracleFinancialServicesApplicationStudioUserManual	OracleFinancialServicesApplicationStudioUserManual	v1	N	OracleFinancialServicesApplicationStudioUserManual	OracleFinancialServicesApplicationStudioUserManual	OracleFinancialServicesApplicationStudioUserManual
2	OracleFinancialServicesApplicationStudioUserManual	OracleFinancialServicesApplicationStudioUserManual	OracleFinancialServicesApplicationStudioUserManual	OracleFinancialServicesApplicationStudioUserManual	v1	N	OracleFinancialServicesApplicationStudioUserManual	OracleFinancialServicesApplicationStudioUserManual	OracleFinancialServicesApplicationStudioUserManual
3	OracleFinancialServicesApplicationStudioUserManual	OracleFinancialServicesApplicationStudioUserManual	OracleFinancialServicesApplicationStudioUserManual	OracleFinancialServicesApplicationStudioUserManual	v1	N	OracleFinancialServicesApplicationStudioUserManual	OracleFinancialServicesApplicationStudioUserManual	OracleFinancialServicesApplicationStudioUserManual
4	OracleFinancialServicesApplicationStudioUserManual	OracleFinancialServicesApplicationStudioUserManual	OracleFinancialServicesApplicationStudioUserManual	OracleFinancialServicesApplicationStudioUserManual	v1	N	OracleFinancialServicesApplicationStudioUserManual	OracleFinancialServicesApplicationStudioUserManual	OracleFinancialServicesApplicationStudioUserManual
5	OracleFinancialServicesApplicationStudioUserManual	OracleFinancialServicesApplicationStudioUserManual	OracleFinancialServicesApplicationStudioUserManual	OracleFinancialServicesApplicationStudioUserManual	v1	N	OracleFinancialServicesApplicationStudioUserManual	OracleFinancialServicesApplicationStudioUserManual	OracleFinancialServicesApplicationStudioUserManual
6	OracleFinancialServicesApplicationStudioUserManual	OracleFinancialServicesApplicationStudioUserManual	OracleFinancialServicesApplicationStudioUserManual	OracleFinancialServicesApplicationStudioUserManual	v1	N	OracleFinancialServicesApplicationStudioUserManual	OracleFinancialServicesApplicationStudioUserManual	OracleFinancialServicesApplicationStudioUserManual
7	OracleFinancialServicesApplicationStudioUserManual	OracleFinancialServicesApplicationStudioUserManual	OracleFinancialServicesApplicationStudioUserManual	OracleFinancialServicesApplicationStudioUserManual	v1	N	OracleFinancialServicesApplicationStudioUserManual	OracleFinancialServicesApplicationStudioUserManual	OracleFinancialServicesApplicationStudioUserManual
8	OracleFinancialServicesApplicationStudioUserManual	OracleFinancialServicesApplicationStudioUserManual	OracleFinancialServicesApplicationStudioUserManual	OracleFinancialServicesApplicationStudioUserManual	v1	N	OracleFinancialServicesApplicationStudioUserManual	OracleFinancialServicesApplicationStudioUserManual	OracleFinancialServicesApplicationStudioUserManual
9	OracleFinancialServicesApplicationStudioUserManual	OracleFinancialServicesApplicationStudioUserManual	OracleFinancialServicesApplicationStudioUserManual	OracleFinancialServicesApplicationStudioUserManual	v1	N	OracleFinancialServicesApplicationStudioUserManual	OracleFinancialServicesApplicationStudioUserManual	OracleFinancialServicesApplicationStudioUserManual
10	OracleFinancialServicesApplicationStudioUserManual	OracleFinancialServicesApplicationStudioUserManual	OracleFinancialServicesApplicationStudioUserManual	OracleFinancialServicesApplicationStudioUserManual	v1	N	OracleFinancialServicesApplicationStudioUserManual	OracleFinancialServicesApplicationStudioUserManual	OracleFinancialServicesApplicationStudioUserManual
11	OracleFinancialServicesApplicationStudioUserManual	OracleFinancialServicesApplicationStudioUserManual	OracleFinancialServicesApplicationStudioUserManual	OracleFinancialServicesApplicationStudioUserManual	v1	N	OracleFinancialServicesApplicationStudioUserManual	OracleFinancialServicesApplicationStudioUserManual	OracleFinancialServicesApplicationStudioUserManual
12	OracleFinancialServicesApplicationStudioUserManual	OracleFinancialServicesApplicationStudioUserManual	OracleFinancialServicesApplicationStudioUserManual	OracleFinancialServicesApplicationStudioUserManual	v1	N	OracleFinancialServicesApplicationStudioUserManual	OracleFinancialServicesApplicationStudioUserManual	OracleFinancialServicesApplicationStudioUserManual
13	OracleFinancialServicesApplicationStudioUserManual	OracleFinancialServicesApplicationStudioUserManual	OracleFinancialServicesApplicationStudioUserManual	OracleFinancialServicesApplicationStudioUserManual	v1	N	OracleFinancialServicesApplicationStudioUserManual	OracleFinancialServicesApplicationStudioUserManual	OracleFinancialServicesApplicationStudioUserManual
14	OracleFinancialServicesApplicationStudioUserManual	OracleFinancialServicesApplicationStudioUserManual	OracleFinancialServicesApplicationStudioUserManual	OracleFinancialServicesApplicationStudioUserManual	v1	N	OracleFinancialServicesApplicationStudioUserManual	OracleFinancialServicesApplicationStudioUserManual	OracleFinancialServicesApplicationStudioUserManual
15	OracleFinancialServicesApplicationStudioUserManual	OracleFinancialServicesApplicationStudioUserManual	OracleFinancialServicesApplicationStudioUserManual	OracleFinancialServicesApplicationStudioUserManual	v1	N	OracleFinancialServicesApplicationStudioUserManual	OracleFinancialServicesApplicationStudioUserManual	OracleFinancialServicesApplicationStudioUserManual
16	OracleFinancialServicesApplicationStudioUserManual	OracleFinancialServicesApplicationStudioUserManual	OracleFinancialServicesApplicationStudioUserManual	OracleFinancialServicesApplicationStudioUserManual	v1	N	OracleFinancialServicesApplicationStudioUserManual	OracleFinancialServicesApplicationStudioUserManual	OracleFinancialServicesApplicationStudioUserManual
17	OracleFinancialServicesApplicationStudioUserManual	OracleFinancialServicesApplicationStudioUserManual	OracleFinancialServicesApplicationStudioUserManual	OracleFinancialServicesApplicationStudioUserManual	v1	N	OracleFinancialServicesApplicationStudioUserManual	OracleFinancialServicesApplicationStudioUserManual	OracleFinancialServicesApplicationStudioUserManual

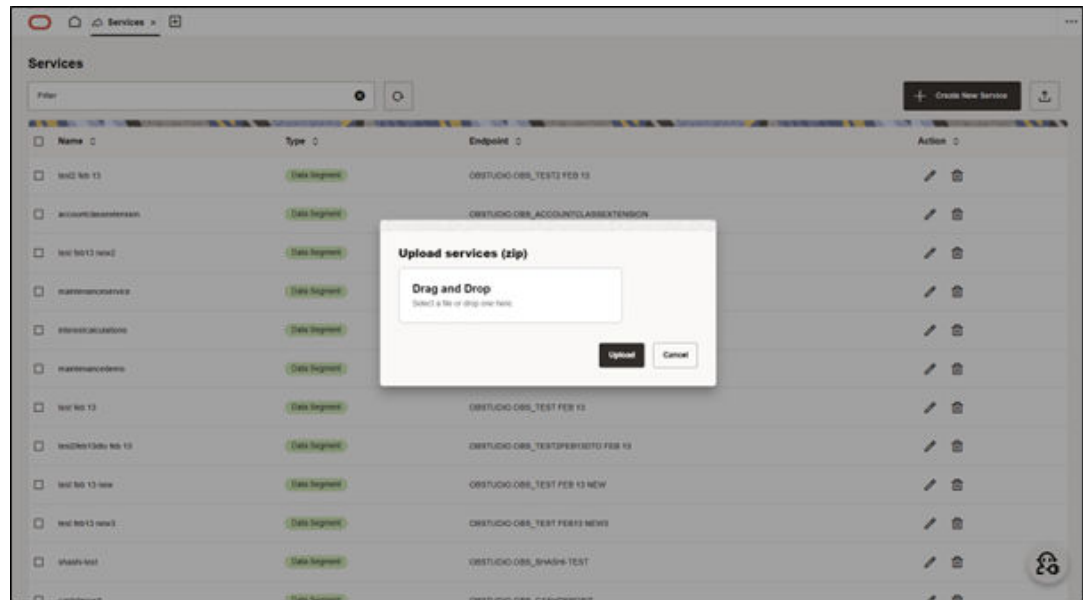
2.4 Import Service

This topic provides information about the process of uploading and integrating a service-related file, configuration, application package, or resource into a system.

Any user can import services into the application by uploading a packaged set of service files. The process is straightforward and can be completed directly from the Services screen:

- Click the **Upload** icon on the **Services** screen to initiate the import workflow.
- Select the ZIP file that contains the service files to be imported. Ensure the ZIP package includes all required service artifacts and is prepared in the expected structure.
- Click **Upload** to upload the zip file. The system will upload and register all services contained in the ZIP file, making them available for use once the upload completes successfully.

Figure 2-8 Import Service



2.5 Download Service

This topic provides information about the process of retrieving and saving a service-related file, configuration, application package, or resource from a system to a local device.

