SAP Fiori Launchpad

THE BEST RUN **SAP**

Content

1 SAP Fiori Launchpad

SAP Fiori launchpad is a shell that hosts apps built with different UI technologies, and provides these apps with services such as navigation, personalization, embedded support, and application configuration.

What is SAP Fiori launchpad?

SAP Fiori launchpad is the entry point to apps on mobile and desktop devices. It gives users the relevant information they need at a glance and centers on their needs and how they work.

The new design of the launchpad displays a panoramic user interface. Users can focus on their daily work and still keep track of what's going on in other areas.
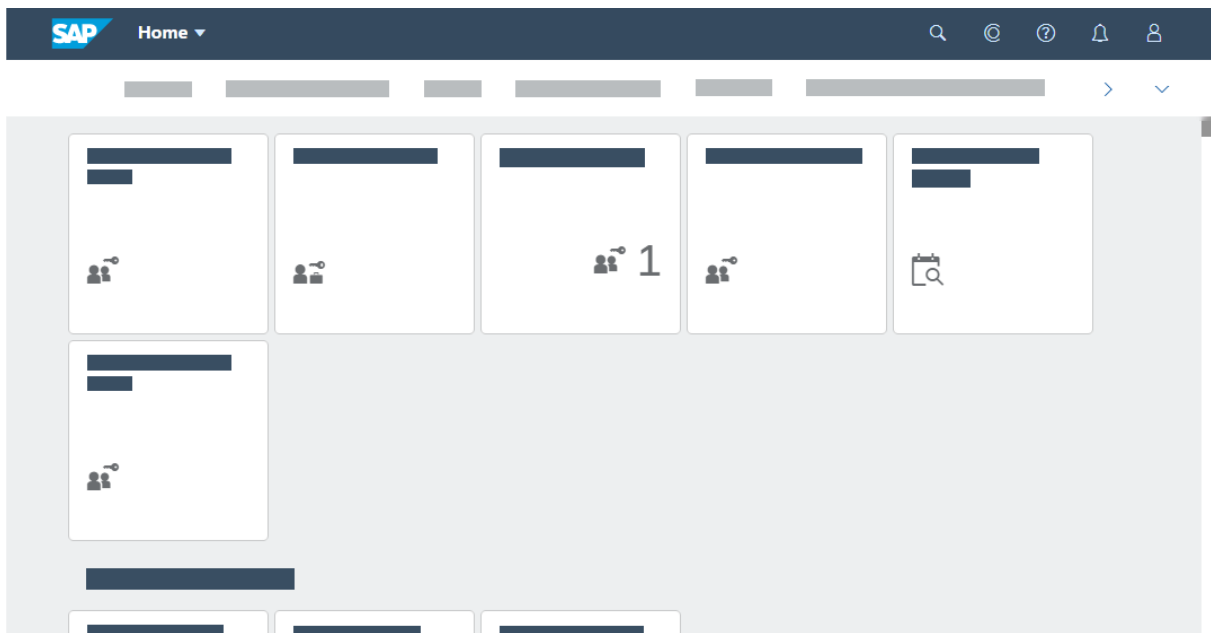
Where can I find more information?

This image is interactive. Hover over each section for a description. Click the highlighted sections for more information.

- Administration Guide [page 8]
- User Guide [page 412]
- Developer Guide [page 518]
- A short description of new features or changes since the previous release [page 5]
- Security Aspects [page 627]
- Troubleshooting [page 631]

More information about the different personas

Persona	What do they do?
Administrators	Administrators set up and configure the launchpad. They also create and configure content and adapt it if necessary to enable end users to optimally use the launchpad for their daily tasks.
End Users	End users carry out their daily tasks in the launchpad. They can search for apps and add them to their home page, personalize their settings, adapt content to suit their business scenario and collaborate with their colleagues.
Developers	Developers can customize the launchpad user interface by using various APIs to create buttons, add user options, and activating other features.

About SAP Fiori

For more information, see *SAP Fiori* on SAP Help Portal at http://help.sap.com/fiori.

Open the Road Map Explorer🔧 and go to the relevant product, e.g. *ERP and Finance*, then *SAP S/4HANA*. Here, you can e.g. search for "Launchpad".

For new users we recommend the tutorial SAP Fiori Launchpad. In this five-minute tutorial you learn how to work with the launchpad and get to know the most important features.

Related Information

SAP Fiori Launchpad Features [page 6]

1.1   What's New

Get an overview of the main new features and enhancements in SAP Fiori launchpad.

See What's New Viewer.

1.2 SAP Fiori Launchpad Features

SAP Fiori launchpad offers many key features that can be used by end users, administrators, and developers.

Key Feature	Use

Intuitive design and layout	A user interface that includes:
	
	• Shell header and footer areas that provide icons and buttons for users to perform various actions, such as the search, a home page, SAP Jam interactions, and app navigation in the header.
	• Two layout options to present the apps in the user's main working area:
	   • Classic home page, displaying app tiles and links arranged in groups.
	   • Pages and spaces: Spaces are used to combine several related pages. The pages display related apps. Pages and spaces can be configured by the customer adminstrator and tailored to suit the needs of the users.
	• User actions menu that provides access to user-related information and actions including personalization, profile settings, an interaction history, and the option to contact support.
	• Notifications that inform users about important business tasks and requests that require their timely action or knowledge.
	
	See: About the Launchpad [page 413] and Using the Launchpad [page 429]

Multi-device support	The responsive patterns and controls of SAPUI5 and the adaptive design of apps allows the launchpad to run on desktops, tablets, smartphones, and hybrid devices. The launchpad and its apps accommodate the resolution, image size, and scripting on-the-fly, as users switch between devices, allowing them to work how and where they want.
Flexible branding	Users can change the visual appearance of the launchpad by selecting a theme of their choice: • SAP provides standard themes, such as SAP Quartz Light and Dark, SAP Belize, SAP Belize Deep, High-Contrast Black, and High-Contrast White. • Organizations can create custom themes based on the standard SAP themes. See: Managing Your Settings [page 509] and Adjusting the Visual Appearance of the Launchpad [page 70]
Enhanced search capabilities	Using SAP Enterprise Search and the app finder tool, users can quickly find apps and objects relevant to their daily tasks. From here, users can add apps to their home page in the launchpad. See: App Finder [page 427] and Searching for Business Objects and Apps [page 433]

Key Feature	Use
Cross-app and in-app navigation	Provides users with a vast array of navigation paths to move between their apps and perform their daily tasks. For example: • Within apps using a hierarchical navigation menu. • Between apps and the group selection bar at the top of the home page. • Opening the home page from any app. • Navigating to related apps. • Navigating to home page groups and catalogs from anywhere in the launchpad using the All My Apps tool. • Keyboard shortcuts to quickly navigate to different areas in the launchpad. See: Navigating Within and Between Apps [page 436], Usability Features in the Launchpad [page 445], and Setting Up Navigation [page 127]
Personalized user experience	Allows users to personalize their own launchpad experience by: • Adding, removing, renaming, and rearranging tiles in groups on the home page. • Adding, removing, renaming, hiding, and moving groups. • Changing the visual appearance and functionality of the launchpad. See: Personalizing the Launchpad [page 496]
Key user adaptation	Key users can change the user interface of SAP Fiori apps directly in the launchpad, intuitively, and without having to write new code. See: Adapting SAP Fiori UIs at Runtime - Key User Adaptation [page 459] and Enabling Key User Adaptation [page 75]
Admin customizations and configurations	Determine runtime behavior and functionality for end users by customizing a multitude of global configuration options in the launchpad's design time, or by integrating supporting SAP products. For example: • Creating and configuring tiles, catalogs, and groups using the SAP Fiori launchpad designer. • Enabling the display of users' profile pictures or avatars in various controls in the SAP Fiori launchpad. • Enabling UI adaptation by users at runtime. • Enabling or disabling optional features for end users, for example, the ability to personalize the home page or contact support. See: Initial Setup of the Launchpad [page 10] and Managing Launchpad Settings [page 29]
UI client integration	Integrate the launchpad with other UI clients, such as SAP Business Client, SAP Enterprise Portal, SAP Fiori client, and Web browsers. For example, you can run the launchpad embedded in the SAP Business Client, which allows you to call transactions in SAP GUI for Windows, and then gain additional features compared to those provided with SAP GUI for HTML. See: Integrating the Launchpad and Launchpad Content with Other UI Clients [page 340]

Key Feature	Use
Development	Develop and extend the launchpad by: • Embedding apps that have been developed with various UI technologies into the launchpad; for example, SAPUI5 Fiori, Web Dynpro ABAP, and SAP GUI for HTML. • Consuming launchpad services in SAPUI5 apps, for example to develop navigation across apps. • Extending the functionality of the launchpad with custom features using APIs, such as adding new elements to the user interface. See: Developing Applications for the Launchpad [page 519] and Extending the Launchpad [page 575]

> i Note
>
> Some end user options listed here are only available if enabled by the launchpad administrator.

1.3 Administration Guide

This guide provides information about administration and configuration tasks, and about the tools used to implement them.

This image is interactive. Hover over each section for a description. Click highlighted sections for more information.
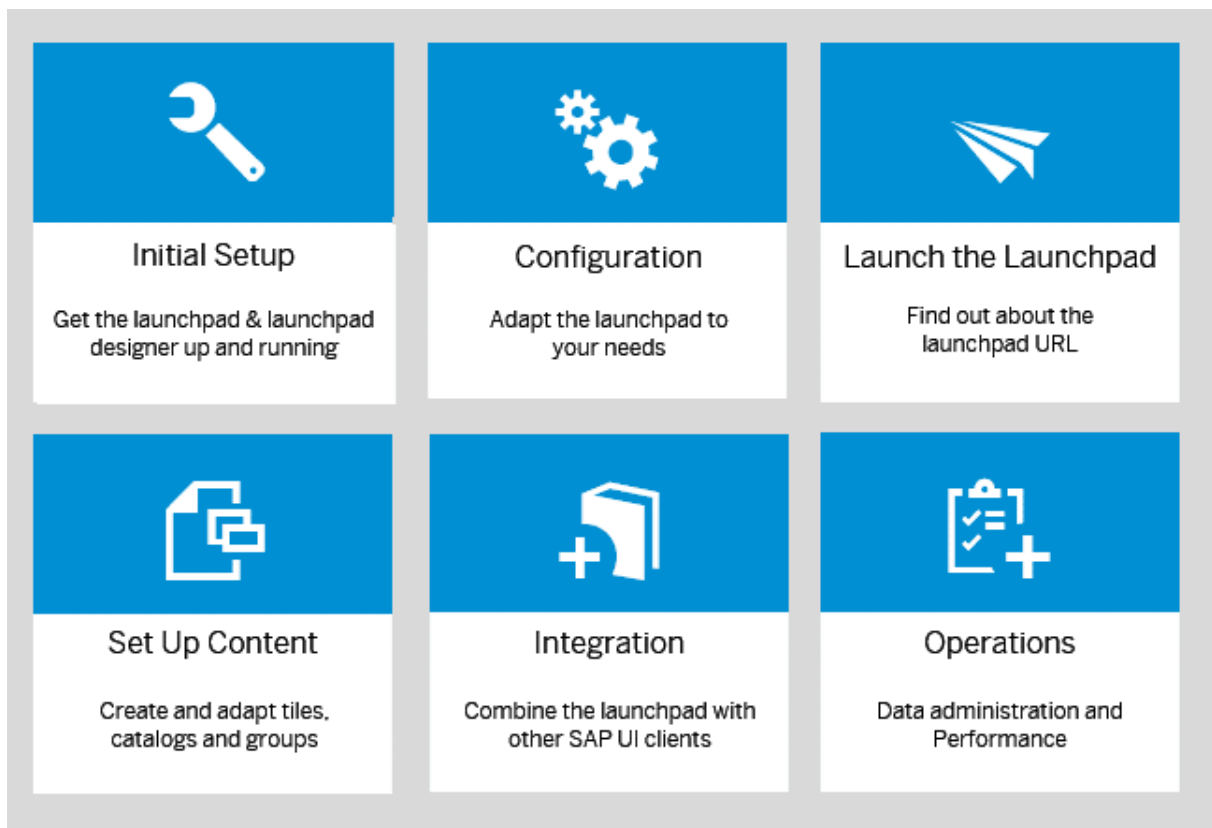
- Initial Setup of the Launchpad [page 10]
- Managing Launchpad Settings [page 29]
- Launching the Launchpad [page 101]
- Setting Up Launchpad Content [page 102]
- Integrating the Launchpad and Launchpad Content with Other UI Clients [page 340]
- Operations [page 355]

Related Information

SAP Fiori Implementation Information
SAP Fiori Launchpad [page 3]
User Guide [page 412]
Developer Guide [page 518]
Security Aspects [page 627]

1.3.1 Initial Setup of the Launchpad

This section describes the initial configuration settings for the launchpad and launchpad designer.

> **i Note**
>
> To properly display SAP transactions in SAP GUI for HTML from the launchpad, the browser needs to display the SAP GUI in standard mode. The system in which the SAP GUI for HTML is started, requires SAP kernel 7.40 patch level 38 or higher.

Related Information

1.3.1.1 Deployment Options

SAP Fiori setup scenarios in fact denote the deployment location of the SAP Fiori front-end UI components with respect to the location of the backend components.

The following initial setup scenarios are valid when a new system is configured:

- **Embedded deployment**
  This scenario features an SAP Fiori front-end server (and the associated components) deployed along with the backend components in the same system.
- **Standalone (hub) deployment**
  This scenario represents a decoupled setup of an SAP Fiori front-end server deployed separately on a standalone system apart from the backend components.

> **i Note**
>
> When you set up a new SAP S/4HANA on-premise system, the recommended setup of SAP Fiori is the embedded scenario.
>
> Additionally, the embedded scenario features various automatic task lists for rapid activation of SAP Fiori content. These task lists facilitate the activation of SAP Fiori for SAP S/4HANA.

Related Information

SAP Fiori Deployment Options and System Landscape Recommendations
Deployment Options

1.3.1.2 Initial Setup for Embedded Deployment

You can perform ABAP system configuration tasks in an automated way by using predefined task lists.

For SAP Fiori, task lists support you in setting up and configuring the communication channels between the client, the front-end, and the back-end servers.

For more information, see Embedded Deployment (Recommended)

1.3.1.3 Initial Setup for Standalone (Hub) Deployment

For more information, see Initial Setup for Standalone (Hub) Deployment

1.3.1.4 Additional Setup Activities

1.3.1.4.1 Configuring SAP Web Dispatcher

The following is to give you an example of how to configure your Web Dispatcher.

- 1 SAP Web Dispatcher as proxy to map in and outgoing requests for different systems
- 1 SAP Gateway server (optional if using external gateway)
- 1 Front-end system

In the example below SAP Web Dispatcher listens at port 443 and transfers:

- all requests which start with */sap/opu/odata/* to the Gateway system
- all requests which start with */sap/public/bc or /sap/bc* to the front-end system

> i Note
>
> The example given is not complete and must be adapted according to your requirements.

For more information on SAP Web Dispatcher, see http://help.sap.com/nw_platform. Select a release. Under *Application Help*, open *Function-Oriented View* and choose ▶ *Application Server* ▶ *Application Server Infrastructure* ▶ *SAP Web Dispatcher* ▶ .

Configuration Settings for SAP Web Dispatcher

When configuring the SAP Web Dispatcher define the example values below according to your requirements. To implement your configuration you can copy the source code below and adapt it according to your business requirements.

```
#https settings
ssl/ssl_lib = <sapcrypto dll>
ssl/server_pse = <pse>
icm/HTTPS/verify_client = 0
ssl/client_pse = <pse>
wdisp/ssl_encrypt = 1
wdisp/ssl_auth = 2
wdisp/ssl_cred = <pse>
wdisp/ssl_certhost = <host name>
icm/HTTPS/forward_ccert_as_header = true

# SAP Web Dispatcher Ports
icm/server_port_0 = PROT=HTTPS,PORT=443,TIMEOUT=120
wdisp/system_conflict_resolution = 1
wdisp/add_clientprotocol_header = 1
wdisp/handle_webdisp_ap_header = 1
wdisp/add_xforwardedfor_header = true

# Configuration for Gateway Server(For example, SAP Business Suite on HANA):
wdisp/system_0 = SID=<GW SID>, MSHOST=<GW MSHOST>, MSPORT=<GW MSPORT>,
SRCSRV=*:443, SRCURL=/sap/opu/odata/
# Configuration for Front-End Server:
wdisp/system_1 = SID=<FE SID>, MSHOST=<FE MSHOST>, MSPORT=<FE MSPORT>,
SRCSRV=*:433, SRCURL=/sap/public/bc/;/sap/bc/
```

> **i Note**
>
> If you want to use the Web GUI applications on the front end servers then you have to append this additional path */sap/public/icmandir/* to the *SRCURL* of the frontend server.

The table explains the example values used in the source code above:

Variable	Description	Example Value
<sapcrypto dll>	Installation path	C:\sapwebdisp\sapcrypto.dll
<pse>	Folder for PSE file	C:\sapwebdisp\sec\PSE_UI2.pse
<host name>	Fully qualified host name of dispatcher	mydispatcher.example.com
<GW SID>	Gateway SID	ABC
<GW MSHOST>	Gateway message server host name	abc.example.com
<GW MSPORT>	Gateway message server port	8120
<FE SID>	Front-end SID	DEF
<FE MSHOST>	Front-end message server host name	def.example.com
<FE MSPORT	Front-end message server port	8131

Related Information

1.3.1.4.1.1 Configuring SAP Web Dispatcher for Search

To enable the search for business objects in the launchpad, you need to make some special settings for the SAP Web Dispatcher.

> **i Note**
>
> You only need to make these settings if your frontend system and client (where you log on) are different from the search backend system and client.

Add the embedded search backend system as follows to the `sapwebdisp.pfl` profile:

```
wdisp/system_<NUM> = SID=<BE SID>, EXTSRV=<BE HOST>:<BE PORT>, SRCSRV=<RD
PORTS>,
SRCURL=/sap/es/ina/;/ENTERPRISE_SEARCH/, CLIENT=052
```

This table explains the variables used in the profile entry above:

Variable	Description	Example Value
`<NUM>`	Number of the profile entry	**8**
`<BE SID>`	SID of the search backend	**ES1**
`<BE HOST>`	Host of the search backend	**searchhost.acme.corp**
`<BE PORT>`	HTTP or HTTPS port of the search backend	**47110**
`<RD PORTS>`	Ports to be redirected	***:54380;*:40280;*:54390**
`<BE CLIENT>`	Client of the search backend	**001**

In addition, create the following external alias in the search backend system using transaction `SICF`:

/enterprise_search/sap/public/bc/icf/logoff pointing to **default_host/sap/public/bc/icf/logoff**

1.3.1.4.1.2 Configuring SAP Web Dispatcher to Dispatch Requests to Different Systems

In SAP Fiori launchpad, you can call OData services on different systems.

For example, this is necessary for catalog federation or if specific OData services need to be called on a system different from the SAP Fiori launchpad front-end server. To achieve this, URLs for OData service calls are

enriched by a segment parameter o having the system on which the service is to be called as the value. This mechanism is as specified for the origin segment parameter by SAP Gateway.

> ℹ Note
>
> Depending on your system landscape, you can set up the segment parameter resolution on the SAP Gateway server, on SAP Web Dispatcher, or on both.

The example URL `https://host:port/sap/opu/odata/MY_SERVICE;o=SYS` can thus be routed to the corresponding system.

Use modification rules on SAP Web Dispatcher to route URLs according to the correct system:

1. Define modification rules as a file (for example, `modification_rule.txt`) with the following content to properly route requests based on the `;o=` origin segment parameter containing the system.

   > ℹ Note
   >
   > The regular expression `*;o=<system>*` below is an example; you can provide a more specific expression as needed.
   >
   > ```
   > if %{PATH} regimatch *;o=SYS* SetHeader x-sap-webdisp-target-sid <FOO>
   > [break]
   > ```
   >
   > ```
   > if %{PATH} regimatch *;o=OTHER* SetHeader x-sap-webdisp-target-sid <BAR>
   > [break]
   > ```

2. Add the modification file to the SAP Web Dispatcher profile file (*.pfl):
   ```
   icm/HTTP/mod_<number> = PREFIX=/,FILE=modification_rule.txt
   ```

   > ℹ Note
   >
   > To ensure that the dispatching works properly, the profile file must contain systems both for `FOO` and `BAR`.

1.3.1.4.1.3 Defining the Host Name of SAP Web Dispatcher

Use this configuration setting to define a URL that points to the SAP Web Dispatcher instead of the current server.

You need to be able to define a URL on the front-end server that does not point to the current system but to the SAP Web Dispatcher. For this, the system administrator defines the host name and port of the SAP Web Dispatcher to be used in the start URL in table `HTTPURLLOC` using *Data Browser* (SE16).

> ℹ Note
>
> This configuration setting is only relevant if you want to launch SAP Fiori launchpad designer from Customizing or *Role Maintenance* (transaction `PFCG`), for example, to view the content of the catalog you assigned to an authorization role in the `PFCG`.

Example

You define one entry that is applicable for both SAP Fiori launchpad and SAP Fiori launchpad designer in table HTTPURLLOC. You define one entry for each protocol.

Sort Key	Protocol	Application	Host	Port
0001	HTTPS	/sap/bc/ui5_ui5*	<webdisp_host>	<webdisp_port_https>
0002	HTTP	/sap/bc/ui5_ui5*	<webdisp_host>	<webdisp_port_http>

1.3.1.4.2 Establish Connections between Front-End Server and Backend

To allow the launchpad to consume catalogs and application descriptors from the backend, you need to establish connections between the front-end server and the backend.

Prerequisites

- You are assigned to authorization role SAP_FLP_ADMIN.
  See

Procedure

Perform the following steps on the front-end server:

1. Decide on a unique logical system alias identifying the back-end system in the target client (e.g. FINASIA or FINCLNT100).
2. Create RFC destinations pointing to the relevant back-end systems:
   a. In Customizing, choose ▶ *SAP NetWeaver* ▶ *Gateway* ▶ *OData Channel* ▶ *Connection Settings* ▶ *SAP NetWeaver Gateway to SAP System* ▶ and launch the IMG activity *Manage RFC Destinations* or directly launch transaction SM59.
   b. For each logical system alias, create the following RFC destinations:
      - Connection type 3: ABAP connection
        This connection type is used to replicate app descriptors from the back-end server to the front-end server. Application descriptors provide information required to access applications from the SAP Fiori launchpad.
        Use the following naming convention: <Logical System Alias>_RFC
      - Connection type H: HTTP(S) connection
        This connection type is used to start back-end applications from the launchpad and retrieve the business data for these applications.

Use the following naming convention: `<Logical System Alias>_HTTP` or `<Logical System Alias>_HTTPS`

> **i Note**
>
> We recommend to use an HTTPS connection.

You want to use the logical system alias `S4FIN` for your integration.

You need to create an RFC destination of connection type 3 (ABAP connection) named `S4FIN_RFC` and RFC destination of connection type H (HTTP(S) connection) named `S4FIN_HTTP`.

3. Select a system alias for the back-end systems to which you want to connect. See Define System Aliases [page 17].

   To ensure consistency, you should also create a Gateway system alias with the same name as the system alias for the SAP Fiori launchpad, see Creating System Alias for Applications.

   Instead of the manual configuration of the system alias, you can also use the task list `SAP_GATEWAY_ADD_SYSTEM_ALIAS`.

4. Map SAP system aliases to logical system aliases, which stand for the physical back-end systems connected to the front-end system. See Map System Aliases to RFC Connections [page 17].

5. Register the back-end server location as safe host in the HTTP_WHITELIST table.

   For more information, go to https://help.sap.com/viewer/p/SAP_NETWEAVER, select your release and search for "Using a Whitelist for Clickjacking Framing Protection".

Related Information

Setting Up Technical Catalogs with the SAP Fiori Launchpad App Manager [page 132]

1.3.1.4.2.1 Manage RFC Destinations

Define remote function call (RFC) destinations from the front-end server for SAP Fiori to the back-end system(s). Additionally define an RFC destination that has the front-end server itself as target for local RFC calls.

1.3.1.4.2.2 Define System Aliases

Display SAP-delivered system aliases or define your own system aliases.

Context

The system aliases are, for example, used for the replication of technical catalogs.

Procedure

1. Start *Maintain Table Views* (transaction `SM30`).
2. In the *Table/View* field, enter the view `/UI2/V_SYSALIAS`.
3. Choose *Display* to display the available system aliases.
4. Choose *Edit* to add your own system alias.

Next Steps

In the next step, you map the system aliases to RFC connections.

See Map System Aliases to RFC Connections [page 17].

1.3.1.4.2.3 Map System Aliases to RFC Connections

Map SAP system aliases to logical system aliases, which stand for the physical back-end systems connected to the front-end system.

Context

In your system landscape, several system aliases may point to one and the same physical source system. Rather than creating RFC and HTTP/S connections to each of these system aliases, you map system aliases to logical system aliases, which stand for the physical back-end systems connected to the front-end system. The logical system is resolved to a concrete RFC or HTTP connection by adding the suffix _RFC or _HTTP/S.

Perform the following steps on the front-end server:

Procedure

1. Use *Configuration of RFC Connections* (transaction SM59) to identify a target system alias that you can use for your mapping.

   Ideally, RFC and HTTP connections are already available for the selected system alias.

2. If necessary, create missing RFC and HTTP connections in transaction SM59.

3. Enter the mapping data:

   a. Start *Maintain Table Views* (transaction SM30).

   b. In the *Table/View* field, enter the view /UI2/V_ALIASMAP and choose *Maintain*.

   c. Choose ▶ *Edit* ❭ *New Entries* ❭.

   d. Provide the following:

Field	Description
Client	Leave blank to define that the system alias is mapped to a different value for all clients.
Source	Enter the source system alias. See step 1 under Establish Connections between Front-End Server and Backend [page 15].
Target System Alias	Enter the target system alias identified in step 1. **i Note** Leave this field blank if the launchpad content that you want to replicate resides on the same system as the front-end server and SAP Gateway.

   e. Save your changes.

Results

Before extracting applications from a backend system, the RFC connection is retrieved from the target system alias that you specified in step 3. At runtime of the launchpad, the source system alias is replaced by the target system alias and then mapped to an existing HTTP connection.

> **i Note**
>
> Changes to the mapping table will cause cache invalidations and recalculation of cache buster tokens. See Cache Buster for Target Mappings and UI2 Services [page 366].

Example

Catalogs for the system aliases S4FIN and S4LO should both be replicated using an existing SM59 connection called ERP_100. The entries in the mapping table look like this:

Source System Alias	Target System Alias
S4FIN	ERP_100
S4LO	ERP_100

1.3.1.4.3 Configuring a Login Screen for the Launchpad

You can configure a login screen for the SAP Fiori launchpad if you want a customized version.

Prerequisites

We recommend that you do this configuration in an external alias that refers to the ICF node `/sap/bc/ui2/flp`. For information on how to create such an external alias, see Customizing the Launchpad URL [page 22].

Context

The SAP Fiori login page is a standard **ABAP** login page. If you want to to adapt it to your specific needs, you can configure your own login page.

> **i Note**
>
> If you create your own login page or did this in a previous version, you need to adapt your ICF settings, when a new ABAP class is available.

The ABAP class /UI2/CL_FIORI3_LOGIN supports Content Security Policy (CSP). CSP is a mechanism web applications can use to mitigate a broad class of content injection vulnerabilities, such as cross-site scripting (XSS). See for more information. In addition, clickjacking framing protection is supported. You can find more information in Clickjacking Framing Protection [page 630]. To use this you need to define an allowlist first. For more information see Using a Whitelist for Clickjacking Framing Protection.

Procedure

1. In transaction `SICF`, enter the ICF node `/sap/bc/ui2/flp` in the field *Reference Service* and confirm.

2. Select your host or service, choose , and navigate to ❙▶ *Error Page* ❭ *Logon Errors* ❭ tab.

3. Select the *System Logon* radio button and choose the *Configuration* button.

   A *System Logon Configuration* dialog box appears. Enter the following:

   a. Under *Settings Selection* section, select the *Define Service-Specific Settings* radio button.

   b. In the *Logon Layout and Procedure* section, select the *Custom Implementation* radio button and enter `/UI2/CL_FIORI3_LOGIN` in the *ABAP Class* field.

   c. To configure the login screen you have the following options:

Dialog Section	Settings
Default	You can define the default values for the client or the language that then are used for the launchpad startup.
Select Display	You can set fields that should be displayed. The user can then e.g. select the client that should be opened or the language that should be used. Click on the button *Adjust Links and Images* to define URLS that lead to a Register or Password Forgotten page. If the links are defined, they will be displayed in the login screen enhanced with a sap-return-url query parameter.

4. You can also choose one of the sign-in methods offered by SAP NetWeaver. See Maintaining Logon Procedures and Creating and Configuring ICF Services for detailed information.

Results

The SAP Fiori logon screen is configured for the SAP Fiori launchpad. Customizing the Login Screen [page 21] shows you additional possibilities to adapt the login page.

1.3.1.4.3.1 Customizing the Login Screen

You can adapt the login screen to your needs, e.g replace the images or change the copyright.

We recommend that you subclass the SAP Fiori login class before you start customizing. You can simply redefine the `init_default_properties` file to provide alternative values to the various properties (e.g. background image, copyright ...). You can change the following properties:

Property Name	Description
img_background	Background image
img_favicon	Favorite icon
img_logo	Logo
img_logo_width	Logo width
label_copyright	Copyright notice
extra_css	Optional URL when you want to load an additional CSS stylesheet
extra_js	Optional URL when you want to load an additional JavaScript file

Example

The following script example from the `init_default_properties` file shows how to change the background image. In the line `m_properties->set( name = 'img_background' value = '/path/to/my/background.png' ).` you enter the path to your new image. In a similar way you can customize the other parameters in the file.

 Sample Code

```
class z_custom_login definition
  public
  inheriting from /ui2/cl_fiori3_login
  final
  create public .
  public section.
  protected section.
    methods init_default_properties
        redefinition .
  private section.
endclass.
class z_custom_login implementation.
  method init_default_properties.
    super->init_default_properties( context ).
    m_properties->set( name = 'img_background' value = '/path/to/my/
background.png' ).
  endmethod.
endclass.
```

1.3.1.4.4 Customizing the Launchpad URL

You can make the launchpad available under a custom URL by creating an external alias.

Procedure

1. On the front-end server, start transaction `SICF`.

2. Select hierarchy type SERVICE, then choose .

3. Choose *External Aliases*.

4. Select a host, then choose ▷ *External Alias* ❯ *Create* ❯.

5. In the *External Alias* field, enter the alias under which you want the launchpad to be available.

   Enter an alias of your choice, for example:

   `/fiori`

6. On the *Trg Element* tab, select the following target handler:

   `/sap/bc/ui2/flp`

Results

After doing the configuration as shown in the example above, you can launch the launchpad using a nice and short URL like this:

`<protocol>://<host>:<port>/fiori/`

> i Note
>
> Note the slash (`/`) at the end of this URL.

Next Steps

If you are using the SAP Web Dispatcher as reverse proxy in your system landscape, see SAP Web Dispatcher for Multiple Systems for information how to customize the login URL in a reverse proxy scenario.

Related Information

Launching the Launchpad [page 101]

1.3.1.5 Manual Setup Without Task Lists

You can manually perform the steps required to set up the launchpad.

However, we recommend you to use the task lists available to perform these steps automatically.

1.3.1.5.1 Activating SAP Gateway OData Services

The launchpad and launchpad designer need a reverse proxy to direct the browser requests to either the front-end system in which the JavaScript sources are stored, or to the SAP Gateway system in which the OData services are located.

Prerequisite

When activating the OData services, the system alias must be the local system.

To configure that a local connection to SAP NetWeaver Gateway is used, proceed as follows:

1. In Customizing (transaction SPRO), choose ▶ *SAP NetWeaver* ❯ *Gateway* ❯ *OData Channel* ❯ *Configuration* ❯ *Connection Settings* ❯ *Connection SAP NetWeaver* ❯ *SAP NetWeaver Gateway to SAP System* ◀ and launch the IMG activity *Manage SAP System Aliases*.
2. Provide the following:

Field	Value
SAP System Alias	Name of the system alias (for example, **LOCAL**).
Description	Descriptive text for the system alias.
RFC Destination	**NONE**
Local GW	Make sure the checkbox is marked to specify that the system alias points to a local SAP Gateway instance.

Procedure

1. Using the system alias defined in the Customizing activity *Manage SAP System Alias*, in Customizing choose ▶ *SAP NetWeaver* ❯ *Gateway* ❯ *OData Channel* ❯ *Administration* ❯ *General Settings* ❯ *Activate and Maintain Services* ◀ and activate the following OData services:
   - */UI2/PAGE_BUILDER_CONF* -This is used by the launchpad designer.
   - */UI2/PAGE_BUILDER_CUST* - This is used by the launchpad designer.

- *   */UI2/PAGE_BUILDER_PERS* - This is used by the launchpad.
- *   */UI2/TRANSPORT*- This is used by the launchpad designer.
- *   */UI2/INTEROP* - This is used by the launchpad and launchpad designer.
- *   *ESH_SEARCH_SRV* - This is used by the launchpad for the search function.
- *   *FDM_SPACE_REPOSITORY_CUST_SRV* - This is used by the launchpad for read and write access to the Manage Launchpage Spaces app.
- *   *FDM_PAGE_REPOSITORY_CUST_SRV* - This is used by the launchpad for read and write access to the Manage Launchpage Pages app.
- *   *FDM_PAGE_RUNTIME_SRV* - This is used by the launchpad to enable users to view spaces and pages.
- *   *FDM_TRANSPORT_SRV* - This is used by the launchpad for transport handling of spaces and pages.

   For information about activating these services, see the SAP Gateway Developer Guide at ▌▶ *http:// help.sap.com/netweaver* ❯ *SAP NetWeaver Gateway* ❯ *Development Information* ❯ *SAP NetWeaver Gateway* ❯ *Developer Guide* ❯ *OData Channel* ❯ *Basic Features* ❯ *Service LifeCycle* ❯ *Activate and Maintain Services* ▌.

2. Call each service once.

1.3.1.5.2   Configuring ICF Nodes

This section describes the configuration of ICF nodes for local and remote gateway deployment option.

This section covers the following topics:

- *   SICF Services [page 24]
- *   External SAP Gateway System has a Different Client to the Front-end ABAP System [page 25]

When you activate the HTTP compression in Internet Communication Framework (ICF) not all document types are compressed. For more information refer to SAP Note 1037677 .

1.3.1.5.2.1 SICF Services

In addition to the OData services mentioned above, the launchpad and launchpad designer also need the SICF services for the application resources from the front-end system.

To activate the SICF services, open transaction *SICF*.Choose *Execute* or press **F8**. The Maintain Service window is displayed. The relevant paths to be activated are as follows:

- *   */default_host/sap/bc/ui2/start_up*
- *   */default_host/sap/bc/ui5_ui5/ui2/ushell*
- *   */default_host/sap/bc/ui2/flp*
- *   */default_host/sap/bc/ui5_ui5/sap/arsrvc_upb_admn*
- *   */default_host/sap/bc/ui5_ui5/sap/ar_srvc_news*
- *   */default_host/sap/bc/ui5_ui5/sap/ar_srvc_launch*
- *   */default_host/sap/public/bc/ui5_ui5/*

- */default_host/sap/public/bc/ui2/*
- */default_host/sap/bc/ui2/nwbc/* for NWBC for HTML
- */default_host/sap/bc/lrep* (For more information, see .)
- */default_host/sap/bc/ui2/app_index* (For more information, see .)

Activate the services mentioned above (either by right-clicking the mouse and selecting *Activate Service* or choosing ▌ *Service/host* ❯ *Activate* ▌ from the menu).

> **i Note**
>
> In the *Activation of ICF Services* dialog box, choose the *Yes* button with *Hierarchy* icon to activate all the child nodes under this service.

For more information about specific SICF services for search UIs including the launchpad, see SAP Library for SAP NetWeaver on SAP Help Portal at http://help.sap.com/netweaver. ▌ *Application Help* ❯ *Function-Oriented View* ❯ *Search and Operational Analytics* ❯ *Embedded Search* ❯ *Activating UI Services* ▌.

After configuring the SICF services, we recommend that you create an external alias for the service `/sap/bc/ui2/flp`. For information on how to do this, see Customizing the Launchpad URL [page 22].

1.3.1.5.2.2 External SAP Gateway System has a Different Client to the Front-end ABAP System

Context

To avoid an incorrect client user credentials request when running the launchpad and launchpad designer, configure the ICF nodes as described below:

Procedure

1. In your ABAP front-end system, access transaction *SICF* and choose *Execute*. The *Maintain Services* screen appears.
2. Choose ▌ *Goto* ❯ *External Alias* ▌.
3. Select *Default Host* and choose ▌ *External Alias* ❯ *Create* ▌.
4. Choose *Trg Element* and navigate to ▌ *SAP* ❯ *BC* ❯ *UI5_UI5* ❯ *SAP* ▌.
5. Double-click the SAP node.
6. In the *External Alias* field, enter */sap/bc/ui5_ui5/sap* and enter a meaningful description.
7. Choose the *Logon Data* tab. Leave the *User* field blank. In the *Procedure* field, select *Required with Logon Data*.

8. Enter the client information as required and save your entries.

9. Log on to your SAP Gateway system, call transaction *SICF* and navigate to the respective service that the launchpad and launchpad designer uses.

10. Double-click the service *PAGE_BUILDER_CONF*.

11. Choose the *Logon Data* tab. In the *Procedure* field, select *Required with Logon Data*.

12. Repeat the steps (10- 11) for the following services - */UI2/PAGE_BUILDER_CUST*, */UI2/ PAGE_BUILDER_PERS*, */UI2/TRANSPORT*,*/UI2/INTEROP*.

1.3.1.5.3   Updating the SAPUI5 Application Index

This index provides an indexing and caching mechanism for information related to apps, components, and libraries in the SAPUI5 ABAP repository and related to components and libraries in the SAPUI5 distribution layer.

You have to make sure that the index is updated using its calculation report `/UI5/APP_INDEX_CALCULATE` whenever the content of the SAPUI5 ABAP repository has changed. For more information whether the report is executed automatically or you have to run it manually, see .

1.3.1.5.4   Scheduling Update of OData Metadata Caching

For fast loading times of SAP Fiori apps, cache buster tokens are used to cache OData metadata (metadata and annotations documents). To ensure that the tokens are up to date, the report `/UI5/ UPD_ODATA_METADATA_CACHE` has to be executed at least every 48 hours.

You can execute the report either periodically or trigger it manually after content changes. Examples for changes that require executing the report:

- Importing a transport
- Implementating an SAP Note
- Changing a service

Prerequisites

- This report is executed by a technical user in the SAP Gateway hub and in the back end. This user needs the authorization *Display (3)* in the authorization objects `/IWFND/ADM` and `/IWBEP/ADM`.
- The ICF node `/sap/opu/utils/iwfnd/bct/GetBackendContextTokens` has to be activated in the remote gateway.

Periodic Scheduling

To perform this configuration task in an automated way, you can schedule it in Customizing under ▶ *SAP NetWeaver* ▶ *UI Technologies* ▶ *SAP Fiori* ▶ *Initial Setup* ▶ *Manual Setup without Task Lists* ▶ *Schedule Update of Caching of OData Metadata* ▶. The default value for the interval of the job execution is *Hourly*.

You can also you use the predefined task list `/UI5/SCHEDULE_JOB_UPDATE_CACHE` and include the report there.

> **i Note**
>
> If you run `/UI5/UPD_ODATA_METADATA_CACHE` periodically or include it in a task list, make sure that the report runs after any import tasks. Please take this into account if you change the interval in the Customizing activity *Schedule Update of Caching of OData Metadata*.
>
> In a productive landscape, you may want to schedule the report to run once per night as the last job after any import jobs. In test and quality systems with more frequent imports, it can make sense to schedule the report to run hourly. This depends on your individual environment.

Related Information

Client-Side Cache [page 364]
Cache Buster for OData Metadata of SAP Fiori Apps [page 370]

1.3.1.5.5 Scheduling Caching of SAP Menu Entries

To ensure fast loading of SAP menu apps in the app finder, run the report `/UI2/EAM_BUILD_CACHE` periodically in the respective back-end systems.

This task is relevant if you enabled the app finder and its ability to display and search within the SAP menu of a particular back-end system.

The report `/UI2/EAM_BUILD_CACHE` reads the SAP menu entries and merges this data with default transaction texts and data of SAP Fiori back-end catalogs. The merged data is persisted and acts as a cache when users display or open SAP menu entries in the app finder.

Run the report `/UI2/EAM_BUILD_CACHE` in the following cases:

- Initially to fill the cache

To always ensure good performance, we recommend that you schedule the report to run periodically.

If there is no cached result, you will get an error message in the app finder. An error message is also displayed in the network trace.

Related Information

SAP Note 2545066
Integrating Applications from SAP Easy Access Menu [page 312]
Issues with SAP Easy Access Menu [page 649]

1.3.1.5.6 Scheduling the Synchronization of User-Independent Cache

The launchpad uses a user-independent cache to accelerate data access for catalogs and groups.

The report `/UI2/PAGE_CACHE_SYNCHRONIZATION` (or transaction `/UI2/SYNC_PBC`) reads all the relevant information from catalogs and groups and updates the user-independent cache. This way, it ensures that all changes of catalogs and groups are displayed correctly in the launchpad runtime, as well as in the design-time tools (e.g. in the launchpad content manager or in the exposure report `UI2/CDM3_EXP_SCOPE`).

We recommend to run the job `/UI2/PAGE_CACHE_SYNC` once a day depending on how often changes are made in the system.

The cache synchronization happens automatically if you make changes in the design-time tools, e.g. creating or modifying technical catalogs in the launchpad app manager, or creating or modifying business catalogs in the launchpad content manager.

In the following cases, you need to run the transaction or the report manually:

- Launchpad texts, such as titles of tiles, target mappings, groups, or catalogs, are translated
- The translations of tiles, target mappings, groups, or catalogs are imported into the system (transport management system)
- Catalog and group changes are transported as `R3TR TABU` object as a direct table change (instead of a `R3TR WDCC`, `R3TR UIAD`, or `R3TR UIAC` object)

1.3.1.5.7 Enabling Context-Sensitive Help

You can enable a context-sensitive help for your users in the launchpad.

How the in-app help is enabled depends on your system:

- For SAP S/4HANA follow the instructions in Enable Context-Sensitive User Assistance for SAP Fiori Launchpad.
- For SAP Marketing follow the instructions in Displaying the SAP Online Documentation.

To ensure that all help content is shown, you also need to set the launchpad parameter ENABLE_HELP to TRUE. See Setting Parameters in SAP Fiori Customizing [page 63] and Launchpad Configuration Parameters [page 29] for additional information.

1.3.2 Managing Launchpad Settings

Several options are available to adapt the SAP Fiori launchpad to your needs, allowing you to enable additional features and change the visual appearance of the launchpad.

Related Information

1.3.2.1   Launchpad Configuration Parameters

You can use configuration parameters to configure different aspects of the launchpad.

The parameters listed in the table below are categorized as follows:

- FLP UI Server Settings
  These parameters can be set only in Customizing.
- FLP UI Client Settings
  These parameters can be set in Customizing or in a target mapping.

If you do not explicitly define a parameter, its default value is taken and triggered in the launchpad. The default values described in the table below are the coded defaults for the ABAP platform. Products, for example SAP S/4HANA, might define different default settings. The default values may be overridden by settings in Customizing. For more information, see Setting Parameters in SAP Fiori Customizing [page 63] and Alternative Ways of Setting Parameters [page 64].

Launchpad Configuration Parameters

Short Key for ABAP Customizing	Path and Parameter for Target Mapping	Category	Description
UI5FLEX_ENABLE _VARIANTS	n/a	FLP UI Server Settings	**Specify if users are allowed to create and adapt views and whether they're allowed to share them (see** Save Your Personal Settings as a View [page 488]**).**
			This is a global setting. There are three options:
			• PUBLIC (default) Users are allowed to create and adapt new views. By selecting the *Public* checkbox, they can also share them with other users.
			• PRIVATE Users are allowed to create and adapt new views, but the *Public* checkbox is not displayed, preventing users from sharing views.
			• NONE Users are not allowed to create or adapt views. They can still set a view as favorite and change their default view. Optionally, they can also choose whether a view should be applied automatically, meaning a search should immediately be triggered when the view is selected.
			This setting supersedes SAP Note 2658662 . If you have implemented this SAP Note, you need to revert it for this launchpad configuration parameter to have an effect.
SUPPORTTICKET	services/ SupportTicket/config/ enabled	FLP UI Client Settings	**Specify whether end users can create support tickets directly from the launchpad.**
			When set to true, the *Contact Support* option is displayed in the user actions menu..
			Default value: false
			See: Configuring the Option to Contact Support [page 95]

Short Key for ABAP Customizing	Path and Parameter for Target Mapping	Category	Description
NAVIGATION_ALL MYAPPS	services/AllMyApps/ config/enabled	FLP UI Client Settings	**Specify whether the All My Apps navigation option is available to end users in the launchpad, and which content it displays.**
NAVIGATION_ALL MYAPPS_SHOWHOM EPAGEAPPS	services/AllMyApps/ configshowHomePageApp s		The All My Apps navigation option allows users to quickly access their home page and catalog apps from anywhere in the launchpad.
NAVIGATION_ALL MYAPPS_SHOWCAT ALOGAPPS	services/ AllMyApps/config/ showCatalogApps		When enabled is set to true, the *All My Apps* button is available in an app's hierarchical navigation menu, and the tool is automatically launched when a user clicks *Home* in the middle of the shell header. When the tool is enabled, you can optionally set showHomePageApps or showCatalogApps to false to hide either home page apps or catalog-assigned apps in All My Apps.

When enabled is set to false, the All My Apps tool is not available to end users.

All My Apps does not support the SAP Easy Access menus (EAM) and non-static, non-dynamic custom tiles.

Default values:

- enabled: true
- showHomePageApps: true
- showCatalogApps: true

See: Navigating Within and Between Apps [page 436]

Short Key for ABAP Customizing	Path and Parameter for Target Mapping	Category	Description
NAVIGATION_GUI _INPLACE	services/ ClientSideTargetResol ution/config/ enableInPlaceForClass icUIs/GUI	FLP UI Client Settings	**Specify whether SAP GUI applications are opened in-place** Set to `true` if you want SAP GUI applications to open in the same browser window and tab. Default value: `false` **i Note** When users navigate to Web Dynpro ABAP or SAP GUI applications through links in notifications, in the Quick Access dialog, or via intent-based navigation triggered by WDA, the target application is always opened in a new browser window or tab.
NAVIGATION_WDA _INPLACE	services/ ClientSideTargetResol ution/config/ enableInPlaceForClass icUIs/WDA	FLP UI Client Settings	**Specify whether Web Dynpro applications are opened in-place** Set to `true` if you want Web Dynpro applications to open in the same browser window and tab. Default value: `false`
NAVIGATION_WDA _COMPATIBILITY MODE	ushell/navigation/ wdaCompatibilityMode		**Specify whether Web Dynpro applications are opened with NWBC for HTML running invisibly in the background.** Default value: `false` `false` means, Web Dynpro applications are started directly as URL. `true` means, Web Dynpro applications are started with NWBC for HTML running invisibly in the background. When your custom WebDynpro-based app doesn't run as expected, please switch back to compatibility mode (`true`). Note that this setting is not considered as default value for apps created in the FLP App Manager or target mappings created in LPD_CUST.

Short Key for ABAP Customizing	Path and Parameter for Target Mapping	Category	Description
NAVIGATION_HOM ETARGET	renderers/fiori2/ componentData/config/ homeNavigationTarget	FLP UI Client Settings	**Decide what happens when a user clicks on the SAP logo (customer logo) in the launchpad shell bar.** Select `first_page`, if users should navigate to the first page in their launchpad. Select `origin_page`, if users should navigate to the last opened page. Also see Shell and Shell Bar [page 414] Default setting: `first_page` Note, that this setting only applies when working with spaces and pages.
SMARTNAVIGATIO N	services/ SmartNavigation/ config/ isTrackingEnabled	FLP UI Client Settings	**Specify whether intent-based navigation links are ranked based on usage data.** Default setting: `false` We recommend that you do not change the default setting. If you set this parameter to `true` user-specific navigation data is tracked in order to provide better search results. We provide a solution to delete this data manually. For more information, see SAP note 2624570 .
APPSTATE_TRANS IENT	services/AppState/ config/transient	FLP UI Client Settings	**Specify whether application state data for navigation is kept in the browser memory only (transient) or stored in the database of the front-end server.** Default setting: `true` Note that when you set the parameter to `false`, you need to ensure that no sensitive or personal data is stored. We recommend that you schedule a report to clean up expired application states on a daily basis. For more information, see Cleanup of Expired Application State [page 360].

Short Key for ABAP Customizing	Path and Parameter for Target Mapping	Category	Description
NOTIFICATIONS	services/ Notifications/config/ enabled	FLP UI Client Settings	**Set this parameter to** true **to activate the Notification UShell service and enable the consumption of push notifications in the launchpad.** Default value: false See Enabling Notifications in the Launchpad [page 78] for more information.
NOTIFICATIONS_ SERVICE_URL	services/ Notifications/config/ serviceUrl	FLP UI Client Settings	**Set this parameter to refer to the Notification OData service root URL (or base URL).** This path is a combination of the hostname, the published Notification Channel service group, repository ID, service name, and the service version. See Enabling Notifications in the Launchpad [page 78] for more information.
NOTIFICATIONS_ POLLING_INTERV AL	services/ Notifications/config/ pollingIntervalInSeco nds	FLP UI Client Settings	The Notification UShell service uses either a WebSocket connection or a polling mechanism to retrieve updates from the SAP Gateway Notification Channel. The WebSocket protocol is the preferred communication method. If the notification service cannot establish a WebSocket connection from the browser to the notification channel, then it will implement the fallback polling policy at an interval (in seconds) that is determined by this parameter. Default value: 60 (seconds) See Enabling Notifications in the Launchpad [page 78] for more information.
NOTIFICATIONS_ UI	renderers/fiori2/ componentData/config/ enableNotificationsUI	FLP UI Client Settings	**Set this parameter to** true **to enable the out-of-the-box UI modules that allow users to display and interact with notifications in the launchpad.** Default value: false See Enabling Notifications in the Launchpad [page 78] for more information.

Short Key for ABAP Customizing	Path and Parameter for Target Mapping	Category	Description
HOMEPAGE_PERSO NALIZATION_HID EGROUPS	renderers/fiori2/ componentData/config/ applications/Shell- home/enableHideGroups	FLP UI Client Set- tings	**Specify whether users can hide groups on the home page or sections on a page in edit mode.** When set to `true`, the *Hide* option is displayed in the group header or section header. Note that per- sonalization must be enabled (see `HOMEPAGE_PERSONALIZATION` below) for this option to take effect. Default value: `false` See: Personalizing the Home Page [page 501] or Personalizing a Page [page 496]
HOMEPAGE_TILES _OPACITY	renderers/fiori2/ componentData/ config/applications/ Shell-home/ enableTilesOpacity	FLP UI Client Set- tings	**Specifies whether tiles are displayed with varying degrees of opacity, according to how often they are used.** Default value: `false` > **ⓘ Note** > The parameter is deprecated. The opacity feature has been removed, so activating the parameter has no effect.
HOMEPAGE_TILES _TEXT_WRAPPING	urenderers/fiori2/ componentData/ config/applications/ Shell-home/ tilesWrappingType	FLP UI Client Set- tings	Specify if the tile titles are hyphenated at the end of the line or if hyphenation is disa- bled. There are two values: • `Hyphenated`: Hyphenation will be used to break words on syllables where possible. • `Normal`: Normal text wrapping will be used. Words won't break based on hy- phenation. Default value is: `Hyphenated` Also see App Finder [page 427].

Short Key for ABAP Customizing	Path and Parameter for Target Mapping	Category	Description
APPFINDER_ENAB LED	renderers/fiori2/ componentData/config/ enableAppFinder	FLP UI Client Settings	**Specify if your users have access to the app finder.** Users can open all the apps in the app finder, which are assigned to them according to their role assignment. If set to false, the app finder will only be enabled if personalization is enabled. Default value: true

Short Key for ABAP Customizing	Path and Parameter for Target Mapping	Category	Description

`APPFINDER_EASY ACCESSMENU`
 	
`APPFINDER_EASY ACCESSMENU_ON_ TABLET`
 	
`APPFINDER_EASY ACCESSMENU_SAP MENU`
 	
`APPFINDER_EASY ACCESSMENU_USE RMENU`
	`renderers/fiori2/ componentData/config/ applications/Shell- home/enableEasyAccess`
	
	`renderers/fiori2/ componentData/config/ applications/Shell- home/ enableEasyAccessOnTab let`
	
	`renderers/fiori2/ componentData/config/ applications/Shell- home/ enableEasyAccessSAPMe nu`
	
	`renderers/fiori2/ componentData/config/ applications/Shell- home/ enableEasyAccessUserM enu`
		FLP UI Client Set- tings	**Specify whether the SAP Easy Access menu tabs (***SAP Menu*** and ***User Menu***) are displayed in the app finder.**
			
			When `APPFINDER_EASYACCESSMENU` is set to `false`, both the *SAP Menu* and *User Menu* tabs are hidden.
			
			When `APPFINDER_EASYACCESSMENU` is set to `true`, the *Catalog* tab is displayed together with the *SAP Menu* and *User Menu* tabs, unless you set `APPFINDER_EASYACCESSMENU_SAPM ENU` or `APPFINDER_EASYACCESSMENU_USER MENU` to `false`.
			
			⚙ Example
			• `APPFINDER_EASYACCESSMEN U=true`: Displays the *Catalog*, *SAP Menu*, and *User Menu* tabs.
			• `APPFINDER_EASYACCESSMEN U=true`, `APPFINDER_EASYACCESSMEN U_SAPMENU=true`, `APPFINDER_EASYACCESSMEN U_USERMENU=false`: Displays the *Catalog* and *SAP Menu* tabs.
			• `APPFINDER_EASYACCESSMEN U=trueAPPFINDER_EASYACC ESSMENU_SAPMENU=false`, `APPFINDER_EASYACCESSMEN U_USERMENU=true`: Displays the *Catalog* and *User Menu* tabs.
			
			When `APPFINDER_EASYACCESSMENU_ON_T ABLET` is set to true, the Easy Access menu is also shown on tablets. Note that `APPFINDER_EASYACCESSMENU` has to be set to true, otherwise this setting has no effect.

Short Key for ABAP Customizing	Path and Parameter for Target Mapping	Category	Description
			Default values: • APPFINDER_EASYACCESSMENU: `true` • APPFINDER_EASYACCESSMENU_S APMENU and APPFINDER_EASYACCESSMENU_U SERMENU: Both inherit the value you set for APPFINDER_EASYACCESSMENU • APPFINDER_EASYACCESSMENU_O N_TABLET: `false` Note that the SAP Easy Access menu is not supported on phones. See: App Finder [page 427] and Integrating Applications from SAP Easy Access Menu [page 312]
APPFINDER_SEAR CH_CATALOG APPFINDER_SEAR CH_EASYACCESSS APMENU APPFINDER_SEAR CH_EASYACCESSU SERMENU	renderers/fiori2/ componentData/ config/applications/ Shell-home/ enableCatalogSearch renderers/fiori2/ componentData/config/ applications/Shell-home/ enableEasyAccessSAPMe nuSearch renderers/fiori2/ componentData/config/ applications/Shell-home/ enableEasyAccessUserM enuSearch	FLP UI Client Settings	**Specify whether the search option is available in the app finder** You can configure the search option for each of the app finder tabs independently (*Catalog*, *SAP Menu*, or *User Menu*). When set to `false` the search option is hidden from the respective tab. Default value for all tabs: `true` See: App Finder [page 427], Personalizing the Home Page [page 501], and Integrating Applications from SAP Easy Access Menu [page 312]

Short Key for ABAP Customizing	Path and Parameter for Target Mapping	Category	Description
APPFINDER_CATALOG_TAGFILTER	renderers/fiori2/ componentData/config/ applications/Shell-home/ enableCatalogTagFilter	FLP UI Client Settings	**Specify whether the *Filter by Tag* option is available in the *Catalog* tab in the app finder** When set to `false` the *Filter by Tag* option is hidden from the *Catalog* tab in the app finder. Default value: `true` See: App Finder [page 427], Personalizing the Home Page [page 501], and Integrating Applications from SAP Easy Access Menu [page 312]
HOMEPAGE_GROUP SELECTIONBAR_MODE	renderers/fiori2/ componentData/ config/applications/ Shell-home/ homePageGroupDisplay	FLP UI Client Settings	**Specify the default mode of the group selection bar in the home page** In the launchpad home page, apps are clustered in groups. These groups are listed in the group selection bar at the top of the home page. This bar can function either as an anchor bar or a tab bar. With this parameter, you specify the default mode of the group selection bar. Valid values: • `scroll` • `tabs`: Sets the tab mode. The home page shows one group at a time. When the user chooses a group, the home page displays only the content for that group. For large amounts of apps, showing one group at a time may be better. See: Home Page [page 422] → Tip With the `enableHomePageSettings` parameter, you can specify whether end users can switch modes in their user settings. If users change the mode, it overrides the default setting defined by you.

Short Key for ABAP Customizing	Path and Parameter for Target Mapping	Category	Description
USERSETTINGS_G ROUPSELECTIONB AR	`renderers/fiori2/ componentData/config/ applications/Shell- home/ enableHomePageSetting s`	FLP UI Client Settings	**Specify whether end users can personalize the group selection bar in the home page** In the *Home Page* setting in the *Settings* dialog box (accessed from the *Settings* entry in the user actions menu), end users can personalize whether the group selection bar in the home page functions as an anchor bar or a tab bar. See `homePageGroupDisplay` for more details group selection bar, the home page scrolls to the specific content of the group. When set to `false`: Sets the setting is hidden from the *Settings* dialog box; meaning that users won't be able to change the display mode of their home page. The display mode of their home page will be fixed as defined by the `homePageGroupDisplay` parameter. Default value: `true` See: Managing Your Settings [page 509]
SPACES	`ushell/spaces/enabled`	FLP UI Client Settings	**Home If you want to enable spaces and pages for your users, set this parameter to true. False means that the home page is shown.** Default value: `false` See: Enabling Spaces [page 302] and Managing Launchpad Spaces and Pages [page 269]
SPACES_CUSTOM_ HOME	n/a	FLP UI Server Settings	**Note: This parameter is only for internal use. Please don't enable this parameter, as the feature is currently not available on this platform.** Default value: `false`
SPACES_CUSTOM_ HOME_COMPONENT _ID	`ushell/homeApp/ component/ SPACES_CUSTOM_HOME_CO MPONENT_ID`	FLP UI Client Settings	**Note: This parameter is only for internal use. Please don't enter a value, as the feature is currently not available on this platform.**

Short Key for ABAP Customizing	Path and Parameter for Target Mapping	Category	Description
`SPACES_ENABLE_ USER`	`ushell/spaces/ configurable`	FLP UI Client Settings	**If your users should be able to switch between the spaces and the classic home page, set the parameter SPACES_ENABLE_USER to true. Users then can decide in the user settings, which layout they prefer.** Default value: `false` See: Enabling Spaces [page 302] and Managing Launchpad Spaces and Pages [page 269]
`SPACES_MYHOME`	n/a	FLP UI Client Settings	**Enable the My Home space for all your users.** Default value: `true` Note that this setting is only relevant, when you have enabled spaces for your users with the parameters SPACES or SPACES_ENABLE_USER. See: Enabling Spaces [page 302] and Managing Launchpad Spaces and Pages [page 269].
`SPACES_SORT_CR ITERION`	n/a	FLP UI Client Settings	**Specify how the spaces are sorted for the users in the navigation bar.** `title` indicates that spaces are sorted according to the space title. `id` indicates sorting according to the space ID. Default value: `title` See: Sorting the Spaces [page 300] for a detailed description and additional sorting options.

Short Key for ABAP Customizing	Path and Parameter for Target Mapping	Category	Description
USERSETTINGS_S ET_LANGUAGE	renderers/fiori2/ componentData/config/ enableSetLanguage	FLP UI Client Settings	**With this parameter you can enable your users to change the launchpad language themselves in the user settings.** Default value: `true` The language that is used for the launchpad is determined by four different settings. The following list is sorted by the priority the parameters will be set. 1. ICF service configuration (see Configuring ICF Nodes [page 24]) 2. URL parameter "sap-language" or header field with that name (see Customizing the Launchpad URL [page 22]). 3. Cookie "sap-usercontext" (containing "sap-language"). E.g. set when a language is selected during sign-in. 4. The user's default logon language configured in the user settings (see Managing Your Settings [page 509]). When you enable the parameter, your users can set the language themselves at step 4. Note that settings with higher priority overwrite settings with lower ones. E.g. will a parameter in the URL overwrite the user setting.

Short Key for ABAP Customizing	Path and Parameter for Target Mapping	Category	Description
HOMEPAGE_PERSO NALIZATION	`renderers/fiori2/ componentData/config/ enablePersonalization`	FLP UI Client Settings	**Specify whether users can personalize the content displayed in the home page or pages (when working in the spaces mode).** When set to `false`, the end user cannot perform the following actions: • Move, rename, delete, reset, or create groups or sections • Move or remove tiles When `false`, the app finder is also disabled for users. Default value: `true` Note, that this setting doesn't influence My Home. When My Home is enabled (parameter SPACES_MYHOME), users can edit it anytime. See: Personalizing the Home Page [page 501] and Personalizing a Page [page 496]
USERSETTINGS_S ET_THEME	`renderers/fiori2/ componentData/config/ enableSetTheme`	FLP UI Client Settings	**Specify whether users can select a different theme in the *Settings* dialog box in the *Appearance* section.** When set to `false`, the end user can see the current theme, but cannot select a different one. When set to `true`, the end user can select the theme to use. Default value: `true` See: Managing Your Settings [page 509]. For more information on the launchpad theme settings, see Setting Themes for the Launchpad [page 70].

Short Key for ABAP Customizing	Path and Parameter for Target Mapping	Category	Description
DARK_MODE	ushell/darkMode/ enabled	FLP UI Client Settings	**Specify if users can enable the dark mode in the launchpad.** Set to `true` to enable users to enable dark mode settings in the user settings dialog. This is only supported if the users have selected a Quartz or Horizon theme. When they use a current browser that supports automatic color scheme detection, they can decide that the current dark mode setting in the operating system settings will be reflected in the launchpad. If they use a browser that does not support automatic color scheme detection, they can enable or disable the dark mode to switch easily between the dark and the light Quartz or Horizon theme. You can check the MDN page compatibility matrixInformation published on non-SAP site to derive the list of supported browsers. Support for this feature also depends on the underlying operating system support as the same browser can be used on different operating systems. See: Managing Your Settings [page 509]
SEARCH	renderers/fiori2/ componentData/config/ enableSearch	FLP UI Client Settings	**Specify whether the search option is displayed in the launchpad shell bar.** Set to `false` to hide the search option from the shell header bar. Default value: `true` See: Searching for Business Objects and Apps [page 433]
DEFAULT_SEARCH _SCOPE_APPS	renderers/fiori2/ componentData/config/ esearch/ defaultSearchScopeApp s	FLP UI Client Settings	Specifies the default search scope. • `false` (default): default search scope is All • `true`: default search scope is Apps See: Searching for Business Objects and Apps [page 433]

Short Key for ABAP Customizing	Path and Parameter for Target Mapping	Category	Description
`SEARCH_BUSINES S_OBJECTS`	`renderers/fiori2/ componentData/ config/esearch/ searchBusinessObjects`	FLP UI Client Settings	Specifies if business objects are included in the search scope. • `true` (default): search for business objects is activated • `false`: search for business objects is deactivated. Users can only search for apps. See: Searching for Business Objects and Apps [page 433]
`SEARCH_SCOPE_W ITHOUT_ALL`	`renderers/fiori2/ componentData/ config/esearch/ searchScopeWithoutAll`	FLP UI Client Settings	Specifies whether the *All* connector is displayed in the 🔍 search bar in the shell header. • `false` (default): The *All* connector is used. • `true`: The *All* connector is not used. The default search scope is Apps. See: Searching for Business Objects and Apps [page 433]

`HEADER_APPFIND ER_ACTION`
 	
`HEADER_CONTACT SUPPORT_ACTION`
 	
`HEADER_EDITHOM EPAGE_ACTION`
 	
`HEADER_USERSET TINGS_ACTION`
	`renderers/fiori2/ componentData/config/ moveAppFinderActionTo ShellHeader`
	
	`renderers/fiori2/ componentData/config/ moveContactSupportAct ionToShellHeader`
	
	`renderers/fiori2/ componentData/config/ moveEditHomePageActio nToShellHeader`
	
	`renderers/fiori2/ componentData/config/ moveUserSettingsActio nToShellHeader`
		FLP UI Client Settings	Move one of the four standard actions from the user actions menu to the shell bar by setting the respective configuration parameter to `true`.
			
			Default value for all four parameters: `false`
			
			See: Customizing the Shell Bar [page 73]

Short Key for ABAP Customizing	Path and Parameter for Target Mapping	Category	Description
HOMEPAGE_BACKG ROUNDSHAPES	renderers/fiori2/ componentData/config/ enableBackGroundShape s	FLP UI Client Set- tings	**This parameter specifies whether back- ground shapes are displayed in themes. For a better performance, the parameter should be set to false to hide background shapes.**
			If you use the SAP Belize themes, or cus- tom themes based on the Belize theme, set the parameter to true. SAP recommends to use SAP Fiori-3-based themes also as a basis for custom themes.
			Default value: `false`
NAVIGATION_REC ENTACTIVITY	renderers/fiori2/ componentData/config/ enableRecentActivity	FLP UI Client Set- tings	**Specify whether end users can see and access a list of apps and objects in the Quick Access dialog that they have re- cently worked with and used the most.**
			Set to `false` to stop collecting data about the user's most recent and most used items, and hide the *Recent Activity* and *Frequently Used* lists from the user actions menu.
			Default value: `true`
			See: User Actions Menu [page 424]
USERSETTINGS_D EFAULTPARAMETE RS	renderers/fiori2/ componentData/config/ enableUserDefaultPara meters	FLP UI Client Set- tings	**Specify whether end users can view and edit default parameter values**
			Set to `true` to allow end users to view and edit default parameter values in the *Settings* dialog box.
			Default value: `false`
			See: Maintaining Your Default User Values [page 514]

Short Key for ABAP Customizing	Path and Parameter for Target Mapping	Category	Description
USERSETTINGS_C ONTENTDENSITY	renderers/fiori2/ componentData/config/ enableContentDensity	FLP UI Client Settings	**Specify whether end users can view and toggle the** *Optimized for Touch Input* **option** This setting is intended for users of hybrid devices that combine both touch and mouse events. Set to `false` to hide the *Optimized for Touch Input* option from the *Settings* dialog box. Default value: `true` See: Managing Your Settings [page 509]
USER_IMAGE	renderers/fiori2/ componentData/config/ enableUserImage	FLP UI Client Settings	**Specify whether a user profile image is displayed in the launchpad.** This parameter is only relevant if you use SAP Jam integration. Default setting: `false` We recommend that you do not change the default setting. If you set this parameter to `true`, user profile images from SAP Jam will be displayed in the launchpad without asking users for consent. Local data privacy regulations may be relevant.
HOMEPAGE_ANIMA TION_MODE	renderers/fiori2/ componentData/config/ animationMode	FLP UI Client Settings	**Specified the degree of animation of UI elements when end users interact with the launchpad** **i Note** The animation mode is no longer necessary and available as the Me Area was removed. This parameter is deprecated and settings have no effect.

Short Key for ABAP Customizing	Path and Parameter for Target Mapping	Category	Description
NAVIGATION_GUI _STATEFUL_CONT AINER	renderers/fiori2/ componentData/config/ statefulApplicationCo ntainer/GUI	FLP UI Client Settings	**Specify whether to use a stateful application container for SAP GUI for HTML apps.** A stateful application container improves navigation performance when launching SAP GUI apps. Default value: `false` ⓘ Note This feature requires a minimum kernel version. For more information, see SAP note 2657777👉. See: Improving Navigation Performance for SAP GUI Applications [page 374]
HOMEPAGE_TILES _SIZE	renderers/fiori2/ componentData/config/ sizeBehavior	FLP UI Client Settings	**Specify the size of tiles on the home page or on the pages in spaces mode.** Valid values: • `Responsive`: The size of the tiles is set to `regular`, unless the screen width gets smaller than 375 px (for example on small mobile devices), in which case the size of the tiles is set to `small`. • `Small`: The tile size is fixed to `small` regardless of the available screen width.

Short Key for ABAP Customizing	Path and Parameter for Target Mapping	Category	Description
HOMEPAGE_TILES_SIZE_ENABLE_USER	`renderers/fiori2/ componentData/config/ sizeBehaviorConfigura ble`	FLP UI Client Settings	**Specify if users are allowed to set the tile size on their home page or on the pages in spaces mode.** Default value: `false`. ⓘ Note For custom tiles you need to enable the tile size change to ensure that the tiles are shown correctly. If your custom tiles do not support size change, you should set this option to `false`. See: Implementing a Custom Tile Type [page 578] and Enabling Tile Size Change for Custom Tiles [page 597]. See: Managing Your Settings [page 509]
APPFINDER_DISPLAYMODE	`renderers/fiori2/ componentData/ config/applications/ Shell-home/ AppFinderDisplayMode`	FLP UI Client Settings	**Specify how to display apps in the app finder.** Valid values: • `Tiles`: apps are displayed in a square container. • `AppBoxes`: apps are displayed in a more compact rectangular container. ⓘ Note This setting does not affect custom tiles. They are always displayed as tiles.
SESSION_TIMEOUT_STOP_DATA_REFRESH	`renderers/fiori2/ componentData/config/ sessionTimeoutTileSto pRefreshIntervalInMin utes`	FLP UI Client Settings	**Specify the time a user has to be inactive before the tiles stop sending polling data refresh requests.** After a defined idle time the ABAP server terminates the session for security reasons. Enter the time in minutes. The default value is 15 minutes. If you enter -1, you disable the timeout. See Configuring the Automatic Sign-Out [page 88] for additional information.

Short Key for ABAP Customizing	Path and Parameter for Target Mapping	Category	Description
SESSION_TIMEOUT_INTERVAL_IN_MINUTES	renderers/fiori2/component-Data/config/ `sessionTimeoutIntervalInMinutes`	FLP UI Client Settings	**Specify the timeout period in minutes according to platform configuration for session timeout.** The maximum value is 71582 minutes. If you enter a larger value, it automatically falls back to the allowed maximum value. If you enter -1, you disable the timeout. The setting is disabled by default. See Configuring the Automatic Sign-Out [page 88] for additional information.
SESSION_TIMEOUT_REMINDER_IN_MINUTES	renderers/fiori2/component-Data/config/ `sessionTimeoutReminderInMinutes`	FLP UI Client Settings	**Enter the time before session timeout when the keep alive popup window should be displayed.** The default value is 5 minutes. If you enter 0, no dialog will be displayed.

Short Key for ABAP Customizing	Path and Parameter for Target Mapping	Category	Description
XHRLOGON_MODE	xhrLogon/mode	FLP UI Client Settings	**Specify the mode for XMLHttpRequest (XHR) logon.** Possible values are `frame` (this is the default), `reload` and `logoffAndRedirect`. Usually, you don't have to change this parameter. Check the settings, if you have configured an authentication method that is HTML-based (e.g. SAML2-based or form-based log-in) and experience problems with authentification requests triggered by SAP Fiori launchpad or SAP Fiori apps based on SAPUI5. By default, the XHR logon uses an iFrame to display the logon page. Only if this is not possible due to restrictions of the logon page, it can be configured to a different mode. Example 1: If the involved logon page cannot be loaded inside an iFrame and the user session on the ABAP front-end server times out, the SAP Fiori launchpad shows a dialog with a white screen. Solution: Select XHR logon mode `reload`. Example 2: You are using a reverse proxy (e.g. SAP Web Dispatcher) for routing XHR requests to one or several back-end systems and a different logon mechanism like SAP logon tickets for authentication with these back-end systems. If the user session on the ABAP back-end server times out while the user session on the front-end server is not timed out, an XHR request to the back-end server does not lead to a re-logon to the back-end system. If you have configured xhrlogon mode 'reload', the browser continues endlessly showing a pop-up that the session has timed out and a reload is required. Solution: Select XHR logon mode `logoffAndRedirect`. Then a logoff to

Short Key for ABAP Customizing	Path and Parameter for Target Mapping	Category	Description
			the front-end server with a redirection to the current page is triggered in case of an expired session on the back-end system.
HOMEPAGE_FEATU REDGROUP	ushell/home/ featuredGroup/enable ushell/home/ featuredGroup/ frequentCard ushell/home/ featuredGroup/ recentCard	FLP UI Client Settings	**Set this parameter to true, if the Featured Group should be displayed as first group on the launchpad home page.** Default value: `false`. See Home Page [page 422] for additional information. With the parameters `frequentCard` or `recentCard` you can show only one of the cards. i Note This feature is experimental. Experimental features are not part of the officially delivered scope that SAP guarantees for future releases - this means that experimental features may be changed by SAP at any time for any reason without notice. Experimental features are NOT FOR PRODUCTIVE USE. You may not demonstrate, test, examine, evaluate or otherwise use the experimental features in a live operating environment or with data that has not been sufficiently backed up.

Short Key for ABAP Customizing	Path and Parameter for Target Mapping	Category	Description
FLP_CLIENT_ROLE	n/a	FLP UI Server Settings	**Specify the role of the client in your system.** ℹ **Note** The parameter is deprecated and has been replaced by the parameter SYS_INFO_TENANT_ROLE. Valid values: • **P**: Production client • **T**: Test client This setting overwrites the value of the *Client Role* field in *Display View "Clients": Overview* (transaction SCC4). Users can display the client role in the *About* dialog in the user actions menu. See Information About App and Launchpad Version [page 443].
FLP_EAM_ALIASES	n/a	FLP UI Server Settings	**Specify a comma-separated list of system aliases for the role-independent enablement of the SAP Easy Access menu.** ⁘ **Example** `LOCAL,FIORI` The specified aliases will be used to retrieve entries from the SAP Easy Access menu in the related back-end systems. See Integrating Applications from SAP Easy Access Menu [page 312]. You can hide the SAP Easy Access menu tabs (SAP Menu and User Menu) in the app finder using the parameters `enableEasyAccess`, `enableEasyAccessSAPMenu`, or `enableEasyAccessUserMenu`.
ENABLE_HELP	`renderers/fiori2/ componentData/config/ enableHelp`	FLP UI Client Settings	**To ensure that all help content is shown, the parameter ENABLE_HELP has to be set to TRUE.**

Short Key for ABAP Customizing	Path and Parameter for Target Mapping	Category	Description
THEMING_DEFAUL T_THEME	n/a	FLP UI Server Settings	To ensure compliance with accessibility standards, please make sure you provide a high-contrast theme in your list ofChange the default theme for SAP Fiori launchpad by entering a theme ID. The specified theme is applied to the launchpad homeTo ensure compliance with accessibility standards, please make sure you provide a high-contrast theme in your list of custom themes.

The theme should support the design guideline SAP Fiori 2.0 or SAP Fiori 3.

You have the following options:

- Set a different SAP standard theme. The default theme is `sap_fiori_3` (Quartz theme)
- Set a custom theme created with UI theme designer

For further information on creating custom themes, see .

For more information on the launchpad theme settings, see Setting Themes for the Launchpad [page 70]. |
| THEMING_HIDE_S AP_THEMES | n/a | FLP UI Server Settings | When the parameter is set to TRUE*Settings*, dialog of SAP Fiori Launchpad.

> i Note
>
> custom themes. |

Short Key for ABAP Customizing	Path and Parameter for Target Mapping	Category	Description
EXPOSURE_SYSTE M_ALIASES_MODE	n/a	FLP UI Server Set-tings	Specify whether the system aliases should be cleared or kept. When nothing is set, the system aliases are kept as defined in the target mapping. You have the following options: • CLEAR - the system aliases are cleared during exposure. Only the system aliases used for replication are being removed. • KEEP - the system aliases are not cleared and kept as defined in the target mapping. After the exposure, the administrator has to do a system alias mapping in the launchpad on SAP BTP. ⁱ Note In an embedded scenario, we recommend to set the parameter to CLEAR. Note that only the system aliases that are used for replication are being removed. For further information on the embedded deployment scenario, see Deployment Options [page 10].
INPUTFIELD_HIS TORY	apps/ inputFieldHistory/ enabled	FLP UI Client Set-tings	Enable this parameter so that the system saves the values a user has entered into a field in an app (only for SAPUI5 apps using smart controls (SmartField and Smart-FilterBar) that are supporting a history). When a user sets the cursor in a field, the last five entries are shown as a value list. Default value: false You can delete the history with transaction /UI2/FLP_DEL_PERS. See Delete Personalization Data [page 355] for more information. Users can enable or disable the history and clear their own entered values. See User Actions Menu [page 424].

Short Key for ABAP Customizing	Path and Parameter for Target Mapping	Category	Description
UI5_PLACEHOLDE R_SCREEN	`apps/placeholder/ enabled`	FLP UI Client Settings	Enable the parameter to render placeholder screens and improve the perceived performance when loading. We recommend to enable the parameter unlesss problems with the placeholder loading arise. Default value: `false`
UI5_ASYNC_LOAD ER	n/a	FLP UI Server Settings	**Specify whether asynchronous loading for SAPUI5 modules and preload files is used by the SAP Fiori launchpad.** ⓘ Note This option requires that the apps follow the latest UI5 best practices for SAPUI5 applications. For more details see Best Practices for App Developers🔗. Default value: `false`
IE_DEPRECATION _WARNING	`renderers/fiori2/ componentData/config/ enableIEDeprecationWa rning`	FLP UI Client Settings	ⓘ Note The parameter is deprecated. The launchpad is no longer supported on Microsoft Internet Explorer or Microsoft Edge Legacy🔗 versions. So activating the parameter has no effect. Decide if your users see a warning message when they open the launchpad in the Microsoft Internet Explorer. `true` enables the warning; `false` disables it. You should ensure that all users work with a supported browser.
SYS_INFO_SYSTE M_NAME	`services/Container/ adapter/config/ systemProperties/ systemName`	FLP UI Client Settings	**Specify a system name that should be shown in the *About* dialog, for example a customer-specific system name.** If nothing specific is defined here, the *About* dialog shows the standard system settings. Also see Information About App and Launchpad Version [page 443].

Short Key for ABAP Customizing	Path and Parameter for Target Mapping	Category	Description
SYS_INFO_SYSTE M_ROLE	services/Container/ adapter/config/ systemProperties/ systemRole	FLP UI Client Settings	**Specify the system role that should be shown in the *About* dialog.** Also see Information About App and Launchpad Version [page 443].
SYS_INFO_TENAN T_ROLE	services/Container/ adapter/config/ systemProperties/ tenantRole	FLP UI Client Settings	**Specify the tenant role that should be shown in the *About* dialog, for example "test" or "production".** If nothing specific is defined here, the *About* dialog shows the SAP entry. Also see Information About App and Launchpad Version [page 443].
SYS_INFO_PRODU CT_NAME	services/Container/ adapter/config/ systemProperties/ productName	FLP UI Client Settings	**Specify the product name that should be shown in the *About* dialog, for example a customer-specific product name.** If nothing specific is defined here, the *About* dialog shows the product name defined by SAP or defined by a task list (see Initial Setup for Embedded Deployment [page 11]). Also see Information About App and Launchpad Version [page 443].
SYS_INFO_HIDE_ PRODUCT_VERSIO N	n/a	FLP UI Server Settings	**Specify whether end users are allowed to see the product version in the user actions menu of their launchpad.** When set to `true`, the *Product Version* field is hidden in the *About* dialog of the user actions menu. Default value: `false` (the product version is displayed in the *About* dialog in the user actions menu). See Information About App and Launchpad Version [page 443].

Short Key for ABAP Customizing	Path and Parameter for Target Mapping	Category	Description
EXPOSURE_SUPPORT_LSAPI_TO_IBN_NAV	n/a	FLP UI Server Settings	Enables the launchpads on SAP BTP as consumers of content exposure to support cross-application navigation for Web Dynpro or SAP GUI applications using ABAP class CL_LSAPI_MANAGER.
			Applications based on Web Dynpro ABAP or SAP GUI do not use intent-based navigation for cross-application navigation. Instead, they use other navigation methods, such as object-based navigation or call transaction. You can use class CL_LSAPI_MANAGER to convert these navigation methods to intent-based navigation. See Cross-Application Navigation for ABAP-based Apps [page 537].
			Set the parameter to true if the launchpads on SAP BTP as consumers of content exposure should support this navigation method using ABAP class CL_LSAPI_MANAGER. See Exposing Launchpad Content to SAP Business Technology Platform [page 340].
PFCG_INCLUDE_G4BA	n/a	FLP UI Server Settings	Include OData v.4 services with the object type G4BA (backend) in *Role Maintenance* (transaction PFCG).
			Default value: true.
			See Assign Business Catalogs to Roles [page 330].
PFCG_INCLUDE_HTTP	n/a	FLP UI Server Settings	Include HTTP services with the object type HTTP (backend) in *Role Maintenance* (transaction PFCG).
			Default value: true.
			See Assign Business Catalogs to Roles [page 330].
PFCG_INCLUDE_INA1	n/a	FLP UI Server Settings	Include Information Access (InA) services with the object type INA in *Role Maintenance* (transaction PFCG).
			Default value: true.
			See Assign Business Catalogs to Roles [page 330].

Short Key for ABAP Customizing	Path and Parameter for Target Mapping	Category	Description
PFCG_INCLUDE_IWSG	n/a	FLP UI Server Settings	Include OData v.2 services with the object type IWSG (frontend) in *Role Maintenance* (transaction PFCG). Default value: true. See Assign Business Catalogs to Roles [page 330].
PFCG_INCLUDE_IWSV	n/a	FLP UI Server Settings	Include OData v.2 services with the object type IWSV (backend) in *Role Maintenance* (transaction PFCG). Default value: true. See Assign Business Catalogs to Roles [page 330].
PFCG_INCLUDE_SAPC	n/a	FLP UI Server Settings	Include ABAP Push Channel (APC) services with the object type SAPC (backend) in *Role Maintenance* (transaction PFCG). Default value: true. See Assign Business Catalogs to Roles [page 330].
PFCG_INCLUDE_TRAN_GUI	n/a	FLP UI Server Settings	Include SAP GUI-based transactions (backend) in *Role Maintenance* (transaction PFCG). Default value: true. See Assign Business Catalogs to Roles [page 330].
PFCG_INCLUDE_TRAN_UI5	n/a	FLP UI Server Settings	Include UI5-based transactions (frontend) in *Role Maintenance* (transaction PFCG). Default value: false. See Assign Business Catalogs to Roles [page 330].

Short Key for ABAP Customizing	Path and Parameter for Target Mapping	Category	Description
PFCG_INCLUDE_W DCA	n/a	FLP UI Server Settings	Include Web Dynpro applications with the object type WDCA (backend) in *Role Maintenance* (transaction PFCG). Default value: true. See Assign Business Catalogs to Roles [page 330].
PFCG_INCLUDE_W DYA	n/a	FLP UI Server Settings	Include Web Dynpro applications with the object type WDYA (backend) in *Role Maintenance* (transaction PFCG). Default value: true. See Assign Business Catalogs to Roles [page 330].
EXPOSURE_HOMEP AGE_STRUCTURE	n/a	FLP UI Server Settings	Specify which layout you want to use to structure your launchpad. You can choose between: • Groups - The exposed launchpad content is structured in groups. • Groups and Spaces/Pages - The exposed launchpad content is either structured in groups or in spaces and pages. This is the default value. • Spaces/Pages - The exposed launchpad content is structured in spaces and pages. See Exposing Launchpad Content to SAP Business Technology Platform [page 340].

Short Key for ABAP Customizing	Path and Parameter for Target Mapping	Category	Description
EXPOSURE_LOG_L EVEL	n/a	FLP UI Server Settings	Define which log messages should be stored in the exposure log.
			This setting considers the overall exposure status of an entity. This means that if you, for example, select **2 - Show entities with warnings and errors**, messages from successfully exposed entities will not be stored in the log. This is the default value to prevent your log file from growing too big. If you increase the log level, logs for successfully exposed entities or even requests to the API will be logged.
			You can choose between:
			• 1 - Show entities with errors • 2 - Show entities with warnings and errors • 3 - Show entities with errors, warnings and success messages • 9 - Show all entities including API calls
			See Manage Launchpad Content for Exposure [page 343].
FLP_START_WITH _SSO	n/a	FLP UI Server Settings	Start the launchpad or design-time tools with single sign-on. If the setting is set to `true`, single sign-on is enabled with the browser you have set in your control settings (Microsoft Edge or Internet Explorer). If this setting is set to `false`, single sign-on is not enabled and you will start in your browser that you have configured as default browser.
			The default is `false`.
PFCG_INCLUDE_T RAN_TILE_ONLY	n/a	FLP UI Server Settings	Include tile only-based transactions (frontend) in *Role Maintenance* (transaction `PFCG`).
			Default value: `false`.
			See Assign Business Catalogs to Roles [page 330].

Short Key for ABAP Customizing	Path and Parameter for Target Mapping	Category	Description
PFCG_INCLUDE_T RAN_URL	n/a	FLP UI Server Settings	Include URL app-based transactions (frontend) in *Role Maintenance* (transaction PFCG). Default value: `false`. See Assign Business Catalogs to Roles [page 330].
PFCG_INCLUDE_T RAN_URLT	n/a	FLP UI Server Settings	Include transactions with app type: Other App Types (frontend) in *Role Maintenance* (transaction PFCG). Default value: `false`. See Assign Business Catalogs to Roles [page 330].
PFCG_INCLUDE_T RAN_WCF	n/a	FLP UI Server Settings	Include WebClient UI-based transactions (backend) in *Role Maintenance* (transaction PFCG). Default value: `false`. See Assign Business Catalogs to Roles [page 330].
PFCG_INCLUDE_T RAN_WDA	n/a	FLP UI Server Settings	Include Web Dynpro-based transactions (backend) in *Role Maintenance* (transaction PFCG). Default value: `false`. See Assign Business Catalogs to Roles [page 330].
EXPOSURE_REPOS ITORY_EXPIRATI ON_DAYS	n/a	FLP UI Server Settings	Specify after how many days unused content stored in the exposure repository should be automatically removed. The default value is `100` (days). To disable the automatic clean up, you can set the value to `0`.

Related Information

Setting Parameters in SAP Fiori Customizing [page 63]
Alternative Ways of Setting Parameters [page 64]

1.3.2.2   Setting Launchpad Parameters

The recommended method for configuring the SAP Fiori launchpad is to set parameters in ABAP Customizing, but other methods are supported as well.

Related Information

1.3.2.2.1   Setting Parameters in SAP Fiori Customizing

You can configure the launchpad by setting parameters in Customizing on the front-end server.

Setting parameters in Customizing is the recommended way of configuring the launchpad.

Default Values, Cross-Client Settings and Client-Specific Settings

All parameters in the SAP Fiori launchpad configuration have default values. For a complete list of available launchpad configuration parameters, see Launchpad Configuration Parameters [page 29].

This list describes the coded default value for each parameter.

In Customizing, two types of settings are available:

- **Cross-client settings** are product-specific settings provided by SAP. If a parameter is configured in the cross-client settings, this overrides the coded default value in your system.
- If you configure **client-specific settings**, these override coded default values as well as cross-client settings

How to Set Configuration Parameters

In *SAP Reference IMG* (transaction SPRO), choose ▶ *SAP NetWeaver* ❯ *UI Technologies* ❯ *SAP Fiori* ❯ *SAP Fiori Launchpad Settings* ▶. Under this path the following Customizing activities are available:

- To display **system-wide** parameter values provided by SAP, choose *Display Cross-Client Settings* (transaction /UI2/FLP_SYS_CONF).

These settings are read-only.

- To change **client-specific** parameter values, choose *Change Client-Specific Settings* (transaction `/UI2/ FLP_CUS_CONF`).
These settings are **not** automatically recorded in transports, but you can manually add them to Customizing transports.

The configuration screen has the following sections:

Section	Description
Launchpad Configuration	In this section, you can set values for launchpad configuration parameters.
	When adding new entries, you can select a parameter from the value help. The value help shows the available parameter IDs and their descriptions.
	The data type of each parameter is displayed for your information. Set a property value as required.
	For more information on each parameter, see Launchpad Configuration Parameters [page 29].
Launchpad Plug-Ins	In this section, you can define which SAP Fiori launchpad plugins should be active. *Active* means that the plugin is loaded at runtime.
Configuration	In this section, you can set specific parameter values for each plugin.