Users can download one or more services from the Services screen as a single packaged ZIP file for offline storage, sharing, or migration across environments. Follow these steps:

1. Select the services you want to download from the list (you can typically choose one or multiple services, depending on your requirement).
2. Click the **Download** icon to initiate the export process.
3. Save the generated ZIP file to the appropriate location on your system, based on your local folder structure

Figure 2-9 Download Service

Name	Type	Endpoint	Action
currencycodes	Data Segment	OBSTUDIO_OBS_CURRENCYCODES	[Edit] [Delete]
budget15	Data Segment	OBSTUDIO_OBS_TESTJAN15	[Edit] [Delete]
test feb 13	Data Segment	OBSTUDIO_OBS_TEST2 FEB 13	[Edit] [Delete]
accountclassextension	Data Segment	OBSTUDIO_OBS_ACCOUNTCLASSEXTENSION	[Edit] [Delete]
test feb 13 new2	Data Segment	OBSTUDIO_OBS_TEST FEB 13 NEW2	[Edit] [Delete]
maintenanceinfo	Data Segment	OBSTUDIO_OBS_MAINTENANCEINFO	[Edit] [Delete]
interestcalculation	Data Segment	OBSTUDIO_OBS_INTERESTCALCULATION	[Edit] [Delete]
maintenanceinfo	Data Segment	OBSTUDIO_OBS_MAINTENANCEINFO	[Edit] [Delete]
test feb 13	Data Segment	OBSTUDIO_OBS_TEST FEB 13	[Edit] [Delete]
test feb 13 new	Data Segment	OBSTUDIO_OBS_TEST FEB 13 NEW	[Edit] [Delete]

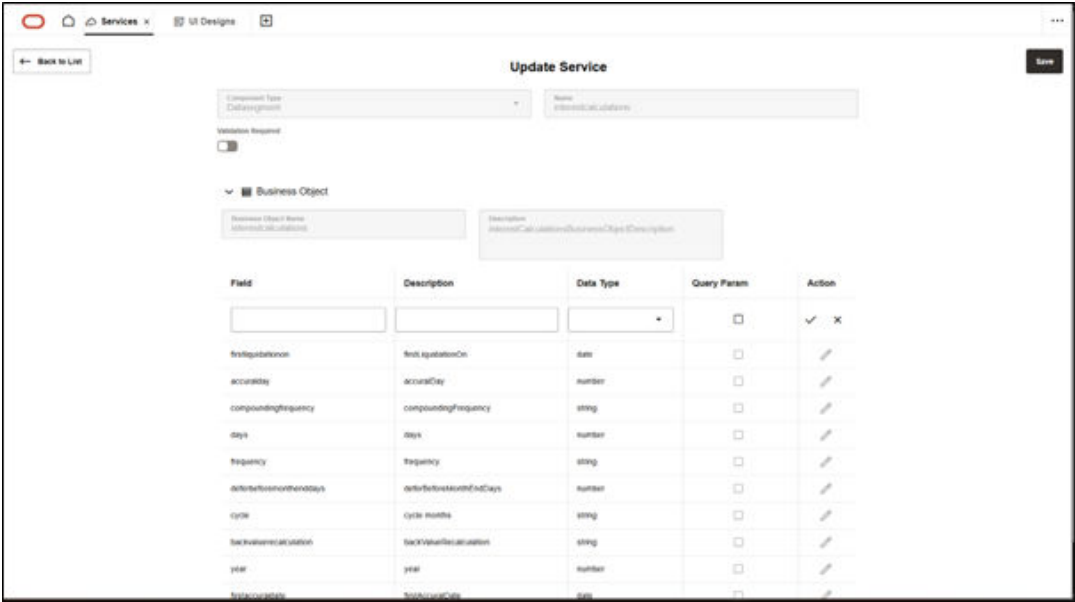
2.6 Edit Service

This topic provides information about the process of modifying or updating an existing service, application, or resource within a system.

To update an existing service, users can use the Edit option available on the service record. The edit workflow is designed to allow controlled changes while protecting the integrity of fields that have already been defined.

1. Click the **Edit** action for the service you want to modify. This will open the service in edit mode.
2. Add new fields as needed. Existing fields cannot be modified (for example, you cannot change the data type of an existing field), but you can extend the service by adding additional fields.
3. Once you have added the required fields, click **Save** to persist your changes and update the service definition.

Figure 2-10 Update Service

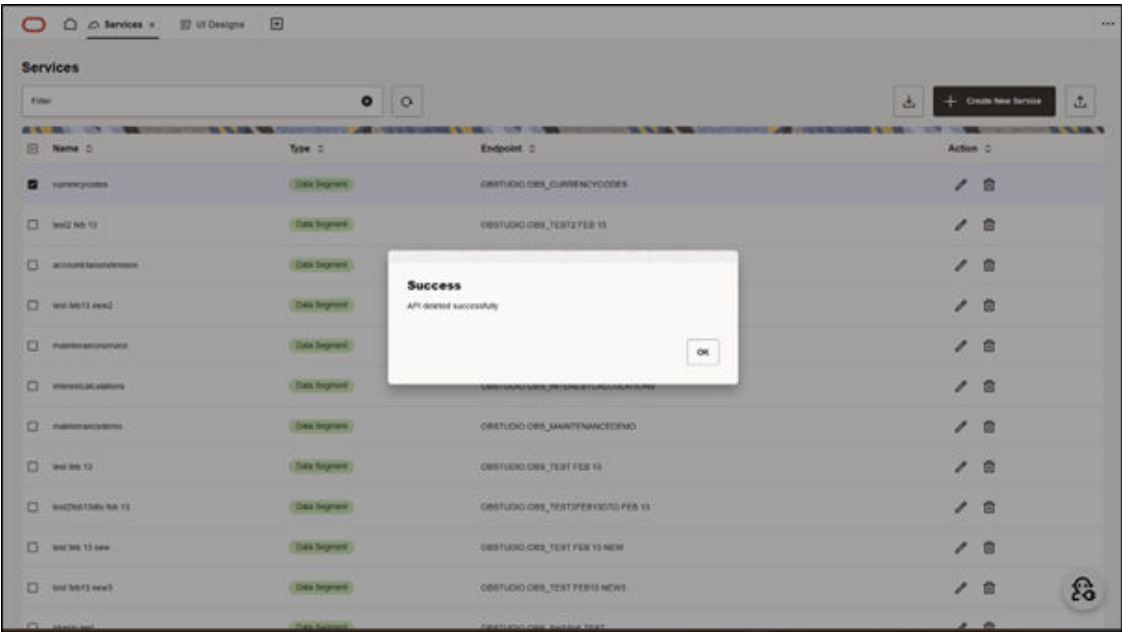


2.7 Delete Service

This topic provides information of the process of removing or deactivating an existing service, application, or resource within a system.

To delete a service, locate it in the Services list and click the Delete action corresponding to that service. Once selected, the service is removed, and all associated entries/records for that service are deleted as part of the operation.

Figure 2-11 Delete Service



3

UI Designs – User Interface Designs

This topic provides information about user interface designs used to define the visual layout, interactive elements, and overall user experience of an application.

The UI Designs module in Oracle Financial Studio (OFS) enables users to create, maintain, and govern UI design artifacts used by applications. It provides capabilities to build new designs, manage versions, and promote changes in a controlled manner based on access and lifecycle rules.

- [Introduction](#)
This section provides an overview of the topic, outlining its purpose, key functionalities, and relevant concepts to help users understand the scope and usage within the system.
- [Page Controls](#)
This topic provides information about page controls.
- [UI Design Patterns](#)
This topic provides information about UI design patterns used to define reusable solutions for common user interface challenges.

3.1 Introduction

This section provides an overview of the topic, outlining its purpose, key functionalities, and relevant concepts to help users understand the scope and usage within the system.

The User Interface Designs page is used to manage UI design artifacts in OFS. From this page, you can view existing designs, create new designs, import designs, open a design for editing, and publish versions.

Navigation: Dashboard → User Interface Designs.

Figure 3-1 User Interface Designs

<input type="checkbox"/> Component Name	Pattern	Published Version	BASE	Actions
<input type="checkbox"/> testing-publish	Standalone	N/A	N	
<input type="checkbox"/> flow-test-files	Flow	N/A	N	
<input type="checkbox"/> test-widget-comp	Widget	N/A	N	
<input type="checkbox"/> testing-payload-flow	Flow	N/A	N	
<input type="checkbox"/> fsgbu-ob-cmc-test-feb12	Datasegment	N/A	N	
<input type="checkbox"/> fsgbu-ob-obcfpm-ds-test-ms	Standalone	N/A	N	
<input type="checkbox"/> testing-generate-file	Standalone	N/A	N	
<input type="checkbox"/> fsgbu-ob-dda-testing-generate-ai-agent	Datasegment	1.0.0	N	
<input type="checkbox"/> fsgbu-ob-cmc-sm-maintenance-rso	Maintenance	N/A	N	
<input type="checkbox"/> fsgbu-ob-dda-crm-sm-cash-deposit	Maintenance	1.0.1	N	
<input type="checkbox"/> fsgbu-ob-cmc-mn-summary-test-gcs	Maintenance	N/A	Y	
<input type="checkbox"/> fsgbu-ob-dda-crm-ds-cash-deposit	Standalone	1.0.0	Y	
<input type="checkbox"/> fsgbu-ob-cmc-vp-test-old	Standalone	1.0.0	N	

3.2 Page Controls

This topic provides information about page controls.

Filter

Use Filter to quickly locate a UI design in the list. Enter a keyword (for example, full or partial Component Name) and the grid updates to show only matching records.

Refresh

Use Refresh to reload the page data from the server. This is useful after creating, importing, or publishing a design, or when multiple users are working in the same environment, and you want the latest list.

Import Screens

Use Import Screens to bring UI design artifacts into OFS from a supported import package/file. This is typically used to:

- Move designs between environments when import/export is the configured promotion method.
- Restore designs from a backup/export.
- Share designs across teams.

Create New Design

Use Create New Design to create a new UI design artifact. Typically, you will:

- Enter a Component Name (as per naming standards).
- Select a Pattern (Standalone, Flow, Widget, Datasegment, Maintenance).

- Create the initial design and open it in the editor for configuration.
- Publish the design when it is ready and approved.

3.3 UI Design Patterns

This topic provides information about UI design patterns used to define reusable solutions for common user interface challenges.