How to Differentiate Between SAP-Delivered Settings and Custom Settings

In *Change Client-Specific Settings* (transaction `/UI2/FLP_CUS_CONF`), SAP-delivered settings and custom settings can be distinguished by the entry in column *Origin*. If the setting is predefined by SAP, the value S is displayed. Predefined settings can neither be deleted nor edited. If the setting is custom-made, the column is empty. Custom settings will override SAP-delivered settings. If a custom setting is deleted, it will fall back to the previous setting, if there is one.

Related Information

Alternative Ways of Setting Parameters [page 64]

1.3.2.2.2   Alternative Ways of Setting Parameters

The recommended method for configuring the SAP Fiori launchpad is to set parameters in ABAP Customizing, but other methods are supported as well. This section gives an overview of these methods and explains which setting takes precedence if a parameter has been defined in more than one place.

The following table shows the configuration methods and their priorities:

Priority	Configuration Method	Description
1	URL parameters	Parameters that you provide in a URL override any other parameters. ℹ Note The set of parameters that you can configure in the URL is different from the set of parameters that you can configure in ABAP Customizing or in a target mapping. For more information, see Setting Parameters in the Launchpad URL [page 66].
2	Target Mappings	In a target mapping, you can set parameters for a specific role. Parameters that you define in a target mapping override any parameters defined in Customizing. For more information, see Setting Parameters in a Target Mapping [page 68].
3	Customizing	This configuration method is only available on ABAP-based systems. Parameters that you define in Customizing on the front-end server override any parameters defined in the `FioriLaunchpad.html` file. For more information, see Setting Parameters in SAP Fiori Customizing [page 63] and Setting the Default Theme in SAP Business Client Customizing (Deprecated) [page 67].

Many parameters can only be configured using a subset of the methods described above. For more details about which parameters are available, see the respective documentation.

Related Information

Setting Parameters in SAP Fiori Customizing [page 63]
Launchpad Configuration Parameters [page 29]
Adjusting the Visual Appearance of the Launchpad [page 70]
Enabling Key User Adaptation [page 75]
Configuring the Option to Contact Support [page 95]
Configuring Sign-Out (Logout) [page 82]

1.3.2.2.2.1 Setting Parameters in the Launchpad URL

You can change the behavior of the SAP Fiori launchpad by appending specific parameters to the launchpad URL.

The standard URL for accessing the launchpad is as follows:

```
https://<server>:<port>/sap/bc/ui2/flp/FioriLaunchpad.html
```

You can add parameters using the following syntax:

```
https://<server>:<port>/sap/bc/ui2/flp/FioriLaunchpad.html?<parameter>=<value>
```

The following table shows the parameters that can be used and provides information about the possible values.

Parameter	Description
sap-language	This parameter allows you to set the display language of launchpad texts. Example: sap-language=en > **i Note** > If a value is set for this parameter, it overrides other language settings (see Determining the Logon Language) given the specified language is installed on the back-end server.
sap-theme	This parameter allows you to run the SAP Fiori launchpad with a custom theme. For more information, see .
sap-ushell-config-url	This parameter allows you to specify the location of a launchpad configuration file that contains a set of parameters.

> → Tip
>
> You can use the following URL parameter value as an aid when troubleshooting issues in the launchpad: sap-ui-debug=true. This will open the launchpad in debug mode. Additional launchpad URL parameters relating to the cache buster are also available for performance troubleshooting. For more information, see Cache Buster for SAP Fiori Launchpad and SAP Fiori Apps [page 367].

> → Tip
>
> You can also apply SAPUI5 parameters to the launchpad URL; however, keep in mind that in some cases this may have an adverse effect on the optimal operation of the SAP Fiori launchpad. The reason being that URL parameters override certain default SAPUI5 configuration parameters that are set automatically by the launchpad according to server-side or user settings (for example, user language and themes).

> You can find a list of URL-relevant parameters for SAPUI5 in (relate only to parameters whose URL attribute is marked as Yes).

Related Information

1.3.2.2.2.2 Setting the Default Theme in SAP Business Client Customizing (Deprecated)

You can define the default theme for the launchpad in SAP Business Client Customizing.

> **i** Note
>
> The parameter `THEME` has been replaced by the parameter `THEMING_DEFAULT_THEME` which you can set in the ABAP Customizing. The settings made for the THEME parameter in SAP Business Client customizing are not migrated and have to be manually added in the ABAP customizing. See Launchpad Configuration Parameters [page 29].

SAP delivers some parameters with default values via transaction `/UI2/NWBC_CFG_SAP`.

You can change these default values in transaction `/UI2/NWBC_CFG_CUST`. In *Customizing*, choose ▷ *SAP NetWeaver* ▷ *UI Technologies* ▷ *SAP NetWeaver Business Client* ▷ and launch the IMG activity *Assign Parameter Values*. The system always prefers the values that you have defined over the default values.

Parameters relevant for the launchpad have the path filter `SAP_FLP`:

Parameter Name	Translatable	Description
THEME	no	Enter a theme ID to set a different default theme for the launchpad.
		The theme should support the design guideline SAP Fiori 2.0 or SAP Fiori 3.
		You have the following options:
		• Set a different SAP standard theme
		• Set a custom theme created with UI theme designer. For further information on creating custom themes, see .
		The specified theme is applied to the launchpad home page and content embedded in the launchpad.

Related Information

1.3.2.2.2.3 Setting Parameters in a Target Mapping

You can configure the launchpad in a modification-free and role-based way by setting parameters in a target mapping.

Target mappings allow you to configure the launchpad without modifying any files delivered by SAP. You can create multiple target mappings with different configurations and assign these to different roles. In the launchpad designer, configure the target mapping as follows:

Detailed Configuration Description

Parameter	Value
Semantic Object	**Shell**
	This value needs to be written exactly like this.
Action	**bootConfig**
	This value needs to be written exactly like this.
Application Type	**SAPUI5 Fiori App**
Title	Enter a title.
URL	Leave this field empty.
ID	Enter an ID, for example **Z_FLP_CONFIG**.
Parameters	In the *Name* column, enter the complete path of each parameter, separated by slashes, for example **renderers/fiori2/componentData/config/ enablePersonalization**.
	Enter the value for each configuration parameter into the *Default Value* column.
	i Note
	You can find the target parameters in the table in Launchpad Configuration Parameters [page 29] Note that target mapping is only possible for client settings.

Example

SAP ships the following catalogs containing sample target mappings:

- SAP: Configuration for enabling notifications (`/UI2/CONFIG_NOTIFICATION`)
- SAP: Configuration for disabling personalization of the home page (`/UI2/CONFIG_PERS_OFF`)

1.3.2.2.2.4 Setting Non-Transportable Parameters

You can define parameters that should not be transported to other systems.

You can change these default values in transaction `/UI2/FLP_BE_SYS`. In *Customizing*, choose ▶ *SAP NetWeaver* ❯ *UI Technologies* ❯ *SAP Fiori* ❯ *Configuring SAP Fiori Launchpad* ❯ *Cross-Client Settings* ◀ and launch the IMG activity *Non-Transportable Settings*.

> **i Note**
>
> The settings are not transported to other systems.

Parameter Name	Value
SYSTEM_SCENARIO	You can use this parameter to specify whether your system is based on a hub or embedded deployment of the SAP Fiori front-end server. This setting allows the launchpad design-time tools to offer an optimized logic in case you have implemented an embedded deployment. Choose one of the two values: • **Hub**: The SAP Fiori front-end server is deployed as standalone system apart from the back-end server. • **Embedded**: The SAP Fiori front-end server is deployed along with the back-end server as part of the same system setup.

Related Information

[Migrate SAP Fiori Front-End Server from Standalone to Embedded System Deployment]

1.3.2.3 Adjusting the Visual Appearance of the Launchpad

You can adjust the visual appearance of the SAP Fiori launchpad by setting different themes or applying flavors.

Themes

You can change the look and feel of the launchpad homepage and launchpad content by choosing a different SAP standard theme or a custom theme. To create and edit custom themes for the launchpad, you use the UI theme designer. For example, you can adjust the color scheme to your branding colors or display a custom logo in the launchpad header bar, instead of the SAP logo.

Flavors

For SAP GUI transactions and Web Dynpro ABAP applications, SAP Screen Personas offers additional options to change the visual appearance. A flavor is always linked to a specific SAP GUI transaction or Web Dynpro ABAP application. On the other hand, one particular SAP transaction/application can have an infinite number of flavors. For example, you can define one flavor for administrators and another for end users. You can use flavors to modify images, hide or insert UI elements, or add events. To create and edit flavors for the launchpad, you use SAP Screen Personas.

Related Information

SAP Screen Personas

1.3.2.3.1 Setting Themes for the Launchpad

You can change the look and feel of the launchpad homepage and launchpad content by choosing a different SAP standard theme or a custom theme created with the UI theme designer.

To set a theme, you specify its ID. There are several options for setting themes:

- Administrators can change the SAP default theme for the launchpad in two ways:
  - in *ABAP Customizing* using the parameter `THEMING_DEFAULT_THEME`
    See .
  - using the URL parameter `sap-theme`
    See .
- Users can change the theme in the personalization settings, provided this option was not disabled by an administrator by setting the parameter `USERSETTINGS_SET_THEME` to `false`.

See Managing Your Settings [page 509] and Launchpad Configuration Parameters [page 29].

Setting Themes from a Different Domain

You can also provide a theme that is located in a different domain for your users. This applies to themes that are e.g. located on a different server or that are referenced with a URL parameter.

1. If you want to define a new theme, use the UI theme designer (see for a detailed description). Setting Themes in the SAPUI5 Demo Kit gives you further information on remote themes.
2. If you want to use a theme from a different domain, you need to add the theme URL to the allowlist to ensure that the theme can be loaded. For security reasons themes that are not on the allowlist cannot be used. You can maintain the allowlist with UCON maintenance (transaction UCONCOCKPIT) or in the database table HTTP_WHITELIST. Read HTTP Allowlist Scenario: Process and Managing HTTP Allowlists for a detailed description.

> **i Note**
>
> If the theme is not on the allowlist, it is not available in the system. If it is set as a default, the default user theme will be loaded instead.

3. Set your custom theme as default theme (see above).

Related Information

Adjusting the Visual Appearance of the Launchpad [page 70]

1.3.2.3.2   Assigning Flavors to Launchpad Content

You can assign flavors created with SAP Screen Personas to Web Dynpro ABAP applications or SAP GUI transactions that are integrated in the launchpad.

Prerequisites

In order for the desired flavor to be displayed to a user or group of users, you have to make sure that it is assigned to these users or to their authorization role.

To check to which user or role a flavor is assigned, you use transaction /PERSONAS/ADMIN. For more information, see *Flavor Maintenance* under http://help.sap.com/personas.

Context

A flavor defines a specific personalization of screens of a particular application. An application can have more than one flavor, with each screen looking different if so desired.

You can assign flavors to the following application types in the launchpad:

- Web Dynpro ABAP application
- SAP GUI transaction

Procedure

1. Find out the ID of the flavor you want to assign.

   Use the   (*Deep Link*) button in the flavor manager and copy the value of the *sap-personas-flavor* parameter.

2. In the launchpad designer, create a target mapping or select an existing target mapping and choose *Configure*.

3. In the *Parameters* section, add a new row with the following data:

Field Name	Value
Name	`sap-personas-flavor`
Default value	Enter the flavor ID.

Results

The selected flavor is applied to the target application if the application user or his role is assigned to the flavor.

Related Information

1.3.2.4 Customizing the Shell Bar

The right-hand side of the shell bar in SAP Fiori launchpad is customizable, allowing you to add a limited number of icons for actions and custom plug-ins.

> **i Note**
>
> In right-to-left (RTL) languages, the icons mentioned in this topic will be displayed on the left-hand side of the shell bar.

The right-hand side of the shell bar contains reserved slots for the following standard icons:

- Q Search
- ⌂ Notifications

You can add up to three additional icons to the right-hand side of the shell using the following methods:

- Move any of the four standard actions from the user actions menu to the shell bar by setting the following configuration parameters to `true` in in the launchpad customizing (see Launchpad Configuration Parameters [page 29] and Setting Parameters in SAP Fiori Customizing [page 63]). You can also use target mappings if you need a role-based assignment (see Setting Parameters in a Target Mapping [page 68]. Note that the shell bar component is called `Shell`.

Short Keys for ABAP Customizing	Parameters for Target Mappings (Path: `renderers/fiori2/ componentData/ config/`)	Actions
`HEADER_APPFINDER_ACTION`	`moveAppFinderAction ToShellHeader`	App Finder
`HEADER_USERSETTINGS_ACTION`	`moveUserSettingsAct ionToShellHeader`	Settings
`HEADER_EDITHOMEPAGE_ACTION`	`moveEditHomePageAct ionToShellHeader`	Edit Home Page
`HEADER_CONTACTSUPPORT_ACTION`	`moveContactSupportA ctionToShellHeader`	Contact Support

The default value for each of these parameters is `false`.

> **→ Tip**
>
> To prevent clutter in the shell bar, we recommended adding only the frequently-used actions to it.

- Add custom icons or plug-ins via the `addEndItem` renderer API function. This method requires developer assistance. For more information, see Adding Custom Icons to the Shell Bar [page 615].

Restrictions

The following restrictions apply when moving the icons or adding custom icons to the right-hand side of the shell bar:

- The slots used by the standard Search and Notifications icons in the shell bar are reserved. If you disable the standard icons or they are not available on your platform, you cannot use their empty slots for additional icons.
- A maximum of three additional slots are available to you.
- You cannot move the default icons from the shell bar to the user actions menu.
- Any custom icons or user actions you add to the shell are inserted between the Search and Notifications icons.
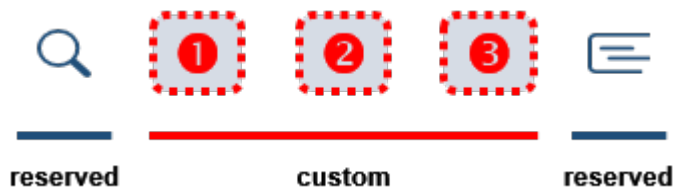
Positioning of reserved and custom slots in the right-hand side of the shell bar

- If more than three icons are added to the shell bar in the empty slots, the following rules apply:
  - User actions take preference over custom icons added via the API method. For example, if you move three actions from the user actions menu to the shell bar and then add one custom icon via the API method, the three actions from the user actions menu will be displayed but not the custom icon.
  - If you move more than three user actions to the shell bar, the actions are moved according to priority (highest to lowest): App Finder > Settings > Contact Support > Edit Home Page
  - If you move a user action to the shell bar and there are no available slots, it will remain in the user actions menu.
- If you move the Edit Home Page action to the shell bar, it will not be shown in the shell bar when the app finder is being used or when a user is navigating in an application.

Related Information

Setting Parameters in a Target Mapping [page 68]

1.3.2.5 Enabling Key User Adaptation

Here's what you as an administrator have to do to enable a key user to change the user interface (UI) of apps in the SAP Fiori launchpad using key user adaptation.

- Assign the key user to the authorization role SAP_UI_FLEX_KEY_USER.

> ### i Note
>
> If you want to define your own roles, make sure to copy the SAP_UI_FLEX_KEY_USER role and create them based on this role.

- Ensure that the key user uses the new FioriLaunchpad.html page.
- Ensure that the key user has the required authorizations for transports.
- Ensure that you have activated the required ICF nodes. For more information, see SICF Services [page 24].

> ### ⚠ Caution
>
> As UI changes made directly in the productive system might interfere with transported changes, we do not recommend that key users use key user adaptation in the productive system. Instead, they should use key user adaptation in a test system and then transport the UI changes to the productive system where all users can access the adapted version of the app. For more information, see Working with App Variants [page 482] under *Publish*.

Here's more information:

How has an app to be built to be used with key user adaptation?	See
What UI changes can a key user make?	See Adapting SAP Fiori UIs at Runtime - Key User Adaptation [page 459]

1.3.2.6 Enabling Personalization of Object Pages (Experimental)

If you want end users to be able to personalize object pages used in apps in the SAP Fiori launchpad, you can enable this experimental feature.

Context

> ### i Note
>
> Experimental features are not part of the officially delivered scope that SAP guarantees for future releases - this means that experimental features may be changed by SAP at any time for any reason without notice.

> Experimental features are NOT FOR PRODUCTIVE USE. You may not demonstrate, test, examine, evaluate or otherwise use the experimental features in a live operating environment or with data that has not been sufficiently backed up.

Here's how you enable the personalization feature:

Procedure

1. In the launchpad designer, in a catalog, create a target mapping with the intent `Shell-plugin` and the following configuration:

Field	What to enter
Semantic Object	**Shell** Please enter exactly this value.
Action	**plugin** Please enter exactly this value.
Application Type	*SAPUI5 Fiori App*

Field	What to enter
Title	Enter a title, for example **Enable Personalization**.
URL	Please leave this field empty.
ID	**sap.ushell.plugins.rta-personalize** Please enter exactly this value.
Information	Optional: Enter a description.
Device Types	Please only select *Desktop*.
Parameters	Enter a parameter with the name **enabled** and the default value **true**.

2. In *Role Assignment* (transaction PFCG), assign the catalog to a role.
3. If you have not already done this, assign the role to the respective users.

Related Information

Personalizing Object Pages (Experimental) [page 494]

1.3.2.7 Disabling Sharing of Views by End Users

You can limit the ability to share views to key users only.

In the standard system, end users can personalize views, and they can share views with other users by setting them to *Public*.

> **i Note**
>
> This function is only available if it is enabled for the respective app.

If you do not want end users to be able to share views, you can limit this ability to key users only.

To limit the ability to share views to key users only, implement SAP note 2658662.

1.3.2.8 Translating UI5 Flexibility Key User Adaptations

When key users adapt the user interface of an application, they can create or change UI texts. You can translate these UI texts into other languages.

If you have access to the back-end system, you can use the standard SAP translation tools to translate UI texts which have been created by key users.

Make sure that you have set up the standard SAP translation tools. For more information, see Setting Up and Coordinating Translation.

Proceed as follows:

1. In *Transport Organizer* (transaction SE10), identify the transport that contains the key user changes.
2. Identify the object names of the changes that contain UI texts.
3. In the *Translation Editor* (transaction SE63), choose *Transport Object*.
4. Enter the object name into the *Transport Object* field.

To search for UI texts that have been created by key users, you can also proceed as follows:

1. In the *Translation Editor* (transaction SE63), choose *Short Texts*.
2. In the *Object Type Selection* dialog box, choose ▶ *A5 User Interface Texts* ▶ *LRDS LRepository Client-Specific Short Texts* ▶.
3. In the *Object Name* field, use the input help to search for text objects that have been created by key users.

For more information on the SAP translation tools, see Translation Tools for Translators.

1.3.2.9   Enabling the Display of User Profile Pictures

As an administrator, you can enable the display of a user's profile picture or avatar in various controls in the SAP Fiori launchpad, e.g. in the shell bar.

This functionality is currently provided by SAP Jam and the collaboration features of its ABAP Social Media Integration (ABAP SMI) component.

Once SAP Jam is integrated and set up as described and users have uploaded an image file to their SAP Jam profile, they will see their picture instead of the default 👤 *(Person Placeholder)* icon.

> **i Note**
>
> - The user's profile picture is neither stored on the SAP Fiori server nor cached by the browser. For every new browser session, the user's current picture is retrieved from SAP Jam and then presented in the UI.
> - Every user is responsible for updating their own profile picture or avatar in their SAP Jam profile.
> - To remove a user's profile picture in the SAP Fiori launchpad, users must remove their profile picture from their SAP Jam profile.

1.3.2.10 Enabling Notifications in the Launchpad

Notifications in enterprise applications are an effective way to make users aware of situations that require their timely action or attention. Examples include workflow scenarios, such as the assignment of new purchase order approvals, or invitations to join SAP Jam groups. Users can receive push notifications in the launchpad and/or e-mail notifications.

SAP Fiori launchpad can consume notifications from the SAP Gateway Notification Channel, which is part of the SAP_GWFND software component. The SAP Gateway Notification Channel is a framework for applications to deliver notifications to end users through various channels.

Notifications are disabled by default in the launchpad, and none of the notification-related UI elements are made visible to the user. This topic describes how to configure the notification parameters in the launchpad to enable the required service and UI elements.

Prerequisites

Before you can enable notifications in the launchpad, you need to configure the SAP Gateway Notification Channel. For more information, see section Notification Channel in the SAP NetWeaver Application Server for ABAP help on SAP Help Portal.

> → Tip
>
> The *Push Channel Settings* topic contains especially important information for setting up a web socket as the push channel.
>
> The *Add Push for Use with SAP Mobile Platform* topic contains information about adding notifications to mobile devices using SAP Mobile Platform.

Configuration Parameters for Enabling Notifications

To enable push notifications in the launchpad, you use need to configure the following parameters. For more information about the ways in which you can configure parameters for the launchpad, see Alternative Ways of Setting Parameters [page 64].

> → Tip
>
> SAP provides a default catalog that contains the relevant target mapping for conveniently enabling notifications in SAP Fiori launchpad: /UI2/CONFIG_NOTIFICATION. By assigning this default catalog to user roles you can quickly configure notifications for your launchpad users.
>
> See: Setting Parameters in a Target Mapping [page 68]

Parameters for Enabling the Launchpad Notification Service

Short Key (for ABAP Customizing)	Path and Parameter (for Target Mapping)	Description
`NOTIFICATIONS`	`services/ Notifications/ config/enabled`	Set this parameter to `true` to activate the Notification UShell service and enable the consumption of push notifications in the launchpad.
		For more information about the service, go to SAPUI5 Demo Kit: Controls and then navigate to ▌▶ *API REFERENCE* ❯ *sap.ushell.services* ❯ *Notifications* ▌
		Default value: `false`
`NOTIFICATIONS_S ERVICE_URL`	`services/ Notifications/ config/ serviceUrl`	Set this parameter to refer to the Notification OData service root URL (or base URL). This path is a combination of the hostname, the published Notification Channel service group, repository ID, service name, and the service version.
		Format: `http(s)://<hostname of Notification Hub server>/<ICF root node>/<published service group namespace>/<service group ID>/<repository ID>/<service namespace>/<service ID>/<service version>/`
		❖ Example `https://pluto/sap/opu/odata4/iwngw/ notification/default/iwngw/notification_srv/ 0001/` Where: • Hostname: `pluto` • ICF root node: `/sap/opu/odata4` • Service group namespace: `iwngw` • Service group ID: `notification` • Repository ID: `default` • Service namespace: `iwngw` • Service ID: `notification_srv` • Service version: `0001`
		The *Configuration for Notification Channel Hub* topic in the *SAP Gateway Foundation* documentation describes how to publish a service group.

Short Key (for ABAP Customizing)	Path and Parameter (for Target Mapping)	Description
NOTIFICATIONS_P OLLING_INTERVAL	services/ Notifications/ config/ pollingInterval InSeconds	The Notification UShell service uses either a WebSocket connection or a polling mechanism to retrieve updates from the SAP Gateway Notification Channel. The WebSocket protocol is the preferred communication method. If the notification service cannot establish a WebSocket connection from the browser to the notification channel, then it will implement the fallback polling policy at an interval (in seconds) that is determined by this parameter. Default value: 60 (seconds) i Note When you set a custom value, you need to consider the optimal balance between network traffic volume and the frequency of notification updates. The lower the polling interval, the more frequent the notification updates, but the higher the load on network traffic over a given timespan. Testing may be required to optimize this setting for your particular setup and environment.

Parameters for Enabling the Notification UI Components in the Launchpad

Once you have configured the parameters to enable the Notification service, you need to configure the following parameters to add the out-of-the-box UI elements to the launchpad.

> i Note
>
> You can also use the SAP Gateway Notification Channel APIs to develop and implement your own UI visualization and end-user interaction with notifications for the launchpad. In this case, do not specify any of these parameters.

Short Key (for ABAP Customizing)	Path and Parameter (for Target Mapping)	Description
NOTIFICATIONS_UI	renderers/ fiori2/ componentData/ config/ enableNotificat ionsUI	Set this parameter to `true` to enable the out-of-the-box UI modules that allow users to display and interact with notifications in the launchpad: • Notifications icon (🔔) in the shell header bar that users use to open the Notifications area. • Notifications area where users can view and interact with their notifications. • A badge over the 🔔 icon to show the number of new notifications received. • Alert popups that immediately show when a high priority notification is received. Users can disable this feature. • A set of notification-specific preferences that each user can personalize. Default value: `false`

Settings for E-Mail Notifications

The e-mail address of the recipient is read from the user master record maintained on the Frontend Server. Please make sure that the users' e-mail addresses are maintained there.

Related Information

1.3.2.11 Configuring Sign-Out (Logout)

You can do the following configuration:

Configuration Activity	Description	More Information
Configure a sign-out screen	After users sign out from the SAP Fiori launchpad, the browser displays a generic sign-out screen. You can configure a custom HTML page that is displayed as the sign-out screen. This is an optional configuration step.	Configure a Sign-Out Screen for the SAP Fiori Launchpad (Optional) [page 83]

Configuration Activity	Description	More Information
Configure sign-out from integrated systems	When users sign out from the SAP Fiori launchpad, it is important that they are also logged out from systems that are integrated with the launchpad. This works out of the box in the most common scenarios. In complex scenarios, more system might be accessed through a reverse proxy, like SAP Web Dispatcher. In this case you can use a custom sign-out page to trigger logout from these systems.	Configuring Logout from Integrated Systems [page 84]
Configure automatic sign-out	For security reasons, users are signed-out from the launchpad after a certain time of inactivity (e.g. no mouse or keyboard action). By default this is set to 30 minutes but you can change the time.	Configuring the Automatic Sign-Out [page 88]

1.3.2.11.1 Configure a Sign-Out Screen for the SAP Fiori Launchpad (Optional)

After users sign out from the SAP Fiori launchpad, the browser displays a generic sign-out screen. You can configure a custom HTML page that is displayed as a sign-out screen.

Procedure

1. On the SAP Fiori front-end server that runs the SAP Fiori launchpad, start transaction *Maintain Services* (transaction `sicf`).
2. On the start screen, choose *External Aliases*.
3. Select a host in the hierarchy that you want to be the root node of the alias, and choose *Create New External Alias*.
4. Enter the following data:
   - *External Alias*: **/sap/public/bc/icf/logoff**
   - *Trg Element*: **/sap/public/bc/icf/logoff**
   - On the ▌ *Error Pages* ❯ *Logoff Page* ▐ tab page: In *Redirect*, enter the URL of the logout page in HTML format.
5. Save your entries.

Next Steps

For SAP NetWeaver 7.31, see SAP Library for SAP NetWeaver on SAP Help Portal at http://help.sap.com/ nw731 ▸ *Application Help* ▸ *Function-Oriented View* ▸ *Application Server* ▸ *Application Server Infrastructure* ▸ *Connectivity* ▸ *Components of SAP Communication Technology* ▸ *Communication Between ABAP and Non-ABAP Technologies* ▸ *Internet Communication Framework* ▸ *Development* ▸ *Server-Side Development* ▸ *Creating and Configuring ICF Services* ▸ *External Aliases* ▸.

For SAP NetWeaver 7.4 and 7.5, see SAP Library for SAP NetWeaver on SAP Help Portal at http://help.sap.com/ nw74 or http://help.sap.com/nw75 : ▸ *Application Help* ▸ *Function-Oriented View* ▸ *Application Server* ▸ *Application Server Infrastructure* ▸ *Functions and Tools of SAP NetWeaver Application Server* ▸ *Connectivity* ▸ *Components of SAP Communication Technology* ▸ *Communication Between ABAP and Non-ABAP Technologies* ▸ *Internet Communication Framework* ▸ *Development* ▸ *Server-Side Development* ▸ *Creating and Configuring ICF Services* ▸ *External Aliases* ▸.

1.3.2.11.1.1 Configuring Logout from Integrated Systems

You may need to configure one or more logout rules and define a custom sign-out page to ensure logging out completely from all integrated systems.

Overview

When using other systems that are integrated with SAP Fiori launchpad, after signing out, an open browser window may still contain session cookies. A user who has access to the open browser window can access these systems without having to authenticate. The solution described in this topic ensures that session cookies of all systems are removed when signing out from SAP Fiori launchpad.

Scenarios Not Requiring Logout Configuration

For the following scenarios, logging out is performed completely and you do **not** need to perform the extra logout configuration described in this topic.

- SAP Fiori launchpad front-end server.
- Enterprise Search system (AS ABAP).
- Systems used to load remote tile catalogs, such as SAP HANA KPI tiles for SAP Smart Business.
- SAP Lumira™ running in SAP Fiori launchpad.

  > **i** Note
  >
  > For releases of SAP Lumira prior to 1.18, it is necessary to perform the configuration as described in SAP Note 2010502 .

- Systems that are accessed to start Web Dynpro ABAP or SAP GUI for HTML applications in SAP Fiori launchpad based on report launchpad customizing with specific application types (not plain URL) - either through SAP Web Dispatcher or directly on the system.

Scenarios Requiring Configuration to Ensure Complete Logout from All Systems

You need to configure one or more logout rules in SAP Web Dispatcher and define a custom logout page to ensure logging out completely from all systems for the following scenarios:

- SAP Gateway systems (hub) that are not installed on the front-end server (other client, server, or system); these are used to connect to ABAP application back-end systems.

  > i Note
  >
  > To keep the complexity of the system landscape to a minimum, we advise having only one SAP Gateway server on the SAP Fiori front-end server.

- SAP HANA or other systems accessed via REST or OData, for example, but not used to load remote catalogs.

  > i Note
  >
  > This does not include SAP Lumira scenarios, as mentioned above.

Related Information

Adding SAP Web Dispatcher Logout Rules [page 85]
Creating a Custom Sign-Out Page [page 86]
Ensuring Complete Logout from Integrated Systems [page 627]

1.3.2.11.1.2 Adding SAP Web Dispatcher Logout Rules

When using other systems that are integrated with SAP Fiori launchpad, after signing out, an open browser window may still contain session cookies. A user who has access to the open browser window can access these systems without having to authenticate. SAP Web Dispatcher logout rules are used with a custom sign-out page to make sure that all system sessions are closed when signing out of SAP Fiori launchpad.

Such a rule leading to a login may look as follows for the scenario where SAP Gateway is not on the front-end server in the SAP Web Dispatcher profile file `sapwebdisp.pfl`:

```
wdisp/system_<rule number> = SID=<Gateway System ID>, EXTSRV=https://<server
host and port>, SRCSRV=*:<Fiori Launchpad port>, SRCURL=/sap/opu/odata/
```

For a corresponding logout rule, it is recommended to use a URL segment parameter to be able to distinguish several systems of the same platform which have the same logout URL:

1. Create a text file and name it `modification_rule.txt`.
2. Add a corresponding modification rule to the file you created in step 1, for example:

```
if %{PATH} regimatch ;o=<Gateway System ID>
SetHeader x-sap-webdisp-target-sid <Gateway System ID> [break]
```

   Replace `<Gateway System ID>` by the system ID (SID) of your SAP Gateway system.
3. Add this modification rule file to the file `sapwebdisp.pfl` using the following path:

```
icm/HTTP/mod_<number> = PREFIX=/,FILE=modification_rule.txt
```

For the scenario where the SAP HANA system is accessed via OData but not used to load remote catalogs, both the rule leading to login and the modification rule for logout look quite similar:

> **i Note**
>
> The parts specific to SAP HANA are marked below in bold.

```
wdisp/system_<number> = SID=<HANA System ID>, EXTSRV=https://<server
host and port>, SRCSRV=*:<Fiori Launchpad port>, SRCURL=/sap/bi/;/sap/
hana/;/sap/ui5/;/sap/vi/;/sap/viz/
```

```
if %{PATH} regimatch ;o=<HANA System ID>
SetHeader x-sap-webdisp-target-sid <HANA System ID> [break]
```

Replace `<HANA System ID>` by the system ID (SID) of your SAP HANA system.

Related Information

1.3.2.11.1.3 Creating a Custom Sign-Out Page

SAP Web Dispatcher logout rules are used with a custom sign-out page to make sure that all system sessions are closed when signing out of SAP Fiori launchpad.

Context

Adapt the following:

Procedure

1. Create an HTML file `logout.html` as in the example below. The sign-out page URL must use SAP Web Dispatcher as its origin.

2. For each ABAP system from which you want to log out, add a `requestLogout` call with the logout URL matching the rule configured on the SAP Web Dispatcher:

   ```
   requestLogout("sap;o=<ABAP System ID>/public/bc/icf/logoff");
   ```

3. For each SAP HANA system from which you want to log out, call `requestHanaLogout` with the system ID from the corresponding SAP Web Dispatcher rule as parameter:

   ```
   requestHanaLogout("<HANA System ID>");
   ```

4. Redirect to the actual sign-out page:

   ```
   document.location = <Your logout page>
   ```

Example

```
<!DOCTYPE html>
<html>
<head>
    <meta http-equiv="Content-Type" content="text/html;charset=UTF-8"/>
    <meta http-equiv="X-UA-Compatible" content="IE=edge"/>
    <!-- Instead of this, you may include a jQuery version from your internal
network if available-->
    <script src="//ajax.googleapis.com/ajax/libs/jquery/1.11.0/jquery.min.js"></
script>
</head>
<body>
    <em id="message">Logout is in progress...</em>
    <script>
    (function () {
        "use strict";
        /*global document, jQuery*/
        var iPending = 0;

        function finishLogout() {
            iPending -= 1;
            if (iPending <= 0) { //logout done for all URLs
                //Change message text on logout page
                document.getElementById("message").innerHTML = "You are logged
out";

                //(3) Client-side redirect
                document.location = "http://www.sap.com"; //REPLACE with your
actual logout page
            }
        }
        function requestHanaLogout(sSystem) {
            iPending += 1;
            jQuery.ajax({
                type: "HEAD",
                url: "/sap;o=" + sSystem + "/hana/xs/formLogin/token.xsjs",
                headers: {
                    "X-CSRF-Token": "Fetch"
                },
            }).done(function (oData, oStatus, oXhr) {
```

```
            jQuery.ajax({
                type: "POST",
                url:  "/sap;o=" + sSystem + "/hana/xs/formLogin/
logout.xscfunc",
                headers: {
                    "X-CSRF-Token": oXhr.getResponseHeader("X-CSRF-Token")
                }
            }).always(finishLogout);
        }).fail(finishLogout);
    }
    function requestLogout(sUrl) {
        iPending += 1;
        jQuery.get(sUrl).always(finishLogout);
    }

    //(1) ABAP platform: Add such a line for each system with the
corresponding logout URL
    requestLogout("/sap;o=<ABAP system ID>/public/bc/icf/logoff");

    //(2) SAP HANA platform: Add such a line for each system with the system
ID from the Web Dispatcher rule
    requestHanaLogout("<HANA system ID>");
    }());
    </script>
</body>
</html>
```

Related Information

[Adding SAP Web Dispatcher Logout Rules \[page 85\]](#)
[Ensuring Complete Logout from Integrated Systems \[page 627\]](#)

1.3.2.11.2 Configuring the Automatic Sign-Out

For security reasons, a user should be signed-out automatically after a certain time of inactivity. You can set the time for automatic sign-out and define if a warning before sign-out is displayed.

When a user has not been active in the system for a specific time, they can be automatically signed out.

The launchpad offers two different possibilities to specify how the automatic sign-out should be handled. We recommend that you only set parameters for one of the solutions, not for both.

Server-Driven Sign-Out

You can use the parameter SESSION_TIMEOUT_STOP_DATA_REFRESH to work with the ABAP-server driven sign-out. (See ,Activating HTTP Security Session Management on SAP NetWeaver AS for ABAP, especially the section about http/security_session_timeout for further information). This is the default setting. With the parameter set here you determine the time a user has to be inactive before applications (e.g. dynamic tiles or notifications) stop sending polling requests for refreshing data. This is necessary because as long as the client sends requests, the server will not terminate the session. After the client has stopped sending requests

the server session-side timeout can apply. The idle time depends on your system configuration. When the ABAP server terminates the security session, the user is logged off from the local ABAP server. Depending on the system and single sign-on configuration the user is either signed-in to the launchpad automatically again via single-sign on or the sign-in screen is displayed.

Enter the time in minutes. The default value is 15 minutes. If you enter -1, you disable the timeout. Check the ABAP server timeout to see when the sign-out will be performed. If you set e.g. 15 minutes for `SESSION_TIMEOUT_STOP_DATA_REFRESH` and the server timeout is set to 30 minutes, the security session will be ended after 45 minutes of inactivity.

UI-Driven Sign-Out

You can also set parameters for a UI-driven sign-out. With this setting, the users are automatically signed-out from the launchpad, open applications in the launchpad and all SAP products that also use the same Identity Provider (depending on configuration).

A few minutes before this timeout, a warning can be displayed, informing them that they will be signed out soon. If several browser tabs with the launchpad are opened, the user activity is monitored in all tabs within the same browser.

You define the times for automatic UI-driven sign-out with two parameters:

- Use the parameter `SESSION_TIMEOUT_INTERVAL_IN_MINUTES` to enter the timeout period in minutes for the security session timeout. The maximum value is 71582 minutes. If you enter -1, you disable the automatic sign-out. This is the default setting.
- With the parameter `SESSION_TIMEOUT_REMINDER_IN_MINUTES` you can set the time when a popup window reminding the user of the automatic sign-out will be displayed. Enter the time before session timeout in minutes. The default value is 5 minutes. If you enter 0, no dialog will be displayed.

See Launchpad Configuration Parameters [page 29] for additional information.

To ensure that the activity and idle time is monitored correctly across several open browser tabs or windows you need to enable a feature in the browser settings. It is called *localStorage* or *DOMStorage* depending on the browser vendor. If the feature is disabled, logoff might be performed earlier and not synchronized between different browser tabs or windows.

An automatic sign-out might also be raised although the user is still active in the following situations:

- an app is opened in a new window and not hosted by the launchpad shell, e.g. a link to SAP SuccessFactors
- the user uses different browser instances (e.g. Chrome and Firefox) in parallel

> i Note
>
> Depending on your system configuration, you might also have defined a time for a server timeout. If users are signed out unexpectedly, you should check both settings.
>
> If you have a SAML setup and configured a single logout there, users are automatically logged off from the Identity Provider and all connected applications after the timeout.

1.3.2.12 Configuring Mapping Data for the Islamic Calendar

The mapping information for the Islamic and Gregorian calendars is calculated using a formula. You can change this default by entering dates based on the observation of the visibility of the new moon.

Prerequisites

In your ABAP system, you have the following authorization:

Authorization Object	Field	Value	Description
S_ADMI_FCD	S_ADMI_FCD	SCP1	Set character sets, languages, and character conversions.

Context

The Islamic calendar is strictly lunar. Therefore, an Islamic year of twelve lunar months does not correspond to the solar year used by most other calendar systems, including the Gregorian calendar. An Islamic year is, on average, about 354 days long, so each successive Islamic year starts about 11 days earlier in the corresponding Gregorian year. For example, the beginning of the Islamic year 1436 corresponds approximately to the Gregorian date 25 October 2014 and the beginning of the Islamic year 1437 corresponds approximately to the Gregorian date 15 October 2015. Each Islamic year has 12 months with either 29 or 30 days per month.

The mapping information for the Islamic and Gregorian calendars is stored in database table `TISLCAL`. The start date of each Islamic month is defined by its corresponding Gregorian date in this table. By default, the table contains date information which is calculated from a formula, the Tabular Islamic Calendar. The months are determined by arithmetic rules rather than by observation of the visibility of the new moon. The calculation is based on a 30-year lunar cycle where the length of the lunar months alternates between 29 and 30 days. Every two or three years, an extra day is added at the end of the year to keep up with the phase of the moon. A leap day is added 11 times in a 30-year cycle.

You can change the default entries calculated according to this algorithm by entering customized dates for table `TISLCAL`.

Procedure

1. In your ABAP system, launch the program `I18N_MAINTAIN_TISLCAL`.
2. Select the date format corresponding to the Islamic calendar which you want to maintain:

   - *Date format A* for Islamic date 1
   - *Date format B* for Islamic date 2

3. Select the range of Islamic years you want to maintain.

4. Choose *Execute*.

   The database table `TISLCAL` containing the mapping information for the Islamic and Gregorian calendars is displayed. Each row in the table contains the mapping data for one month of the Islamic year.

5. To customize the default entries of table `TISLCAL`, select a row and choose *Edit entry*.

6. In the *Edit entry* dialog box, enter a date in the *Gregorian Date (YYYYMMDD)* field. This date defines the start of the Islamic month.

   All Islamic months must have a length of either 29 or 30 days. The length is indicated in the *Days* column.

   Example:

   The Islamic year 1434 is a leap year in the Tabular Islamic Calendar, meaning that the twelfth month has 30 rather than 29 days. To adjust this, you edit the entry with Islamic date *14341201* and enter the value **20131005** in the *Gregorian Date (YYYYMMDD)* field. This defines that the twelfth month starts on 5 October 2013 rather than 6 October 2013 in the corresponding Gregorian year.

7. Choose *Continue*.

8. Choose *Check consistency* to check the consistency of the date you just entered with the date in the previous and following row.

9. Choose *Save to database*.

> **i Note**
>
> To remove the customized mapping information, select one or more rows containing this information and choose *Remove customization*.

Results

If the default is set to the date format you customized (**YYYY/MM/DD (Islamic Date 1)** or **YYYY/MM/DD (Islamic Date 2)**) in *Maintain User Profile* (transaction SU3), the customized calendar data is used for calendar display in the SAP Fiori launchpad.

1.3.2.13 Using Custom Currency Formats

The SAP Fiori launchpad uses currency definitions from the frontend server.

Custom currency formats (e.g. decimal places) are defined in the table TCURX. For consistency, custom currencies have to be maintained in the frontend server and all business backends. They are automatically uploaded to the SAP Fiori launchpad and used automatically by all SAPUI5-based apps. There is no synchronization mechanism between frontend server and business backends.

The currency formats can't be changed in the launchpad directly. If you encounter differences in currency values or errors in currency handling, you should check the table entries. The table can be maintained with transaction OY04. Read e.g. Currency Translation for more background information.

1.3.2.14 Loading SAPUI5 from a Content Delivery Network (CDN)

You can configure the launchpad to load SAPUI5 resources from a content delivery network.

In the default configuration, SAPUI5 resources are loaded from the front-end server. Loading SAPUI5 resources from a content delivery network can improve performance for resources that are not yet in the browser cache, especially if the clients and the front-end server are far apart.

To load SAPUI5 from your custom CDN, you configure the SAPUI5 bootstrapping in the Customizing of your SAP system. For more information, see the documentation for the Customizing activity *Configure SAPUI5 Bootstrapping* in Customizing under ▌ *SAP NetWeaver* ❯ *UI Technologies* ❯ *SAPUI5* ▐ (see Variant for Bootstrapping for Content Delivery Network).

1.3.2.15 Configuring In-Place Navigation for Classic UIs

Configure applications based on Web Dynpro ABAP or SAP GUI for HTML to open in the same browser window and tab.

What's the Default Behavior?

When users navigate to applications, the default behavior is as follows:

- SAPUI5 apps are opened in-place (in the same browser window and tab).
- Applications based on Web Dynpro ABAP or SAP GUI for HTML are opened in a new browser window or tab.
- URL applications are opened in a new browser window or tab.

What Can Be Configured?

You can change the behavior for applications based on Web Dynpro ABAP and SAP GUI for HTML, so these applications are opened in-place.

You can do this configuration per technology: There is a parameter for applications based on Web Dynpro ABAP, and another one for applications based on SAP GUI for HTML.

What are the Restrictions?

In the following cases, applications are always opened in a new browser window or tab, even if you have configured in-place navigation:

- When navigation to an application is triggered from outside the launchpad, for example from a link in a notification e-mail.
- In exceptional cases, application developers can specify that the target of a link in an application is always opened in a new browser window or tab. For such links, the configuration has no effect.

How to Configure In-Place Navigation for Classic UIs

1. In *SAP Reference IMG* (transaction SPRO), choose ▮▶ *SAP NetWeaver* ❭ *UI Technologies* ❭ *SAP Fiori* ❭ *Configuring SAP Fiori Launchpad* ❭ *Client-Specific Settings* ▮.
2. In the section *FLP Configuration*, add either of the following parameters, or both:
   - *NAVIGATION_GUI_INPLACE*
   - *NAVIGATION_WDA_INPLACE*
3. In the *Property Value* column, enter **true**.