OFS supports the following UI design patterns. Choose the pattern based on how the screen/component is intended to be used.

Standalone

Used to build an independent screen/component that can run on its own. Typically used for single-page experiences that do not require multi-step navigation.

Flow

Used to create a multi-step user journey (wizard-like navigation). Suitable when a process is broken into multiple screens/steps (for example, capture → review → confirm).

Widget

Used to create a small, embeddable UI component. Typically used for dashboard tiles, summary panels, or reusable visual blocks embedded within other pages.

Datasegment

Used to design a data-driven section of a maintenance screen, usually focused on capturing or displaying a related set of fields/records. Commonly used when a component represents a logical grouping of data that can be reused across screens.

Maintenance

Used for maintenance-style screens, typically aligned to create/view/update operations. Often used where you need structured layouts for summary/detail style interactions.

4

UI Design Patterns Creation

- [Datasegment Screen](#)
This topic provides information about the Data Segment screen used to view, manage, and configure data segment details within a system.
- [Process Flow Datasegment](#)
This topic provides information about the process flow for a data segment, outlining the sequence of steps involved in creating, managing, updating, and processing data segment operations within the system.
- [Standalone Screen](#)
This topic provides information about the standalone screen used to independently display and manage specific functionalities or data within the system without dependency on other interface components.
- [Maintenance Summary Screen](#)
This topic provides information about the Maintenance Summary screen used to view, manage, and update summary details related to maintenance activities within the system.
- [Import Screens](#)
This topic provides information about the import screens used to upload and integrate files, configurations, application packages, or resources into the system.
- [Edit Component](#)
This topic provides information about the process of modifying or updating an existing component within a system.
- [Publish Component](#)
This topic provides information about the process of making a component available for use within the system.

4.1 Datasegment Screen

This topic provides information about the Data Segment screen used to view, manage, and configure data segment details within a system.

A Data Segment is a reusable, data-focused section of a maintenance RSO. It represents a logical group of fields or a small data capture/display area that can be embedded into maintenance screens.

1. Step 1: Component Creation

Creates the base definition of the data segment.

What you do on this screen:

- Select pattern type as Data Segment.
- Extended data segment: choose Yes/No (based on requirement and what your configuration supports).
- Select product type.
- Enter name and label.

Figure 4-1 Component Creation

The screenshot shows the 'Component Creation' screen in the Oracle Financial Services Application Studio. The interface includes a top navigation bar with 'Back', 'Component Creation', 'Configurations', and 'Create' steps. The main area is titled 'Component Creation' and contains the following fields:

- Pattern Type:** A dropdown menu with 'Datasegment' selected.
- Product Type:** A dropdown menu with 'Common Core' selected.
- Extended Datasegment:** A toggle switch with 'Yes' selected.
- Name:** A text field containing 'Interest-calculations'.
- Label:** A text field containing 'Interest Calculations'.

A 'Next' button is located in the top right corner.

2. Step 2: Endpoint Configuration

Links the data segment to runtime data and defines binding-related settings.

What you typically configure:

- Endpoint selection which has been created in the Service creation step.
- DTO/object mapping or binding configuration.
- The selected Endpoint will be used for data persistence.

Figure 4-2 Endpoint Configuration

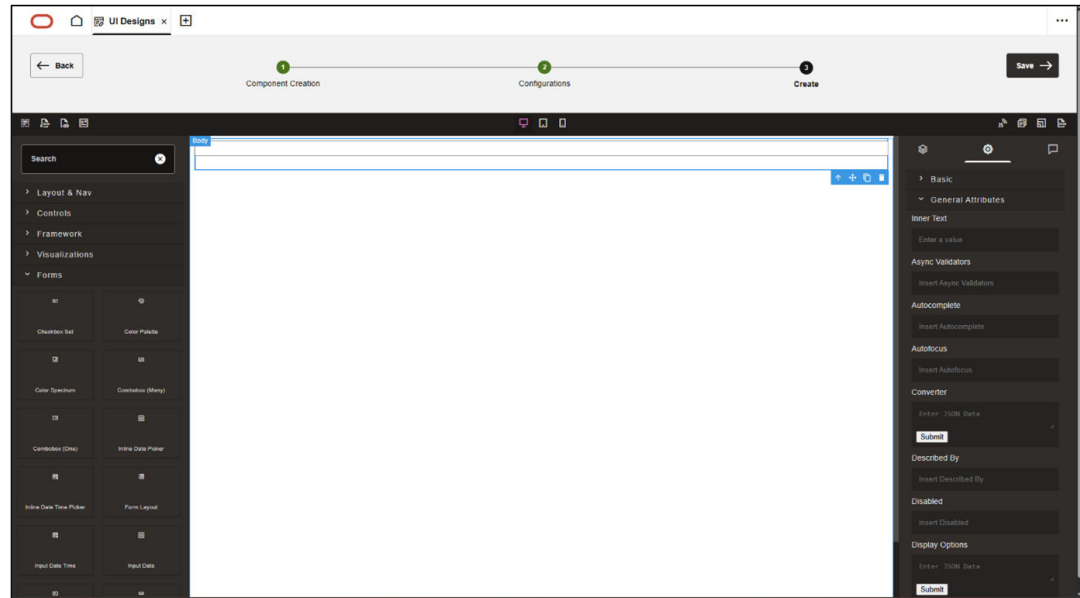
The screenshot shows the 'Endpoint Configuration' screen in the Oracle Financial Services Application Studio. The interface includes a top navigation bar with 'Back', 'Component Creation', 'Configurations', and 'Create' steps. The main area is titled 'Endpoint Configuration' and contains the following fields:

- Extended Datasegment Endpoint:** A dropdown menu with a list of endpoints: 'ORSTUDIO:OB_ACCOUNTCLASSEXTENSION', 'ORSTUDIO:OB_TEST_FIBS_NEWZ', 'ORSTUDIO:OB_MAINTENANCESERVICE', 'ORSTUDIO:OB_INTERESTCALCULATIONS', and 'ORSTUDIO:OB_MAINTENANCEDEMO'.
- Endpoint:** A text field containing 'Interest-calculations'.

A 'Next' button is located in the top right corner.

3. Step 3: Template Design

Designs the reusable UI block in the designer.

Figure 4-3 Template Design

What you typically do in the designer

- Add fields and controls relevant to the data block.
- Configure display behaviour and validations.
- Map UI elements to Service DTO Fields.

This page is the UI design workspace in OFS where you create or update a screen. You build the screen by dragging components from the left panel onto the central canvas and then configure each selected component using the settings panel on the right. It is mainly used to assemble the screen layout, set component properties (such as text, behaviour, and validations), and save the design so it can be published later. All database entries for the generated component are automatically added to the required tables after the user saves a component.

Figure 4-4 Datasegment Component Summary

[illegible]

4.2 Process Flow Datasegment

This topic provides information about the process flow for a data segment, outlining the sequence of steps involved in creating, managing, updating, and processing data segment operations within the system.

A process flow Datasegment is a reusable data-focused UI section intended to be used inside a flow/process journey. It helps standardize data capture/display across flow steps and supports reuse.

1. Step 1: Component Creation

Creates the base definition of a flow-specific Data segment.

What you typically do on this screen:

- Select pattern type as Process Flow Data segment.
- Select product type.
- Enter name and label.

What this step decides:

The component is intended for flow usage rather than as a standalone reusable block (the template and available options may differ based on your OFS setup).

Figure 4-5 Component Creation

The screenshot shows the 'Component Creation' wizard in Oracle Financial Services Application Studio. The wizard is titled 'UI Designs' and has three steps: 1. Component Creation, 2. Configurations, and 3. Create. The 'Component Creation' step is currently active. It contains four input fields: 'Pattern Type' (a dropdown menu with 'Process Flow Datasegment' selected), 'Product Type' (a dropdown menu with 'Common Core' selected), 'Name' (a text field with 'interest-calculations' entered), and 'Label' (a text field with 'Interest Calculations' entered). There are 'Back' and 'Next' buttons at the top of the wizard.

2. Step 2: Endpoint Configuration

Captures the configuration required for the Data Segment to bind correctly in a flow context.

What you typically configure:

- Endpoint/object/DTO settings as required for binding.
- Flow-related configuration options provided by your pattern.

3. Step 3: Template Design

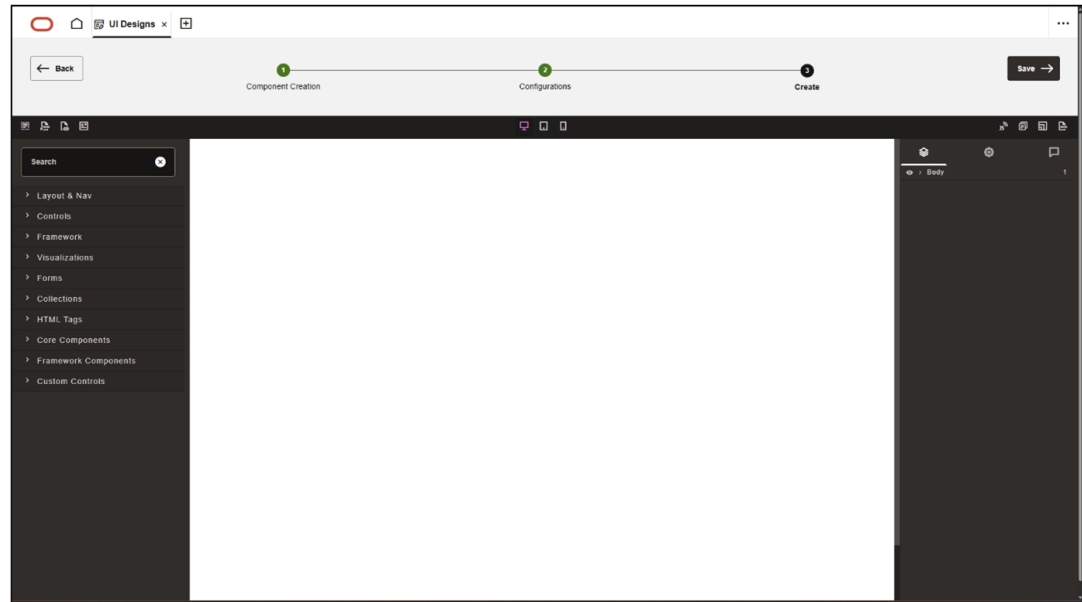
Builds the UI section in the designer editor.