Defining Exceptions to In-Place Navigation on App Level

It is also possible to define exceptions to a general in-place navigation. There are two ways to change the navigation for a specific app:

- with the sap-ushell-next-navmode intent parameter
- with the sap-ushell-navmode intent parameter

Both settings overwrite a general navigation setting.

With the sap-ushell-navmode intent parameter you define how an app is opened. Three values are available:

- `inplace`: Opens the app in the current tab. When this behavior is set in general for all apps as described above it doesn't need to be set on app level as well.
- `explace`: Opens the app in a new tab embedded in the launchpad shell.
- `frameless`: Opens the app in a new tab without the launchpad shell. Set this value for apps that only run in Microsoft Internet Explorer to ensure that they still run in the enterprise compatibility mode of Microsoft Edge.

With the `sap-ushell-navmode` intent parameter you define how navigation within an app is done. In exceptional cases, you can use the `sap-ushell-next-navmode` intent parameter to specify in the target mapping that the target of a link in an application is always opened in a new browser window or tab. This parameter has two allowed values: `inplace` and `explace`.

You can use the parameter to overrule the centrally configured navigation mode for a specific application to ensure that the UI state is kept. If the parameter is set to `explace` in the target mapping of an app, every navigation triggered from this specific app opens ex-place instead of in-place as configured centrally for the launchpad.

> **! Restriction**
>
> The parameter is only intended for custom apps or custom target mappings of products that fulfill specific requirements. For a list of technologies for which the parameter can be used, see SAP note 2677034 .

1.3.2.16 Configuring SAP CoPilot for the Launchpad

SAP CoPilot is a plugin-loaded service operated from the cloud using the SAP Cloud Platform. SAP CoPilot can be integrated into the launchpad for both SAP Fiori on-premise deployment models: Gateway-Hub deployment and Embedded deployment.

For information on how to set up each deployment scenario (with varying options for user authentication and single sign-on), refer to the *SAP CoPilot Configuration Guide* on SAP Help Portal at https://help.sap.com/viewer/p/SAP_COPILOT.

Related Information

Working with the Digital Assistant [page 441]

1.3.2.17 Configuring SAP Mobile Cards for the Launchpad

SAP Mobile Cards is an application that allows you to share content from SAP Fiori-based applications from your desktop computer's SAP Fiori launchpad to your iPhone, which provides access to the most up-to-date business data available. SAP Mobile Cards is now integrated into the Launchpad for both SAP Fiori on-premise deployment models: Gateway-Hub deployment and Embedded deployment.

For information on how to deploy, refer to Fiori Card Settings.

Related Information

Working With SAP Mobile Cards [page 441]

1.3.2.18 Configuring SAP Web Analytics Plugin for the Launchpad

Create and configure the SAP Web Analytics plugin to track the usage of SAP Fiori launchpad.

SAP Web Analytics is a SaaS application on the SAP BTP. It lets you collect, report, and analyze the usage data of your application that helps you to identify meaningful patterns from various digital channels. SAP Web

Analytics can offer insights on how well your application performs in key channels. These insights can lead you to implement critical improvements that can help you optimize the usage to measure goals, drive strategy, and improve the overall usability of your applications.

SAP Web Analytics provides a plugin for SAP Fiori launchpad to track its usage. To enable tracking, SAP Fiori launchpad's customer must add the plugin to their launchpad. The plugin bootstraps SAP Web Analytics tracker every time the launchpad is loaded. For information on how to create and configure the plugin, refer to the SAP Web Analytics help documentation.

1.3.2.19 Configuring the Option to Contact Support

You can allow end users to directly create a support ticket from the launchpad.

If an error occurs when using the SAP Fiori launchpad, you can provide end users with the option to report the error from inside SAP Fiori launchpad using the *Contact Support* option (see Contacting Support [page 442]).

> → Remember
>
> To enable the *Contact Support* option in the user actions menu, set the *SUPPORTTICKET* parameter to `true`. See Launchpad Configuration Parameters [page 29].

You can offer several ticket processing systems to the user:

Ticket System	Description
SAP Solution Manager	The SAP Solution Manager is integrated in the SAP Fiori launchpad and enabled by default.
	See Configuring the Option to Contact Support Via SAP Solution Manager [page 96]
Custom ticket system	You can integrate a custom ticket system in the SAP Fiori launchpad by creating your own BAdI implementation.
	See Configuring the Option to Contact Support Via a Custom Ticket System [page 97].

Data Transmitted

When a user chooses the *Contact Support* option from the launchpad and enters a problem description in the dialog box, relevant technical data is automatically appended to the message. The following data is transmitted to the ticket system:

Data	Description
Subject	(Optional)
	Message title
	If no value is given, the first message text line is used.
Component	Message component
	The system tries to automatically retrieve the component based on hash, URL of the current application, and catalog ID.
	See Determining the Support Message Component [page 101].
Text	Message text
Priority	Priority of the message
	The value is automatically set to 3 (= medium).
Client context	Information from the UI (for example, language or entries from the client console log)
Log ID	Error ID that is used when an error log is created
URL	URL of the application from which the *Contact Support* function was called
Catalog ID	ID of the catalog that includes the tile from which the *Contact Support* function was called
Hash	Hash of the semantic object and action (navigation intent) that were used to navigate to the application from which the *Contact Support* function was called.

1.3.2.19.1 Configuring the Option to Contact Support Via SAP Solution Manager

You need to make configuration settings before an end user can contact support from the launchpad by creating an incident in SAP Solution Manager.

These configuration settings are mandatory to be able to use this option.

Make the following configuration settings:

- Create an RFC connection to the Solution Manager system using *RFC Destinations (Display/Maintain)* (transaction SM59) and enter the following data:
  - *RFC Destination*: Name for the RFC connection (for example, SM_INC_E2E)
  - *Connection Type*: 3
  - *Host*: Host name (for example, hs0311)
  - *System*: System number (for example, 00)
- Maintain the following values for table BCOS_CUST using *Call View Maintenance* (transaction SM30):
  - *Appl.*: OSS_MSG
  - *Targ. Type*: W
  - *RFC Connection*: Enter the name of the RFC connection you provided in transaction SM59.
  - *Client*: CUST620
  - *Version*: 1.0

In addition, you have to make configuration settings in SAP Solution Manager. For more information, see SAP Solution Manager Configuration.

Related Information

SAP Solution Manager Configuration
Contacting Support [page 442]

1.3.2.19.2 Configuring the Option to Contact Support Via a Custom Ticket System

You can integrate a custom ticket processing system into the SAP Fiori launchpad.

Context

The default message channel for the launchpad is the SAP Solution Manager. You can, however, specify a custom ticket processing system as the support channel.

Procedure

1. Create a Business Add-In (BAdI) implementation for the BAdI /UI2/BADI_EMB_SUPP.
   a. Implement the following BAdI interface methods:

Method	Description
CREATE_INCIDENT	Creates a message in the specified ticket processing system.
	You need to implement the relevant interface parameters.
	See Parameters of Interface /UI2/IF_EMB_SUPP [page 99].
IS_REPORTING_ENABLED	Checks if the specified support channel is active.
	i Note This check will be deprecated at some point in the future. We recommend you to only implement the following line: `rv_enabled = abap_true`

b. In the BAdI implementation, specify a filter value for the filter SCENARIO_ID .

> i Note
>
> It is only possible to configure one ticket processing system to be called at runtime. The filter allows you to switch between different BAdI implementations.

2. Register your BAdI implementation to be called at runtime.

   a. Call program /UI2/MAIN_SUPP_CON.
   b. Select *Register for client*.
   c. In the *BAdI Filter value* field, enter the filter value for SCENARIO_ID, which you have specified in step 1.b.
   d. Execute the program.

> i Note
>
> If you want to deregister a BAdI implementation, select *Delete for client* and execute the program. In this case, the default connection to Solution Manager is used, provided the connection to the Solution Manager is enabled. If it is not enabled, the option to contact support will not be available at all in the launchpad.

3. Repeat step 2 for every client in your system in which you want to use your support channel.
4. If you want to offer the feature in additional systems, repeat the above procedure.

Related Information

How to Implement a BAdI
Configuring the Option to Contact Support [page 95]

1.3.2.19.2.1 Parameters of Interface /UI2/IF_EMB_SUPP

When you want to integrate a custom ticket system into the SAP Fiori launchpad, you need to implement parameters of the interface /UI2/IF_EMB_SUPP.

To configure the option to contact support via a custom ticket system, you need to implement method CREATE_INCIDENT of the BAdI /UI2/BADI_EMB_SUPP.

Method CREATE_INCIDENT of interface /UI2/IF_EMB_SUPP has the following parameters:

Importing Parameters

Parameter Name	Type	Description
CATEGORY	CHAR12	(Optional) Default value: SPACE
LANGUAGE	SYLANGU	Logon language of the user who has created the ticket
PRIORITY	CHAR1	(Optional) Priority of the message Default value: 4
PROCESSOR	SYUNAME	(Optional) The user name of the ticket processor Default value: SPACE
REPORTER	SYUNAME	The user name of the user who has created the ticket
SEND_2_SAP_IM	BCOS_TYPES-FLAG	This parameter is obsolete. Default value: SPACE
SUBJECT	TEXT60	Title of the message with max. 60 characters
TWB_CALL	BCOS_TYPES-FLAG	This parameter is obsolete. Default value: SPACE
TYPE_NOTIF	CHAR6	(Optional) Processing type of the message Default value: SLF1
TI_ATTRIBUTES	TY_GT_BCOS_ATTRI	Name value pairs of technical attributes. For example: • SAP application component • Installation number of the ABAP backend system • System ID of the ABAP backend system • Client number

Parameter Name	Type	Description
TI_TEXT_HEADER	TY_GT_BCOS_TEXTH	Text header information
TI_TEXT_LINES	TY_GT_BCOS_TEXTL	Message text
TI_APPX_HEAD	TY_GT_BCOS_APPX	(Optional) Header information of attached files (appendix)
TI_APPX_LINES	TY_GT_SOLI	(Optional) Text of attached files
TI_APPX_LINES_BIN	TY_GT_SOLIX	(Optional) Binary data of attached files
IV_EMAIL_ADDRESS	CHAR255	(Optional) E-mail address of the user who has created the message

Exporting Parameters

Parameter Name	Type	Description
MSG_KIND	BCOS_CUST-OSS_OR_SM	Type of the message target system
MSG_KEY	BCOS_TYPES-MSG_KEY	Message key If the ticket was successfully created, a message key should be passed. If this parameter is not filled, the system assumes that the message creation has failed.
MSG_DEST	BCOS_CUST-DESTINAT	Logical destination
R_CODE	SY-SUBRC	Return value of ABAP statements If R_CODE <> 0, the system assumes that the message was not created.
SAP_KEY	BCOS_TYPES-MSG_KEY	
TE_RETURN	TY_GT_BAPIRET1	Return parameter If the ticket was not created successfully, you can pass a detailed error message text, which is also sent to the SAP Gateway.
EV_EXCEPTION	SYSUBRC	Return value of ABAP statements If EV_EXCEPTION <> 0, the system assumes that the message was not created.

Related Information

1.3.2.19.3 Determining the Support Message Component

If an error occurs when using the SAP Fiori launchpad, users can report the error from inside SAP Fiori launchpad using the *Contact Support* option (if the option has been enabled by the administrator). The system proposes the relevant message component (category) in the back-end system.

When the error occurs, the system automatically determines the relevant message component based on the current application used. This can be the BSP, SAPUI5 application, SAP GUI for HTML, and Web Dynpro ABAP.

If this is not successful, the message component is determined based on the related `LPD_CUST` launchpad.

If this is not successful, the message component is determined based on the catalog ID for the catalog in which the `LPD_CUST` launchpad is used.

If the message component is not determined successfully, the component remains empty.

1.3.3 Launching the Launchpad

In this section you can find the default URLs for launching the SAP Fiori launchpad.

After you set up the launchpad, you can launch it by entering any of the following URLs:

- `https://<server>:<port>/sap/bc/ui2/flp/`
- `https://<server>:<port>/sap/bc/ui2/flp/index.html`
- `https://<server>:<port>/sap/bc/ui2/flp/FioriLaunchpad.html`

If you prefer to use a different URL, you can customize the launchpad URL.

Related Information

1.3.4 Setting Up Launchpad Content

This section provides an overview of how you can consume and, if necessary, adapt launchpad content delivered by SAP or integrate custom content in the launchpad.

The following graphic visualizes how the setup of launchpad content fits into the SAP Fiori Rapid Activation end-to-end process:
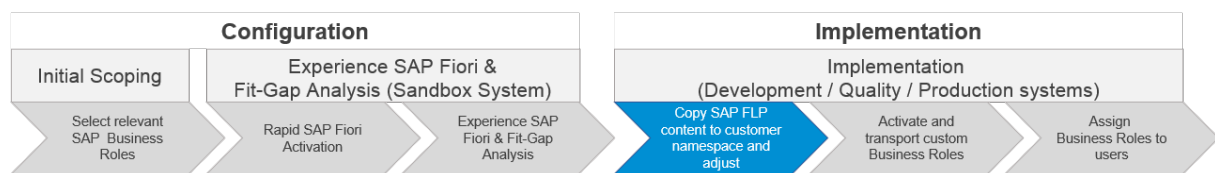
- Typical Scenarios for Managing Launchpad Content [page 120]

For more information on the SAP Fiori Rapid Activation end-to-end process, see Set up and Configure SAP Fiori and Implement SAP Fiori Apps.

About Launchpad Content [page 103]
> This section explains how the different entities required to set up launchpad content relate to one another.

Scopes for Adapting Launchpad Content [page 111]
> As an administrator you can adapt the launchpad content delivered by SAP for the roles within your company.

Tools for Setting Up Launchpad Content [page 112]
> Get an overview of the most important tools required to set up launchpad content.

Best Practices and Typical Scenarios for Setting Up Launchpad Content [page 114]
> This section provides recommendations for consuming and adapting launchpad content.

Setting Up Navigation [page 127]
> You have to perform several steps to add applications as navigation targets to the launchpad.

Setting Up Technical Catalogs [page 132]
> Technical catalogs contain original tiles and target mappings.

Setting Up Business Catalogs [page 238]
> Business catalogs contain a collection of tiles and target mappings relevant for a business role.

Setting Up Launchpad Layout and Structure [page 268]
> There are two options for structuring apps in the launchpad.

Integrating Remote Content [page 308]
> For applications located in the backend (for example, applications based on Web Dynpro ABAP or SAP GUI), you need to define remote function call (RFC) destinations from the ABAP front-end server to the ABAP back-end system(s) to enable integration in the launchpad.

1.3.4.1 About Launchpad Content

This section explains how the different entities required to set up launchpad content relate to one another.

To give launchpad users access to apps, administrators have to maintain several entities. The graphic below provides an overview of the involved entities. The arrows indicate the direction in which these entities are assigned.
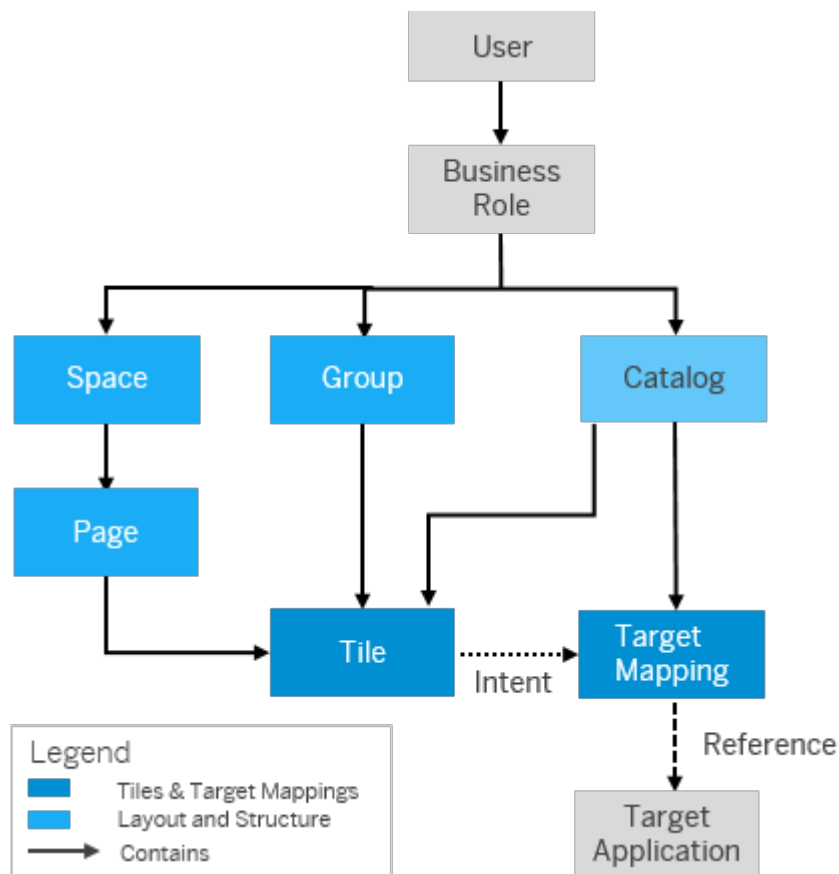
Entity	Description	Related Administration Tasks
Role	Authorizations to run the launchpad and to access launchpad content are assigned to users based on roles.	Assign catalogs and groups to a role which can be assigned to users. See Setting Up Authorization Roles [page 327]
Space	A space is the unit that holds a page. Users may see several spaces in their launchpad. The spaces are displayed in the navigation bar where users switch between the different available spaces.	See Managing Launchpad Spaces and Pages [page 269].

Entity	Description	Related Administration Tasks
Page	A page is the part of the space that contains the apps clustered into different sections. It is shown in the main area of the launchpad.	See Managing Launchpad Spaces and Pages [page 269].
Group	Groups define the grouping, sort order, and general appearance (tile or link) of apps that are initially displayed on the launchpad home page. They consist of tiles (and links) representing a subset of apps. Users that have this role assigned, can view the group and the contained apps on their launchpad homepage. Users can personalize their launchpad by adding or removing apps from the out-of-the-box groups or self-defined groups.	Assign tiles to a group and groups to a role. See Managing Groups [page 304].
Catalog	A collection of apps that are available for a specific role (or for multiple roles). Users that have a particular role assigned, have access to all the apps that are contained in the catalogs assigned to this role. Catalogs are the smallest entities that define the set of apps you want to assign to your users for selection and authorization.	Assign apps (via tiles and target mappings) to a catalog and catalogs to roles.
Tile	A visual representation of an app on the launchpad home page. When a user clicks a tile in their launchpad, an intent (including optional parameters) navigates to the app and opens it.	How you maintain tiles and target mappings depends on the design-time tool: • Launchpad app manager: Allows you to maintain original tiles and target mappings. Tiles and target mappings are maintained in one entity, the launchpad app descriptor item. See Launchpad App Descriptor Items [page 107]. • Launchpad designer: Allows you to maintain original tiles and target mappings as well as references to them.

| Entity	Description	Related Administration Tasks
Target mapping	Mapping of a navigation target to the intent (combination of semantic object and action). The target mapping is a prerequisite for the navigation to an app in the launchpad. It can refer to only one target application which can be built with different UI technologies (such as SAPUI5, Web Dynpro, SAP GUI for HTML). See Intent-Based Navigation [page 108].	Tiles and target mappings are maintained as individual entities. • Launchpad content manager: Allows you to maintain references to tiles and target mappings. Tiles and target mappings are matched to combinations based on their intent. See Tile/Target Mapping Combinations [page 243].

Related Information

Tools for Setting Up Launchpad Content [page 112]
Best Practices and Typical Scenarios for Setting Up Launchpad Content [page 114]

1.3.4.1.1   Tiles

A tile is the visual representation of an app.

The tile definition is used to define what the tile looks like on the launchpad (home) page or in the app finder.

A tile definition consists of texts, icons, and may also include dynamic information such as a count or aggregated figure (e.g. % budget used). Tiles can also be displayed as links. Links can be useful if users need direct access to the app but it is not necessaryto display additional information to the app in a tile.

Tile Types

Tile Type	Description
Static tile	Displays static text and icons that can be configured.

Tile Type	Description
Dynamic tile	Displays text and icons as well as dynamic content read from a data source. The information that is displayed on a dynamic tile can be pulled from the back-end using an OData service. The dynamic information pulled from the backend overwrites the configuration values given in the tile configuration.
Custom tile	Additional tile type provided by SAP (e.g. SAP Smart Business) or tile type defined by customers.

See the following figure for a schematic diplay of the different tiles.
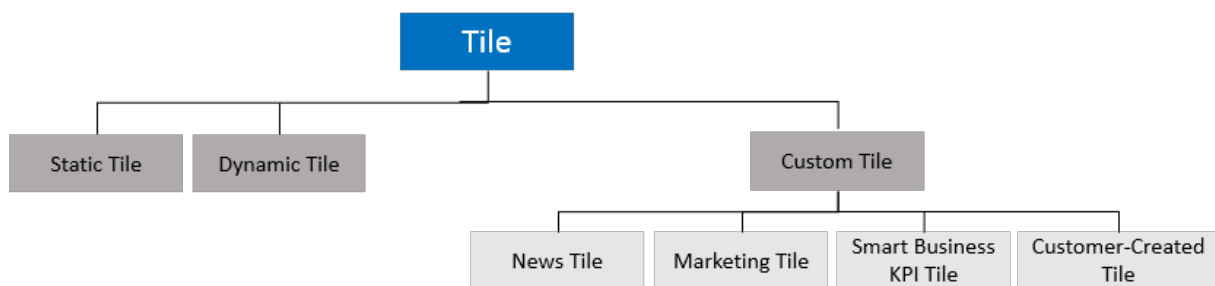
Overview of Tiles

Related Information

1.3.4.1.2 Target Mappings

A target mapping maps a navigation target to the intent (combination of semantic object and action). Thus, it is a prerequisite for the navigation to a target application in the launchpad.

It can refer to only one target application which can be built with different UI technologies (such as SAPUI5, Web Dynpro, SAP GUI for HTML). See Intent-Based Navigation [page 108].

1.3.4.1.3   Launchpad App Descriptor Items

This section provides information about launchpad app descriptor items and tiles.

What is a Launchpad App Descriptor Item?

A launchpad app descriptor item is an entity that consists of a target mapping and one or more tiles.
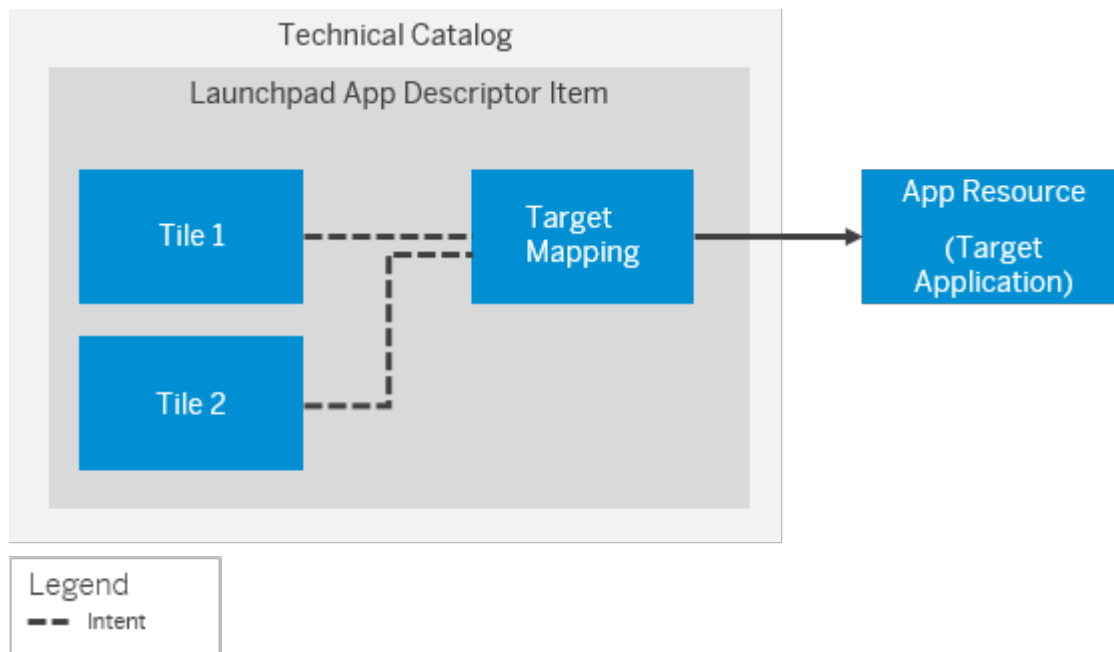
It is possible to only add target mappings if you don't want to launch the app from a tile but from another application (app-to-app navigation). If you want to launch an application of type URL, you can add tiles only.

A launchpad app descriptor item is always assigned to a catalog.

Related Information

Setting Up Technical Catalogs with the SAP Fiori Launchpad App Manager [page 132]
Maintaining Launchpad App Descriptor Items [page 148]

1.3.4.1.4 Intent-Based Navigation

The SAP Fiori launchpad enables you to specify that the same application is launched in different views or modes depending on the end user's role. To allow this, navigation between launchpad applications is based on abstract representations (intents) that are resolved to concrete navigation targets.

Each application within the launchpad has a resource locator (URL) by which it can be loaded. Instead of directly encoding the (technical) name of the target application into the URL hash, the launchpad performs an indirection by so-called intents.

An **intent** is a mechanism that allows users to perform actions on semantic objects (such as navigating to a sales order or displaying a fact sheet), without worrying about the UI technology or technical implementation of the navigation target.

Intent-based navigation is necessary in the following cases:

- Depending on the user's role, a different application or view of an application should be displayed.
- To extend and customize Fiori scenarios, you can change a target without modifying the Fiori application code.
- Different Fiori apps have different life cycles. An app cannot assume that another app exists in a productive environment since it might not have been deployed.
- You would like to use a legacy application and later switch to an SAPUI5 app in a way that is transparent for users.

Intents are comprised of the following parts:

Semantic object	Represents a business entity such as a customer, a sales order, or a product. Using semantic objects, you can bundle applications that reflect a specific scenario. They allow you to refer to objects in a standardized way, abstracting from concrete implementations of these objects. You can either use semantic objects shipped by SAP, or create new semantic objects. See Defining Semantic Objects for Navigation [page 128].
Action	Describes which operation (such as display or approvePurchaseOrders) is intended to be performed on a semantic object (such as Purchase Order or Product).
Semantic object parameters	Define the instance of the semantic object (e.g. by specifying the employee ID).

Intents have the following pattern:

`#<semantic object>-<action>?<semantic object parameter>=<value1>`

> ⁘ Example
>
> The intent `#SalesOrder-displayFactSheet?SalesOrder=27` specifies that a fact sheet for the sales order no.27 is launched.

The intent-based fragment identifier `#SalesOrder-displayFactSheet?SalesOrder=27` might be resolved to the following URL: `https://<server>:<port>/sap/bc/ui5_ui5/ShowEmployeeSalesOrders.html?id=27`

However, the URL of the launchpad hides the actual URL under an intent-based URL.

> ⁂ Example
>
> ```
> https://<server>:<port>/sap/bc/ui5_ui5/ui2/ushell/shells/abap/
> FioriLaunchpad.html#SalesOrder-displayFactSheet?SalesOrder=27
> ```

The same intent can be resolved differently based on the role of the user that triggers the navigation. For example, you may want to define that a manager can display a different version of an employee fact sheet than regular employees. Thus, for managers the intent `#factsheet-display` should resolve to view1 of your application, while for regular employees it should resolve to view2.

The intent is resolved to the concrete navigation target by the launchpad target resolution service. The concrete application targets have to be configured by the administrator. In this configuration (called target mapping), admins map the combination of a semantic object and an action (both defined in the app launcher tile) to the navigation target by specifying launchpad role and instance as well as application alias or ID. This allows any link specifying an intent-based URL to trigger the correct application assigned to the user. Since target mappings are assigned to users as part of a catalog, they can be assigned to authorization roles, while an intent is independent of a role and can therefore be resolved differently based on the role of the user that triggers the navigation.

Navigation Flow

1. You as an administrator have set up navigation. See Setting Up Navigation [page 127].
2. The user logs on to the SAP Fiori launchpad and sees the assigned tiles in the home page.
3. When the user selects a tile, the system checks if an intent with the semantic object/action pair exists in the `PFCG` permissions of the user.
4. If yes, the system resolves the navigation target using the available configuration information and navigates the user to the target application.

Related Information

Intent-based Navigation [page 525]

1.3.4.1.5 Transactions

The transaction represents an application in the SAP Fiori launchpad.

Transactions are available for the following app types:

- SAPUI5 Fiori apps
- SAP GUI apps
- URL apps

- Web Dynpro applications
- WebClientUI applications
- Tile Only

A transaction is an entity in the SAP system that represents an application or an executing program. Each transaction consists of a transaction code, a technical ID, that can be used to call or identify a transaction in the SAP system.

A transaction code can contain up to 20 characters and should always begin with a letter. Permitted characters are letters from A to Z, numbers from 0 to 9, and the underscore. For SAPUI5 Fiori apps, the transaction code is the same as the SAP Fiori ID, for example **F1234**.
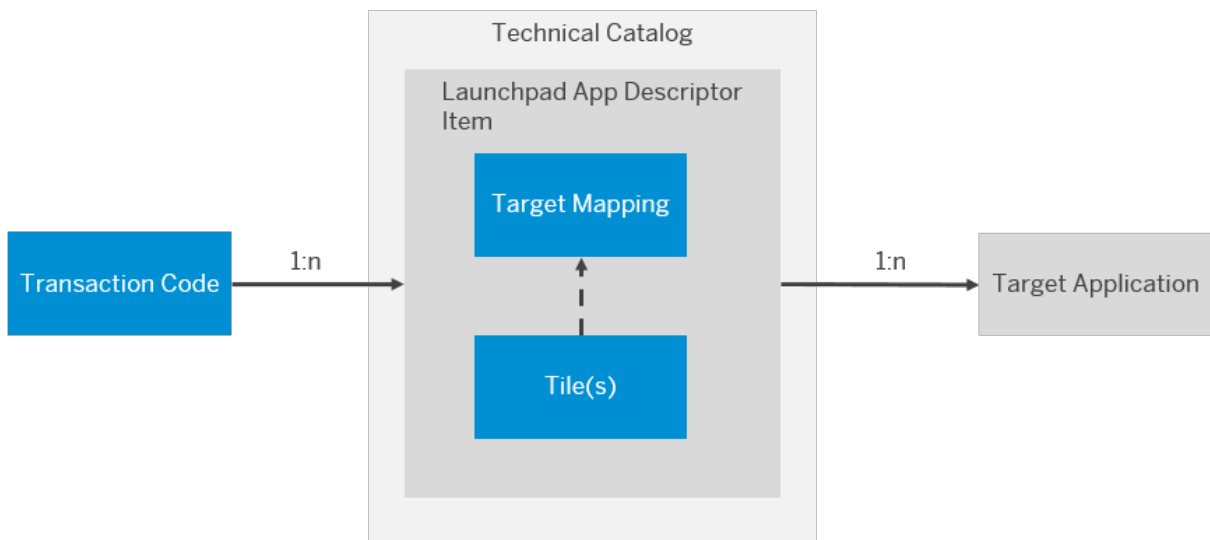
Use Cases

There are two use cases where transactions for SAPUI5 applications are used:

- **Upgrade Support** - Transactions and their status allow you to find out if an app is still in use or if it needs to be replaced. They are assigned to launchpad app descriptor items belonging to an SAP Fiori application. SAP assigns the transaction in the launchpad app manager and maintains the status in *Maintain Authorization Default Values* (transaction SU22). See Adjusting Launchpad Content after an Upgrade [page 266].

  > **i Note**
  >
  > Upgrade support with transaction codes is only supported in an embedded scenario.

- **PFCG Integration** - In *Role Maintenance* (transaction PFCG) you are able to see which transaction codes, hence which SAP Fiori applications, belong to a relevant role. Each transaction code represents an SAP Fiori app. Before that, you could only see the assigned services. See Assign Business Catalogs to Roles [page 330].

Related Information

Launchpad App Descriptor Items [page 107]

1.3.4.2 Scopes for Adapting Launchpad Content

As an administrator you can adapt the launchpad content delivered by SAP for the roles within your company.

Scopes for Launchpad Content

SAP delivers predefined launchpad content like catalogs, spaces and groups. As an administrator you can reuse this content and adapt it to your needs or you can create new content from scratch.

Launchpad content is maintained in the following scopes:

Scope	Description
Configuration	System-wide settings The content delivered by SAP is stored in the configuration scope. We recommend you to maintain your custom technical catalogs in the configuration scope. This allows you to reuse the tiles and target mappings created in the technical catalogs in all clients. i Note SAP technical catalogs can be adapted in the launchpad app manager using the adaptation mode. See Adapting Launchpad App Descriptor Items [page 165].
Customizing	Client-specific settings We recommend you to maintain your business catalogs in the customizing scope. This allows you to manage business catalogs for a selected client and to assign these catalogs to roles. Customizing settings supersede configuration settings, but can be superseded by personalization settings.
Personalization	User-specific settings At launchpad runtime, end users can use the personalization scope to adapt a page or group that contains predefined content and that has been assigned to their user role by an administrator. For example, they can rearrange the tiles in the SAP Fiori launchpad and save their preferences in the personalization layer. Personalization settings supersede customizing and configuration settings.

If the content exists in more than one scope and you change it in a more specific scope (for example, personalization), the changes supersede the content stored in a more generic scope (for example, configuration).

For example, if you make changes to a catalog in the content delivered by SAP and stored in the configuration scope after you personalized the content, these changes are not visible in the personalization scope.

If you delete the whole group or catalog in the personalization scope or in the customizing scope, the system displays the original data of the less specific scope.

If you delete a catalog or group in the customizing scope, the launchpad designer tries to load the object from the layer of the configuration scope. If this is successful, a message will let you know that the object has been reloaded from a lower layer.

Selecting the Scope for a Design-Time Tool

Here is how you access the available design-time tools in the configuration or customizing scope:

- Launchpad app manager
  This tool is always run in the configuration scope.
  See Running the Launchpad App Manager [page 139].
- Launchpad content manager
  This tool can be run in the configuration or in the customizing scope.
  See Running the Launchpad Content Manager [page 240].
- Launchpad designer
  This tool can be run in the configuration or in the customizing scope.
  See Running the Launchpad Designer [page 188].

Related Information

Best Practices and Typical Scenarios for Setting Up Launchpad Content [page 114]
Tools for Setting Up Launchpad Content [page 112]

1.3.4.3   Tools for Setting Up Launchpad Content

Get an overview of the most important tools required to set up launchpad content.

Tool	Major Use Case	More Information
SAP Fiori apps reference library	Explore launchpad content and get key information for each app, including all the technical data you need for installation and configuration.	SAP Fiori Apps Reference Library
Launchpad app manager	Manage technical catalogs with original tiles and target mappings

Adapt launchpad app descriptor items in SAP technical catalogs | Setting Up Technical Catalogs with the SAP Fiori Launchpad App Manager [page 132]Adapting Launchpad App Descriptor Items [page 165] |
| Launchpad content manager | Manage business catalogs with references to tiles and target mappings in technical catalogs | Setting Up Business Catalogs with the Launchpad Content Manager [page 238] |

Tool	Major Use Case	More Information
Launchpad designer	• Manage launchpad structure and layout using groups • Manage custom tiles • Update existing technical catalogs created with the launchpad designer.	Updating Content Created with the Launchpad Designer [page 182]
"Manage Launchpad Spaces" and "Manage Launchpad Pages" app	Manage launchpad structure and layout using spaces and pages	Manage Launchpad Spaces [page 291] Manage Launchpad Pages [page 278]
Role maintenance (transaction PFCG)	Manage authorization roles	Configuring Roles for Catalogs, Spaces and Groups [page 328]

Tool Strategy

- **Launchpad app manager**
  The launchpad app manager is provided for creating technical catalogs. The tool introduces the strategy to represent apps as own ABAP objects, called launchpad app descriptor items, which consist of a target mapping and one or multiple tiles.
  The launchpad app manager has the following advantages over the launchpad designer:
  - SAP content is fully migrated to launchpad app descriptor items, which can be transported independent of the catalog
  - Simplified translation process
  - Improved search function
  - Improved usability

- **Launchpad content manager**
  The launchpad content manager allows a comfortable set up of business catalogs with enhanced search and copy functionalities. It can be used with existing SAP and customer business catalogs without the need to migrate any content.

- **Launchpad designer**
  In previous releases, the launchpad designer was used for creating technical catalogs, business catalogs and business groups. Its recommended usage is now limited to managing existing technical catalogs, managing custom tiles in business catalogs and managing business groups. The management of new technical catalogs and business catalogs was shifted towards the launchpad app manager and the launchpad content manager.

- For structuring the launchpad content, spaces and pages have been introduced, succeeding the home page structuring approach based on business groups.
  The spaces mode offers more flexibility to influence the launchpad layout.
  The set up of spaces and pages is done with the apps *Manage Launchpad Spaces* and *Manage Launchpad Pages*. Major user interface enhancements will be based on the spaces and pages approach. Business groups can still be used in parallel to spaces and pages or as the only structuring approach.

The following graphic visualizes the tool strategy:
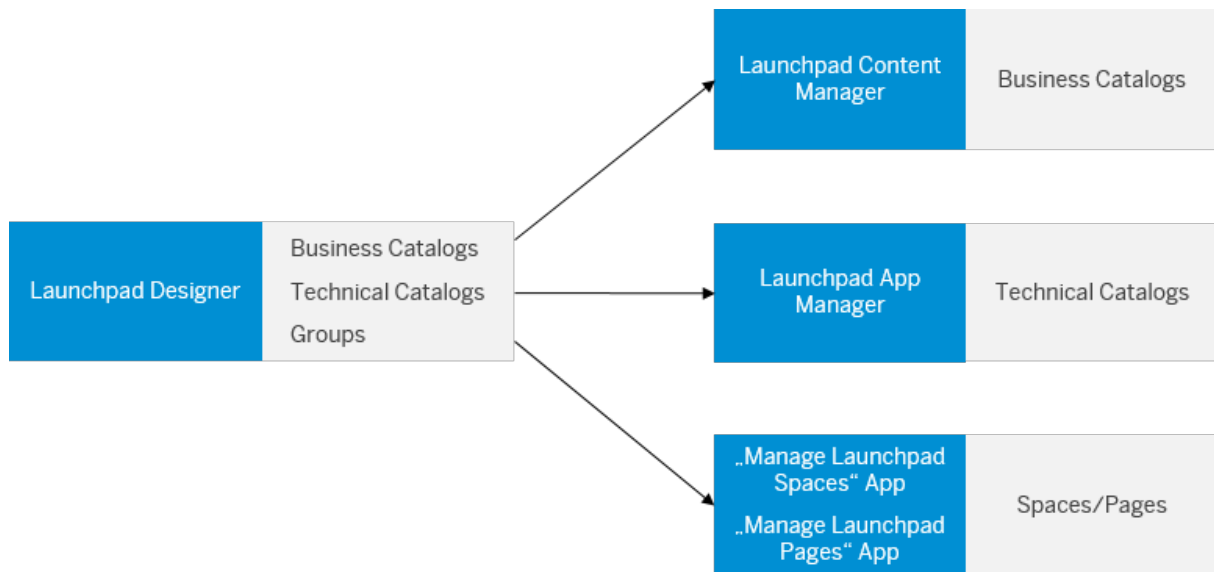
Related Information

Best Practices and Typical Scenarios for Setting Up Launchpad Content [page 114]
About Launchpad Content [page 103]

1.3.4.4 Best Practices and Typical Scenarios for Setting Up Launchpad Content

This section provides recommendations for consuming and adapting launchpad content.

Best Practices for Managing Catalogs [page 115]

Best Practices for Setting Up Launchpad Layout and Structure [page 117]
This section provides recommendations on how to set up the structure and layout for your launchpad content.

Typical Scenarios for Managing Launchpad Content [page 120]
This section contains best practice scenarios on how to use and adapt launchpad content to meet the needs of your company.

Related Information

Best Practices for Setting Up Launchpad Layout and Structure [page 117]
Best Practices for Managing Groups [page 119]
About Launchpad Content [page 103]
Tools for Setting Up Launchpad Content [page 112]

1.3.4.4.1 Best Practices for Managing Catalogs

Overview

The following graphic shows a typical content setup:
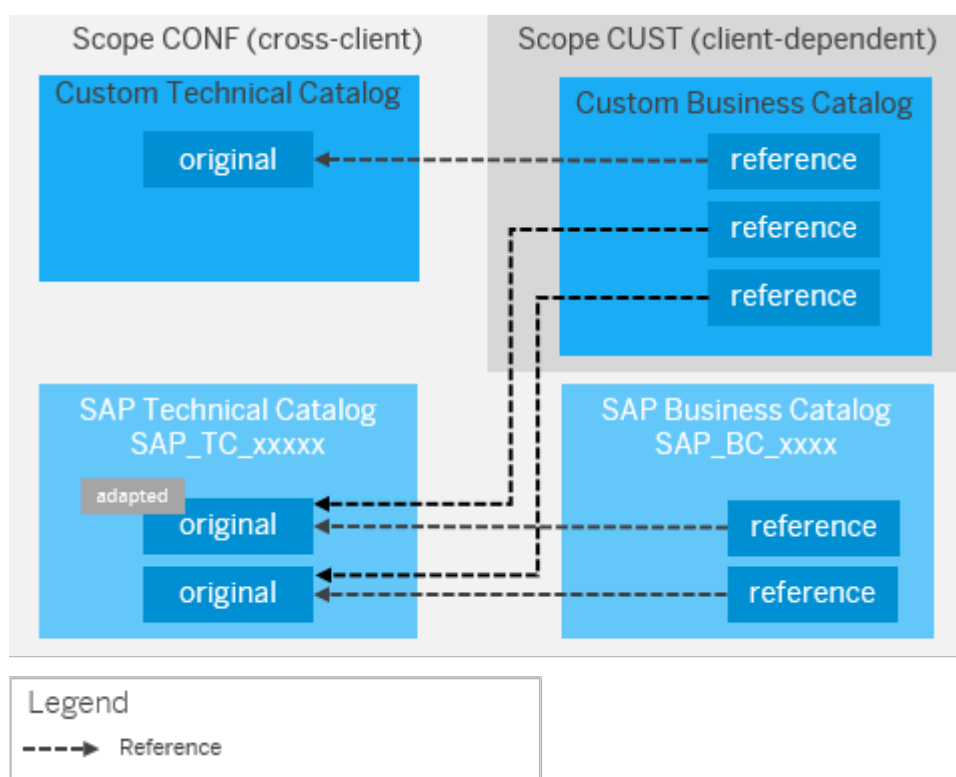
SAP Catalogs

SAP-delivered catalogs are structured as follows:

- SAP technical catalogs (naming convention SAP_TC_<...>)
  Technical catalogs contain original tiles and target mappings relevant for apps per application area.
  Technical catalogs act as a repository of tiles and target mappings.
- SAP business catalogs (naming convention SAP_BC_<...>)
  Business catalogs contain references to the tiles and target mappings defined in technical catalogs. The
  content of a business catalog is a subset of the content of a technical catalog. This subset reflects
  the requirements of a specific business user. Business catalogs are assigned to business roles. These

references exist only on tile/target mapping level; there are no references on catalog level. The references ensure that any changes to the original tile or target mapping can be made centrally in the technical catalogs and are propagated to the business catalogs.

SAP catalogs are always located in the configuration scope (cross-client).

Custom Catalogs

We recommend to structure your custom catalogs using the launchpad app manager and the launchpad content manager:

- Custom technical catalogs
  Custom technical catalogs are created in the configuration scope (client-independent). In your custom business catalogs, you can add references to the tiles and target mappings of your custom technical catalogs.
- Custom business catalogs
  Custom business catalogs are copied from SAP business catalogs and are used as a starting point. The copied catalog contains references to the tiles and target mappings in the SAP technical catalog. The content of the copied catalog can then be adjusted by removing or adding references to the tiles and target mappings in SAP technical catalogs.

Recommendations for Custom Technical Catalogs

- If you want to make changes to SAP-delivered launchpad content, there are the following options:
  - Adapt launchpad app descriptor items directly in SAP technical catalogs. Certain fields, such as the title or the description, can be adapted in the adaptation mode of the launchpad app manager. See Adapting Launchpad App Descriptor Items [page 165].
  - If your desired changes can't be made in the adaptation mode, copy the launchpad app descriptor item to your custom technical catalog and make necessary changes. See Advanced Scenario - Adapt SAP Template Content and Add Your Own Content [page 123].
- Create and change your custom technical catalog in the configuration scope. This allows you to reuse the tiles and target mappings created in the technical catalogs in all clients.