What you typically do in the designer:

- Design the data segment layout (sections, fields, small grids if needed).
- Configure component properties (labels, behaviour, validations).

Keep the data segment focused on the specific data block to support reuse in multiple flow steps:

Figure 4-6 Template Design



multiple flow steps Important note about this page: the designer editor is the same across patterns; the starting structure differs due to init function initialization for Process Flow Data segment.

The init function is the pattern-specific initialization that runs when the Create/Designer page is opened for the first time (or when an existing design is loaded for editing). Its purpose is to generate boiler plate functions which are useful when you plug it into existing maintenance or process flow screens.

Figure 4-7 Process Flow Data segment Component Summary



4.3 Standalone Screen

This topic provides information about the standalone screen used to independently display and manage specific functionalities or data within the system without dependency on other interface components.

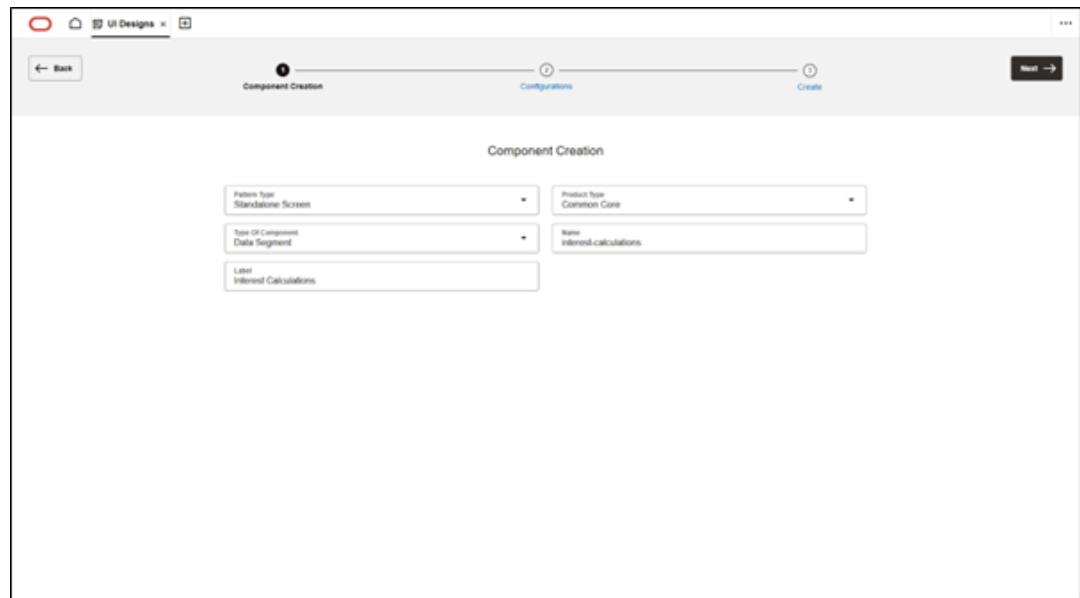
A standalone screen is a single, independent UI page/component. It is used when the user can complete the task in one view without navigating through multiple steps.

1. Step 1: Component Creation

What you do on this screen:

- Select pattern type as Standalone.
- Select product type (module context).
- Enter name (technical identifier) and label (display name shown in the UI).

Figure 4-8 Component Creation

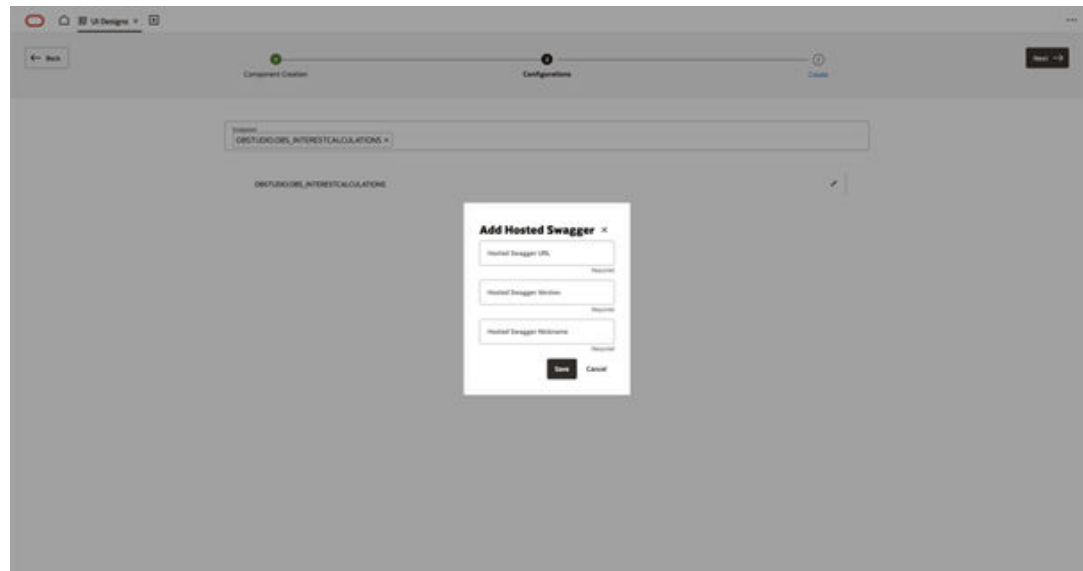


2. Step 2: Endpoint Configuration

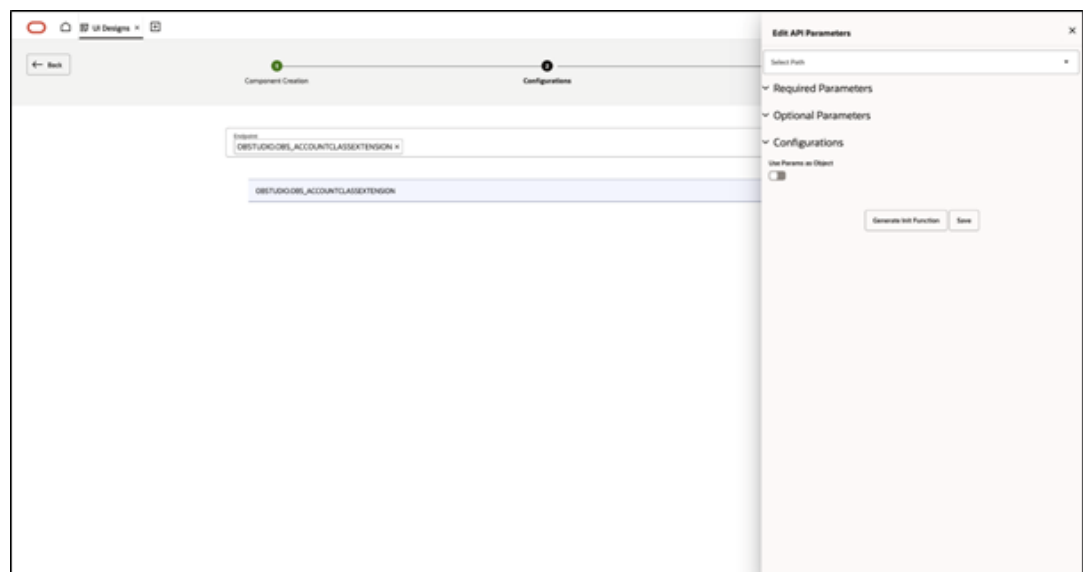
Captures runtime configuration needed for the screen to work with data and behave correctly.

What you typically configure (based on what is enabled in your OFS setup):

- Endpoint or data source selection (where the screen reads/writes data).
- Add a hosted swagger for endpoint mapping.

Figure 4-9 Endpoint Configuration

For each endpoint, add the required and optional parameters for each path in their swagger.

Figure 4-10 Edit API Parameters

The user can then optionally choose to generate the init function based on the configured path and parameters.

3. Step 3: Template Design

Builds the actual UI layout using the designer editor.

What you do on this screen:

- Add containers/sections to define structure.

- Add fields and controls for data capture/display.
- Add tables/grids.
- Configure buttons/actions (for example submit, cancel, next).
- Set properties for each selected component in the right panel (text, behaviour, validations, display options).

4.4 Maintenance Summary Screen

This topic provides information about the Maintenance Summary screen used to view, manage, and update summary details related to maintenance activities within the system.

The maintenance summary screen pattern is used when you need a maintenance RSO flow. Each maintenance RSO Screen contains of different stages with each stage containing its own Data Segment.

1. Step 1: Component Creation

Creates the base definition of the maintenance summary screen.

What you do on this screen:

- Select pattern type as Maintenance Summary Screen.
- Select product type.
- Enter name (technical identifier) and label (display name).

Figure 4-11 Component Creation

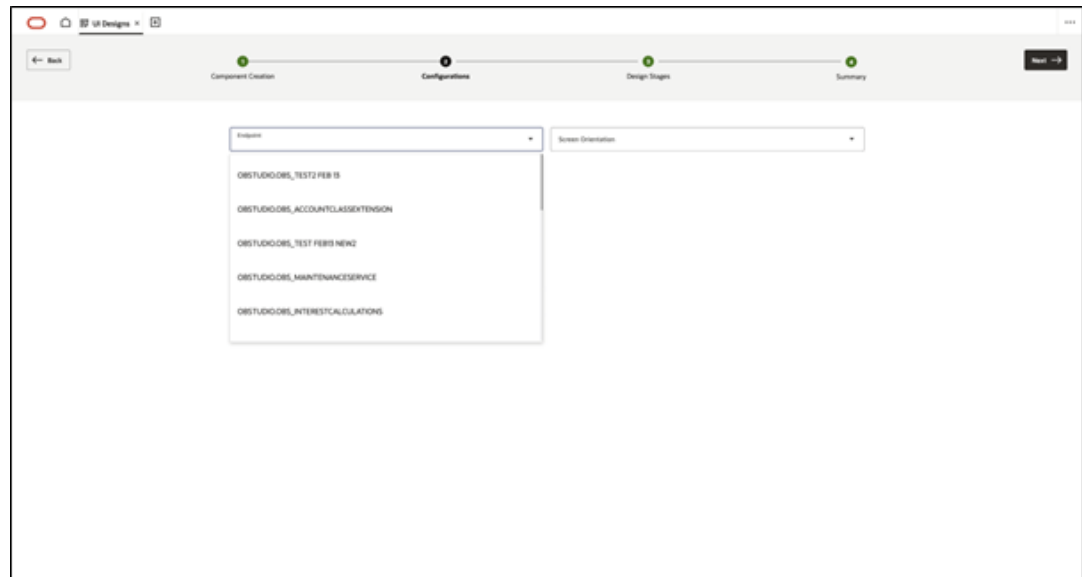
The screenshot displays the 'Component Creation' screen within the Oracle Financial Services Application Studio. At the top, a progress bar indicates four stages: 1. Component Creation, 2. Configurations, 3. Design Stages, and 4. Summary. The 'Component Creation' stage is currently active. Below the progress bar, the screen is titled 'Component Creation'. It contains four input fields arranged in a 2x2 grid: 'Pattern Type' (set to 'Maintenance Summary Screen'), 'Product Type' (set to 'Common Core'), 'Name' (set to 'interest-calculations'), and 'Label' (set to 'Interest Calculations'). A 'Back' button is located on the top left, and a 'Next' button is on the top right.

2. Step 2: Endpoint Configuration

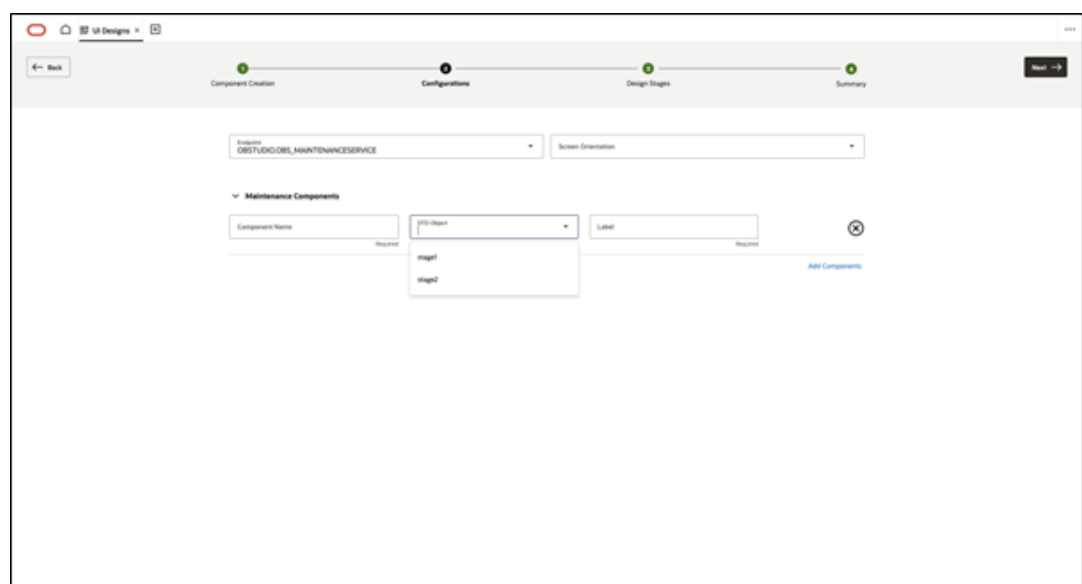
Links the design to its runtime data source and defines maintenance-specific component setup.

What you do on this screen:

- Endpoint: select the service/data endpoint used to fetch and maintain records.
- Screen orientation: choose how the maintenance layout is arranged (for example horizontal).

Figure 4-12 Endpoint Configuration**Maintenance components**

- **Component name:** technical identifier for the maintenance component.
- **DTO object:** select the DTO/entity used for data binding.
- **Label:** display name for the maintenance component.
- **Add components:** add additional maintenance component rows if the design requires more than one component.
- **Delete/remove:** remove a maintenance component row if not needed.

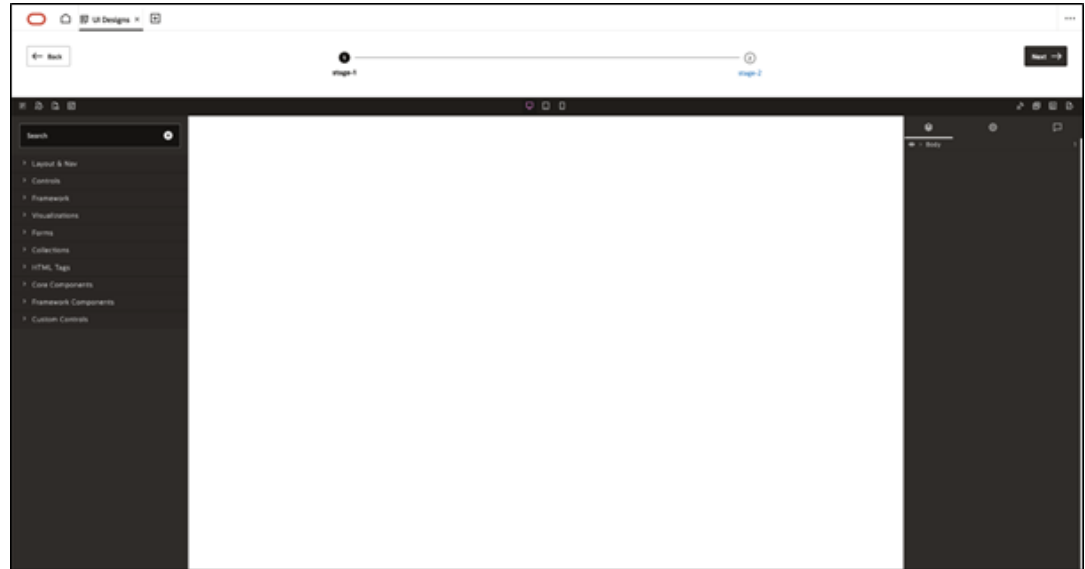
Figure 4-13 Maintenance components**3. Step 3: Template Design**

Builds and refines the maintenance summary UI in the designer.

What you typically do in the designer for maintenance summary:

- Configure the display fields for each stage.
- Add search/criteria fields (if not handled entirely in configuration/summary steps).
- Define action buttons (for example create, edit, view, delete, drill-down).
- Configure navigation to a related detail screen (if your implementation supports it).
- Apply field-level behaviour and validations where required.

Figure 4-14 Template Design



4. Step 4: Summary Screen

Final review before saving.

What you do on this screen (as seen in the summary page screenshot):

- **Endpoint:** Verify the selected endpoint.
- **Query Params (required):** Enter required query parameters expected by the endpoint.
- **Filter configurations:** Enable filter/search configuration using the toggle.
- **Template configurations:** Review or adjust template-level settings.
- **Save:** Save the final design.

Figure 4-15 Summary

4.5 Import Screens

This topic provides information about the import screens used to upload and integrate files, configurations, application packages, or resources into the system.

This dialog is used to import UI design artifacts into OFS from an import file/package. It is typically used to move screens between environments or to restore screens from an exported backup.

Figure 4-16 Import Screens

1. How to use

- a. **Drag and drop:** drop the import file into the drag-and-drop area, or select the file from your system (based on browser behaviour).
- b. **Selected files:** shows the file(s) selected for import.
- c. **Save:** starts the import process and loads the screens into the UI Designs list.
- d. **Close:** exits the dialog without importing.

2. Steps

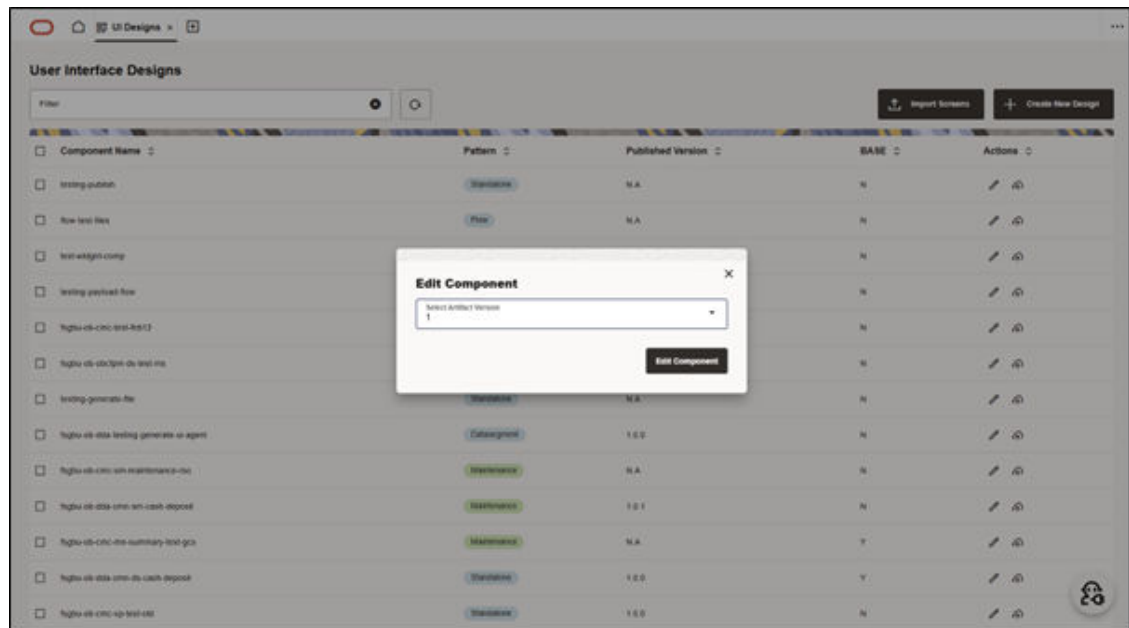
- On the **User Interface Designs** page, click **Import Screens**.
- **Drag and drop** the import file into the dialog (or select it).
- Click **Save** to import.
- Refresh the list to confirm the screens are available.