Recommendations for Custom Business Catalogs

- Minimize the effort in creating your custom business catalogs by copying the SAP business catalogs as a starting point. This will ensure to get a consistent set of apps that supports app-to-app navigation by including the relevant target mappings. This will not necessarily be the case when you select only a set of single apps from the technical catalogs.
- Create and change your custom business catalog in the customizing scope (client-dependent). This allows you to manage business catalogs for a selected client and to assign these catalogs to roles.
- Do not change custom business catalogs that were created in the configuration scope while you work in the customizing scope since this could lead to outdated catalogs.
  See Scopes for Adapting Launchpad Content [page 111]
- The performance of operations on catalog level (e.g. copying a catalog) depends on the size of the catalog. It is recommended to keep the number of tiles/target mappings per catalog as low as possible.
- Some apps (mainly SAP GUI for HTML apps) are not contained in any business catalog and are only delivered as part of technical catalogs. In these cases, you need to select a single app (tile/target mapping) from the technical catalog and add it as a reference to your custom business catalog.

For more information, see Typical Scenarios for Managing Launchpad Content [page 120].

Related Information

About Launchpad Content [page 103]
Scopes for Adapting Launchpad Content [page 111]
Tools for Setting Up Launchpad Content [page 112]

1.3.4.4.2   Best Practices for Setting Up Launchpad Layout and Structure

This section provides recommendations on how to set up the structure and layout for your launchpad content.

The recommended approach for defining launchpad layout and structure is using spaces and pages. See Best Practices for Managing Spaces and Pages [page 117] for tips and tricks and Managing Launchpad Spaces and Pages [page 269] for the detailed concept description.

If you work with the classic groups and home page, read Best Practices for Managing Groups [page 119] for tips and tricks and Managing Groups [page 304] for the detailed concept description.

Best Practices for Managing Spaces and Pages [page 117]
Read this for tips on creating and handling spaces and pages.

Best Practices for Managing Groups [page 119]
Groups define the grouping, sort order, and general appearance (tile or link) of apps that are initially displayed on the SAP Fiori launchpad home page for a user.

1.3.4.4.2.1 Best Practices for Managing Spaces and Pages

Read this for tips on creating and handling spaces and pages.

Context

SAP delivers predefined spaces and pages for selected business roles. You can use them as model to explore the new layout concept. Also see Working with Predefined Spaces and Pages [page 275].

Recommendations

When switching to spaces or creating your own spaces, we recommend adhering to the following guidelines:

Consider using SAP-delivered spaces and pages:

Structuring recommendations when creating your own spaces and pages:

- Keep it **simple and lean** in the beginning: Create one space per business role. If this is not suitable for the way your roles are designed, you can also think about either assigning one space to more roles or assigning several spaces to one role.
- **Less is more**: Aim for lean pages. Only add those apps to a page that the users work with regularly. If you allow users to personalize their pages, they can add further apps to the pages themselves.
- A lean launchpad is easier to navigate, and quicker to load. This is especially true for supervisor and managerial roles, where you are most likely to include dynamic data tiles such as Smart Business **KPIs** on your page. A few well-chosen KPIs help your managers focus on maintaining key thresholds. For example, keeping within their allocated budget, or reaching sales revenue targets. Each KPI comes at a small performance cost of an additional call to retrieve current KPI data. So, while useful and while some of this data can be temporarily cached to minimize these data calls, the number of dynamic data tiles should be kept to the essentials.
- Consider creating **two or more pages** for one space if there is a large number of relevant apps. In the SAP-delivered spaces and pages, the pages reflect the tasks within the work contexts of the user groups.
- SAP recommends including **no more than five** sections on one page. A section should contain between three and seven apps. Ideally, there are not more than 25 apps on the whole page.
- Reflect **"Day in the Life of"** working patterns of your users: Focus on the user group for this page and choose the appropriate apps required by them to fulfill their daily tasks. Structure the pages according to the sequence of tasks performed by the users.
- Apply the **"Insight-to-Action Approach"**: Offer initial orientation on the page at the top-level entry point ("Insight") and from there offer drill down to the necessary tasks ("Action"). The idea is to first help users identify what they need to work on and then guide the navigation to the actions resulting from the insights.
- Aim to be **coherent and consistent** across all your business roles, e.g. use consistent titles for spaces, pages and sections etc.
- Aim to create **reusable** pages with a common layout for usage in different spaces.
- You can influence how spaces are **sorted** in the user interface as described in Sorting the Spaces [page 300].
- When **testing** your own spaces, adhere to real-world conditions and keep the number of roles small. You should not assign too many roles to one user.

Related Information

How to Create and Assign Spaces and Pages [page 271]
https://experience.sap.com/fiori-design-web/sap-fiori-launchpad-spaces/

1.3.4.4.2.2 Best Practices for Managing Groups

Groups define the grouping, sort order, and general appearance (tile or link) of apps that are initially displayed on the SAP Fiori launchpad home page for a user.

Context

> **i Note**
>
> Creating new home pages based on groups is no longer recommended. Instead, use the "Manage Launchpad Spaces" and "Manage Launchpad Pages" apps to define the launchpad layout and structure. See Managing Launchpad Spaces and Pages [page 269].

SAP delivers business groups that help you maintaining your own, role-specific groups:

Business group, naming convention <...>_BCG_<...>

Business groups contain a set of applications from a business catalog that are displayed to a user by default on the home page of the SAP Fiori launchpad. Users can further personalize groups by adding or removing apps from the SAP Fiori launchpad, provided the group is not locked. For example, you can use a locked business group to ensure certain important apps are always displayed to users, and unlocked business groups that users can rearrange to improve their personal productivity.

Groups can reference multiple apps from one or more catalogs. Groups do not grant authorizations. If a user is not authorized for an app provided by the group, the unauthorized app will not appear on the user's home page.

Groups contain the optional sub-entities:

- Tiles – the apps that should appear as tiles on the home page, in the order in which they will appear
- Links – the apps that should appear as links on the home page, in the order in which they will appear

> **i Note**
>
> On the SAP Fiori launchpad home page, tiles are displayed before the links that belong to the same group.

Recommendations

- Do not use SAP-delivered business groups directly by assigning them to roles. Instead, we recommend you to create your own business groups in your customer namespace.
- Business groups should contain applications which are regularly used, e.g. on a daily or weekly basis. Rarely used applications can be left in the app finder or found via the SAP Fiori search or *Home* button.
- Consider reducing the number of apps shown on the home page. We recommend you to aim for a maximum number of 64 tiles on the home page (which corresponds to 8 groups with 8 tiles each). This helps to prevent users from being overloaded with content and it avoids performance issues on the launchpad.
- Aim for a good default home page; users can always personalize the home page at any time.

- Avoid overpopulating the home page – scrolling vertically or horizontally can be cumbersome for the user. Consider turning on the tab mode for the group selection bar at the top of the home page so that it only shows one group at a time. See parameter HOMEPAGE_GROUPSELECTIONBAR_MODE under Launchpad Configuration Parameters [page 29].
- Take advantage of link lists. Include apps here that are relevant, though maybe not as important as others. Link lists help save space on the UI.
  See Adding Tiles to and Removing Tiles from Group [page 307].
- Business roles delivered by SAP contain business groups which represent best practice job tasks. These business groups have generic titles, such as *Sales Contract* or *Data Migration*. We recommend referring to this naming convention and adapting your own business groups accordingly.
- Make sure group and tile names are meaningful to the business users who will use them. Involve end users who will perform the tasks of the relevant business role.
- Consider putting the most frequent and urgent tasks in the first groups towards the top of the home page – they should be the easiest to reach every time you log in.
- Consider putting apps that trigger actions (insight to action) in a prominent place in your groups, i.e. at the beginning of a group.
- Aim for only adding entry point apps to business groups: These combine activities, such as *Manage Activities*, instead of single apps for *Create Activities* or *Display Activities*. This approach saves space on the UI, avoids redundancies and eases access. Consider that many apps can be reached via forward navigation.

Related Information

1.3.4.4.3 Typical Scenarios for Managing Launchpad Content

This section contains best practice scenarios on how to use and adapt launchpad content to meet the needs of your company.

Basic Scenario - Use SAP Content as Delivered [page 121]
  Adapt the launchpad content delivered by SAP to the needs of your company by recombining it in your custom business catalogs.

Advanced Scenario - Adapt SAP Template Content and Add Your Own Content [page 123]

1.3.4.4.3.1 Basic Scenario - Use SAP Content as Delivered

Adapt the launchpad content delivered by SAP to the needs of your company by recombining it in your custom business catalogs.
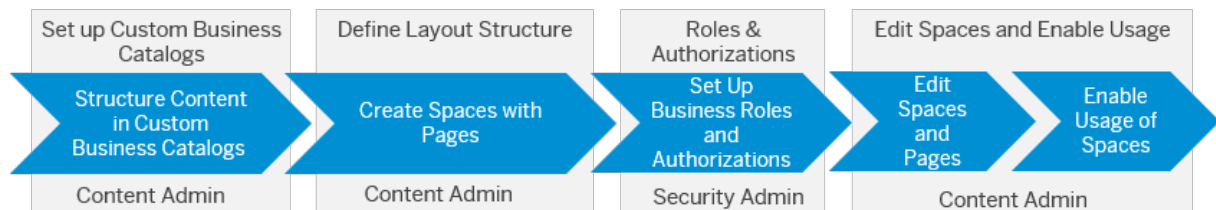
- Structure SAP Content in Custom Business Catalogs [page 121]
- Create Spaces and Pages [page 122]
- Set Up Roles and Authorizations [page 122]
- Edit Spaces and Pages [page 122]
- Enable Usage of Spaces [page 123]

Prerequisites

Use the SAP Fiori apps reference library to search for relevant apps and find an SAP business catalog that is suitable as a template for your own business catalog. See SAP Fiori apps reference library.

Use task list `SAP_FIORI_CONTENT_ACTIVATION` to generate test users and business roles in a test system. See Task List SAP_FIORI_CONTENT_ACTIVATION.

For more information on the SAP Fiori Rapid Activation end-to-end process, see Set up and Configure SAP Fiori and Implement SAP Fiori Apps.

1.3.4.4.3.1.1 Structure SAP Content in Custom Business Catalogs

In the launchpad content manager, copy an SAP-delivered business catalog into your customer namespace and add or remove references to tiles and target mappings as needed.

Set up your business catalogs

Use the launchpad content manager to copy SAP business catalogs into your customer namespace.

1. In your ABAP system, start transaction `/UI2/FLPCM_CUST` to open the launchpad content manager in the customizing scope. This allows you to make client-specific changes.
   See Running the Launchpad Content Manager [page 240].

2. In the *Catalogs* tab, use the *Search Catalogs* field to find the SAP business catalog you want to use as a template.
   See Searching for Launchpad Content [page 242].
3. Copy the catalog. Your copied catalog will include references to the tiles and target mappings in the SAP catalog.
   See Copying Catalogs [page 249].

You can now adjust the content of the copied catalog.

Remove apps from your business catalog

In the launchpad content manager, select your business catalog and remove the references to the tiles and target mappings you don't need.

See Adding and Removing Catalog Content [page 250].

Add apps to your business catalog

In the launchpad content manager, select your business catalog and add references to the tile/target mapping combinations that are missing.

See Adding Reference Tiles and Target Mappings to a Catalog [page 250].

1.3.4.4.3.1.2 Create Spaces and Pages

In the Manage Launchpad Spaces app, create a space with a page.

See Scenario 2: Create and Assign Spaces and Pages in a Bottom-Up Process [page 274].

1.3.4.4.3.1.3 Set Up Roles and Authorizations

In Role Maintenance (transaction PFCG), assign catalogs and spaces (or groups) to users' authorization roles to define which users have access to them.

See Configuring Roles for Catalogs, Spaces and Groups [page 328].

1.3.4.4.3.1.4 Edit Spaces and Pages

To have the apps displayed on the user's entry pages, add them to pages (assigned to spaces) or groups.

See Editing a Page [page 283] and Editing Groups [page 306].

1.3.4.4.3.1.5 Enable Usage of Spaces

To switch to the new layout based on spaces, you need to enable the spaces in the launchpad configuration parameters.

You can either enable it for all users or allow users to switch themselves.

See Enabling Spaces [page 302].

1.3.4.4.3.2 Advanced Scenario - Adapt SAP Template Content and Add Your Own Content
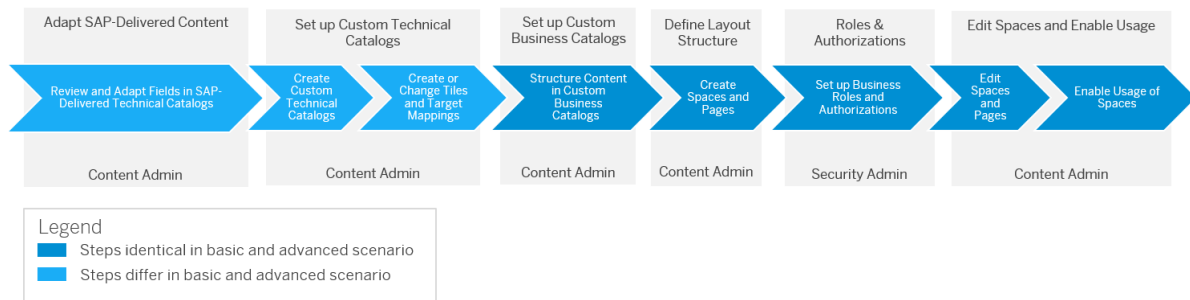
- Adapt Tiles and Target Mappings in SAP-Delivered Technical Catalogs [page 124]
- Create Custom Technical Catalogs [page 124]
- Create or Change Tiles and Target Mappings [page 124]
- Structure SAP Content and Custom Content in Custom Business Catalogs [page 125]
- Create Spaces and Pages [page 122]
- Set Up Roles and Authorizations [page 122]
- Edit Spaces and Pages [page 122]
- Enable Usage of Spaces [page 123]

Prerequisites

Use the SAP Fiori apps reference library to search for relevant apps and find an SAP business catalog that is suitable as a template for your own business catalog. See SAP Fiori apps reference library.

Use task list `SAP_FIORI_CONTENT_ACTIVATION` to generate test users and business roles in a test system. See Task List SAP_FIORI_CONTENT_ACTIVATION.

For more information on the SAP Fiori Rapid Activation end-to-end process, see Set up and Configure SAP Fiori and Implement SAP Fiori Apps.

1.3.4.4.3.2.1 Adapt Tiles and Target Mappings in SAP-Delivered Technical Catalogs

Use the launchpad app manager to adapt tiles and target mappings in SAP-delivered technical catalogs.

Some target application fields and tile fields in a launchpad app descriptor item can be adapted directly in the SAP-delivered technical catalog.

The adaptation mode in the launchpad app manager saves the step of having to copy the launchpad app descriptor items first in order to make the adaptations in your custom technical catalog.

Adaptations that can be made directly are, for example, adapting the title or the description.

See Adapting Launchpad App Descriptor Items [page 165].

1.3.4.4.3.2.2 Create Custom Technical Catalogs

Use the launchpad app manager to create technical catalogs to which you can add your own custom tiles and target mappings.

Create your technical catalogs in the customer namespace. We recommend to use the naming conventions for technical catalogs (<...>_TC_<...>).

In an embedded scenario, you maintain the launchpad app descriptor items in technical catalogs of type *standard*.

In a hub scenario, we recommend to split the technical catalogs according to the app types:

- App types SAPUI5 Fiori app, SAPUI5 Fiori app on SAP BTP, URL app, Tile only:
  Maintain the launchpad app descriptor items in technical catalogs of type *standard* on the front-end server.
- App types Web Dynpro Application, Transaction and WebClient UI Application:
  Maintain the launchpad app descriptor items in technical catalogs (no catalog type selected) in the backend system and replicate them to the front-end server.

See Setting Up Technical Catalogs with the SAP Fiori Launchpad App Manager [page 132] and Customizing Editing Options in the Launchpad App Manager [page 135].

1.3.4.4.3.2.3 Create or Change Tiles and Target Mappings

Use the launchpad app manager to create your own tiles and target mappings. Use the launchpad app manager or the launchpad designer to change your existing tiles and target mappings or to adapt SAP-delivered tiles and target mappings.

Create tiles/target mappings

If there is no suitable tile/target mapping combination for your app, add the original tiles/target mappings to a technical catalog in the configuration scope. You have the following options:

- Add the tiles/target mappings to an existing technical catalog.
  Use the tool with which you created the technical catalog.

- Launchpad app manager: See Maintaining Launchpad App Descriptor Items [page 148].
- Launchpad designer: See Creating and Configuring Tiles and Target Mappings [page 202].
- Create a new technical catalog to which you add the tiles/target mappings.
  We recommend to use the launchpad app manager for this. See Maintaining Launchpad App Descriptor Items [page 148].

Change tiles/target mappings

If you want to make changes to tile/target mapping combinations (e.g. changing the tile title or the parameters passed between related apps), you should make these changes in the technical catalogs where the original tiles/target mappings reside. This avoids breaking references in your business catalogs.

1. If the original tile/target mapping resides in a custom technical catalog, you can directly change it using the tool with which you created the technical catalog.
   - Launchpad app manager: See Launchpad App Descriptor Items [page 107].
   - Launchpad designer: See Creating and Configuring Tiles and Target Mappings [page 202].
2. If the original tile/target mapping resides in an SAP technical catalog and the fields can be adapted in the launchpad app manager, you can directly make changes using the adaptation mode in the launchpad app manager. See Adapting Launchpad App Descriptor Items [page 165] to find out which fields can be adapted in an SAP technical catalog.
3. If the original tile/target mapping resides in an SAP technical catalog and the fields cannot be adapted in the launchpad app manager, you cannot change it directly. You need to first make it available in your own technical catalog by creating a new tile/target mapping based on the existing one. See Copying Launchpad App Descriptor Items From Another Catalog [page 168].

1.3.4.4.3.2.4 Structure SAP Content and Custom Content in Custom Business Catalogs

In the launchpad content manager, copy an SAP-delivered business catalog into your customer namespace and add or remove references to tiles and target mappings as needed.

Set up your business catalogs

Use the launchpad content manager to copy SAP business catalogs into your customer namespace.

1. In your ABAP system, start transaction `/UI2/FLPCM_CUST` to open the launchpad content manager in the customizing scope. This allows you to make client-specific changes.
   See Running the Launchpad Content Manager [page 240].
2. In the *Catalogs* tab, use the *Search Catalogs* field to find the SAP business catalog you want to use as a template.
   See Searching for Launchpad Content [page 242].
3. Copy the catalog. Your copied catalog will include references to the tiles and target mappings in the SAP catalog.
   See Copying Catalogs [page 249].

For tiles and target mappings referring to custom apps, you can simply create an empty business catalog and add tiles and target mappings to it. See Creating Catalogs [page 248].

Remove apps from your business catalog

In the launchpad content manager, select your business catalog and remove the references to the tiles and target mappings you don't need.

See Adding and Removing Catalog Content [page 250].

Add apps to your business catalog

In the launchpad content manager, select your business catalog and add references to the tile/target mapping combinations that are missing.

See Adding Reference Tiles and Target Mappings to a Catalog [page 250].

1.3.4.4.3.2.5 Create Spaces and Pages

In the Manage Launchpad Spaces app, create a space with a page.

See Scenario 2: Create and Assign Spaces and Pages in a Bottom-Up Process [page 274].

1.3.4.4.3.2.6 Set Up Roles and Authorizations

In Role Maintenance (transaction PFCG), assign catalogs and spaces (or groups) to users' authorization roles to define which users have access to them.

See Configuring Roles for Catalogs, Spaces and Groups [page 328].

1.3.4.4.3.2.7 Edit Spaces and Pages

To have the apps displayed on the user's entry pages, add them to pages (assigned to spaces) or groups.

See Editing a Page [page 283] and Editing Groups [page 306].

1.3.4.4.3.2.8 Enable Usage of Spaces

To switch to the new layout based on spaces, you need to enable the spaces in the launchpad configuration parameters.

You can either enable it for all users or allow users to switch themselves.

See Enabling Spaces [page 302].

1.3.4.5 Setting Up Navigation

You have to perform several steps to add applications as navigation targets to the launchpad.

Prerequisites

To allow Web Dynpro ABAP applications to be started from the launchpad, you need to make sure the server runtime for SAP NetWeaver user interface (UI) services is installed in the back end. This is required when there is a separation between front-end server and back-end system. To install the server runtime for SAP NetWeaver user interface (UI) services, you install product instance *Integration Services: Provider* and activate ICF node `/default_host/sap/bc/ui2/nwbc`. For more information about installing the product instance, see chapter "Software Units" in the Master Guide for User Interface Add-On for SAP NetWeaver. For more information about activating ICF nodes, go to https://help.sap.com/viewer/p/SAP_BUSINESS_CLIENT, select your version and search for "Active Services in ICF (UI Services) ".

If you configure Web Dynpro ABAP or SAP GUI applications to run on a system other than the launchpad front-end server, you need to switch off the login cross-site request forgery (XSRF) protection on the ICF node */sap/bc/ui2/nwbc* for NWBC for HTML. Otherwise it might not be possible for end users to log on with a launchpad URL pointing to this Web Dynpro ABAP or SAP GUI application. See SAP Note 1617090 .

Context

For more background information on navigation concepts in the launchpad, see Intent-Based Navigation [page 108].

The following application types can be launched from the launchpad:

- SAPUI5
- Web Dynpro ABAP
- SAP GUI for HTML
- WebClient UI applications (SAP CRM applications)
- SuccessFactors applications
- Any web application

Navigation between launchpad applications is based on intents. To set up navigation, perform the following steps:

Procedure

1. Identify an application that you would like to use as a navigation target.
2. Identify a semantic object you would like to refer to. You can either choose one of the semantic objects shipped by SAP, or create a new semantic object.

3. Create a catalog or use an existing one.

4. Add a target mapping. In the target mapping configuration, you map an intent to a navigation target.

5. (Optional) Create and configure a tile. In the tile configuration, you enter meta data (for example title, description and icon) and define the navigation by specifying semantic object and action.

   You can skip this step if you don't want to launch the app from a tile but from another application (app-to-app navigation).

6. Assign the catalog that contains the tile and the target mapping to a role in role maintenance (transaction PFCG).

Related Information

1.3.4.5.1 Defining Semantic Objects for Navigation

If the semantic objects shipped by SAP do not suit your needs, you can define your own semantic objects.

Context

A semantic object represents a business entity such as a customer, a sales order, or a product. Using semantic objects, you can bundle applications that reflect a specific scenario. They allow you to refer to objects in a standardized way, abstracting from concrete implementations of these objects. For more information, see Intent-Based Navigation [page 108].

To allow intent-based navigation to a given target application, you specify a semantic object and an action when you create tiles and target mappings.

You can either use semantic objects shipped by SAP, or create new semantic objects:

Task	Transaction	View	Tables
Display semantic objects shipped by SAP	/UI2/SEMOBJ_SAP	/UI2/V_SEMOBJ	/UI2/SEMOBJ
			/UI2/SEMOBJT
Display or create custom semantic objects	/UI2/SEMOBJ	/UI2/V_SEMOBJC	/UI2/SEMOBJC
			/UI2/SEMOBJCT

To create a custom semantic object, proceed as follows:

Procedure

1. Go to customizing (transaction SPRO).

2. Choose *SAP Reference IMG*.

3. Go to ▌❭ *UI Technologies* ❭ *SAP Fiori* ❭ *Setting Up Launchpad Content* ❭ *Setting Up Technical Catalogs*
   ❭ *Define Custom Semantic Objects for Navigation* ❭

4. Choose ⊕ to open the table for semantic objects.

5. Choose *New Entries*.
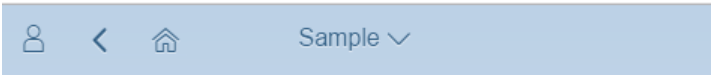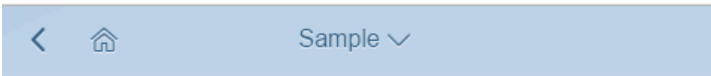
   By defining an entry with the same key, you can overwrite the attributes of a semantic object that was
   delivered by SAP.

1.3.4.5.2 Configuring the Header of Applications and the Launchpad Shell Bar

You can configure the header bar of an application to hide some of the UI elements provided by the shell, or not to display the header bar and application title at all.

Configuring the Header Bar or Launchpad Shell Bar

It is possible to configure the header bar state to hide some of the UI elements provided by the shell, or not to display at all. You do this in the URL, using the `sap-ushell-config` URL parameter with one of the following values:

Parameter Value (State)	Description	Relevant to
`standalone`	Displays only the ⚇ *(user actions menu)* button or the user photo/avatar, ‹ *(Back)* button, company logo, and the application title. ⚇ ‹ **SAP** Sample ⌄ The ⚇ button or the user photo/avatar opens an options menu that contains only the following default application-specific menu items: *Contact Support* and *About*. It also shows custom application-specific menu items they have been developed for the app. > **i Note** > This specific *standalone state* should not be confused with the broader concept of running apps in the *standalone mode* that was supported in older versions. This setting simply defines one of the possible states of the header bar for an app.	Applications only
`embedded`	Same as `standalone`, except it displays the ⌂ *(Home)* button instead of the company logo. ⚇ ‹ ⌂ Sample ⌄	Applications only
`merged`	Displays only the application's title, and the ‹ *(Back)* and ⌂ *(Home)* buttons. The launchpad shell bar items, such as the ⚇ *(user actions menu)* button or the user photo/avatar, company logo, search bar, and ☰ *(Notifications)* button, are not shown. ‹ ⌂ Sample ⌄ When navigating to the launchpad, it is not possible to personalize the home page. This means that the user cannot move or remove tiles.	Applications and the launchpad

Parameter Value (State)	Description	Relevant to
headerless	Does not display the launchpad shell bar or the application title. When navigating to the launchpad, it is not possible to personalize the home page. This means that the user cannot move or remove tiles.	Applications and the launchpad

> **i Note**
>
> When running an application in standalone mode, the *Save as Tile* option in the footer bar of an application is disabled.

You add the `sap-ushell-config` parameter before the hash sign.

This parameter is intended for hosting an app, for example, in SAP Enterprise Portal.

> **⁘ Example**
>
> The following example shows a URL that calls the launchpad shell with no header:
>
> ```
> http://<server>:<port>/sap/bc/ui2/flp?sap-client=120&sap-ushell-
> config=headerless#Shell-home
> ```

1.3.4.5.2.1 Identifying the Component Name

This section describes how to identify the SAPUI5 component name based on the location of the application's resource file.

Context

If you know the location of an SAPUI5 application in the BSP repository, you can look up the component name in the `Component.js` file.

Procedure

1. Enter the URL of the `Component.js` file of your SAPUI5 application into your web browser.

   The URL of the Component.js. file has the following syntax:

   ```
   http://<server>:<port>/<path>/Component.js
   ```

   For example, if the path of your SAPUI5 application is `/sap/bc/ui5_ui5/sap/crm_opprtnty`, the URL would be as follows:

   ```
   http://<server>:<port>/sap/bc/ui5_ui5/sap/crm_opprtnty/Component.js
   ```

Replace `<server>` and `<port>` with the server and port of your SAP Fiori launchpad installation.

The code of the `Component.js` file is displayed.

2. In the code of the `Component.js` file, locate a line starting with `jQuery.sap.declare`.

This line has the following syntax:

```
jQuery.sap.declare("<componentName>.Component")
```

For example, if the part between the brackets reads (`"cus.crm.opportunity.Component"`), the component name is `cus.crm.opportunity`.

1.3.4.6 Setting Up Technical Catalogs

Technical catalogs contain original tiles and target mappings.

The tiles and target mappings can be referenced from business catalogs. Technical catalogs are not intended to be assigned directly to users.

You have the following options:

- Use the predefined technical catalogs provided by SAP
- Create your own technical catalogs.

Related Information

Best Practices for Managing Catalogs [page 115]

1.3.4.6.1 Setting Up Technical Catalogs with the SAP Fiori Launchpad App Manager

The SAP Fiori launchpad app manager allows you to maintain technical catalogs with launchpad app descriptor items.

Prerequisites

You are assigned to the role `SAP_FLP_ADMIN`.

See Configuring Roles with Launchpad Start Authorizations [page 327].

Context

> **i Note**
>
> In previous versions, the launchpad app manager was called Mass Maintenance Tool (MMT). The tool was renamed due to its extended feature scope.

For more information on the different entities required to set up launchpad content, see About Launchpad Content [page 103].

Setup in a Standalone (Hub) Scenario

The following graphic shows the content setup with the launchpad app manager in a standalone (hub) scenario:
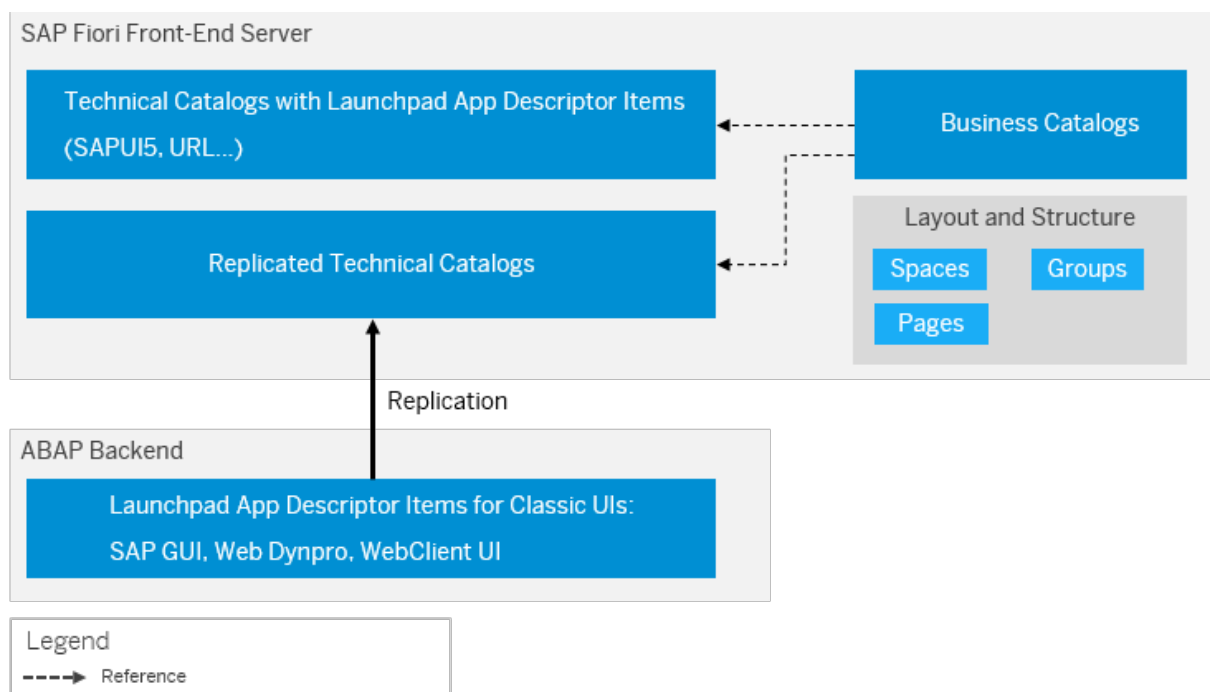
In a standalone (hub) scenario, you create the launchpad app descriptor items on the front-end server or in the ABAP backend, depending on the respective app type:

- You create technical catalogs with launchpad app descriptor items for SAPUI5 Fiori apps on the front-end server. These launchpad app descriptor items can directly be referenced from business catalogs and spaces/pages or groups.
- You create launchpad app descriptor items for classic UIs (applications based on Web Dynpro, SAP GUI for HTML and WebClient UI technology) in the ABAP backend. These launchpad app descriptor items have to be replicated to the front-end server.
  During the replication step, technical catalogs containing these launchpad app descriptor items are created on the front-end server. The replicated catalogs cannot be edited or assigned to an authorization role.
  The launchpad app descriptor items in the replicated technical catalogs can be referenced from business catalogs and spaces/pages groups.

Setup in an Embedded Scenario

The following graphic shows the content setup with the launchpad app manager in an embedded scenario:
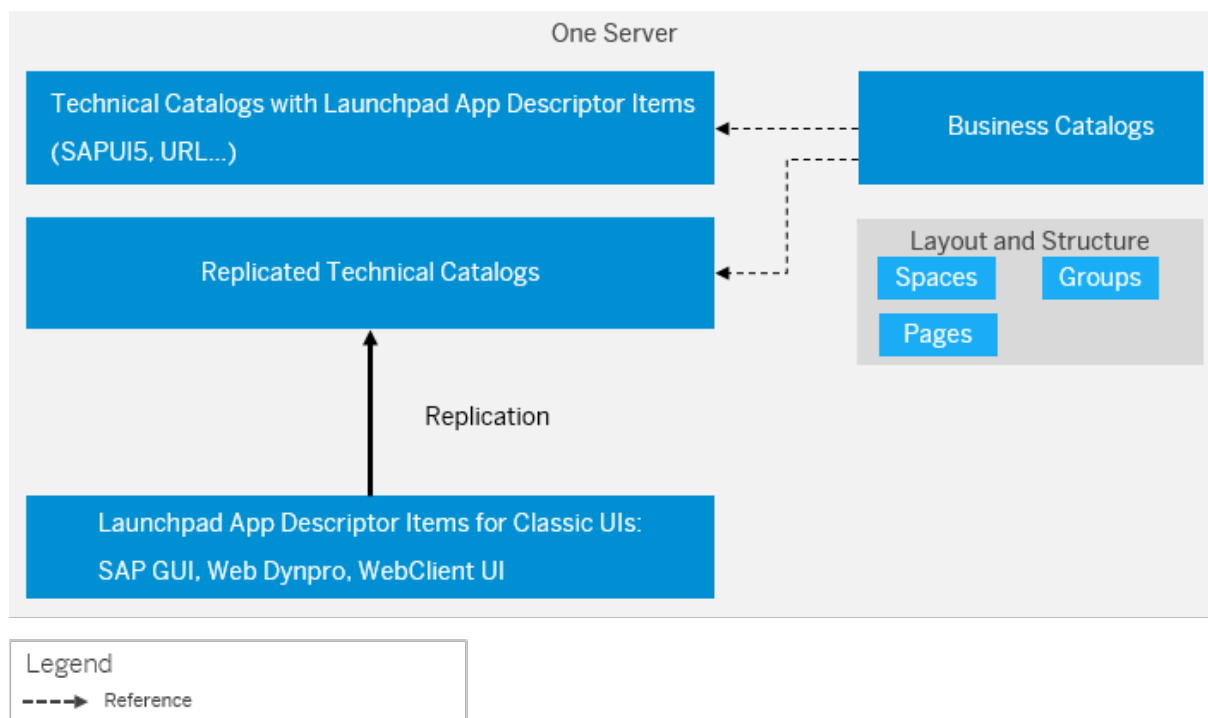
In an embedded scenario, you create all your launchpad app descriptor items on the same server. For technical reasons, you have to replicate the launchpad app descriptor items for classic UIs to make use of them.

See Setting Up the Replication of Back-End Catalogs [page 168].

Procedure

1. Define which catalog types can be maintained in the launchpad app manager. See Maintain Allowed Catalog Types for Launchpad App Manager [page 136].
2. Create a launchpad application descriptor item for each application you want to integrate in the launchpad.
   - Add your launchpad application descriptor item to an existing technical catalog.
   - If there is no catalog that fulfills your requirements, you can create your own technical catalogs and add your launchpad application descriptor items to them.

   See Maintaining Launchpad App Descriptor Items [page 148].
3. If the technical catalog was created in a back-end component (catalog type *replicable*), you need to replicate the catalog to the front-end server.

   See Setting Up the Replication of Back-End Catalogs [page 168].

   For catalogs of type *standard*, you can skip this step.
4. Create business catalogs referencing the tiles and target mappings in the technical catalog.
   See Setting Up Business Catalogs [page 238].
5. To have the apps displayed on the user's entry page/s, add them to pages (assigned to spaces) or groups.
   See Configuring Roles for Catalogs, Spaces and Groups [page 328].

6. In transaction `PFCG`, assign spaces, groups and catalogs to users' authorization roles to define which users have access to them.

    See Configuring Roles for Catalogs, Spaces and Groups [page 328].

Related Information

Setting Up Technical Catalogs [page 132]
Best Practices for Managing Catalogs [page 115]

1.3.4.6.1.1 Customizing Editing Options in the Launchpad App Manager

In certain scenarios, it makes sense to disable certain app types and catalog types in the launchpad app manager.

Related Information

Maintain Allowed App Types for Launchpad App Manager [page 135]
Maintain Allowed Catalog Types for Launchpad App Manager [page 136]

1.3.4.6.1.1.1 Maintain Allowed App Types for Launchpad App Manager

Define a list of allowed application types that can be maintained in the launchpad app manager.

Context

If you leave the table empty, launchpad app descriptor items for all application types can be maintained in the launchpad app manager.

In an embedded deployment, we recommend you leave the table empty to allow all app types.

In a hub deployment, we recommend you restrict the application types that can be maintained with the launchpad app manager as follows:

- Allow the following application types on the front-end server:
  - *SAPUI5 Fiori App*

- *SAPUI5 Fiori App on* SAP BTP
- *Tile Only*
- *URL App*
- Allow the following application types in the back-end system:
  - *Transaction*
  - *Web Dynpro Application*
  - *WebClient UI Application*

Procedure

1. In SAP Fiori Customizing (transaction SPRO), choose ▶ *SAP NetWeaver* ❯ *UI Technologies* ❯ *SAP Fiori* ❯ *Setting Up Launchpad Content* ❯ *Setting Up Technical Catalogs* ◀ and launch the IMG activity *Maintain Allowed Application Types for Launchpad App Manager*.
2. Choose *New Entries*.
3. Select the relevant app types using the drop-down menu.

Related Information

Maintaining Launchpad App Descriptor Items [page 148]
Maintain Allowed Catalog Types for Launchpad App Manager [page 136]

1.3.4.6.1.1.2 Maintain Allowed Catalog Types for Launchpad App Manager

Define which catalog types can be maintained in the launchpad app manager.

Context

If you launch the IMG activity *Maintain Allowed Catalog Types for Launchpad App Manager* you are presented with a table in which you define which catalog types can be maintained in the launchpad app manager.

Settings for Embedded Deployment

The following graphic visualizes the recommended settings for an embedded deployment:
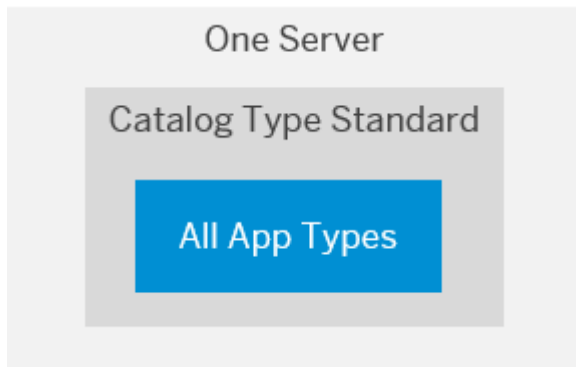
Allow catalog type *standard*.

When a user enters a new catalog ID in the *Catalog Entry* screen of the launchpad app manager, a new catalog with a corresponding UIAC object of type *standard* is created.

Settings for Hub Deployment

The following graphic visualizes the recommended settings for a hub deployment:
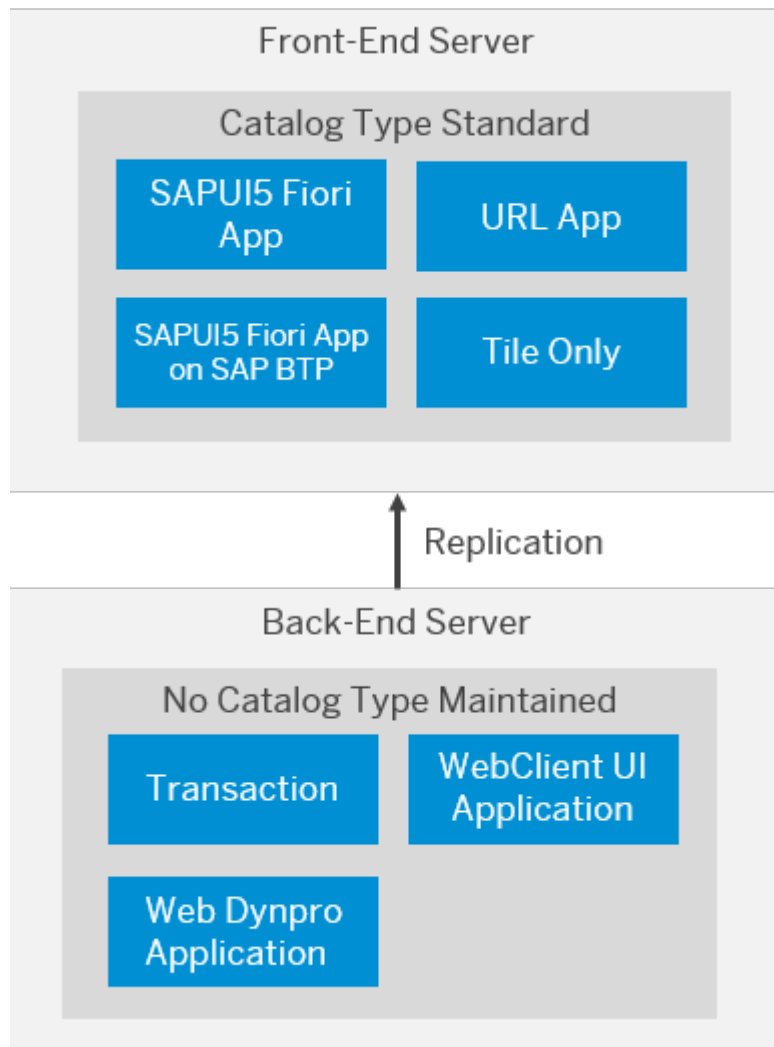
Split your technical catalogs according to the app type:

- On the front-end server, allow catalog type *standard*.
  This catalog type should be used to maintain launchpad app descriptor items for the following app types:
  - *SAPUI5 Fiori App*
  - *SAPUI5 Fiori App on SAP Business Technology Platform (Beta)*
  - *Tile Only*
  - *URL App*

  When a user enters a new catalog ID in the *Catalog Entry* screen of the launchpad app manager, a new catalog with a corresponding UIAC object of type *standard* is created.
- In the back-end system, leave the table empty.
  This setting should be used to maintain launchpad app descriptor items for the following app types:
  - *Transaction*
  - *Web Dynpro Application*
  - *WebClient UI Application*

When a user enters a new catalog ID in the *Catalog Entry* screen of the launchpad app manager, a new catalog without corresponding UIAC object will be created

Note that this option requires you to replicate the launchpad app descriptor items to the front-end server. See Setting Up the Replication of Back-End Catalogs [page 168].

Procedure

To set the catalog type for the launchpad app manager, proceed as follows:

1. In SAP Fiori Customizing (transaction SPRO), click *SAP Reference IMG*.

2. Choose ⫸ *SAP NetWeaver* ⟩ *UI Technologies* ⟩ *SAP Fiori* ⟩ *Setting Up Launchpad Content* ⟩ *Setting Up Technical Catalogs* ⫷ and launch the IMG activity *Maintain Allowed Catalog Types for Launchpad App Manager*.

3. Choose *New Entries*.

4. Select the *Standard Catalog* catalog type using the drop-down menu.

   Please do not set catalog type *Replicable Catalog* since it is deprecated.

Related Information

Maintain Allowed App Types for Launchpad App Manager [page 135]

1.3.4.6.1.2 Running the Launchpad App Manager

The launchpad app manager is always launched in the configuration scope, allowing you to make system-wide changes.

1. Log on to your ABAP system.

2. To launch the launchpad app manager, in Customizing (transaction SPRO), choose ⫸ *SAP NetWeaver* ⟩ *UI Technologies* ⟩ *SAP Fiori* ⟩ *Setting Up Launchpad Content* ⟩ *Setting Up Technical Catalogs* ⫷ and launch the IMG activity *Manage Technical Catalogs with SAP Fiori Launchpad App Manager*.

Transaction /UI2/FLPAM is started.

Related Information

Scopes for Adapting Launchpad Content [page 111]
Create Custom Technical Catalogs [page 124]

1.3.4.6.1.3 Searching for Catalogs and Launchpad App Descriptor Items

When you enter the launchpad app manager, you have three different views to search for content or to maintain catalogs or launchpad app descriptor items:

Depending on which catalog types you have maintained in Customizing (transaction SPRO), you either see the catalog entry page or the search page:

- If you have maintained no catalog type, you see the catalog entry page and catalogs **without** corresponding UIAC object are created.
- If you have maintained the catalog type *Standard*, you start on the search page and catalogs **with** a corresponding UIAC object are created.

You can switch between different views by clicking *Switch to Facet Filter*, *Switch to Search*, or *Switch to Catalog Entry* in the top left corner.

Catalog Entry View in the Launchpad App Manager
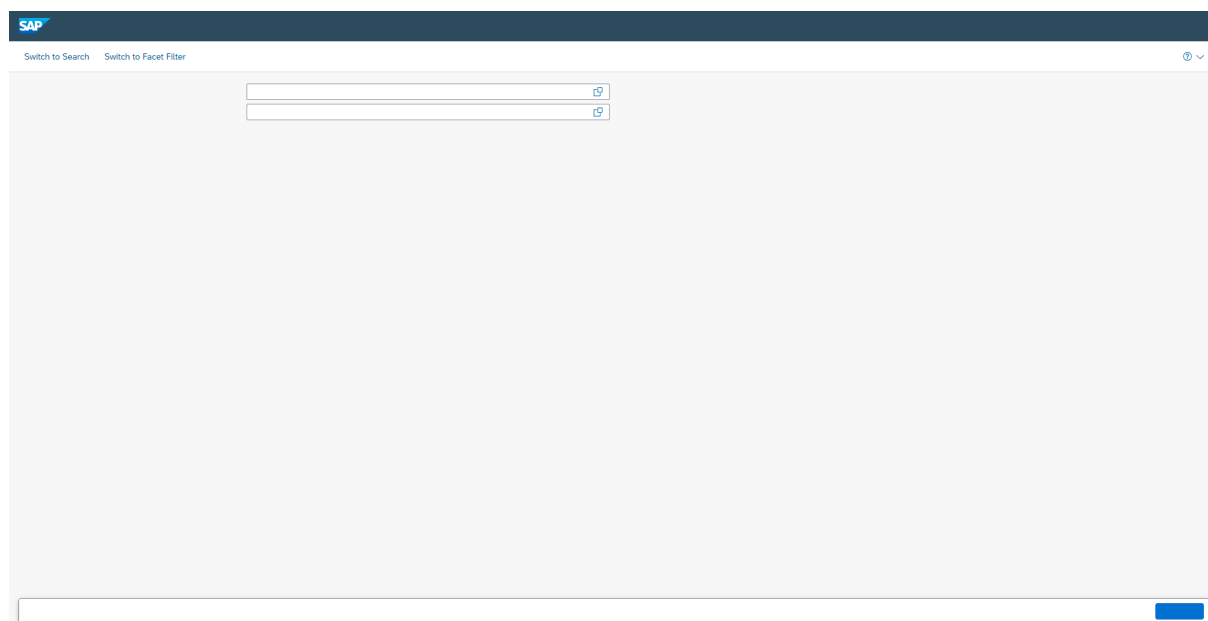
In the catalog entry view, you can search for an existing catalog. You can either enter the catalog ID or press F4 to select a catalog from the list.

Search View in the Launchpad App Manager
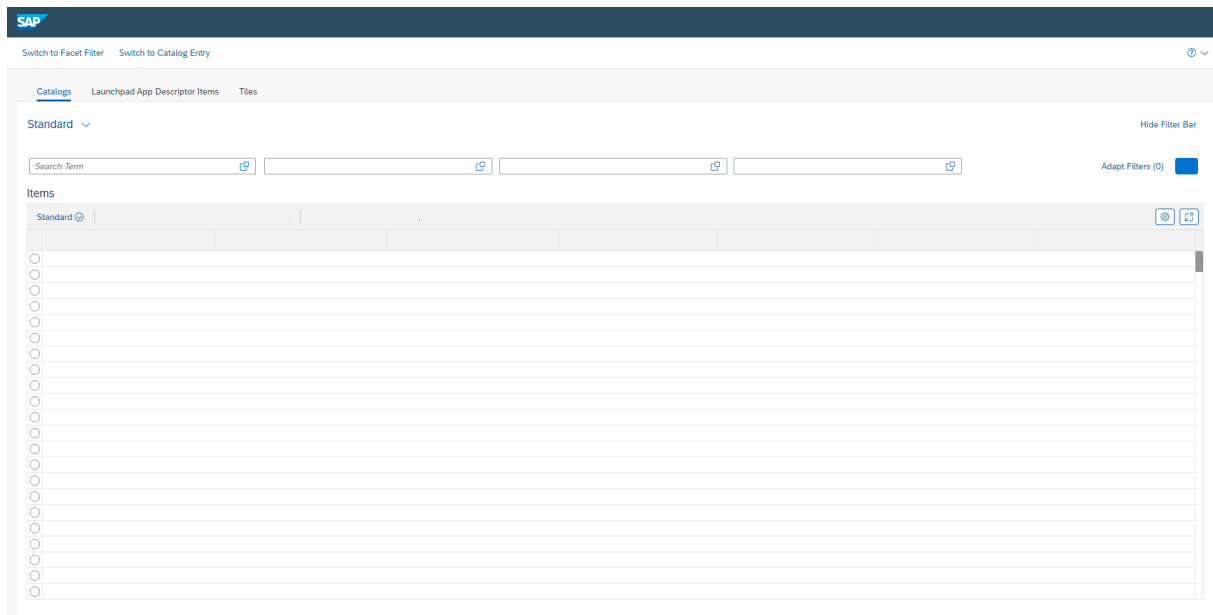
In the search view, you can search for the following entities:

- *Catalogs*
- *Launchpad App Descriptor Items*
- *Tiles*

If you click on the respective tab, you are presented with a table with the relevant items listed. Above the table, you can see fields and functions to specify your search. Standard search criteria depend on the respective tab. You also have the option to create, edit, or delete catalogs. If you want to learn more about how to maintain catalogs, see Maintaining Technical Catalogs [page 142].

You can hide the search criteria to provide more space on the screen.

Facet Filter in the Launchpad App Manager
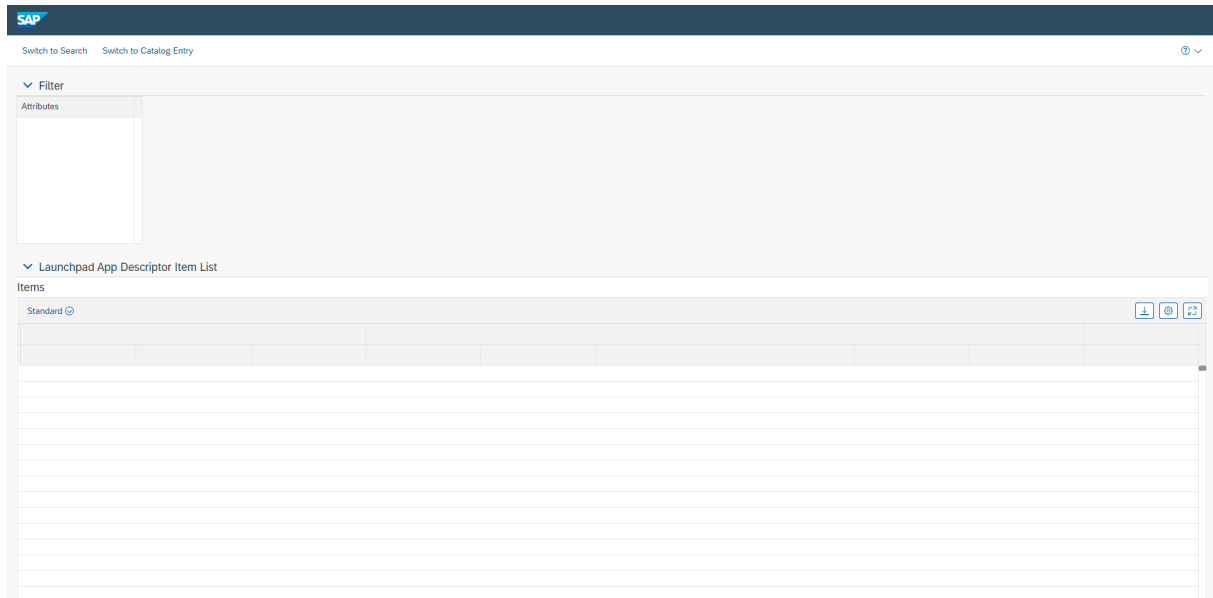
In the facet filter view, you can search for specific launchpad app descriptor items. You can specify relevant field attributes, such as app type, semantic object, additional parameters, and more.

In the table below, you get a list of the launchpad app descriptor items that match your criteria. You can click on the catalog ID that directly opens the catalog in which the launchpad app descriptor item is stored.

Related Information

[Defining Selection Criteria for Searching (Filtering)](#)

1.3.4.6.1.4 Maintaining Technical Catalogs

Prerequisites

- You are assigned to the back-end authorization role `SAP_FLP_ADMIN`.

Context

You can create, edit, and delete technical catalogs in the launchpad app manager or by using the transaction `/UI2/FLPCAT`:

- Creating Technical Catalogs [page 143]
- Maintaining Catalog Titles [page 145]
- Deleting Technical Catalogs [page 146]
- Maintaining Technical Catalogs Using the Transaction /UI2/FLPCAT [page 146]

1.3.4.6.1.4.1 Creating Technical Catalogs

Context

You have three options to create a technical catalog:

- Create a technical catalog using the catalog entry page in the launchpad app manager.
- Create a technical catalog via the *Search* page in the launchpad app manager.
- Create a technical catalog using the transaction `/UI2/FLPCAT`. See Maintaining Technical Catalogs Using the Transaction /UI2/FLPCAT [page 146].

> **i Note**
>
> Depending on which catalog types you have maintained in the Customizing table, you either see the catalog entry page or the search page when launching the launchpad app manager. See Searching for Catalogs and Launchpad App Descriptor Items [page 140].

Creating Technical Catalogs on the Catalog Entry Page

Procedure

To create a catalog on the catalog entry page in the launchpad app manager, you need to do the following:

> **i Note**
>
> When you create a catalog directly in the launchpad app manager, you don't have the option to add a catalog title. This step needs to be carried out in a subsequent step. See Maintaining Catalog Titles [page 145].

1. Launch the launchpad app manager. See Running the Launchpad App Manager [page 139].

2. In the *Catalog ID* field, enter a new ID to create your own catalog.

> **i Note**
>
> The ID should not have more than 20 characters.
>
> To learn about the naming conventions for technical catalogs, see Best Practices for Managing Catalogs [page 115].

3. (Optional) Enter the ID of an existing semantic object to define that all launchpad app descriptor items added to the selected catalog are assigned to the same semantic object. If you choose not to do that, you later have to choose a semantic object for each launchpad app descriptor item individually.

4. Choose *Continue*.

   You have successfully created your catalog.

   You can now continue with filling your catalog with launchpad app descriptor items. See Maintaining Launchpad App Descriptor Items [page 148].

Creating Technical Catalogs on the Search Page

Procedure

Alternatively, you have the option to create a catalog directly via the *Search* page in the launchpad app manager.

1. Launch the launchpad app manager. See Running the Launchpad App Manager [page 139].
2. In the top left corner, click *Search*.
3. Choose *New Catalog*.

   Depending on the catalog types you have maintained in the Customizing table in SAP Fiori Customizing (transaction SPRO), the button might be called differently. See Maintain Allowed Catalog Types for Launchpad App Manager [page 136].

4. In the window that opens, fill out the following fields:

Field	Description
Catalog ID	Enter a new ID.
	> **i Note** > > The ID should not have more than 20 characters. > > To learn about the naming conventions for technical catalogs, see Best Practices for Managing Catalogs [page 115].
Language	This field is already filled.
Catalog Title	Choose a title for your catalog.
Package	Select a package for your catalog.

Field	Description
Transport Request	Select a transport request for your catalog.
Create Empty Catalog Only	Check this box if you don't plan on filling the catalog with launchpad app descriptor items right away. If you deselect this checkbox, empty catalogs are deleted automatically.

5. Choose *Save*.

   You have successfully created your catalog.

   You can now continue with filling your catalog with launchpad app descriptor items. See Maintaining Launchpad App Descriptor Items [page 148].

1.3.4.6.1.4.2 Maintaining Catalog Titles

Context

To add or change the title of a technical catalog, you need have two options:

- Change the title directly in the launchpad app manager
- Change the title in the transaction `/UI2/FLPCAT`. See Maintaining Technical Catalogs Using the Transaction /UI2/FLPCAT [page 146].

Change the Title in the Launchpad App Manager

Procedure

1. In SAP Fiori Customizing (transaction SPRO), choose ▶ *SAP NetWeaver* 〉 *UI Technologies* 〉 *SAP Fiori* 〉 *Setting Up Launchpad Content* 〉 *Setting Up Technical Catalogs* ◀ and launch the IMG activity *Manage Technical Catalogs with SAP Fiori Launchpad App Manager*.

   Transaction `/UI2/FLPAM` is launched.
2. In the top left corner, switch to the *Search* view.
3. Select the relevant catalog.
4. Choose *Change Catalog Title*.
5. In the window that opens, enter a new catalog title or overwrite the existing one.
6. Choose *Save*.

1.3.4.6.1.4.3 Deleting Technical Catalogs

Context

To delete a technical catalog, you need to delete its content. Once the catalog has no content assigned anymore, it is automatically deleted. It is possible now, however, to create empty catalogs. To delete an empty catalog, you have two options:

- Delete a technical catalog in the launchpad app manager.
- Delete a technical catalog in the transaction `/UI2/FLPCAT`. See Maintaining Technical Catalogs Using the Transaction /UI2/FLPCAT [page 146].

Delete an Empty Technical Catalog in the Launchpad App Manager

Procedure

1. Launch the launchpad app manager. See Running the Launchpad App Manager [page 139].
2. Depending on your catalog type settings, you either start in the catalog entry view or in the search view. Switch to the search view by clicking *Switch to Search* in the top left corner.
3. Select the catalog you want to delete.
4. Choose *Delete Catalog*.

   You have successfully deleted your catalog.

1.3.4.6.1.4.4 Maintaining Technical Catalogs Using the Transaction /UI2/FLPCAT

Context

Another option to maintain technical catalogs is by using the transaction `/UI2/FLPCAT`:

- Create a technical catalog using the transaction `/UI2/FLPCAT`.
- Maintain the catalog title using the transaction `/UI2/FLPCAT`.
- Delete a technical catalog using the transaction `/UI2/FLPCAT`.

Creating Technical Catalogs Using the Transaction /UI2/ FLPCAT

Procedure

To create a catalog with the transaction `/UI2/FLPCAT`, you need to do the following:

1. In SAP Fiori Customizing (transaction SPRO), choose ▶ *SAP NetWeaver* ❯ *UI Technologies* ❯ *SAP Fiori* ❯ *Setting Up Launchpad Content* ❯ *Setting Up Technical Catalogs* ◀ and launch the IMG activity *Maintaining Technical Catalogs*.

   Transaction `/UI2/FLPCAT` is launched.

2. Choose *Create Catalog*.

3. In the window that opens, fill out the following fields:

Field	Description
Catalog ID	Enter a new ID.
	i Note The ID shouldn't have more than 20 characters. To learn about the naming conventions for technical catalogs, see Best Practices for Managing Catalogs [page 115].
Catalog Title	Choose a title for your catalog.
Package	Select a transport request.

4. Choose *Continue*.

   You have successfully created your catalog.

   You can now continue with filling your catalog with launchpad app descriptor items. See Maintaining Launchpad App Descriptor Items [page 148].

Maintaining Catalog Titles Using the Transaction /UI2/ FLPCAT

Procedure

To add or change the title of a technical catalog, you need to do the following steps:

1. In SAP Fiori Customizing (transaction SPRO), choose ▶ *SAP NetWeaver* ❭ *UI Technologies* ❭ *SAP Fiori* ❭ *Setting Up Launchpad Content* ❭ *Setting Up Technical Catalogs* ❬ and launch the IMG activity *Maintaining Technical Catalogs*.

   Transaction `/UI2/FLPCAT` is launched.
2. Select the relevant catalog.
3. Choose *Change Catalog Title*.
4. In the window that opens, enter a new catalog title or overwrite the existing one.
5. Choose *Continue*.

Deleting Technical Catalogs Using the Transaction `/UI2/FLPCAT`

Procedure

This function is used to delete **empty** catalogs that are not needed. If the catalog you want to delete still contains applications, you need to access the launchpad app manager and delete the content of your catalog first. Once you save the changes, the catalog is deleted automatically. See Deleting Technical Catalogs [page 146].
Perform the following steps to delete an empty catalog.

1. In SAP Fiori Customizing (transaction SPRO), choose ▶ *SAP NetWeaver* ❭ *UI Technologies* ❭ *SAP Fiori* ❭ *Setting Up Launchpad Content* ❭ *Setting Up Technical Catalogs* ❬ and launch the IMG activity *Maintaining Technical Catalogs*.

   Transaction `/UI2/FLPCAT` is launched.
2. Select the relevant catalog.
3. Choose *Delete Catalog*. The catalog must be empty to perform this step.

   You have successfully deleted your catalog.

1.3.4.6.1.5 Maintaining Launchpad App Descriptor Items

Applications require a launchpad app descriptor item if they are to be called from the SAP Fiori launchpad.

Prerequisites

- You are assigned to the backend authorization role SAP_FLP_ADMIN.

Procedure

Perform the following steps:

1. In the *Catalog ID* field, enter the catalog ID of an existing catalog or create a new one. See Maintaining Technical Catalogs [page 142].

2. Choose *Continue*.

3. Switch to edit mode and select a transport request.

   You are presented with a table in which to enter the information of your launchpad app descriptor.

4. To create a new launchpad app descriptor item, choose *Insert*.

   The launchpad app descriptor items you create using the launchpad app manager are transportable objects with connections to the Note Assistant (transaction SNOTE) and the Correction Workbench (transaction SCWB). An application descriptor service on the ABAP server is consumed by the launchpad to retrieve the relevant information.

   > ### i Note
   >
   > Performance of operations on catalog level (e.g. copying a catalog) depend on the size of the catalog. It is recommended to keep the number of launchpad app descriptor items per catalog as low as possible.

5. Choose the *App Type* that you require:

   - **SAPUI5 Fiori App**
   - **SAPUI5 Fiori App on SAP BTP (Beta)**
     See Integrating SAP BTP Extension Apps (Beta) [page 351].
   - **Tile Only**
   - **Transaction**
   - **URL App**
   - **Web Dynpro Application**
   - **WebClient UI Application**

   > ### i Note
   >
   > Whether the app types listed above are displayed for the system to which you are logged on depends on the settings in the corresponding customizing table. See Maintain Allowed App Types for Launchpad App Manager [page 135].

6. Provide the following settings.

   Each app type has different settings. Filter for your app type in the below table to display all relevant fields and a description of the settings.

Settings for All Application Types

Field Name	App Type	Description	Field Type
Semantic Object	SAPUI5 Fiori App SAPUI5 Fiori App on SAP BTP (Beta) Transaction URL App Web Dynpro Application WebClient UI Application	Semantic object to be referred when clicking the tile, for example **Order**. The following characters are allowed: uppercase letters [A-Z], lowercase letters [a-z], numbers [0-9]. The first character must be an uppercase or lowercase letter. If the semantic objects shipped by SAP do not suit your needs, you can define your own semantic objects. See Defining Semantic Objects for Navigation [page 128]. For more information on intent-based navigation, see Intent-Based Navigation [page 108].	Mandatory
Action	SAPUI5 Fiori App SAPUI5 Fiori App on SAP BTP (Beta) Transaction URL App Web Dynpro Application WebClient UI Application	For the action name, choose a verb or a short phrase starting with lower case and without blanks. Examples are **display**, **create**, or **release**. The following characters are allowed: uppercase letters [A-Z], lowercase letters [a-z], numbers [0-9] and underscores [_]. The first character must be an uppercase or lowercase letter or an underscore. For more information on intent-based navigation, see Intent-Based Navigation [page 108].	Mandatory

Field Name	App Type	Description	Field Type
Target Application Title	SAPUI5 Fiori App SAPUI5 Fiori App on SAP BTP (Beta) Tile Only Transaction URL App Web Dynpro Application WebClient UI Application	Enter a title that will appear as a title for the smart link.	Optional
SAPUI5 Component ID	SAPUI5 Fiori App SAPUI5 Fiori App on SAP BTP	Choose a SAPUI5 component ID from the list. Example: `cus.sd.myquotations`. For app type *SAPUI5 Fiori App*, see Identifying the Component Name [page 131].	Mandatory
SAP Fiori ID	SAPUI5 Fiori App SAPUI5 Fiori App on SAP BTP (Beta) URL App	Displays the SAP Fiori ID of your application.	Optional
Suppress Tile	SAPUI5 Fiori App SAPUI5 Fiori App on SAP BTP (Beta) Tile Only Transaction URL App Web Dynpro Application WebClient UI Application	Click this checkbox to suppress a tile. Tiles that are only used for internal navigation can be suppressed.	Optional

Field Name	App Type	Description	Field Type
Parameters Maintained	SAPUI5 Fiori App SAPUI5 Fiori App on SAP BTP (Beta) Tile Only Transaction URL App Web Dynpro Application WebClient UI Application	Key-value pairs defining parameters for the semantic object, for example `orderID=4711`. If you enter multiple parameters, separate them with an ampersand (&), for example `orderID=10000&custID=c82200`.	Optional
Has Parameter Defaults	SAPUI5 Fiori App SAPUI5 Fiori App on SAP BTP (Beta) Tile Only Transaction URL App Web Dynpro Application WebClient UI Application	This checkbox is automatically ticked when you fill out the relevant fields *Parameter Name* and *Default Value* under the **Parameters** tab.	Optional
Has Parameter Filters	SAPUI5 Fiori App SAPUI5 Fiori App on SAP BTP Tile Only Transaction URL App Web Dynpro Application WebClient UI Application	This checkbox is automatically ticked when you fill out the relevant fields *Parameter Name* and *Filter Value* under the **Parameters** tab.	Optional
Additional Parameters Allowed	SAPUI5 Fiori App SAPUI5 Fiori App on SAP BTP (Beta) Tile Only Transaction URL App Web Dynpro Application WebClient UI Application	Select the checkbox to allow passing additional parameters that are not defined in the table to the target.	Optional

Field Name	App Type	Description	Field Type
Device Types	SAPUI5 Fiori App SAPUI5 Fiori App on SAP BTP (Beta) Tile Only Transaction URL App Web Dynpro Application WebClient UI Application	Select the device types that are suitable for running the target application: • *Desktop* • *Tablet* • *Phone* The selected device types determine the devices on which to display the tiles using this target mapping. By default, all types are selected. If you deselect a type, the tiles using this target mapping are not displayed on that device.	Optional
Application Component ID	SAPUI5 Fiori App SAPUI5 Fiori App on SAP BTP (Beta) Tile Only Transaction URL App Web Dynpro Application WebClient UI Application	Choose the relevant component used for customer incidents. The ACH component is determined based on the corresponding entry in the app descriptor (manifest.json). It can be overwritten by the sap-ach parameter in the signature.	Optional
Launchpad App Descriptor Item ID	SAPUI5 Fiori App SAPUI5 Fiori App on SAP BTP (Beta) Tile Only Transaction URL App Web Dynpro Application WebClient UI Application	Displays the launchpad app descriptor item ID for your application.	Read-Only

Field Name	App Type	Description	Field Type
Target Mapping ID	SAPUI5 Fiori App SAPUI5 Fiori App on SAP BTP (Beta) Tile Only Transaction URL App Web Dynpro Application WebClient UI Application	Displays the target mapping ID for your application.	Read-Only
Target Mapping Information	SAPUI5 Fiori App SAPUI5 Fiori App on SAP BTP (Beta) Tile Only Transaction URL App Web Dynpro Application WebClient UI Application	Add additional information to the target mapping. The text you enter here won't be translated.	Optional
System Alias	SAPUI5 Fiori App on SAP BTP (Beta) URL App	Specify a system alias of the target system for your application. For more information on how to configure a system alias, see Configuring Remote Systems [page 308].	Mandatory
Target URL	Tile Only URL App	Enter the URL of the application. You can enter any URL.	Mandatory
Transaction Code	Transaction	Enter the transaction code which starts the application.	Mandatory
Web Dynpro Application	Web Dynpro Application	This ID refers to your Web Dynpro ABAP application.	Mandatory
Configuration ID	Web Dynpro Application	This configuration ID refers to specially configured Web Dynpro ABAP applications.	Optional

Field Name	App Type	Description	Field Type
Flavor ID	Web Dynpro Application	This flavor ID refers to personalized and customized Web Dynpro ABAP applications. See Assigning Flavors to Launchpad Content [page 71].	Optional
WebClient Application ID	WebClient UI Application	Displays the ID that refers to your WebClient Application.	Mandatory
WDA Apps Integration Mode	Web Dynpro Application	Mode in which the Web Dynpro application is started, that is, with or without NWBC for HTML running invisibly in the background. Values can be set to **Direct**, **Compatible**, or **System Default**. • *Direct:* If the application specific value for the integration mode is set to `Direct`, the value which is returned by the replication API is `CompatibilityMode=FALSE`. The application is started without NWBC for HTML. • *Compatible:* If the application specific value for the integration mode is set to `Compatible`, the value which is returned by the replication API is `CompatibilityMode=TRUE`. The application is started with NWBC for HTML. • *System Default:* If the customizing value is not configured, the *Direct* mode is used as default value to start the application.	Mandatory

7. Choose *Check selected Launchpad App Descriptor Items* to ensure you have completed all mandatory fields.

Next Steps

After you have maintained the navigation-related fields, you can continue adding tiles for the visualization of your application if needed. See Maintaining Tiles [page 156].

Related Information

Setting Up Technical Catalogs with the SAP Fiori Launchpad App Manager [page 132]
Translate Texts from Launchpad App Manager [page 411]

1.3.4.6.1.5.1 Maintaining Tiles

Add one or more tiles to your launchpad app descriptor item.

After you have maintained your launchpad app descriptor item in the SAP Fiori launchpad app manager, you can map one or multiple tiles by choosing ▌▶ *Tiles* ❯ *Add Tile* ▌ in the lower window.

> i Note
>
> Mapping multiple tiles to a launchpad app descriptor item is only supported for the following app types:
>
> - SAPUI5 Fiori App
> - SAPUI5 Fiori App on SAP BTP

The following tile types can be created:

- Static Tiles [page 157]
- Dynamic Tiles [page 158]
- News Tiles [page 162]

To create and configure a tile using a custom tile type, you use the launchpad designer. See Implementing a Custom Tile Type [page 578].

Related Information

Maintaining Launchpad App Descriptor Items [page 148]
Maintaining Parameters [page 164]
Tiles [page 105]

1.3.4.6.1.5.1.1 Static Tiles

Static tiles are used to launch applications from the home page. This topic describes which parameters you need for configuring a static tile.

Procedure

1. In the *Tiles* tab, choose ▶ *Add Tile* ❯ *Static* ▮.
2. Maintain the following configuration parameters:

Setting	Description
Title	Enter a title that will be displayed on the tile.
Tile Subtitle	Enter a subtitle that will be displayed on the tile.
Tile Information	Enter additional information that will be displayed at the bottom of the tile.
Tile Keywords	Tag a tile with keywords so users can find it more easily using the search function in the tile catalog.
	If you want to specify several keywords, use either space or comma as separator.
Tile Icon	Select an icon by using the value help.
	After you have selected an icon, the system inserts a URL starting with the sap-icon:// protocol.
	These icons are based on SAPUI5 icon fonts. For more information, see .

Related Information

Tiles [page 105]

1.3.4.6.1.5.1.2 Dynamic Tiles

In contrast to regular (static) tiles, the information that is displayed on a dynamic tile can be pulled from a back-end system using an OData service. This topic describes the parameters for configuring dynamic tiles.

Procedure

1. In the launchpad app manager, open your technical catalog.
2. In the *Tiles* tab of your technical catalog, choose ▶ *Add Tile* ❯ *Dynamic* ▶ .
3. Configure the following parameters:

Configuration Parameters	Description
Title	Enter a title that will be displayed on the tile.
Tile Subtitle	Enter a subtitle that will be displayed on the tile.
Tile Information	Enter additional information that will be displayed at the bottom of the tile.
Tile Keywords	Tag a tile with keywords so users can find it more easily using the search function in the tile catalog.
	If you want to specify several keywords, use either space or comma as separator.
Tile Icon	Select an icon by using the value help.
	After you have selected an icon, the system inserts a URL starting with the sap-icon:// protocol.
	These icons are based on SAPUI5 icon fonts. For more information, see .
Unit of Value	Enter the unit to be displayed below the number for example, USD.
Refresh Interval in Seconds	Enter a number of seconds after which dynamic content is read from the data source again and the display is refreshed.
	The launchpad uses a default of 0 seconds unless you enter a higher value in this field.
	The launchpad will fall back to 10 seconds, if the value entered is > 0 and < 10. If the value entered is 0, the dynamic tile is updated only once on load. If the value is >= 10, the value is taken as refresh interval. The maximum value for the refresh interval is 2147483 seconds.

Configuration Parameters	Description
Service URL	Enter a URL of an OData service from which data should be read.
	The response is expected in JSON format.
	When the service is called, the values that are provided by the service override the values that have been configured manually in the tile details.
	Note that the service is executed on the home page only. On the admin page, sample data is displayed instead.

> → Tip
>
> To read the number of entities from an OData service, use the $count parameter in the *Service URL*. If you are using a filter to limit the number of entries, use the $inlinecount=allpages parameter in the *Service URL* to display both the number after the filter and the number from the m:count property.

For more information about the service API, see Response Values for Dynamic Tiles [page 213].

> i Note
>
> If the response is a numerical value over 1,000, the front end scales the number accordingly and adds a unit indicator (K, M, or B).
>
> Note that values higher than 999B are not supported and might result in misleading values.

Configuration Parameters	Description
Data Source Path **(Not Recommended)**	Enter a URL of an OData service from which data should be read.
	For back-end catalogs, the service URL can be split into the fields *Service URL* and *Data Source Path*. During the replication, the fields are concatenated and represent the service URL.

> ✢ Example
>
> **Service URL:** `/sap/opu/odata/sap/`
>
> **Data Source Path:** `E2EIE/PlansSet/$count`
>
> **Runtime:** `/sap/opu/odata/sap/E2EIE/PlansSet/$count`

We recommend that you enter the complete service URL in the *Service URL* field.

4. Choose *Save*.

1.3.4.6.1.5.1.2.1 Response Values for Dynamic Tiles

Learn how to create the OData service for a dynamic app launcher tile.

When working with the classic home page, the dynamic information that is pulled from the back-end overwrites the configuration values given in the tile configuration. Some tile functionality (for example, the state arrow) can only be configured by dynamic information (and not in the tile configuration).

When working with spaces and pages, the configuration values are saved independently from the response values. If some response values are undefined, the configuration value is used. Empty strings should not be used as this might have side effects. If you want to have an empty value instead of the configuration value, you define the response value <RESET>. This will not change the configuration value.

To feed an app launcher tile with dynamic content, you have to create an OData service that returns the configuration properties as in the following example structure:

```
{
                "d": {
                                "icon": "sap-icon://travel-expense",
                                "info": "Quarter Ends!",
                                "infoState": "Critical",
                                "number": 43.333,
                                "numberDigits": 1,
                                "numberFactor": "k",
                                "numberState": "Positive",
                                "numberUnit": "EUR",
                                "stateArrow": "Up",
                                "subtitle": "Quarterly overview",
                                "title": "Travel Expenses"
                }
}
```

The launchpad supports OData V2 and OData V4 for dynamic tiles. In general, OData V4 works similar to V2. The only relevant difference is that the additional "d" level is skipped in the response payload's JSON. So for example, `oResponseData.d.number` is simplified to `oResponseData.number`. The above payload example looks like this for OData V4:

```
{
    "icon": "sap-icon://travel-expense",
    "info": "Quarter Ends!",
    "infoState": "Critical",
    "number": 43.333,
    "numberDigits": 1,
    "numberFactor": "k",
    "numberState": "Positive",
    "numberUnit": "EUR",
    "stateArrow": "Up",
    "subtitle": "Quarterly overview",
    "title": "Travel Expenses"
}
```

Properties

Property	Description	Default Values
icon	Enter an `sap-icon://` URL, for example **sap-icon://cart**. You can look up the names of the available icons in tile configuration. For more information, see Static App Launcher Tiles [page 207].	""
info	Text to be displayed at the bottom of the tile.	""
infoState	The color of the tile is adapted according to the value of this property. The precise color depends on the theme that you have selected in UI theme designer. Allowed values: **Negative**, **Neutral**, **Positive**, **Critical**. Also see enum sap.m.ValueColor	ValueColor.Neutral
number	Number to be displayed in the top right corner of the tile.	"..."
numberDigits	Number of digits to be displayed following the decimal separator (decimal point or decimal comma, depending on the language settings).	4
numberFactor	A factor for scaling numbers, for example, for displaying large numbers like 1.000.000 (-> number = 1 and numberFactor="M") or for percentages (number = 22.2 and numberFactor = "%").	""
numberState	The color of the number is adapted according to the value of this property. The precise color depends on the theme that you have selected in UI theme designer. Allowed values: **Negative**, **Neutral**, **Positive**, **Critical**. Also see enum sap.m.ValueColor	ValueColor.Neutral
numberUnit	Unit to be displayed below the number, for example, **USD**.	""
stateArrow	Displays an arrow indicating a trend. Allowed values: **None**, **Up**, **Down**. Also see enum sap.m.DeviationIndicator	DeviationIndicator.None
subtitle	Subtitle to be displayed below the tile title.	""

Property	Description	Default Values
targetParams	List of key-value-pairs separated by ampersands.	""
	When the application is launched (by clicking on it), these parameters are passed to the application as business parameters (if semantic object-based navigation is used) or as URL parameters (if URL-based navigation is used).	
	If any parameters have been entered in the *Parameters* field in the tile configuration, the parameters passed by the OData service are appended to the list of parameters to be passed to the application.	
targetURL	URL that opens the app.	""
title	Title to be displayed in the tile.	""

> **i Note**
>
> If you want to show aggregated numbers, the server has to aggregate the numbers and provides the result in the JSON format described here. Also see OData Aggregation.

Related Information

1.3.4.6.1.5.1.3 News Tiles

Context

Configure a custom news tile which can monitor and retrieve news articles from up to 10 RSS feeds.

Procedure

1. In the *Tiles* tab, choose ▌ *Add Tile* ❯ *News Tile* ❭.
2. To configure your news tile, click *Open Custom Tile Editor* and maintain the following configuration parameters:

Configuration Parameters	Description
NewsTile Title	Enter a title for your *News Tile*.
Tile Default Image	Enter a URL to set the default image for the *News Tile*.
	You can set this parameter to select an alternate default image to display on the *News Tile*. By default, the *News Tile* application provides 12 default images and the *News Tile* cycles through these default images in sequence.
Always Use Default Image	When this checkbox is set, the *News Tile* ignores any image that accompanies with the RSS feed article. The following order of precedence is used:

	- **When Always Use Default Image is not set**
	  - Image from the RSS Article (if present)
	  - Image from the RSS Channel (if present)
	  - Image from Tile Default Image (if set)
	  - Image from one of the twelve (12) default images
	- **When Always Use Default Image is set**
	  - Image from Tile Default Image (if set)
	  - Image from one of the twelve (12) default images

Article Cycle Interval (secs)	Enter an integer (minimum is 5). The default value is 7000.
	This parameter controls the rate at which the articles cycle through the *News Tile*.
Article Refresh Interval	Select a refresh interval from the dropdown box. The default value is 15 minutes.
	This parameter controls the rate at which the *News Tile* requests new articles from the *Article Feeds*.
Article Feeds	Enter the http URL of the RSS feed. You can configure 10 RSS feeds. The *News Tile* monitors the RSS feeds and retrieves new articles based on the *Article Refresh Interval*.
	i Note If the URL references an external feed, the feed must be `CORS` compliant. If the URL references an internal feed, the feed must originate from the same server and port as the SAP Fiori launchpad.
Inclusion Filters	You can configure 5 inclusion filters by entering a filter text. The *News Tile* filters the feeds and includes any articles that contain the same text in the *Title* of the article.
Exclusion Filters	You can configure 5 exclusion filters by entering a filter text. The *News Tile* filters the feeds and excludes any articles that contain the same text in the *Title* of the article.

3. Click *Done* to save your changes.

1.3.4.6.1.5.2 Maintaining Parameters

Define mandatory and optional parameters for the target mapping of your launchpad app descriptor item.

Users can set the parameters in ▌▶ *Parameters* ❯ *Add Parameters* ❯ in the lower window of the SAP Fiori launchpad app manager.

When defining a parameter, the following rules apply:

- Parameter names must be unique.
- For a mandatory parameter, you can enter a string value or select the value *Regular Expression* in *Filter Type* to indicate that the value is a regular expression.
- For an optional parameter, you can define a default value.
- For the URL parameter name, enter the target name as it is used by the target application (optional).

Select the *Additional Parameters* checkbox to allow passing additional parameters that are not defined in the table to the target.

Configure Default Values

These parameters enable applications to be launched with user-specific default values rather than fixed values. These default values can be overridden, for example if the previous application or the tile that launches the application explicitly passes another value.

> **i Note**
>
> You can't use space in a parameter default. To insert a space, you must enter **/space**.

To configure a default value, enter the following parameters:

Parameter	Value
Parameter Name	Enter a unique name for your parameter.
Default Value	Set your preferred single default value.
Filter Type	The filter type *Value* is automatically filled.
Rename To	Rename the parameter. If for example, the parameter name is `ReceivingCostCenter` and you rename it to `CostCenter`, the application opens with the `CostCenter` parameter.

Configure Filter Values

To configure a filter value, enter the following parameters:

Parameter	Value
Parameter Name	Enter a unique name for your parameter.
Mandatory	This checkbox must be ticked if you want to configure a filter value.
Filter Value	Enter a string value or select *Regular Expression* in *Filter Type*.
Filter Type	Choose between the following: • *Value* - to define a value • *Regular Expression* - to indicate that the value is a regular expression
Rename To	Rename the parameter. If for example, the parameter name is `ReceivingCostCenter` and you rename it to `CostCenter`, the application opens with the `CostCenter` parameter.

Related Information

Maintaining Launchpad App Descriptor Items [page 148]

1.3.4.6.1.6 Adapting Launchpad App Descriptor Items

Adapt launchpad app descriptor items in the technical catalog.

Context

> i Note
>
> When you set up your launchpad content, we recommend following our best practices and reference SAP-delivered launchpad app descriptor items catalogs in business catalogs. See Basic Scenario - Use SAP Content as Delivered [page 121].
>
> The SAP-delivered launchpad content, however, is **read-only**. If you require changes to the launchpad content that cannot be done in the adaptation mode of the launchpad app manager, you need to copy the launchpad app descriptor items to a custom technical catalog. See Advanced Scenario - Adapt SAP Template Content and Add Your Own Content [page 123].

The adaptation mode in the launchpad app manager allows you to make a few adaptations in the launchpad app descriptor items of SAP-delivered technical catalogs and directly reference to them from custom business

catalogs. This has the advantage that you don't have to create a custom technical catalog in order to make changes.

The changes you make apply to all clients (cross-client adaptation).

It is possible to adapt the following fields:

Group of Fields	Field
Target Application Fields	• Target Application Title • Target Mapping Information • Device Type - Desktop, Tablet, Phone
Tile Fields	• Tile Title • Tile Subtitle • Tile Icon • Tile Keywords • Tile Information • Refresh Interval in Seconds

Procedure

1. Open the transcation `/UI2/FLPAM_ADAPT` to directly launch the adaptation mode in the launchpad app manager. You can also launch the launchpad app manager and then choose *Switch to Adaptation Mode* on top of the application. See Running the Launchpad App Manager [page 139].
   A banner on top of the application indicates that you're in the adaptation mode.
2. Open the technical catalog that you want to adapt and choose *Edit*.
3. Select the relevant launchpad app descriptor item from the list and in the details view, navigate to the *Adaptation* tab.
4. Now you see a list of fields that you're allowed to adapt. Mark the checkbox in the *Adapt* column of a field that you want to change.
5. In the column *New Field Value*, enter the new value for your field.
6. Save your changes.
   After you have saved the changes, you can check and release the corresponding transport request in transaction SE09.
7. Now you can continue with Adapt Fields in Target Languages [page 167] or Translate Adapted Fields [page 167].

Find and Reset Adapted Fields

If a field has been adapted, it is indicated by the suffix **(Adapted)**.

To find adapted fields, use the column *Adapted* on catalog level or on launchpad app descriptor level. If you don't see this field, you can add it by using the ⚙ (Settings) button.

To reset an adapted field, deselect the checkbox in the *Adaptation* tab of your launchpad app descriptor item and save your changes. You can also choose *Reset* and undo the changes that you made in the adaptation mode.

Adapt Fields in Target Languages

Adaptations can be done in other target languages in the launchpad app manager directly. For that, log in to the respective target language and adapt the fields as described above. The launchpad app manager is available in four standard language versions: **English**, **German**, **French**, and **Japanese**. For any other language, the fallback is **English**.

The icon ⓘ in the *Information* column of the *Adaptation* tab indicates in which languages the adapted text is available. The languages are displayed in the tooltip when you hover over the icon.

If you have made the adaptations only in your source language, but not in another target language, then the original texts (not the adapted texts) will be displayed for the other languages.

Example 1

1. You change the tile title *Title* to *Title A* in your source language **EN**.
2. You log into the app manager with language **DE** and change the title *Titel* to *Titel A*.

Result: The adapted titles *Title A* for **EN** and *Titel A* for **DE** are shown in your application.

Example 2

1. You change the tile title *Title* to *Title A* in your source language **EN**.
2. You don't change the title in your target language **DE** and leave the original title *Titel*.

Result: For **EN**, the adapted title *Title A* is displayed in your application. For **DE**, the original title *Titel* is shown.

Translate Adapted Fields

Use the *Translation Editor* (transaction SE63) to translate adapted fields into another target language:

1. Switch to the adaptation mode in the launchpad app manager.
2. Select the relevant launchpad app descriptor item.
3. Choose *Translate Tile Texts* to open the selected object for translation in the short text editor (transaction SE63). Object name **SUI_UIAAT** and source and target language are already set.
4. Choose *Edit*.
5. Enter the ID of the relevant launchpad app descriptor item and choose *Execute*.
   This takes you to the short text editor. See Short Text Editor.

Related Information

Advanced Scenario - Adapt SAP Template Content and Add Your Own Content [page 123]

1.3.4.6.1.7 Copying Launchpad App Descriptor Items From Another Catalog

Copy launchpad app descriptor items from another technical catalog.

Context

To add launchpad app descriptor items to a catalog, you first search for a catalog and then select the launchpad app descriptor items you want to copy.

Procedure

1. In the launchpad app manager, open the catalog to which you want to add the launchpad app descriptor items.
2. Click *Edit*.
3. In the toolbar in the upper window, click *Copy from Other Catalog*.
4. In the dialog box, search for the catalog from which you want to copy your launchpad app descriptor items. Then select one or multiple launchpad app desscriptor items you want to add.
5. Choose *Copy*.

   The launchpad app descriptor items you selected are copied to your catalog.

1.3.4.6.2 Setting Up the Replication of Back-End Catalogs

You need to replicate launchpad app descriptor items created in a back-end component to the front-end server.

To enable the access from the SAP Fiori launchpad to applications using Web Dynpro, SAP GUI for HTML and WebClient UI technology, you have to replicate information about these applications to the front-end server. Each application has a launchpad app descriptor which provides this information.

SAP delivers predefined launchpad app descriptors for classic UIs (applications based on Web Dynpro, SAP GUI for HTML and WebClient UI technology) in back-end components. You can also create your own custom launchpad app descriptors in the backend.

> **i Note**
>
> The replication step is also required in an embedded deployment scenario.

Only after the replication, you can see the catalogs in the launchpad design-time tools.