4.6 Edit Component

This topic provides information about the process of modifying or updating an existing component within a system.

This dialog is used to open an existing UI design component for editing. It allows you to choose which saved artifact version you want to edit before launching the designer.

Figure 4-17 Edit Component



Steps

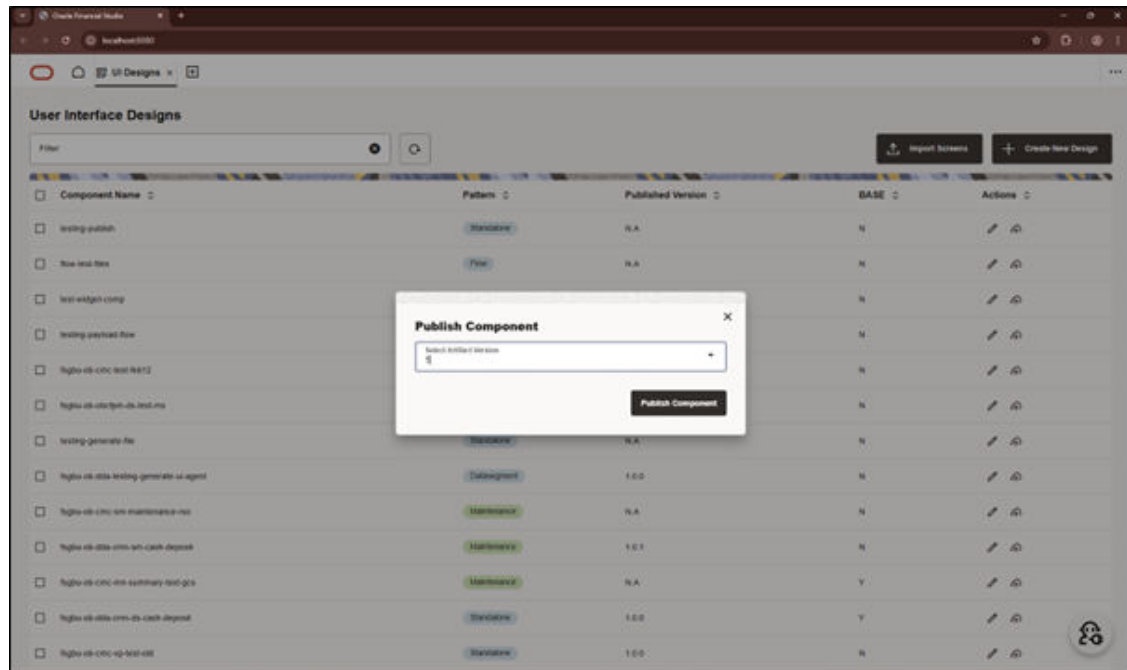
- On the **User Interface Designs** list page, click the **edit** (pencil) icon for the required component.
- Select the artifact version to edit.
- Click **Edit Component** to open the design in the designer page.
- Save your changes in the designer and publish again.

4.7 Publish Component

This topic provides information about the process of making a component available for use within the system.

This dialog is used to publish a selected UI design version, so it becomes the published version of the component and is available for use as per the runtime/lifecycle configuration.

Figure 4-18 Publish Component



Steps

- On the **User Interface Designs** list page, click the **publish** (upload) icon for the required component.
- Select the artifact version to publish.
- Click **Publish Component**.
- Confirm the Published Version column is updated in the list page after refresh.

Audit Logs

This topic provides information about audit logs used to track and record system activities, user actions, and configuration changes within the application.

The Audit Logs section provides a centralized view of all API activity associated with OFS-generated components. It captures and displays every API call initiated from OFS as well as from the App Shell, across all components created using OFS. These logs help teams monitor service usage, validate end-to-end integration, and support operational troubleshooting by offering traceability into when APIs were invoked and how OFS-based components are interacting with the backend services.

- [Tabular View](#)
This topic provides information about the tabular view used to display data in a structured table format within the system.
- [Visual Overview](#)
This topic provides information about the visual overview.

5.1 Tabular View

This topic provides information about the tabular view used to display data in a structured table format within the system.

Figure 5-1 Tabular View

Audit Logs

Filter

Filter

Tabular View

Visual Overview

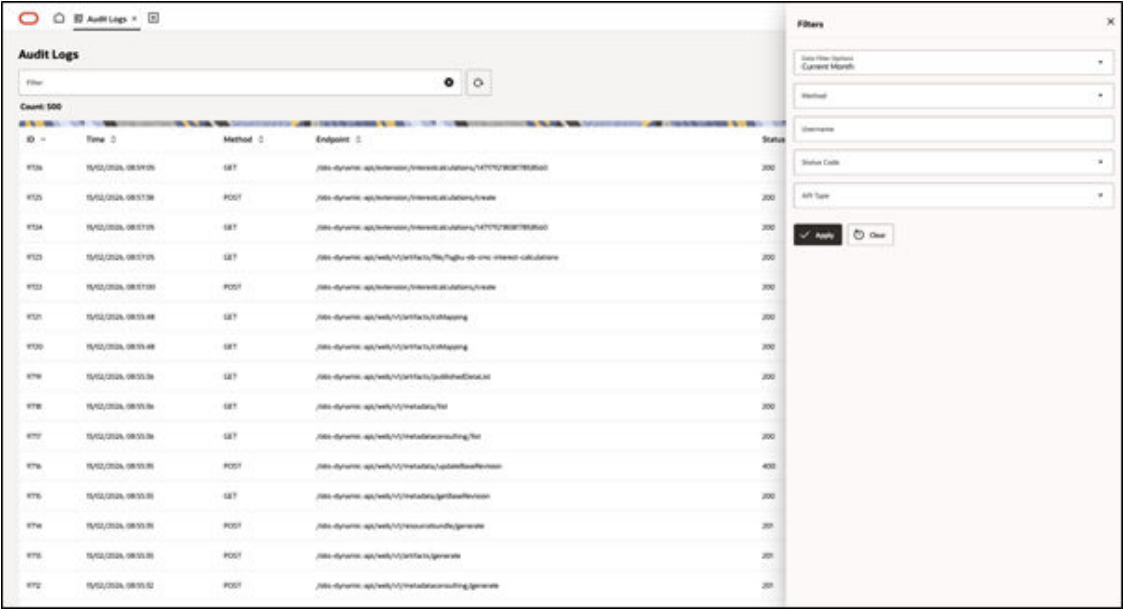
Count: 100

ID	Time	Method	Endpoint	Status	API Type	Duration (ms)	Details
R09	15/02/2024, 08:04:05	GET	/info-dynamic-app/demonstration/environments/all/collections?TPTXZBQD7W0GSAQ	200	Success	45	Details
R05	15/02/2024, 08:07:38	POST	/info-dynamic-app/demonstration/environments/all/collections/crashable	200	Success	19	Details
R04	15/02/2024, 08:07:05	GET	/info-dynamic-app/demonstration/environments/all/collections?TPTXZBQD7W0GSAQ	200	Success	43	Details
R22	15/02/2024, 08:07:05	GET	/info-dynamic-app/web/vv/api/stats/mq/higher-ids-view-mixed-collections	200	Failure	3	Details
R23	15/02/2024, 08:07:00	POST	/info-dynamic-app/demonstration/environments/all/collections/crashable	200	Success	0	Details
R01	15/02/2024, 08:05:48	GET	/info-dynamic-app/web/vv/api/stats/mapping	200	Failure	3	Details
R00	15/02/2024, 08:05:48	GET	/info-dynamic-app/web/vv/api/stats/mapping	200	Failure	3	Details
R18	15/02/2024, 08:05:34	GET	/info-dynamic-app/web/vv/api/stats/publishedData.txt	200	Failure	24	Details
R19	15/02/2024, 08:05:34	GET	/info-dynamic-app/web/vv/api/stats/test	200	Failure	12	Details
R07	15/02/2024, 08:05:34	GET	/info-dynamic-app/web/vv/api/stats/connecting-test	200	Failure	36	Details
R16	15/02/2024, 08:05:35	POST	/info-dynamic-app/web/vv/api/stats/updatedDescription	400	Failure	4	Details
R15	15/02/2024, 08:05:35	GET	/info-dynamic-app/web/vv/api/stats/generateDescription	200	Failure	4	Details
R14	15/02/2024, 08:05:35	POST	/info-dynamic-app/web/vv/api/stats/build/generate	201	Failure	50	Details
R13	15/02/2024, 08:05:35	POST	/info-dynamic-app/web/vv/api/stats/generate	201	Failure	4	Details
R12	15/02/2024, 08:05:32	POST	/info-dynamic-app/web/vv/api/stats/connecting/generate	201	Failure	10	Details

The first section is a tabular view that provides a consolidated list of all API calls made for OFS-generated components. Each row represents an individual API invocation and includes key execution details such as the timestamp (when the call was made), the API method (for example, GET/POST/PUT/DELETE), the endpoint that was invoked, the response status, and

the API type indicating whether the request originated from the App Shell or directly from OFS. The table also captures performance information, including the duration taken to receive the response. A dedicated Details option is available for each entry, allowing users to review the request and response payloads and view any exceptions or error information captured during execution. To help locate specific calls quickly, the table includes filtering capabilities that allow users to filter results using any of the available columns.

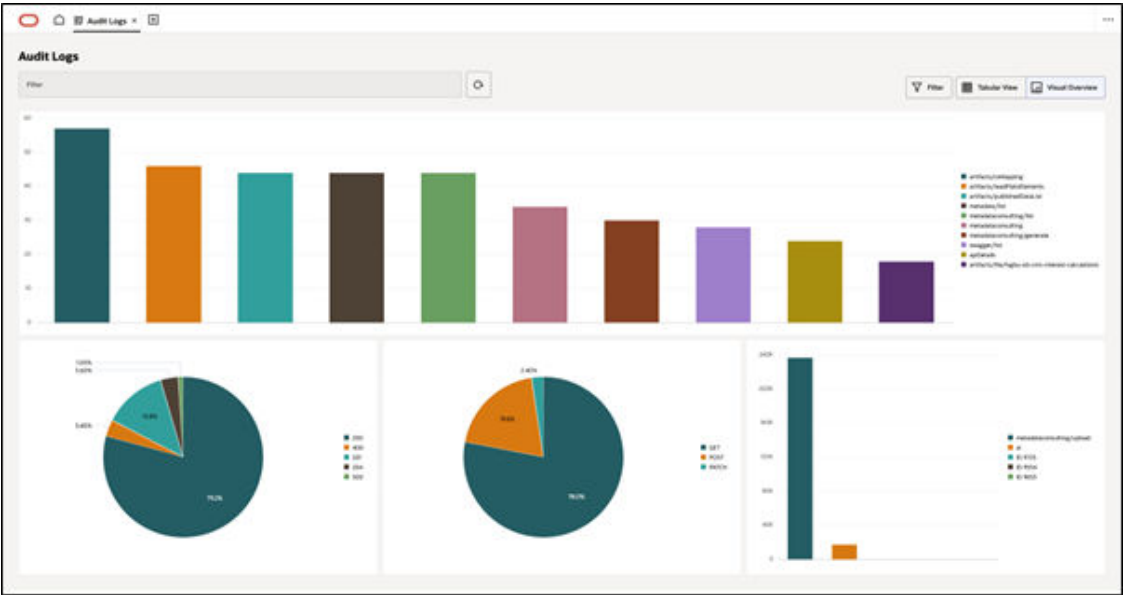
Figure 5-2 Filters



5.2 Visual Overview

This topic provides information about the visual overview.

Figure 5-3 Graphical View



The second section is the Visual Overview, which provides a high-level, graphical summary of the API activity captured in the audit logs. Instead of reviewing individual entries one by one, users can use this dashboard-style view to quickly understand overall usage patterns, identify heavily utilized services, and spot potential reliability or performance concerns briefly. The graphs in this section are designed to make trends easy to interpret and to support faster operational analysis and decision-making.

The Visual Overview includes the following insights:

- **Top 10 Most Hit APIs:** Displays the ten endpoints that receive the highest number of requests. This helps teams understand which services are most actively used across OFS-generated components and can be useful for capacity planning, usage validation, and identifying critical endpoints that may require closer monitoring.
- **Most Hit API Methods:** Shows the distribution of requests by HTTP method (such as GET, POST, PUT, and DELETE). This view helps characterize how the services are being used—whether the traffic is primarily read-heavy, write-heavy, or a balanced mix of operations.
- **Most Common Response Status Codes:** Highlights the response outcomes (for example, 2xx, 4xx, 5xx), allowing users to quickly gauge the overall health of API executions and to detect patterns that may indicate frequent client-side validation issues or server-side failures.
- **Top 5 Slowest APIs:** Identifies the endpoints with the highest response times, enabling teams to quickly focus on potential performance bottlenecks. This can support troubleshooting efforts by indicating where latency is most pronounced and which services may require optimization or deeper investigation.

6

Integrated AI Assistant

This topic provides information about the integrated AI assistant used within the system to support users with guidance, automation, and intelligent recommendations.

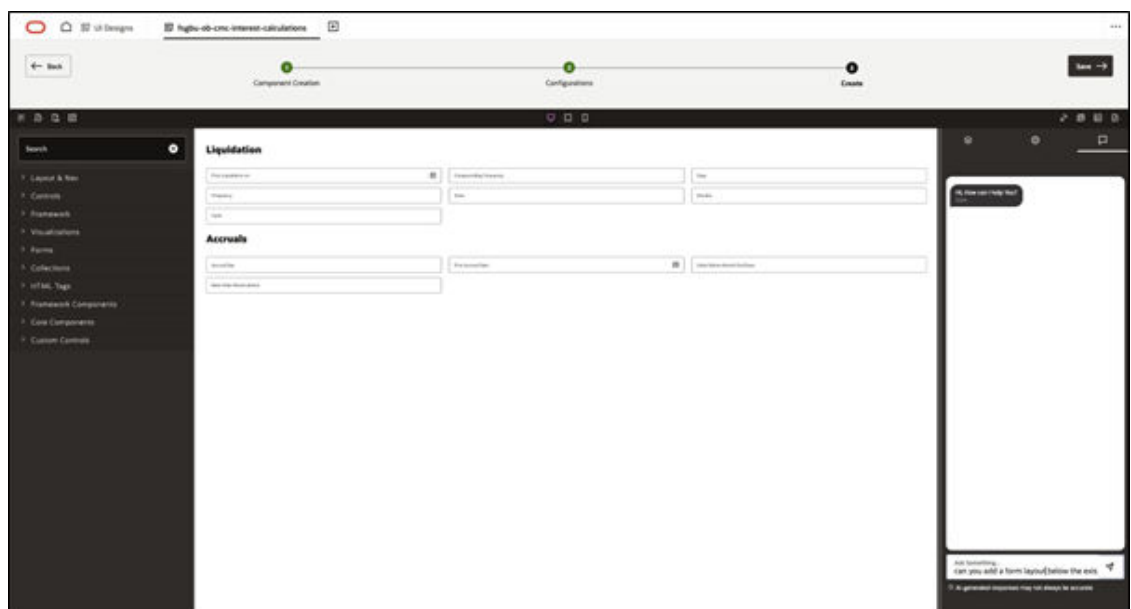
AI integration in OFS is designed to simplify and accelerate the creation of user interfaces and workflows by reducing the amount of manual effort required from users. By analysing the provided requirements and context, OFS can intelligently recommend and generate UI components, streamline key configuration steps, and promote consistent design patterns across implementations. These AI-driven capabilities help shorten overall development and setup time while improving standardization and reducing the likelihood of configuration errors. In addition, AI-powered mapping automatically associates UI elements with the appropriate data fields or service parameters, minimizing repetitive manual mapping and enabling faster, more reliable end-to-end integration.

- [Chat based UI design generation](#)
This topic provides information about chat-based UI design generation used to create user interface designs through conversational interactions.
- [Chat based UI and Service workflow configuration](#)
This topic provides information about chat-based UI and service workflow configuration used to set up user interfaces and service processes through conversational interactions.

6.1 Chat based UI design generation

This topic provides information about chat-based UI design generation used to create user interface designs through conversational interactions.

Figure 6-1 Chat based UI design generation



One of the key AI-driven capabilities in OFS is chat-based UI design generation, which helps users create UI templates quickly with minimal manual setup. From the Create Design Template page, users can enter prompts in natural language to describe the type of screen they want to build—for example, the purpose of the form, the kind of information to capture, and the overall layout expectations. Based on this input, the AI Assistant analyses the prompt and automatically generates a corresponding UI template, providing a ready-to-use starting point and reducing the need to build the structure from scratch.

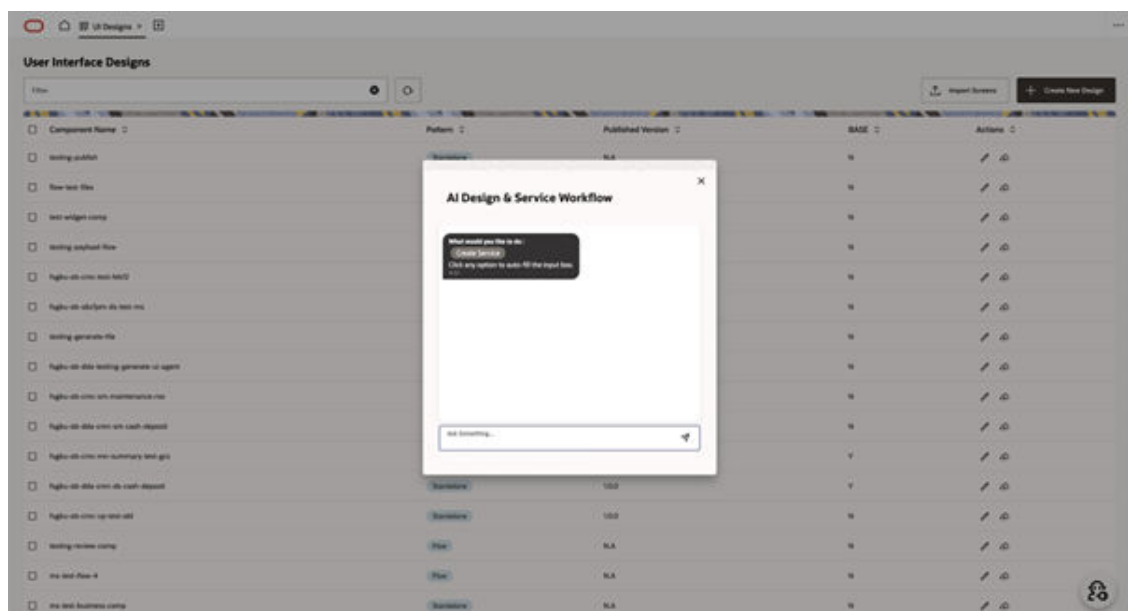
This approach significantly streamlines early-stage UI development by accelerating initial scaffolding and encouraging consistent patterns in how screens are created. Once the template is generated, the remaining user effort is primarily focused on mapping—associating the fields exposed by the selected endpoint with the generated UI components so that each UI element is correctly bound to the appropriate request/response attributes for data persistence.