Replication in a Standalone (Hub) Scenario

The following graphic shows the replication setup in a standalone (hub) scenario:
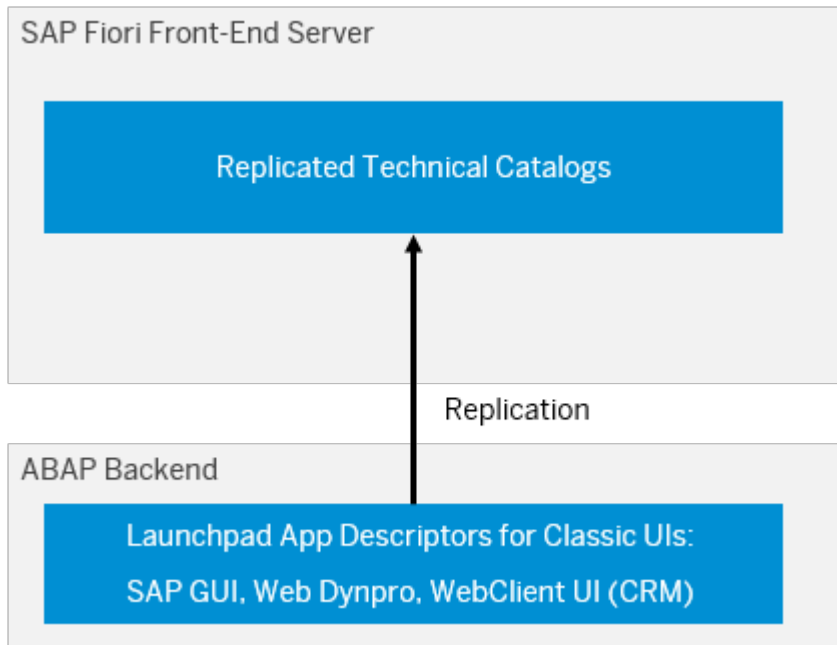
Launchpad app descriptors for classic UIs are created in the ABAP backend.

During the replication step, technical catalogs containing these launchpad app descriptor items are created on the front-end server. The replicated catalogs cannot be edited or assigned to an authorization role.

The launchpad app descriptors can be referenced from business catalogs.

Replication in an Embedded Scenario

The following graphic shows the replication setup in an embedded scenario:
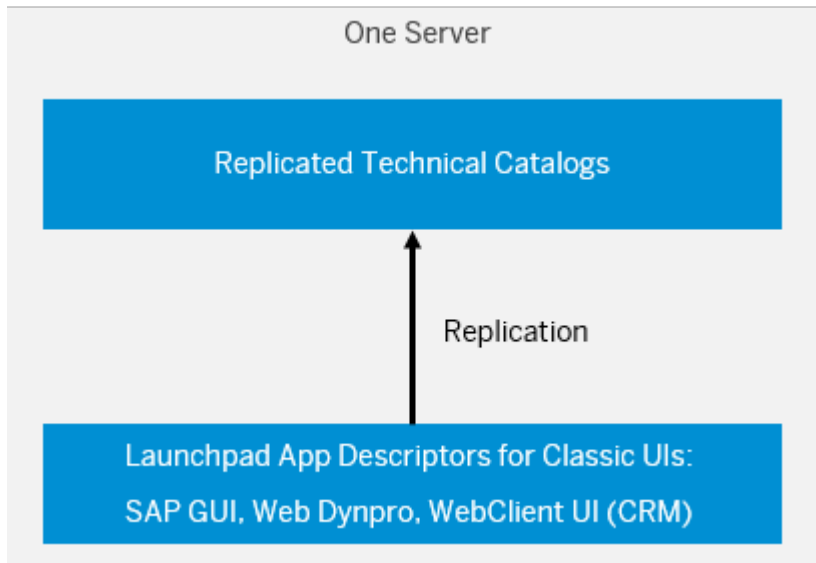
One Server

Replicated Technical Catalogs

↑

Replication

Launchpad App Descriptors for Classic UIs:
SAP GUI, Web Dynpro, WebClient UI (CRM)

Launchpad app descriptor items are created on the front-end server and the system alias mapping points to the local front-end server.

During the replication step, the data of the technical catalogs is read directly from original technical catalog. A copy of the technical catalog as WDCC object is no longer created.

Procedure

Perform the following steps on the front-end server:

1. Establish connections to back-end systems.
   If you add your launchpad application descriptor items to an existing catalog, you can skip this step.
   See Establish Connections between Front-End Server and Backend [page 15].
2. Assign system aliases to the technical catalog.
   If you add your launchpad application descriptor items to an existing catalog, you can skip this step.
   See Maintain Catalogs to Replicate [page 171].
   In an embedded scenario, your catalogs need to be defined as **local**.
   See Define System Alias Mapping as Local [page 172].
3. Replicate the catalog to the front-end server.
   See Replicating Back-End Catalogs [page 173].
4. You can also schedule the replication reports to run periodically.
   See Schedule the Replication of Back-End Catalogs [page 179].

Results

- In a hub scenario: As a result of the replication, read-only technical catalogs are created on the front-end server.
  Naming convention:
  ```
  X-SAP-UI2-ADCAT:<ID of technical catalog in BE>:<System Alias of BE>
  ```
  Example: X-SAP-UI2-ADCAT:SAP_TC_FIN1:S4FIN.

- In an embedded scenario: As a result of the replication, the data is read form the original catalog and no copy as WDCC object is needed. In addition, no subsequent replications are needed after the initial replication.

1.3.4.6.2.1 Maintain Catalogs to Replicate

To ensure that catalogs are only replicated for specific system aliases, you assign the relevant system alias to the catalog.

Context

A backend system connected to a frontend system usually contains technical catalogs assigned to different software components. The catalogs are replicated to the frontend server for a particular system alias which is usually associated with a software component. For example SAP S/4HANA catalogs associated with Financials are assigned to system alias S4FIN. The IDs of the replicated catalogs have the following syntax: `<catalog ID>:<system alias>`

Procedure

Perform the following steps on the front-end server:

1. Start *Maintain Table Views* (transaction `SM30`)
2. In the *Table/View* field, enter the view `/UI2/VC_ALIASCAT`.
3. Enter the relevant system alias for your catalog.
4. Mark the check box in *Skip Replication* if you don't want to replicate the catalog.

Example

You have two technical catalogs assigned to different software components:

- SAP_TC_FIN1
- SAP_TC_LO1

To define that catalog SAP_TC_FIN1 is replicated for system alias S4FIN and catalog SAP_TC_LO1 for system alias S4LO, you enter the following entries in the table:

Catalog ID	System Alias
SAP_TC_FIN1	S4FIN

Catalog ID	System Alias
SAP_TC_LO1	S4LO

The replicated catalogs will have the following IDs:

- SAP_TC_FIN1:S4FIN
- SAP_TC_LO1:S4LO

The mapping of the catalogs to the system aliases avoids duplicates like the following:

- SAP_TC_FIN1:S4FIN
- SAP_TC_FIN1:S4LO
- SAP_TC_LO1:S4FIN
- SAP_TC_LO1:S4LO

Related Information

Setting Up Technical Catalogs with the SAP Fiori Launchpad App Manager [page 132]

1.3.4.6.2.2 Define System Alias Mapping as Local

Define system alias mapping as **local** to replicate the back-end catalogs in an embedded scenario.

Prerequisites

> i Note
>
> This step is only supported in an embedded scenario.

- You have established connections to back-end systems. See Establish Connections between Front-End Server and Backend [page 15].

Context

During the replication step in an embedded scenario, the data can be read directly from the original technical catalogs. There is no need to copy the technical catalog as a WDCC object. For this method to work, the technical catalog needs to exist on the front-end server and the system alias mapping should point to the local front-end server.

Procedure

1. Perform the steps as described in Map System Aliases to RFC Connections [page 17].
2. A local back-end catalog is defined as **local** when the following conditions are met:

Field	Value
Client	Leave this field empty
Source System Alias	Enter the relevant system alias
Target System Alias	Leave this field empty

A back-end catalog is **not** defined as local if the catalog system alias points directly to an entry in *Configuration of RFC Connections* (transaction SM59) even though it points to the local system.

Results

As a result, the replication reports as mentioned in Replicating Back-End Catalogs [page 173] will read the data and then perform a page cache synchronization for local back-end catalogs.

As a next step, continue with:

- Replicating Back-End Catalogs [page 173]
- Schedule the Replication of Back-End Catalogs [page 179]

1.3.4.6.2.3 Replicating Back-End Catalogs

To make the back-end catalogs visible in the launchpad and in the launchpad design-time tools, you need to replicate them to the frontend server.

The following replication reports are offered to fit different use cases:

Report	Description	More Information
/UI2/GET_APP_DESCR_REMOTE_ALL	Allows you to replicate technical catalogs for all maintained system aliases at once. This report is suitable if you want to reuse the whole set of technical back-end catalogs provided by SAP.	Replicate Technical Catalogs for all Available System Aliases [page 174]
/UI2/GET_APP_DESCR_REMOTE	Allows you to only replicate the technical catalogs which are referenced by business catalogs.	Replicate Technical Catalogs per Business Catalog [page 178]

Report	Description	More Information
/UI2/GET_APP_DESCR_REMOTE_DEV	Allows you to replicate technical catalogs using a specified system alias. This report is suitable if you have created single custom technical catalogs in the backend.	Replicate Technical Catalogs for a Single System Alias [page 175]

Related Information

Setting Up Technical Catalogs with the SAP Fiori Launchpad App Manager [page 132]

1.3.4.6.2.3.1 Replicate Technical Catalogs for all Available System Aliases

Replicate technical catalogs for all maintained system aliases at once.

Context
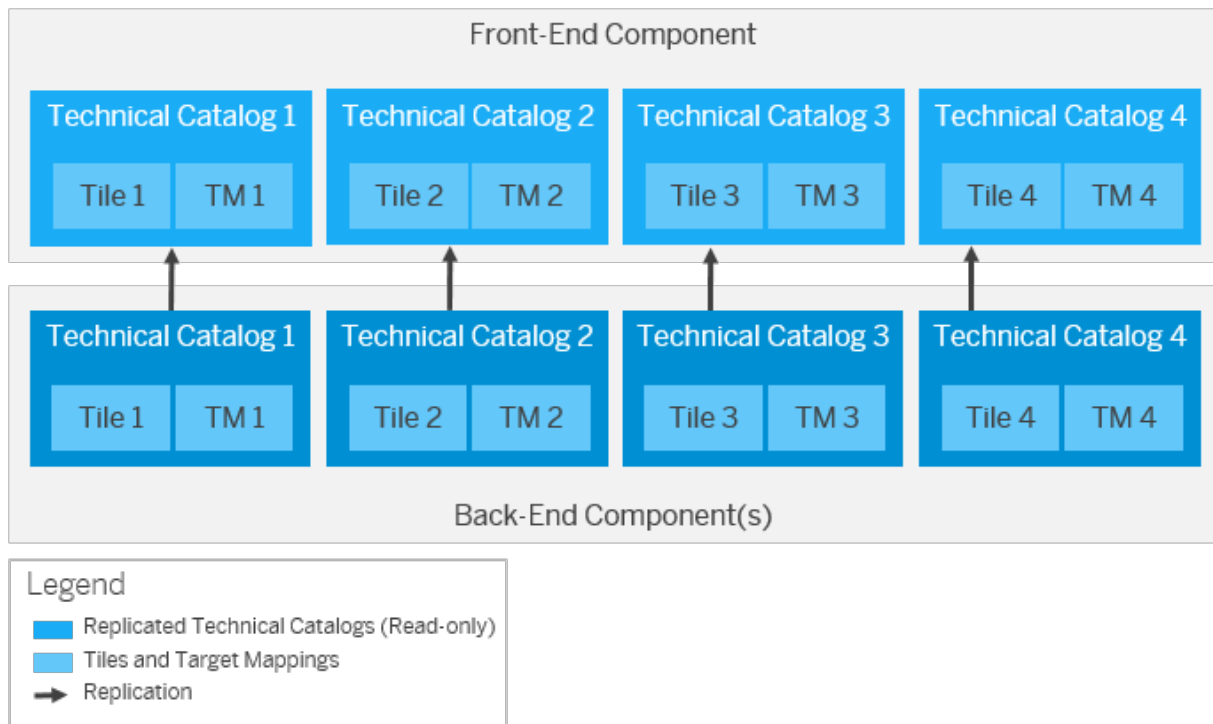
The technical catalogs are replicated for all system aliases maintained in the table /UI2/V_SYSALIAS.

This report is suitable if you want to reuse the whole set of technical back-end catalogs provided by SAP.

Procedure

1. Launch the report `/UI2/GET_APP_DESCR_REMOTE_ALL`.
2. Choose a *Replication Mode*:

   - *Delta replication*: The selected catalogs are only replicated if changes were made since the last replication.
   - *Full replication*: All selected catalogs are replicated.

3. (Optional) Mark the *Add Transaction Code to Tile* checkbox if you want to have the transaction code displayed in the tile. The transaction code of the SAP GUI transaction would then be automatically retrieved and displayed in the *Tile Information* field. Note that if the *Tile Information* field is already filled, the information won't be overwritten.
4. Mark the *Test mode* checkbox and choose *Execute*. The catalogs will not be replicated.

   For catalogs that could not be replicated some error information is displayed in the application log.

5. If the log does not contain any errors, deselect the test mode and run the report.

1.3.4.6.2.3.2 Replicate Technical Catalogs for a Single System Alias

Replicate technical catalogs to the front-end server using a single system alias.

Prerequisites

You are assigned to authorization role `SAP_FLP_ADMIN`. See Configuring Roles with Launchpad Start Authorizations [page 327].

Context

This report allows you to replicate technical catalogs using a specified system alias.

Front-End Component

Technical Catalog 1

Tile 1 TM 1

Technical Catalog 1

Tile 1 TM 1

Back-End Component

Legend
Replicated Technical Catalog (Read-Only)
Tiles and Target Mappings
Replication
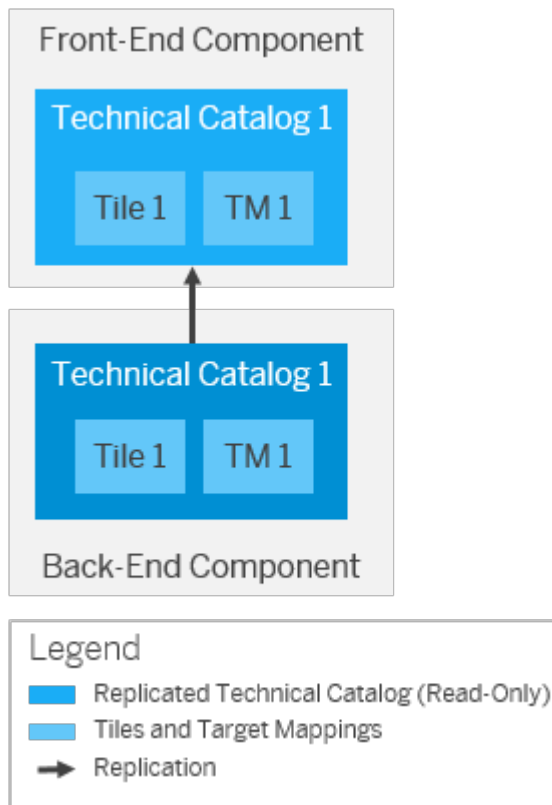
The user needs to specify a system alias. The technical catalogs for which the system alias is maintained are replicated from the source system to the target system. The source and target systems are defined in the system alias mapping table /UI2/V_ALIASMAP. In an embedded scenario, the source and target system would be identical.

This report is suitable if you have created single custom technical catalogs in the backend.

Procedure

Perform the following steps on the front-end server:

1. Launch the program `/UI2/GET_APP_DESCR_REMOTE_DEV`.

2. In the *System Alias* field, enter the name of the system alias for which the catalogs should be replicated.

   This is an alias maintained in view `/UI2/V_SYSALIAS` or `/UI2/VC_SYSALIAS`. See Establish Connections between Front-End Server and Backend [page 15].

3. In the *Catalog ID* field, enter the name of the technical catalog which you want to replicate.

4. Choose a *Replication Mode*:

   • Delta replication: The selected catalogs are only replicated if changes were made since the last replication.
   • Full replication: All selected catalogs are replicated.

5. (Optional) Mark the *Add Transaction Code to Tile* checkbox if you want to have the transaction code displayed in the tile. The transaction code of the SAP GUI transaction would then be automatically

retrieved and displayed in the *Tile Information* field. Note that if the *Tile Information* field is already filled, the information won't be overwritten.

6. Mark the *Test mode* checkbox and choose *Execute*.

   The catalogs will not be replicated. A log is displayed.

7. If the log does not contain any errors, unmark the *Test mode* checkbox and choose *Execute*.

Results

On the front-end server, the program creates a technical catalog with the following naming convention:

`X-SAP-UI2-ADCAT:<ID of technical catalog in BE>:<System Alias of BE>`

Example: X-SAP-UI2-ADCAT:SAP_TC_FIN1:S4FIN.

Based on the application descriptor which you defined in the launchpad app manager(see Maintaining Launchpad App Descriptor Items [page 148]), each application contained in the backend catalog produces the following:

- An app launcher tile containing the intent, the texts (title, subtitle, keywords and information) and the icon maintained in the launchpad app manager
- A target mapping containing the intent, the parameters and the title maintained in the launchpad app manager. The HTTP connection, which is associated with the system alias is entered in the *System Alias* field.

You can display the replicated catalogs in the launchpad designer but you cannot edit them there since the original data is located in the back-end.

You create business catalogs based on the technical catalogs. See Setting Up Technical Catalogs with the SAP Fiori Launchpad App Manager [page 132].

Related Information

Setting Up Technical Catalogs with the SAP Fiori Launchpad App Manager [page 132]
Setting Up the Replication of Back-End Catalogs [page 168]

1.3.4.6.2.3.3 Replicate Technical Catalogs per Business Catalog

Replicate technical catalogs to the front-end server using business catalog IDs.

Prerequisites

You are assigned to authorization role SAP_FLP_ADMIN. See Configuring Roles with Launchpad Start Authorizations [page 327].

Context

This report allows you to replicate technical catalogs which are referenced by a business catalog.
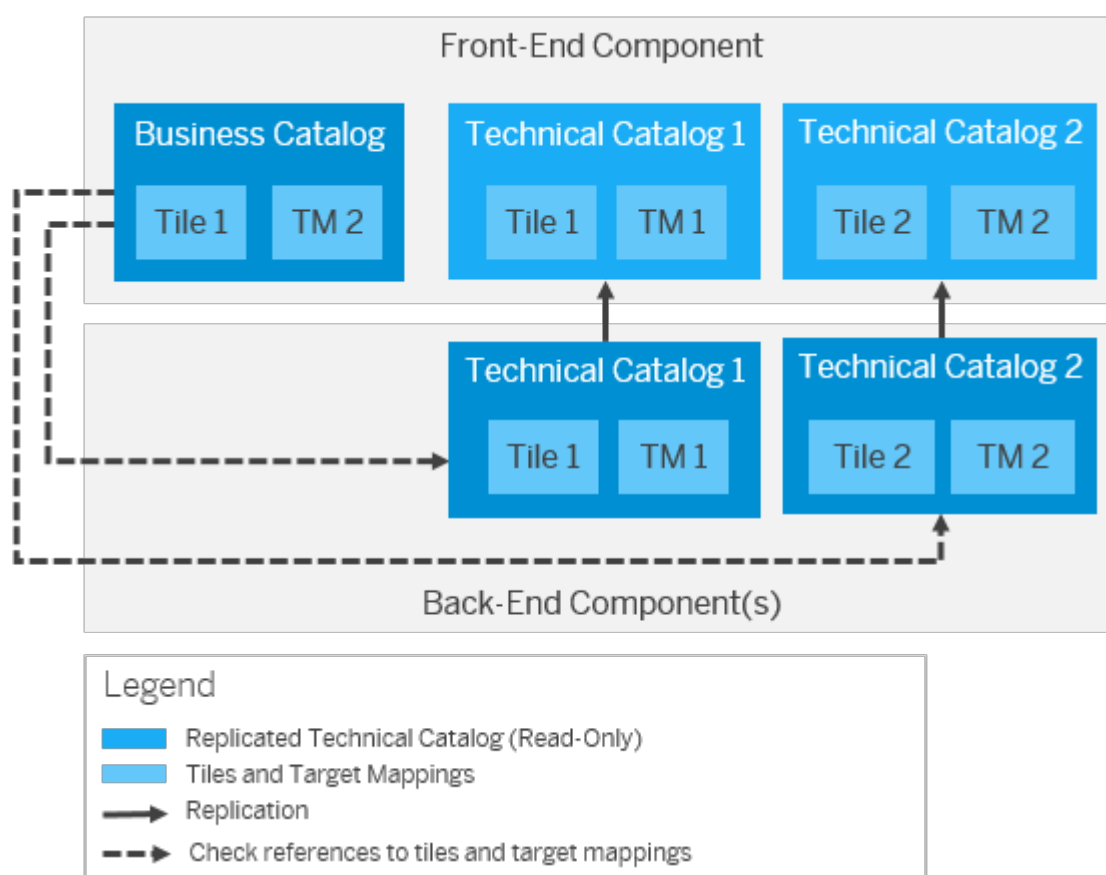
The user can specify a business catalog ID. The report checks the business catalog to see which referenced tiles and target mappings it contains. In a next step, the relevant technical catalogs which contain the original tiles and target mappings are replicated to the front-end server.

This report is suitable if you want, for example, to reuse a specific business catalog that you have found in the SAP Fiori apps reference library.

Procedure

Perform the following steps on the front-end server:

1. Launch the program `/UI2/GET_APP_DESCR_REMOTE`.
2. (Optional) In the *Business Catalog ID* field, enter the ID of the business catalog for which the catalogs should be replicated. If you leave this field blank, all technical catalogs referenced by any business catalog available in this system will be replicated.
3. Choose a *Replication Mode*:

   - Delta replication: The selected catalogs are only replicated if changes were made since the last replication.
   - Full replication: All selected catalogs are replicated.

4. (Optional) Mark the *Add Transaction Code to Tile* checkbox if you want to have the transaction code displayed in the tile. The transaction code of the SAP GUI transaction would then be automatically retrieved and displayed in the *Tile Information* field. Note that if the *Tile Information* field is already filled, the information won't be overwritten.
5. Mark the *Test Mode* checkbox and choose *Execute*.

   The catalogs will not be replicated. A log is displayed.
6. If the log does not contain any errors, unmark the *Test mode* checkbox and choose *Execute*.

1.3.4.6.2.4 Schedule the Replication of Back-End Catalogs

Use *Define Job* (transaction SM36) to schedule the replication of technical catalogs.

1. Launch *Define Job* (transaction SM36).
2. In the *Job Name* field, enter one of the following reports:

   - `/UI2/GET_APP_DESCR_REMOTE_ALL`
   - `/UI2/GET_APP_DESCR_REMOTE_DEV`
   - `/UI2/GET_APP_DESCR_REMOTE`

1.3.4.6.3   Deleting Back-End and Replicated Catalogs

To delete catalogs you created in the backend using the launchpad app manager and replicated to the frontend, you first delete the catalog in the back-end server and then the replicated catalog in the frontend server.

> **i Note**
>
> If you delete a catalog, make sure that its content is not referenced by any other catalog anymore. See References and their Originals [page 244].

Perform the following steps:

- Delete the back-end catalogs in the backend using the launchpad app manager. See Delete Single Back-End Catalogs Using the Launchpad App Manager [page 180].

- Delete the replicated catalog in the frontend using the report `/UI2/DELETE_APP_DESCR_CATS`. See Delete Replicated Catalogs for a Replication System Alias [page 181].

Related Information

Setting Up Technical Catalogs with the SAP Fiori Launchpad App Manager [page 132]

1.3.4.6.3.1 Delete Single Back-End Catalogs Using the Launchpad App Manager

Context

To delete a back-end catalog using the launchpad app manager, you only need to delete its content. Once the catalog has no content assigned anymore, it will be automatically deleted from the back-end server.

Procedure

Perform the following steps in the backend system:

1. Launch the transaction `/UI2/FLPAM`. See Running the Launchpad App Manager [page 139].
2. In the *Catalog ID* field, enter the ID of an existing catalog you want to delete.
3. Choose *Continue*.

   In the subsequent window, you can see the content that is assigned to your selected catalog.
4. Switch to *Edit* mode and select a transport request.
5. Select all line items in the presented table and click 🗑 (Delete).
6. Click *Save*.

   Your catalog has no content assigned anymore, when the table is empty. Your catalog in the backend system will be deleted without further action.

Next Steps

After you have successfully deleted the catalog in the back-end server, you need to delete its replicated version in the frontend server.

Related Information

1.3.4.6.3.2 Delete Replicated Catalogs for a Replication System Alias

Prerequisites

- Make sure that you have maintained the replication system alias in `/UI2/V_SYSALIAS or /UI2/VC_SYSALIAS`.
- Note that the entered alias name is used for the selection of the catalogs in the local system.

Context

If a catalog has been deleted in the back-end system, its replication still exists in the frontend server. To delete the replicated catalog in the frontend server, you need to perform specific actions in the report `/UI2/DELETE_APP_DESCR_CATS`.

The report offers the following features:

- Remove all replicated back-end technical catalogs for a given replication system alias.
- Remove single replicated back-end technical catalogs for a given catalog ID from a selected replication system alias.

> i Note
>
> Note that replication is a client-neutral process and deletion will affect all clients as catalogs are replicated and deleted from the configuration scope.

Procedure

By default, the *Test mode* checkbox field is ticked. Uncheck the field to make the changes effective. Perform the following steps:

1. In *ABAP Reporting* (transaction SA38), enter `/UI2/DELETE_APP_DESCR_CATS` in the *Program* field and choose (Execute).

   The report is launched.

2. In the *Replication System Alias* field, enter your system alias referring to the back-end system catalogs you want to delete.

    This function deletes all replicated catalogs for the replication system alias, if you skip **Step 3**.

3. Optionally, you can delete single replicated catalogs for a specified replication system alias:

    Enter the catalog ID in the *Back-end Technical Catalog ID* field.

4. Choose ⊕ (Execute).

Results

A list of the catalogs you deleted is presented to you. In case your selected catalogs couldn't be deleted successfully, the display log indicates it with a warning or with an error message. In that case, check your entered parameters.

Related Information

Setting Up Technical Catalogs with the SAP Fiori Launchpad App Manager [page 132]
Maintaining Launchpad App Descriptor Items [page 148]

1.3.4.6.4 Updating Content Created with the Launchpad Designer

You use the SAP Fiori launchpad designer to update existing groups, catalogs, tiles and target mappings.

Prerequisites

Before you can run the launchpad designer, you need to make several prerequisite configurations. See Initial Setup of the Launchpad Designer [page 186].

Feature Scope

The SAP Fiori launchpad designer should be used for the following tasks:

- Update existing catalogs, tiles and target mappings created with the launchpad designer.

  > i Note
  >
  > Creating catalogs, tiles and target mappings with the launchpad designer is no longer recommended. Instead, use the launchpad app manager. See Setting Up Technical Catalogs with the SAP Fiori Launchpad App Manager [page 132].

- Manage custom tiles
- Update existing groups to define the classic home page of the SAP Fiori launchpad.

> **i Note**
>
> Creating new home pages based on groups is no longer recommended. Instead, use the "Manage Launchpad Spaces" and "Manage Launchpad Pages" apps to define the launchpad layout and structure. See Managing Launchpad Spaces and Pages [page 269].
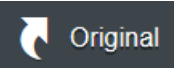
Browser Support

The launchpad designer is designed to optimally work on desktop. This application can be launched in any of the following browsers:

- Internet Explorer 9 and above
- Safari
- Mozilla Firefox
- Chrome

Icons Used in the Launchpad Designer

Icon	Name	Description
⚙	Settings	You use this to display the Transport Request Information dialog box.
⊕	Add Icon	You use this icon to add a tile to a group or catalog and create a catalog or group.
🗑	Trash Icon	You use this icon to delete a catalog, group, or delete tiles from a catalog or group.
+	Add Tile	You use this icon to add tiles to a catalog or group.
✓	Tile already added	You use this to indicate that the tile was already added to the group.
←	Back	Navigate to the home page.
⠿	Catalogs	You use this icon to display the list of available catalogs.
≡	Groups	You use this icon to display the list of available groups.
+	Add Tile Template	You use this to add the tile template to the group.
✏	Edit Title	You use this to edit the title of the existing catalog and group.

Icon	Name	Description
	Outdated	You use this to indicate that changes have been made in the Configuration mode and the Customization mode is outdated for catalog or group. This serves as an alert for the user to delete the catalog or group in the customization mode and then, refresh the page to get the latest changes made in the Configuration mode for the group or catalogs in the Customization mode.
	Information	You use this to display the additional information for the selected tile in the group. You get the following information: • Tile Instance ID • Catalog ID • Tile ID
	Copy	You use this to copy a catalog.
	Drop Zone	You use this to drag and drop the catalog in the *Catalog Collection* section.
	Search	You use this to search a catalog or a group.
	Unpin	You use this to unpin the catalog in the *Catalog Collection* section.
	Pin	You use this to pin the catalog in the *Catalog Collection* section.
	Display More	You use this to display all tiles of the selected catalog in the *Catalog Collection*.
	Display Associated Catalogs	You use to display associated catalogs for an app launcher and target mapping for an intent.
	Intent Confirm	You use this to display the status of an intent.
	Intent Alert	You use this to display the status of an intent.

Icon	Name	Description
✔	Accept	You use this to indicate that a particular configuration parameter in the target mapping table is true.
	Target Mappings	You use this icon to indicate a target mapping tab on the screen.
⚠	Alert	You use this to indicate outdated applications in the target mapping table.
✖	Decline	You use this to indicate that a particular configuration parameter in the target mapping table is false.
	Tiles	You use this icon to indicate a tiles tab on the screen.
	Tiles	You use this icon to indicate a tiles tab on the screen (list view).
Original	Original	You use this icon to navigate from a reference item to the original tile or target mapping.

Flow of Usage

The flow of usage for launchpad designer is mentioned below:

- Administrators would first use the launchpad designer to configure tiles, catalogs, groups, and transport them to target systems.
- Administrators would then communicate the various tile instances, catalog names, and group name to the PFCG administrator so that the tiles, catalogs, and groups can be assigned to roles.
- The administrator can then test the actual rendering of the groups in the launchpad runtime.

Related Information

1.3.4.6.4.1 Initial Setup of the Launchpad Designer

Before you can run the SAP Fiori launchpad designer, you need to make several settings for the configuration.

You use the SAP Fiori launchpad designer to manage catalogs, groups, and tiles.

Related Information

1.3.4.6.4.1.1 Configuring Login Screen for Launchpad Designer

The SAP Fiori login page is the customized standard ABAP login page. You can configure SAP Fiori login page to launch the SAP Fiori launchpad designer.

Context

The class /UI2/CL_FIORI3_LOGIN supports Content Security Policy (CSP). CSP is a mechanism web applications can use to mitigate a broad class of content injection vulnerabilities, such as cross-site scripting (XSS). See for more information. In addition, clickjacking framing protection is supported. You can find more information in Clickjacking Framing Protection [page 630].

To configure the login page, proceed as follows:

Procedure

1. In your ABAP front-end system, access transaction SICF and choose *Execute*. The *Maintain service* screen appears.

2. In the *Filter Details* section, enter `arsrvc_upb_admin` in the *ServiceName* field.

3. Navigate to ▶ *default_host* ❯ *sap* ❯ *bc* ❯ *ui5_ui5* ❯ *sap* ❯ *arsrvc_upb_admin* ❯ and double-click on
   `arsrvc_suite_pb`. A *Create/Change* screen appears.

4. Choose ✏️, navigate to ▶ *Error Page* ❯ *Logon Errors* ❯ tab.

5. Select `System Logon` radio button and choose *Configuration* button. A *System Logon Configuration* dialog
   box appears. Enter the following:

   a. Under *Settings Selection* section, select *Define Service-Specific Settings* radio button.

   b. Under *Logon Layout and Procedure* section, select *Custom Implementation* radio button and
      enter `/UI2/CL_FIORI3_LOGIN` in the *ABAP Class* field.

   c. To configure the login screen you have the following options:

Dialog Section	Settings
Default	You can define the default values for the client or the language that then are used for the launchpad startup.
Select Display	You can set fields that should be displayed. The user can then e.g. select the client that should be opened or the language that should be used. Click on the button *Adjust Links and Images* to define URLS that lead to a Register or Password Forgotten page. If the links are defined, they will be displayed in the login screen enhanced with a sap-return-url query parameter.

   a. Save your entries.

Results

The SAP Fiori logon screen is configured for the SAP Fiori Lauchpad Designer. Customizing the Login Screen
[page 21] shows you additional possibilities to adapt the login page.

1.3.4.6.4.1.2 Protecting Access to Launchpad Designer

You can restrict access to the launchpad designer at the OData service level.

The administrator using the launchpad designer should have the authorization role *SAP_UI2_ADMIN_700*
assigned.

The launchpad user must **not** have authorization to run */UI2/PAGE_BUILDER_CONF* and */UI2/
PAGE_BUILDER_CUST* OData service.

The launchpad user must have the authorization for:

- */UI2/INTEROP*

- *_/UI2/LAUNCHPAD_*
- *_/UI2/PAGE_BUILDER_PERS_*

For more information about authorizations, see http://help.sap.com/nw-uiaddon/ *Security Information* *Security Guide* *Authorizations* .

Related Information

Activating SAP Gateway OData Services [page 23]

1.3.4.6.4.1.3 Running the Launchpad Designer

After the initial setup has been done, you can run the launchpad designer.

The launchpad designer can be launched with the following URLs:

https://<server>:<port>/sap/bc/ui5_ui5/sap/arsrvc_upb_admn/main.html

https://<server>:<port>/sap/bc/ui5_ui5/sap/arsrvc_upb_admn/main.html ?sap-client=<client>&scope = <CONF/CUST>

We recommend the following regarding scope and client:

- For client-independent content, use the configuration scope (`CONF`) with client 000.
- For client-dependent content, use the customizing scope (`CUST`) with the client in which the content should be available.

The launchpad designer is started in **CUST** mode if the scope parameter is omitted.

The client information for the relevant scope is displayed in the launchpad designer header next to the ⚙ icon:

- A specific client (for example, 000) for the customizing scope
- *Client: ALL* for the configuration scope

If you have any problems with loading the page, see Troubleshooting for launchpad designer [page 655].

Related Information

Scopes for Adapting Launchpad Content [page 111]
Initial Setup of the Launchpad Designer [page 186]

1.3.4.6.4.2 Creating and Configuring Catalogs

This section provides information about catalogs.

What is a catalog?

A catalog is a collection of apps that are available for a specific role (or for multiple roles). Administators assign apps (via tiles and target mappings) to a catalog and catalogs to roles. Users that have a particular role assigned, have access to all the apps that are contained in the catalogs assigned to this role.

Related Information

1.3.4.6.4.2.1 Creating or Deleting Catalogs

Context

A catalog is a list of tiles and target mappings (actions). You have to launch the launchpad designer to create or delete the catalog. The catalog names along with the number of tiles appear on the left panel. The catalog has to be assigned to an authorization role in the back end to which the users can be assigned.

For more information on target mapping, see

Creating Catalog

1. Choose the *Create Catalog* add icon on the left panel.

   A *Create Catalog* dialog box appears.
2. Enter the title and the ID.
3. Choose *Save*.
   A catalog with the specified title without any tiles is created.

Deleting Catalog

1. Choose any catalog from the catalog list on the left panel.
2. Drag and drop the selected catalog in the trash zone.

   A *Confirmation* dialog box appears.
3. Choose *OK*.

   The selected catalog is deleted from the catalog list.

1.3.4.6.4.2.2 Copying Catalogs

When you copy a catalog, a new catalog containing references to the tiles and target mappings in the original catalog is created.
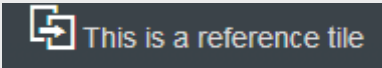
Copying catalogs allows you to adapt predefined catalogs delivered by SAP. You create a copy of the catalog using the configuration or the customizing scope and then adapt it to your needs using the customizing scope.

After copying the predefined content, any changes to the tile or target mapping configuration in the original tiles or target mappings are reflected in the references.

You can then make changes to the copied catalog using the customizing scope. The changes are then valid only for the selected client. For example, you can change the title of a catalog or the icons for app launcher tiles in a catalog. You can add tiles or change navigation targets.

You can make changes to the catalog (for example, changing the title or adding tiles) without breaking the reference of the tiles and target mappings. However, once you have made changes in the tile or target mapping configuration, the connection with the original content is lost. Any changes in the original for the parts that you have modified in the copy are no longer visible in the copied content. For example, if a tile changes in the original catalog after you made the copy, this change is still visible in the referenced tile for the copy, unless you have modified its tile configuration. Here, the reference is decoupled from the original.

> **i Note**
>
> - When you display the configuration screen of a tile of a copied catalog, you receive a message
>
>   ⤒ This is a reference tile   in the footer to indicate that you are in the reference tile.
> - You can make a copy of a copied catalog.

> **⚠ Caution**
>
> Be aware that a tile or target mapping in a copied catalog does not contain any translated texts, once it is decoupled from the original. When you copy and change content, the translation is lost and it is necessary to translate the content again.

You can copy a catalog as follows:

1. Launch the SAP Fiori launchpad designer in configuration scope or customizing scope.

2. Drag and drop the catalog from the left side panel into the (*New Catalog with References*) zone.
   A *Copy Catalog* dialog box appears.

3. Enter title and ID for the new catalog and choose *Copy*.
   A new catalog with the updated details appears on the left side panel and the tiles of the copied catalog on the content area.

Related Information

Scopes for Adapting Launchpad Content [page 111]
Copying Tiles or Target Mappings [page 233]
Managing Outdated Catalogs or Groups [page 191]

1.3.4.6.4.2.3 Managing Outdated Catalogs or Groups

In the SAP Fiori launchpad designer, administrators can update catalogs or groups with the latest changes in the configuration scope.

Context

Changes to catalogs or groups are delivered to the customer in the configuration scope. The launchpad designer allows customers to make changes to these catalogs or groups in the customizing scope. In this case, a copy of the catalog or group is created in the customizing scope. This copy has the same ID as the original catalog or group in the configuration scope.

The copy in the customizing scope is now decoupled from the content updates of the original catalog in the configuration scope. When there is an update for a catalog or group delivered to the customer in the configuration scope, it is not reflected in the corresponding catalog or group in the customizing scope. The inconsistency between the configuration scope and the customizing scope results in the catalog or group becoming outdated in the customizing scope. Even changes made to titles in the configuration scope could lead to catalogs or groups in the customizing scope becoming outdated.

> **i Note**
>
> Changing catalogs that were created in the configuration scope while you work in the customizing scope is not recommended since this can lead to outdated catalogs. See Best Practices for Managing Catalogs [page 115].

The (outdated) icon is displayed against a group or catalog to indicate that it is outdated. You can select the catalog or group with the icon to display the tiles of the outdated catalog or group. The outdated tiles are represented with a red border.

Procedure

As a customer, you can choose to update to the latest changes of the configuration scope or continue working with the existing changes of the customizing scope.

> **i Note**
>
> If you continue working with the customizing scope, you do not receive the latest changes provided in the configuration scope.

To update the customizing scope to be consistent with the configuration scope, proceed as follows:

1. Choose the catalog or group with the outdated icon in the left-hand side panel.
   Tiles along with outdated tiles appear in the content area. The outdated icon also appears in the content area.

2. Choose the link next to the (outdated) icon in the content area.
   A dialog box with a confirmation message to reset the customization scope appears.
3. Choose *OK*.

Result

The outdated catalog or group in customizing is deleted. The customizing scope of the catalog or group is consistent with the configuration scope. The outdated icon for the outdated catalog or group is removed.

Related Information

Scopes for Adapting Launchpad Content [page 111]

1.3.4.6.4.2.4 Accessing Remote Catalogs

A remote catalog contains a collection of tiles provided from the remote server.

> **i Note**
>
> Remote catalogs were made available for integration scenarios with SAP Jam and SAP Smart Business on SAP Business Suite powered by SAP HANA. Remote catalogs are now in maintenance mode and should **not** be used any more.

From the front-end **ABAP** server, you want to access tiles provided by SAP Jam and SAP Smart Business on SAP Business Suite powered by SAP HANA. The other servers, from which you want to access the tiles or catalogs, are the remote servers. The tiles are located in the catalog of the remote servers. These tiles are not defined in the front-end server and the configuration exists in the remote servers.

To render these tiles from the remote server, you create a remote catalog in the front-end server, using either a configuration scope or a customization scope. This remote catalog contains a collection of tiles provided from the remote server. The front-end server does not load the tiles but only provides a reference to the catalog in the remote server.

Remote catalogs allow you to access tile configurations from remote systems. To make sure SAP Fiori applications can be started from the launchpad, the application code has to be installed on the ABAP front-end server and the target mappings for launching the applications have to be configured on the ABAP front-end server.

See SAP Note 2390564 .

> **i Note**
>
> You may need to authenticate to the remote server. The authentication mechanism used depends on the system landscape setup. Only those tiles that the end user has access to are displayed. For remote catalogs, the SAP Web dispatcher configuration for the corresponding catalog should be in place.

Once you have created a remote catalog, you can perform the following:

- View tiles of a remote catalog
- Assign tiles from remote catalogs to groups
- Edit title of a remote catalog
- Delete or copy a remote catalog

You can connect to the remote catalogs and assign remote catalogs to a user's authorization role. For more information, see Assigning Catalogs to Roles [page 330].

Related Information

1.3.4.6.4.2.4.1 Creating or Editing Remote Catalogs

Context

You can launch the launchpad designer either in the customizing or configuration scope to perform the following:

To Create

1. Choose ⊕ on the left-hand side panel.

   A *Create Catalog* dialog box appears.
2. Select *Remote* and enter the following details:
   - *Title*
     Title of the remote catalog in the front-end server.
   - *ID*
     Domain ID in the front-end server.
   - *Remote ID*
     ID of the catalog used in the remote system. In contrary to domain ID, the remote ID is specific to the catalog repository on the remote server.
   - *System alias*
     The system alias is added to the base URL to retrieve tiles from the remote catalog. This way you can define rules in SAP Web Dispatcher to route these requests to the correct remote system.
     For example,
     To dispatch remote catalog URLs in SAP Web Dispatcher, proceed as follows:

     1. Define modification rules as a file (for example, `modification_rule.txt`) with the following content to route requests based on the `;o segment` parameter containing the system:
        ```
        if %{PATH} regimatch *;o=foo* SetHeader x-sap-webdisp-target-sid <FOO>
        [break]
        if %{PATH} regimatch *;o=bar* SetHeader x-sap-webdisp-target-sid <BAR>
        [break]
        [break]
        ```

        > ⓘ Note
        >
        > The regular expression `*;o=<system>*` is just an example, you can provide a more specific expression as needed.

     2. Add the modification file (`modification_rule.txt`) to the SAP Web Dispatcher profile file (`*.pfl`)
        ```
        icm/HTTP/mod_<number> = PREFIX=/,FILE=modification_rule.txt
        ```

        > ⓘ Note
        >
        > - The profile file must contain systems for **FOO** and **BAR** so that the above dispatching works.
        > - To avoid potential issues with system alias maintained for **SAP Gateway Service**, it is recommended to use only "upper case" characters for system alias.

- *Base URL*

  Service root URI of the catalog OData service on the remote system. The **Unified Shell** uses the base URL with system alias and remote ID combined together to load the tiles in the remote catalog. Typically, this URL looks like in the following example:`/sap/hba/r/sb/core/odata/runtime/ SMART_BUSINESS.xsodata;o=<systemAlias>/Catalogs('<remoteID>')/Chips`. A set of known BaseURLs is available in **Value Help** field (F4) in the *Create Remote Catalog* dialog box.

3. Save the details.

   A newly created remote catalog is displayed.

To Edit

1. Choose any remote catalog from the catalog list on the left panel.

2. Choose ![edit icon] (*Edit Title*) icon on the title bar.

   An *Edit Catalog* dialog box with the remote catalog details appears.

3. Edit the following remote catalog details:
   - *Title*
   - *Remote ID*
   - *System alias*
   - *Base URL*

4. Save the details.

   A remote catalog with the new title along with other edited details should appear. If there are any changes made to **Remote ID**, **System Alias**, or **Base URL** is changed, the *Remote Catalog* screen is refreshed with the new tile content.

Related Information

Accessing Remote Catalogs [page 193]
Copying or Deleting Remote Catalogs [page 195]
Assign Business Catalogs to Roles [page 330]

1.3.4.6.4.2.4.2 Copying or Deleting Remote Catalogs

You can launch the launchpad designer either in the customization or configuration scope to copy or delete the remote catalog.

Context

To Delete

- Drag and drop the remote catalog from the left-hand side panel into the (trash zone).