At present, chat-based UI generation is available for a limited set of form-oriented OJET components, primarily covering commonly used input controls and form layout structures.

6.2 Chat based UI and Service workflow configuration

This topic provides information about chat-based UI and service workflow configuration used to set up user interfaces and service processes through conversational interactions.

Figure 6-2 AI Design and Service workflow



The second AI-driven capability in OFS is chat-based service and UI creation with automated configuration, designed to minimize manual steps and accelerate end-to-end setup. Instead of separately defining a service, creating DTO fields, building a UI, and then performing field-by-field mapping, users can describe their requirements in a conversational chat flow. The user's primary task is to specify the service they want to create and define the necessary DTO field structure through the chat interface (for example, listing the fields, their data types, and any key attributes required for the use case).

Based on this information, the AI automatically generates the service artifacts and creates a corresponding UI that is aligned with the service definition. The generated UI includes the required input components for the DTO fields (for example, text inputs for string fields, number

inputs for numeric fields, and date inputs for date fields), providing an immediately usable form-like screen. In addition to creating the UI structure, the feature also performs automatic mapping, linking each UI element to its respective DTO field so that data entered on the screen is correctly bound to the service payload for downstream processing and persistence.

By automating both creation and configuration, this capability reduces repetitive work, lowers the chance of mapping or configuration errors, and helps maintain consistency between the backend service contract and the front-end experience. It enables users to move more quickly from an initial requirement to a functional service-and-UI implementation that can then be refined or extended as needed.

7

OBMA App-Shell

This topic provides information about the application shell used in OBMA to support the overall framework and navigation structure of the application.

Once a component is published in OFS, it becomes available for consumption within the OBMA App Shell. Publishing packages the component in a form that is ready for integration, enabling it to be plugged into the App Shell and used as part of an end-to-end application flow. This allows teams to take components created and configured in OFS, along with their associated UI templates, service bindings, and persistence behaviour, and seamlessly reuse them within the broader OBMA application framework.

Steps to Add Data Segment to Maintenance RSO Screen.

1. **View Data Segment Screen:** Authorize the published Data Segment.

Figure 7-1 View Data Segment

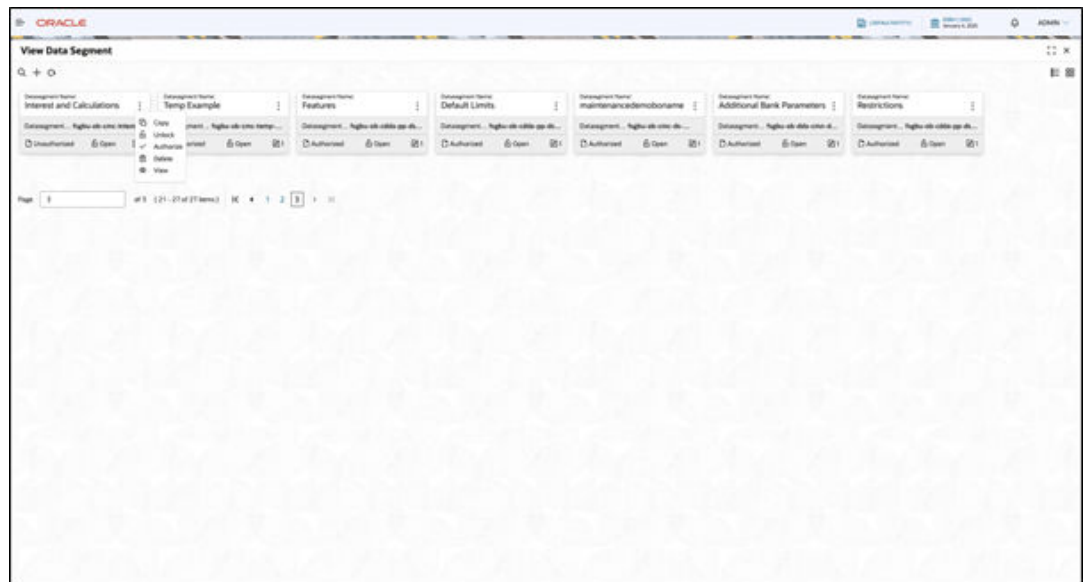
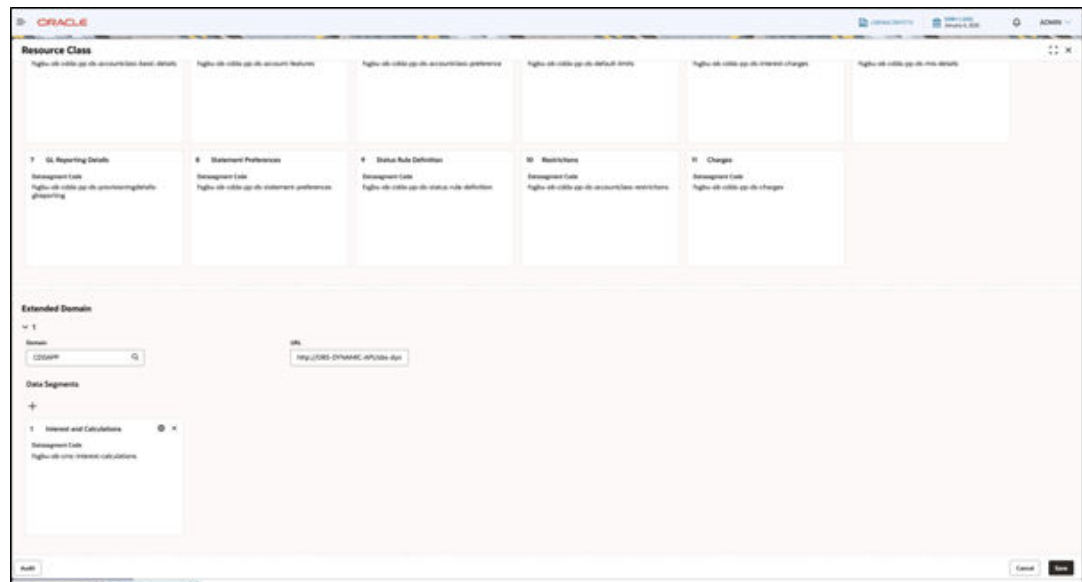


Figure 7-2 View Data Segment_Authorize



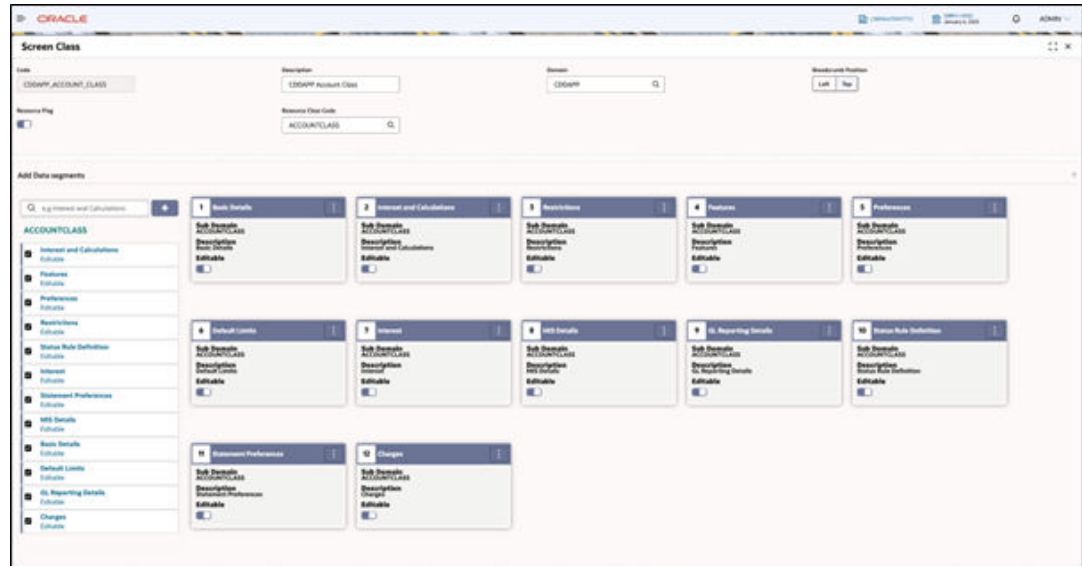
2. **Resource Class Screen:** In the resource class page, add the published data segment in the extended domain section. The URL for the extended domain should be the url of the OFS extension API. For eg. "http://OBS-DYNAMIC-API/obs-dynamic-api/extension/interestcalculations" for Data Segment Interest Calculations.

Figure 7-3 Resource Class Screen



3. **Screen Class Screen:** In the screenclass screen, drag and drop the data segment to the position that you want it to be in the maintenance flow.

Figure 7-4 Screen Class



4. **Account Class:** Create a new account class in which you will see the data segment present at the position where you added it.

Glossary

Index

A

Audit Logs, [1](#)

C

Chat based UI and Service workflow configuration, [2](#)

Chat based UI design generation, [1](#)

D

Datasegment Screen, [1](#)

Delete Service, [8](#)

Download Service, [6](#)

E

Edit Component, [12](#)

Edit Service, [7](#)

I

Import Screens, [12](#)

Import Service, [5](#)

Integrated AI Assistant, [1](#)

Introduction, [1](#)

M

Maintenance Summary Screen, [9](#)

O

OBMA App-Shell, [1](#)

Overview, [1](#)

P

Page Controls, [2](#)

Process Flow Datasegment, [4](#)

Publish Component, [13](#)

S

Service API Creation, [1](#)

Service creation for Data Segment, [2](#)

Service for Individual Component, [3](#)

Service for Maintenance Summary Component, [4](#)

Standalone Screen, [7](#)

T

Tabular View, [1](#)

U

UI Design Patterns, [3](#)

UI Design Patterns Creation, [1](#)

UI Designs – User Interface Designs, [1](#)

V

Visual Overview, [2](#)