To Copy

Procedure

1. Drag and drop the remote catalog from the left-hand side panel into the (copy zone). A *Copy Catalog* dialog box appears.
2. Enter the *Title* and *ID* for the new remote catalog and choose *Copy*. A remote catalog with the same remote configuration details (remote ID, system alias, base URL) as that of the original catalog is created.

Related Information

1.3.4.6.4.2.4.3 Working with SAP HANA Catalogs

Context

You can view SAP HANA catalogs in the catalog list only when you initiate the add tile to group process. You can add tiles from the selected SAP HANA catalog to the group.

Procedure

1. Choose *Groups*. You can see all the groups and tiles associated to each group.
2. Choose the *Add Tile* icon on the *Groups*.
3. Select the HANA catalog from the catalog list. This displays all the tiles associated to the HANA catalog.
4. Choose the *Add Tile* icon on the tile you want to add to the group. You get a message "Tile added successfully." You can add multiple tiles from the HANA catalog to the group.
5. Choose the *Back* button. You can now see the tile added to the group.

For more information, see SAP Library for Fiori documentation on SAP Help Portal at http://help.sap.com/fiori.

For more information, about KPI Modeler documentation see SAP Library on SAP Help Portal at http://help.sap.com/ssb. Choose ▶ *Application Help* ▶ *Tools* ▶ *KPI Modeler* ▶.

1.3.4.6.4.2.4.4 Working with SAP Jam Catalogs

In order to create tiles for your SAP Jam groups on the launchpad, you must create a remote catalog specifically for SAP Jam. To do this, you will need to use the social media catalog ID `SMCatalog` as the ID of your remote catalog.

For more information about SAP Jam catalogs, see:

1.3.4.6.4.2.5 Verifying Catalogs for Targets and Tiles

You want to verify the catalogs for target mappings (targets) and app launcher tiles.

Prerequisites

You understand how target mapping and intents works.

For more information, see About Navigation [page 108] and Configuring Target Mappings [page 217].

Context

If you are unable navigate to an application in SAP Fiori launchpad, it may imply that there is no matching target mapping assigned to the user's role. In this case, you must assign the catalog having the matching target mapping to the user or create a matching target to the catalog that is assigned to the user.

The SAP Fiori launchpad designer identifies matching targets for every tile in a catalog or a set of catalogs. If there are no matching targets, you can create a target mapping with the intent that is missing in the same catalog or any other preferred catalogs. This catalog or set of catalogs can then be assigned to the user's authorization role.

To confirm that the catalogs assigned to the user or role have a valid intent, proceed as follows:

Procedure

1. To add the catalogs to the *Catalog Collection*, drag and drop a catalog in the drop zone. A *Catalog Collection* is a section available in the left-hand side panel to collect the catalogs for verification. This section is available only for the *Catalogs* tab in the left-hand side panel. Using the pin and unpin icons, you can pin and unpin the catalogs in the *Catalog Collection*. By default, the catalogs are pinned. When you unpin, you can remove the catalog only from the *Catalog Collection* section.

2. Select a catalog from the *Catalog Collection*. A list of intents with intent status for the catalogs in the *Catalog Collection* appears in the content area. The following icons are available for the status of an intent:

Status	Description
✔	Indicates that the catalogs in the *Catalog Collection* contain at least one app launcher tile and one target mapping for that particular intent.
⚠	Indicates that the catalogs in the *Catalog Collection* don't contain an app launcher tile tor a target mapping for that particular intent.

3. Select an intent from the list to view the following:

View	Description
Catalog Title and ID	A target mapping table and an app launcher tile table appear in the content area. Both the tables display the title and ID of the catalogs from the *Catalog Collection* for the selected intent.
Associated Catalogs	Choose the 🔧 icon in the target mapping table or app launcher tile table to view the associated catalogs of the selected intent from the catalogs list in left-hand side panel.
Tiles of the selected catalog	Tiles that contain the selected intent in the selected catalog appear in the content area below the *Intent* and *Status* list. To view all the tiles of the selected catalog and to add a tile in the catalog, choose ➠ (*Display More*) icon.

1.3.4.6.4.2.5.1 Creating Missing App Launcher Tiles or Target Mappings

You can perform a consistency check for the catalogs present in the *Catalog Collection*. The result of the consistency check displays details about the app launcher tiles and target mappings within the catalog list. If there are missing app launcher tiles or target mappings, you can create them with the additional buttons displayed in the respective app launcher tile and target mapping tables. This ensures that administrators have a collection of catalogs with validated apps.

Creating App Launcher Tiles

If for a selected intent only target mappings are present and no app launcher tiles are created in the *Intents for Catalogs in Catalog Collection* list, you can proceed as follows:

1. Select the intent for which the app launcher tile is to be created or with the ⚠ *Status Alert* icon.
2. Choose *Create App Launcher*. A *Create App Launcher* dialog box appears.
3. Select either *App Launcher Static* or *App Launcher Dynamic* radio buttons.
4. Choose *Create*. Depending on the selection, a *Configure 'App Launcher - Static'* or *Configure 'App Launcher – Dynamic'* screen appears with a pre-filled semantic object and action.
5. Enter the details in the configuration screen and choose the *Save* button. A *Confirmation* dialog box appears.
6. Choose the *OK* button. An app launcher for the selected intent is created.

Creating Target Mapping

If for a particular intent no target mappings are created or only app launcher tiles are created, you can proceed as follows:

1. Select the intent for which the target mapping is to be created or with the ⚠ *Status Alert* icon.
2. Choose *Create Target Mapping* button from the *Catalogs containing target mappings* table. A *Configure 'Target Mapping'* screen with a pre-filled semantic object and action fields appears.
3. Enter the details in the configuration screen and choose *Save*. A *Confirmation* dialog box appears.
4. Choose the *Save* button. A *Confirmation* dialog box appears.
5. Choose the *OK* button. A target mapping for the selected intent is created.

> **i Note**
>
> If more than one catalog is present in the *Catalog Collection*, the missing target mapping or app launcher tile is created in the selected catalog only.

1.3.4.6.4.2.6 Editing the Title of Catalogs

You can change the title of a catalog or a group.

Procedure

1. Choose the catalog for which you want to edit the title.
2. Choose the *Edit Title* icon from the header. For groups, choose the *Edit Group* icon.
   A dialog box appears.
3. Enter or adapt the title.
4. Choose *Save*.

1.3.4.6.4.2.7 Searching for Catalogs

You want to search for a particular catalog from the list available in the left-hand side panel. A search box with a search icon is provided in the left-hand side panel for searching.

To search for a catalog, proceed as follows:

1. On the launchpad screen, choose the *Catalogs* tab in the left-hand side panel.

   A search box with the search icon can be found in the left-hand side panel.
2. Enter the title or ID of the catalog in the search box and choose the *Search* icon.

   A catalog with the title or ID entered in the search field appears in the left-hand side panel.

> → Tip
>
> You can also perform these steps to search for a group.

> i Note
>
> If the title or ID entered in the search box does not match with any of the titles or IDs of the catalogs or groups title or ID, a "**No results found**" message appears.

1.3.4.6.4.2.8 Navigating to Catalogs

You can navigate to a catalog using the following:

> → Tip
>
> You can also perform these steps to navigate to a group.

- **i** (information) button

  The tiles in a group are supported with the information button for navigation. The button can be found on the top right-hand corner of a tile. The information button provides the details of a tile in a group. You can navigate from a tile to a catalog using this button. When you click the information button, a popover with the following details appears:

  - *Tile instance ID*

    Contains instance ID of a tile
  - *Catalog ID*

    Contains the ID of a catalog from which the tile originated
  - *Tile ID*

    When you choose the catalog ID, the catalog from which the tile originated is displayed. Using the browser back or application back button, you can navigate to the group from which the catalog was displayed. You can perform any catalog-specific functions on the catalog.

- **Hash (#) parameter**

  The SAP Fiori launchpad designer supports hash-based navigation. Information required to load a catalog or a group is encoded in the URL, that is, after the hash (#) character called as a hash parameter. You can explicitly enter the required hash parameter in the URL after the hash (#) character for a catalog or a group that you want to navigate to. The hash parameter contains a valid ID of a group or a catalog.

  The format for the hash parameter in the URL is as follows:

  - For catalogs - *#/Catalog/ID*
  - For groups - *#/Group/ID*

    For example, **http(s) ://< server>:<port>/.../ arsrvc_upb_admn/main.html#/Catalog/ID**

  If the entered ID in the hash parameter is valid, the system displays the corresponding catalog or group for the ID.

  Hash navigation is not supported for the following screens:

  - Adding a tile to a catalog
  - Adding a tile to a group
  - Configuration screen of a tile in a catalog

    The above-mentioned screens in the SAP Fiori launchpad designer are typically represented by a hash (#) with no parameters.

For more information on navigation using semantic objects and actions, see About Navigation [page 108].

1.3.4.6.4.2.9 Language Check for Catalogs

When you open the launchpad designer and try to change a catalog, a verification is run to determine whether your logon language is the same as the original language of the catalog. If the languages are not the same, the catalog is displayed as read-only. In other words, you can view only the tiles or target mappings in it.

> **→ Tip**
>
> You can also perform this step for a group.

1.3.4.6.4.3 Creating and Configuring Tiles and Target Mappings

This section provides information about tiles and target mappings.

What is a tile?

A tile is a visual representation of an app on the launchpad home page. When a user clicks a tile in their launchpad, an intent (including optional parameters) navigates to the app and opens it. Adminstrators create tiles in catalogs.

What is a Target Mapping?

The target mapping maps a navigation target to the intent (combination of semantic object and action). The target mapping is a prerequisite for the navigation to an app in the launchpad. It can refer to only one target application which can be built with different UI technologies:

- SAPUI5
- Web Dynpro
- SAP GUI for HTML
- WebClient UI (CRM)
- plain URL

Administrators create target mappings in catalogs by defining the intent, the navigation target, the device types suppported by the target, and additional parameters.

> **i Note**
>
> Tiles and target mappings are part of catalogs. A lot of the functions that are available within catalogs apply to both tiles and target mappings, such as creating a reference, the delete and where-used functions or configuration.

Related Information

Setting Up Navigation [page 127]
Creating or Deleting Tiles or Target Mappings [page 203]

1.3.4.6.4.3.1 Creating or Deleting Tiles or Target Mappings

Context

The tiles and target mappings are visualized as tabs in the catalogs view. You can use these tabs for viewing, adding, removing, configuring, or copying tiles or target mappings. The table or list views for tiles and target mappings also allow you to sort and filter table entries in the columns. The tabs display the number of tiles or target mappings of the selected catalog.

When an app is deprecated this is displayed in the tile configuration. The configure view and the target mapping view show a message concerning the deprecation and also indicate the ID of the successor app, if a successor is available.

> **i Note**
>
> An empty catalog is not shown in the catalog collection.

Tiles

You can choose the tile templates from the Universal Fiori Launchpad Catalog, `/UI2/FLPD_CATALOG`. The following are the standard tile templates stored in the Universal Fiori Launchpad Catalog:

- Dynamic AppLauncher
- Static AppLauncher
- News

You can enhance the standard tile template list by adding a new tile template to the Universal Fiori Launchpad Catalog in the back end. For more information, see Assigning a CHIP to the Universal Fiori Launchpad Catalog [page 596].

Adding Tiles

> **i Note**
>
> Due to performance reasons, we recommend you to not add more than 200 tiles to one catalog.

1. Choose any catalog from the catalog list.

   The tile or tiles associated with the selected catalog appear in the content area on the screen.

2. Choose ✚ *Add Tile* icon.

The *Select a Tile Template* screen appears with various tile templates. Tiles contain a title and a description for the tile.

3. Choose the tile template that you want to use.

   An instance of the tile template is created in the selected catalog along with the other tiles. The count of the tiles for the selected catalog on the left panel increases by 1.

> **i Note**
>
> You can also add a new tile by choosing *Create* in the tiles list view.

Deleting Tiles

1. Choose the tile that you want to delete from the selected catalog.
2. Drag and drop the selected tile into the trash zone.

   > **i Note**
   >
   > You can also delete a tile by choosing *Delete* in the tiles list view.

   The selected tile is deleted and removed from the catalog.

> **i Note**
>
> A **where-used check** is performed when you try to delete a tile. If the tile is still referenced by a tile in another catalog, a warning appears that it is not recommended to continue without checking the referencing object first.
>
> The check can also be performed by simply choosing *Where-Used* in the tiles list view and ensures the following:
>
> - The tile is not referenced by a tile in another catalog
> - The tile is not used in a group

Further Tasks

You can also do the following for a tile:

- Configure tiles
  You can open the configuration screen for the tile and add the required data.
- Create a reference
  You can create a reference tile in another catalog.
- Navigate to original tile (tiles list view)
  You can navigate from a reference back to the original tile.

Sorting and filtering can be done by right-clicking the column titles in the table.

 Target Mappings

You can view the target mappings inside the Target Mappings tab on the screen. You can also view the number of target mappings displayed on the tab for a selected catalog. When you choose the Target Mappings tab, a table appears on the screen. The target mapping table contains a number of columns, such as:

Parameter	Description
Semantic Object	Displays the semantic object of the target mapping
Action	Displays the action of the target mapping
Navigation Type	Displays whether the navigation target application is either an **SAPUI5-based** Fiori application or any **other** SAP Fiori application
Device Types • Desktop • Phone • Tablet	Displays a ✔ (tick) icon if the application is compatible for that particular device type or displays a ✖ (decline) icon if it is not compatible
Outdated	Displays a ✔ (tick) icon if the target mapping is outdated or displays a ✖ (decline) icon if the target mapping is not outdated
Reference	Displays a ✔ (tick) icon if the target mapping is copied or displays a ✖ (decline) icon if the target mapping is not copied
Create Target Mapping	Adds target mappings to the catalog. On choosing any table entry, the following action buttons are displayed at the footer of the target mappings screen: • _Create Reference_ You can create a reference for the selected target mapping and add it to a different catalog. • _Configure_ You can open the configuration screen for the target mapping and add the required data. • _Delete_ You can delete the selected target mapping. • _Where-Used_ You can perform a where-used check. • _Original_ You can navigate from a reference back to the original target mapping.

> **i Note**
>
> Inside the Target Mappings tab, you can show or hide columns and change their order by selecting *Configure Columns*. You can also change the column order using drag and drop.
>
> Sorting and filtering can be done by right-clicking the column titles in the table.

Adding Target Mappings

1. Choose *Create Target Mapping* button present below the target mapping table. A *Configure: 'Target Mapping'* screen appears.

   > **i Note**
   >
   > A target mapping is created irrespective of whether you save or cancel the configuration changes of the target mapping.

2. Enter the details and choose the *Save* button. A *Confirmation* dialog box appears.
3. Choose the *OK* button.
   A target mapping is added to the tile.

Removing Target Mappings

1. Choose the entry from the table for which you want to delete the target mapping.
2. Choose the *Delete* button. A *Confirmation* dialog box appears.
3. Choose the *OK* button.
   The target mapping for the selected catalog is deleted.

   > **i Note**
   >
   > A **where-used check** is performed when you try to delete a target mapping. If the target mapping is still referenced by another target mapping, a warning appears that it is not recommended to continue without checking the referencing object first.
   >
   > The check can also be performed by simply choosing *Where-Used*.

Related Information

Copying Tiles or Target Mappings [page 233]
Configuring Target Mappings [page 217]

1.3.4.6.4.3.2 Configuring Tiles

Context

You can configure tiles only in catalogs. Once you have added the tile template to the catalog, you can configure the tile as follows:

1. Choose the tile you want to configure.

   A configuration details screen appears. You can view the instance ID of the tile you want to configure.
2. Enter the details.
3. Choose *Save* and then *OK*.

> i Note
>
> In the configuration details screen, you enter the details such as title, description, icon, or news feeds. In app launcher tiles, you need to define the intent, which means to specify the semantic object, action, and parameters.

Related Information

Static App Launcher Tiles [page 207]
Dynamic App Launcher Tiles [page 210]
News Tile [page 215]
Defining Semantic Objects for Navigation [page 128]
Configuring Target Mappings [page 217]
Creating or Deleting Tiles or Target Mappings [page 203]

1.3.4.6.4.3.2.1 Static App Launcher Tiles

App launcher tiles are used to launch applications from the home page. This topic describes the parameters for configuring static app launcher tiles.

> i Note
>
> You can modify only the content of translatable parameters using the same logon language that was active when the catalog was created. For more information about translating customer-created tiles, refer to the related links below.

General

Parameter	Description	Translatable
Title	Title to be displayed on the tile.	Yes
Subtitle	Subtitle to be displayed below the tile title.	Yes
Keywords	You can tag a tile with keywords so users can find it more easily using the search function in the tile catalog. If you want to specify several keywords, use either space or comma as separator.	Yes
Icon	Use the value help to select an icon. After you have selected an icon, the system inserts a URL starting with the `sap-icon://` protocol. These icons are based on SAPUI5 icon fonts. For more information, see .	No
Information	Text to be displayed at the bottom of the tile.	Yes

Navigation

Parameter	Description
Use Semantic Object Navigation	Deactivate this checkbox if you want to define the navigation target using a simple URL rather than a semantic object.
Semantic Object	Semantic object to be referred when clicking the tile, for example **Order**. This semantic object must be defined in Customizing. For more information, see Defining Semantic Objects for Navigation [page 128].
Action	Operation that is intended to be performed on the semantic object when the user clicks on the tile, for example **display**.
Parameters	Key-value pairs defining parameters for the semantic object, for example **orderID=4711**. If you enter multiple parameters, separate them with an ampersand (**&**), for example **orderID=10000&custID=c82200**.
Target URL	You can enter any URL, for example **http://help.sap.com**. Note that you have to provide a complete URL including the protocol (typically **http://** or **https://**) > ℹ Note > If you choose to use semantic object navigation, the target URL will be automatically computed and inserted into this field.

Tile Actions

Parameter	Description	Translatable
Menu Item	The name of the menu item that represents the action.	Yes
Target Type, *Navigation Target*, *Action*	The *Target Type* is one of the following: • *URL* - In the *Navigation Target* column, enter the target URL for the action. In this case, the *Action* column is disabled. • *Intent* - In the *Navigation Target* column, select the semantic object from the list. In this case, use the *Action* column to enter the action.	No
Icon	The icon to display next to the menu item.	No

i Note

To see tile actions in the launchpad, you must enable them in the launchpad configuration.

→ Tip

Use the *Add* and *Remove* buttons at the bottom of the table to add and to remove actions.

Special Case: App Variants

If you create a tile for an app variant, create a target mapping first (see Configuring Target Mappings [page 217]). Then create a static app launcher tile using the following values:

Parameter	Description
Title	Title of the app variant as provided by the key user or developer
Subtitle/Icon/Information	Entries as provided by the key user or developer
Semantic Object	Same value as for the target mapping
Action	Same value as for the target mapping
Parameters	`save-appvar-id=` followed by the app variant ID provided by the key user. Example: `save-appvar-id=customer.sap.ui.fl.testtest.id_1234567890123_45`

→ Tip

Consider adding the new tile for the app variant to an appropriate group so that your users don't have to use the app finder to add it to their home page. For more information, see Adding Tiles to and Removing Tiles from Group [page 307].

Related Information

1.3.4.6.4.3.2.2 Dynamic App Launcher Tiles

In contrast to regular (static) app launcher tiles, the information that is displayed on a dynamic app launcher tile can be pulled from a back-end system using an OData service. This topic describes the parameters for configuring dynamic app launcher tiles.

The dynamic information pulled from the back end overwrites the configuration values given in the tile configuration. Some tile functionality (for example, the state arrow) can only be configured by dynamic information (and not in the tile configuration).

> **i Note**
>
> You can modify only the content of translatable parameters using the same logon language that was active when the catalog was created. For more information about translating customer-created tiles, refer to the related links below.

General

Parameter	Description	Translatable
Title	Title to be displayed on the tile.	Yes
Subtitle	Subtitle to be displayed below the tile title.	Yes
Keywords	You can tag a tile with keywords so users can find it more easily using the search function in the tile catalog. If you want to specify several keywords, use either space or comma as separator.	Yes
Icon	Use the value help to select an icon. After you have selected an icon, the system inserts a URL starting with the `sap-icon://` protocol. These icons are based on SAPUI5 icon fonts. For more information, see .	No
Information	Text to be displayed at the bottom of the tile.	Yes
Number Unit	Enter the unit to be displayed below the number, for example, **USD**.	No

Dynamic Data

Parameter	Description
Service URL	URL of an OData service from which data should be read.
	The response is expected in JSON format.
	When the service is called, the values that are provided by the service override the values that have been configured manually in the tile details.
	Note that the service is executed on the home page only. On the admin page, sample data is displayed instead.
	→ Tip To read the number of entities from an OData service, use the `$count` parameter in the *Service URL*. If you are using a filter to limit the number of entries, use the `$inlinecount=allpages` parameter in the *Service URL* to display both the number after the filter and the number from the `m:count` property.
	For more information about the service API, see Response Values for Dynamic Tiles [page 213].
	i Note If the response is a numerical value over 1,000, the front end scales the number accordingly and adds a unit indicator (K, M, or B).
Refresh Interval	Number of seconds after which dynamic content is read from the data source again and the display is refreshed.
	The launchpad uses a default of 0 seconds unless you enter a higher value in this field.
	The launchpad will fall back to 10 seconds, if the value entered is > 0 and < 10. If the value entered is 0, the dynamic tile is updated only once on load. If the value is >= 10, the value is taken as refresh interval. The maximum value for the refresh interval is 2147483 seconds.

Navigation

Parameter	Description
Use Semantic Object Navigation	Deactivate this checkbox if you want to define the navigation target using a simple URL rather than a semantic object.
Semantic Object	Semantic object to be referred when clicking the tile, for example **Order**.
	This semantic object must be defined in Customizing. For more information, see Defining Semantic Objects for Navigation [page 128].
Action	Operation that is intended to be performed on the semantic object when the user clicks on the tile, for example **display**.

Parameter	Description
Parameters	Key-value pairs defining parameters for the semantic object, for example **orderID=10000**. If you enter multiple parameters, separate them with an ampersand (**&**), for example **orderID=10000&custID=c82200**.
Target URL	You can enter any URL, for example **http://help.sap.com**. Note that you have to provide a complete URL including the protocol (typically **http://** or **https://**) > i Note > If you choose to use semantic object navigation, the target URL will be automatically computed and inserted into this field.

Tile Actions

Parameter	Description	Translatable
Menu Item	The name of the menu item that represents the action.	Yes
Target Type, *Navigation Target*, *Action*	The *Target Type* is one of the following: - *URL* - In the *Navigation Target* column, enter the target URL for the action. In this case, the *Action* column is disabled. - *Intent* - In the *Navigation Target* column, select the semantic object from the list. In this case, use the *Action* column to enter the action.	No
Icon	The icon to display next to the menu item.	No

> i Note
>
> To see tile actions in the launchpad, you must enable them in the launchpad configuration.

> → Tip
>
> Use the *Add* and *Remove* buttons at the bottom of the table to add and to remove actions.

Related Information

1.3.4.6.4.3.2.2.1 Response Values for Dynamic Tiles

Learn how to create the OData service for a dynamic app launcher tile.

When working with the classic home page, the dynamic information that is pulled from the back-end overwrites the configuration values given in the tile configuration. Some tile functionality (for example, the state arrow) can only be configured by dynamic information (and not in the tile configuration).

When working with spaces and pages, the configuration values are saved independently from the response values. If some response values are undefined, the configuration value is used. Empty strings should not be used as this might have side effects. If you want to have an empty value instead of the configuration value, you define the response value <RESET>. This will not change the configuration value.

To feed an app launcher tile with dynamic content, you have to create an OData service that returns the configuration properties as in the following example structure:

```
{
                "d": {
                                "icon": "sap-icon://travel-expense",
                                "info": "Quarter Ends!",
                                "infoState": "Critical",
                                "number": 43.333,
                                "numberDigits": 1,
                                "numberFactor": "k",
                                "numberState": "Positive",
                                "numberUnit": "EUR",
                                "stateArrow": "Up",
                                "subtitle": "Quarterly overview",
                                "title": "Travel Expenses"
                }
}
```

The launchpad supports OData V2 and OData V4 for dynamic tiles. In general, OData V4 works similar to V2. The only relevant difference is that the additional "d" level is skipped in the response payload's JSON. So for example, `oResponseData.d.number` is simplified to `oResponseData.number`. The above payload example looks like this for OData V4:

```
{
    "icon": "sap-icon://travel-expense",
    "info": "Quarter Ends!",
    "infoState": "Critical",
    "number": 43.333,
    "numberDigits": 1,
    "numberFactor": "k",
    "numberState": "Positive",
    "numberUnit": "EUR",
    "stateArrow": "Up",
    "subtitle": "Quarterly overview",
    "title": "Travel Expenses"
}
```

Properties

Property	Description	Default Values
icon	Enter an sap-icon:// URL, for example **sap-icon://cart**. You can look up the names of the available icons in tile configuration. For more information, see Static App Launcher Tiles [page 207].	""
info	Text to be displayed at the bottom of the tile.	""
infoState	The color of the tile is adapted according to the value of this property. The precise color depends on the theme that you have selected in UI theme designer. Allowed values: **Negative**, **Neutral**, **Positive**, **Critical**. Also see enum sap.m.ValueColor	ValueColor.Neutral
number	Number to be displayed in the top right corner of the tile.	"..."
numberDigits	Number of digits to be displayed following the decimal separator (decimal point or decimal comma, depending on the language settings).	4
numberFactor	A factor for scaling numbers, for example, for displaying large numbers like 1.000.000 (-> number = 1 and numberFactor="M") or for percentages (number = 22.2 and numberFactor = "%").	""
numberState	The color of the number is adapted according to the value of this property. The precise color depends on the theme that you have selected in UI theme designer. Allowed values: **Negative**, **Neutral**, **Positive**, **Critical**. Also see enum sap.m.ValueColor	ValueColor.Neutral
numberUnit	Unit to be displayed below the number, for example, **USD**.	""
stateArrow	Displays an arrow indicating a trend. Allowed values: **None**, **Up**, **Down**. Also see enum sap.m.DeviationIndicator	DeviationIndicator.None
subtitle	Subtitle to be displayed below the tile title.	""

Property	Description	Default Values
targetParams	List of key-value-pairs separated by ampersands. When the application is launched (by clicking on it), these parameters are passed to the application as business parameters (if semantic object-based navigation is used) or as URL parameters (if URL-based navigation is used). If any parameters have been entered in the *Parameters* field in the tile configuration, the parameters passed by the OData service are appended to the list of parameters to be passed to the application.	""
targetURL	URL that opens the app.	""
title	Title to be displayed in the tile.	""

> **i Note**
>
> If you want to show aggregated numbers, the server has to aggregate the numbers and provides the result in the JSON format described here. Also see OData Aggregation.

Related Information

1.3.4.6.4.3.2.3 News Tile

Context

To configure the news tile, you have to maintain the following configuration parameters:

Configuration Parameters	Description
Tile Default Image	It is a URL (sets the default image for the *News Tile*). You can set this parameter to select an alternate default image to display on the *News Tile*. By default, the *News Tile* application provides 12 default images and the *News Tile* cycles through these default images in sequence.

Configuration Parameters	Description
Always Use Default Image	It is a checkbox.
	When this parameter is set, the *News Tile* ignores any image that accompanies with the RSS feed article. The following order of precedence is used:
	• **When parameter is set**
	• Image from Tile Default Image (if set)
	• Image from one of the twelve (12) default images
	• **When parameter is not set**
	• Image from the RSS Article (if present)
	• Image from the RSS Channel (if present)
	• Image from Tile Default Image (if set)
	• Image from one of the twelve (12) default images
Article Cycle Interval (secs)	It is an integer (minimum is 5). The default value is 5.
	This parameter controls the rate at which the articles cycle through the *News Tile*.
Article Refresh Interval	It is a dropdown box. The default value is 15 minutes.
	This parameter controls the rate at which the *News Tile* requests new articles from the *Article Feeds*.
Article Feeds	You can configure 10 RSS feeds. The *News Tile* monitors the RSS feeds and retrieves new articles based on the *Article Refresh Interval*.
	i Note If the URL references an external feed, the feed must be `CORS` compliant. If the URL references an internal feed, the feed must originate from the same server and port as the Fiori launchpad.
Feed #1 – Feed #10	It is a URL.
	The http URL of the RSS feed.
Inclusion Filters	You can configure 5 inclusion filters. The *News Tile* filters the feeds and includes any articles that contain the same text in the *Title* of the article.
Filter #1 – Filter #5	It is a text.
	The filter text that is compared to the *Title* of the article. If the text is found, the article is included in the list of articles.
Exclusion Filters	You can configure 5 exclusion filters. The *News Tile* filters the feeds and excludes any articles that contain the same text in the *Title* of the article.
Filter #1 – Filter #5	It is a text.

Configuration Parameters	Description
	The filter text that is compared to the *Title* of the article. If the text is found, the article is excluded from the list of articles.

> ℹ Note
>
> The following restrictions apply when configuring parameters:
>
> - The feed URLs are limited to the following source:
>   - Internal sources (same URL and port as of the SAP Fiori launchpad application)
>   - Any external `CORS` compliant source.
> - The URL format should follow the `https://[server]:[port]` pattern.
>
>   > ℹ Note
>   >
>   > The URLs using "`feed://`" are not supported.
>
> - The UI5 URL validation requires the tilde '~' character to be replaced by the sequence '&#126'. For example, in the path "`...filterID=content~tag`", the tilde should be replaced by "`...filterID=content&#126tag`."
> - Bookmarking and direct navigation to the list of feed articles is not supported.

1.3.4.6.4.3.3 Configuring Target Mappings

In the target mapping configuration, you map a navigation target to the combination of a semantic object and an action, also known as an intent.

Context

To create a target mapping, you define the intent, the navigation target, the device types supported by the target, and additional parameters in the SAP Fiori launchpad designer.

> ℹ Note
>
> You cannot create target mappings for WebClient UI apps. Please use the launchpad app manager instead. (See Maintaining Launchpad App Descriptor Items [page 148] for further information.) When the target mapping is set, the WebClient UI reference can be handled the same way as other references.

> ℹ Note
>
> A target mapping is not necessarily unique at runtime. The same intent, parameter, and system alias can occur in more than **one** target, depending on how you define your catalogs and groups and assign the related roles. For example, one target mapping can occur in several catalogs and can be assigned to roles that are all assigned to the same user. Make sure that you structure your target mapping based on your specific needs.

If you already know the configuration procedure and want to go directly to the information about **settings** for certain target mapping types, use the following links:

Procedure

1. To create a new target mapping, select *Create Target Mapping*. To change an existing mapping, select *Configure*.
   The target mapping configuration page is displayed.
2. Create the *Intent* to be mapped to a target. For more information about intent-based navigation and what an intent is, see Intent-Based Navigation [page 108].

Parameter	Description
Semantic Object	Semantic object as defined in an app launcher tile.
	The following characters are allowed: uppercase letters [A-Z], lowercase letters [a-z], numbers [0-9]. The first character must be an uppercase or lowercase letter.
Action	Action as defined in an app launcher tile. For the action name, choose a verb or a short phrase starting with lower case and without blanks. Examples are `display`, `create`, or `release`.
	The following characters are allowed: uppercase letters [A-Z], lowercase letters [a-z], numbers [0-9] and underscores [_]. The first character must be an uppercase or lowercase letter or an underscore.

!Restriction

The length of the semantic object is limited to 30 characters and the length of the action to 50 characters.

3. Select the application type that you require.

   - If you want to create or change target mappings for an **SAP Fiori app**, select *SAPUI5 Fiori App*. You can only use this function for SAP Fiori apps. Do not use this function for other SAPUI5 applications.
     For **SAPUI5 app variants**, you also select this application type, but you need to provide different settings (see below).

- If you want to create or change target mappings for **Web Dynpro ABAP**, select *Web Dynpro*.
- If you want to create or change target mappings for **SAP GUI**, select *Transaction*.

> **i Note**
>
> Do not create target mappings to launch the SAP Easy Access menu (transaction `SMEN` or `WEBGUI`). Instead, you can configure that the launchpad retrieves entries from the SAP Easy Access menu and displays them in the app finder. See Integrating Applications from SAP Easy Access Menu [page 312] and SAP Notes 2786421 and 2557255 .

- If you want to create or change target mappings for SuccessFactors applications or other web applications, select *URL*.

> **i Note**
>
> The option *SAP Fiori App using LPD_CUST* should not be used to create new target mappings.

4. Provide the following settings.

   Each application type has different settings. Click on the titles below to display a description of the settings for your application type.

   Settings for SAPUI5 Fiori Apps

Parameter	Description
Title	Title of the SAPUI5 application
URL	• If you reference the SAPUI5 component name in the *ID* field (see below), enter the base URL of the SAPUI5 application. > **i Note** > The URL is server-relative. It must point to an SAP Fiori app. This field is optional for SAP Fiori applications located in the SAPUI5 ABAP repository. • If you reference the ID of the application descriptor in the *ID* field (see below), leave this field empty.

ID	Enter one of the following:
	• Component name of an SAPUI5 application
	  Example: `cus.sd.myquotations`
	  See Identifying the Component Name [page 131].
	• ID of the application descriptor of an SAPUI5 application.
	  See the description of the attribute `ID` in the `sap.app` namespace under .

Settings for SAPUI5 App Variants (created via key user adaptation or an adaptation project)

Parameter	Description

> ℹ Note
>
> When you create the target mappings, we recommend that you use the same catalog as for the original app. If you use a different catalog, make sure that users have the required permissions. See also Working with App Variants [page 482]. After you have created the target mapping, create a static app launcher tile.

For some of the following parameters, you have two options (A and B). Make sure that for each target mapping, you always use **either** Option A **or Option B**, but don't mix them.

Parameter	Description
Semantic Object	• Option A: Same value as in the original app* • Option B: A unique value of your choice
Action	• Option A: Same value as in the original app* • Option B: A unique value of your choice
Application Type	*SAPUI5 Fiori App*
Title	Title of the app variant as provided by the key user or developer
ID	ID of the app variant as provided by the key user or developer

*If you used Option A for the *Semantic Object* **and the** *Action*, **also add a new** *Parameter* **with the following values. If you went for Option B, you don't need this parameter:**

Parameter	Description
Name	**save-appvar-id** (also needed to create the app launcher tile later)
Mandatory	[Select]
Value	ID of the app variant as provided by the key user or developer

To configure a tile for an app variant, see Static App Launcher Tiles [page 207].

Settings for SAPUI5 Fiori Apps using LPD_CUST (Deprecated)

Parameter	Description
Launchpad Role	Key of a launchpad role as defined in *LPD_CUST*
Launchpad Instance	Key of a launchpad instance as defined in *LPD_CUST*
Application Alias or *Application ID*	Specify a value for only one of these properties: • Application alias as defined in *LPD_CUST* • Application ID as defined in *LPD_CUST*

Parameter	Description

If you want to change any of these entries, you need to do that in transaction *LPD_CUST*. For more information , see Changing LPD_CUST Entries for Navigation Targets [page 228].

Settings for Web Dynpro Applications

Parameter	Description
Title	Title of the application
Application	ID of the Web Dynpro application
Configuration	ID of the Web Dynpro application configuration
System Alias	Alias of the target system
	For more information on how to configure a system alias, see Configuring Remote Systems [page 308].

Settings for Transactions

Parameter	Description
Title	Title of the transaction
Transaction	Transaction code
System Alias	Alias of the target system
	For more information on how to configure a system alias, see Configuring Remote Systems [page 308].

Settings for URL Applications

Parameter	Description
Title	Title of the application
URL	URL of the application
	You can enter any URL.
System Alias	Alias of the target system
	For more information on how to configure a system alias, see Configuring Remote Systems [page 308].

5. You can specify the device types and business parameters supported by the target.

Parameter	Description
Information	Important or helpful comments and dependencies.

Parameter	Description
Device Types	The device types that are suitable for running the target application: *Desktop*, *Tablet*, *Phone*. The selected device types determine the devices on which to display the tiles using this target mapping. By default, all types are selected. If you deselect a type, the tiles using this target mapping is not displayed on that device.
Parameters	Define mandatory and optional parameters for the target mapping. The parameters are validated during the resolving of the navigation target, to prevent opening a target without providing the necessary mandatory parameters. When defining a parameter, the following rules apply: • Parameter names must be unique. The following characters are allowed: uppercase letters [A-Z], lowercase letters [a-z], numbers [0-9], minus [-] and underscores [_]. The first character should be an uppercase or a lowercase letter. A minus or an underscore should not be used as fist character. • For a mandatory parameter, you can enter a string value or select the checkbox in the *Is Regular Expression* column to indicate that the value is a regular expression. • For an optional parameter, you can define a default value. • For the URL parameter name, enter the target name as it is used by the target application (optional). Select the *Allow additional parameters* checkbox to allow passing additional parameters that are not defined in the table to the target.

6. Choose *Save*.

Related Information

1.3.4.6.4.3.3.1 Identify the Newest Target Mapping

You can identify the most updated version of a target mapping.

Context

When creating a target mapping, you can view similar target mappings from all the catalogs in the system, to make sure that the most updated version of a target mapping is being used as the target.

Procedure

1. In the Catalogs view, select the *Target Mapping* tab.
   The target mapping page is displayed, with a table that lists the existing target mappings of the selected catalog.

2. Choose *Show similar target mappings*.

   The *Target Mapping Versions* table is added to the bottom of the page. The table lists, for all catalogs, the target mappings,that contain the same intents as the target mappings that are defined for the original catalog. Use this table to locate the most updated version of a target mapping.

3. Locate the intent that you want to configure.

4. Use the content of the *Information* and the *Created On* columns to locate the most recent target mapping.

5. Copy the value from the *Target mapping ID* column.

6. Navigate to the catalog of the selected target mapping by clicking the catalog name.

7. Paste the *Target mapping ID* in the search field at the top of the page and search. The corresponding target mapping is displayed (in the Target Mapping tab).

8. Select the mapping and choose *Copy* to add this target mapping to the original catalog, by selecting the catalog from the list.

9. Navigate to the original catalog.
   The target mapping is added to the end of the *Target Mappings* table. The value of the *Referenced* column is a checkmark, which indicates that this mapping is a copy.

Next Steps

You can now delete the older version of the target mapping from the original catalog to ensure that there are no duplicate targets in the same catalog.

Related Information

[Configuring Target Mappings \[page 217\]](#)

1.3.4.6.4.3.3.2 Configuring Navigation Parameters with User-Specific Default Values

For SAP applications that can be launched with parameters, the launchpad provides the capability to pass user-specific default values rather than fixed values.

For example, if an application is launched with a parameter that refers to a user-specific default value for "cost center", the application can filter a list according to the cost center of the user who launched the application.

If supported by the applications assigned to users, they can not only enter a single value per parameter, but also sets of values and value ranges.

Parameters Referring to User-Specific Default Values in Target Mappings

Target mappings that are shipped by SAP may contain parameters that refer to user-specific default values.

In the launchpad designer, the names of these values look like this:

- `%%UserDefault.<parameterName>%%` for single-value parameters
- `%%UserDefault.extended.<parameterName>%%` for sets of values and value ranges

These parameters enable applications to be launched with user-specific default values rather than fixed values. Users can set their preferred values in the *Default Values* dialog box. These default values can be overridden, for example if the previous application or the tile that launches the application explicitly passes another value.

Default Values Dialog Box

Users can access the *Default Values* dialog box by choosing ▎▶ *User Preferences* ❯ *Default Values* ❳ from the Options menu of the SAP Fiori launchpad.

In this dialog, users see fields for all default values that are used in target mappings assigned to them.

For parameters that support sets of values and value ranges (`UserDefaults.extended`), the user can enter a primary value directly in the *Default Values* dialog box, as well as sets of values and value ranges in the *Additional Values* dialog box. Applications that only accept a single value receive the primary value. Applications that accept sets of values receive a set of values consisting of the primary value and additional values.

SAP application areas, like financials, can provide initial values as well as a value help. The data for the initial values and the value help originate from plug-ins, which are provided by SAP application areas as well.

Plug-ins for Parameters Referring to User-Specific Default Values

In the *Default Values* dialog box, initial values and the value help from the application system are only available if the user is assigned to the respective plug-in for user-specific default values.

SAP application areas provide such plug-ins for the SAP Fiori launchpad. These plug-ins are SAPUI5 components that are instantiated during the startup of the SAP Fiori launchpad.

Plug-ins for parameters referring to user-specific default values are displayed in the launchpad designer as target mappings with the following configuration:

Parameter	Values
Semantic Object	Shell
Action	plugin
Parameters	Name: sap-ushell-plugin-type
	Value: UserDefaults

You can assign plug-ins to users via catalogs, just like regular target mappings. For more information, see Activating Plug-Ins on the ABAP Platform [page 606].

OData Services

The plug-ins provided by SAP application areas typically use OData services to get initial values and a value help from the application system. For the initial values and the value help to work, it is important that these services are active and that all relevant users have the required authorizations for using these services. For more information on the required administration tasks, see the documentation for the respective application.

1.3.4.6.4.3.3.3 Passing User Settings to URL Applications

You can pass the values of user-specific parameters to applications that are integrated into the launchpad with application type "URL".

Application developers can then use these parameters in the target application. The effect of the parameters depends on the target application.

If you do the configuration as described below, the parameters are propagated to the target application as part of the application URL.

Entering Parameters in a Target Mapping

In the target mapping for your URL application, you can map user environment parameters to the respective URL parameter names. Enter the URL parameter names into the *Name* column, and the corresponding user environment parameters into the *Default Value* column, as shown in the example below:

Supported Parameters

The following parameters are supported:

Parameter Name	Default Value	Description
sap-ui-legacy-date-format	**%%User.env.sap-ui-legacy-date-format%%**	Only relevant if the SAP Fiori launchpad runs on the ABAP platform.
		Fixed value representing the date format on the ABAP platform. You can find a list of the possible values and their descriptions in the respective ABAP system. In transaction SE11, display the domain XUDATFM, then go to the *Value Range* tab.

Parameter Name	Default Value	Description
`sap-ui-legacy-number-format`	`%%User.env.sap-ui-legacy-number-format%%`	Only relevant if the SAP Fiori launchpad runs on the ABAP platform. Fixed value representing the number format for decimal notation on the ABAP platform. You can find a list of the possible values and their descriptions in the respective ABAP system. In transaction SE11, display the domain XUDCPFM, then go to the *Value Range* tab. Note that the first fixed value in the list is a space character.
`sap-ui-legacy-time-format`	`%%User.env.sap-legacy-time-format%%`	Only relevant if the SAP Fiori launchpad runs on the ABAP platform. Fixed value representing the time format on the ABAP platform. You can find a list of the possible values and their descriptions in the respective ABAP system. In transaction SE11, display the domain XUTIMEFM, then go to the *Value Range* tab.
`sap-language`	`%%User.env.sap-language%%`	The user's logon language, for example EN, ZH or ZF. Note that on the ABAP platform, the value of this parameter is composed of two uppercase letters. You can look up the list of possible values in transaction SU01, on the *Defaults* tab.
`sap-accessibility`	`%%User.env.sap-accessibility%%`	Indicates whether accessibility mode is active for the respective user. If accessibility mode is active, the value of this parameter is "X".
`sap-statistics`	`%%User.env.sap-statistics%%`	Indicates whether performance statistics in response headers are active. If performance statistics are active, the value of this parameter is "true".
`sap-theme-name`	`%%User.env.sap-theme-name%%`	The name of the theme, without the theme root.

Parameter Name	Default Value	Description
sap-theme	**%%User.env.sap-theme%%**	The name of the theme, including the theme root if the theme originates from a different server than the front-end server.
sap-theme-NWBC	**%%User.env.sap-theme-NWBC%%**	The name of the theme including the theme root. This parameter is primarily used for Web Dynpro applications and transactions. Depending on your system landscape, the `sap-theme-NWBC` parameter may include the theme root when the `sap-theme` parameter does not.

1.3.4.6.4.3.3.4 Changing LPD_CUST Entries for Navigation Targets

You can change existing entries for navigation targets in *Launchpad Customizing* (transaction `LPD_CUST`).

Context

Existing `LPD_CUST` entries can be reused in the target mapping configuration in the launchpad designer (application type *SAP Fiori App using LPD_CUST*). If you want to change any of these entries, you need to do that in transaction `LPD_CUST`.

> **i Note**
>
> Creating new `LPD_CUST` entries for the target applications to be launched from the SAP Fiori launchpad is not supported. Please use the SAP Fiori launchpad designer (see Configuring Target Mappings [page 217]) or the launchpad app manager (see Setting Up Technical Catalogs with the SAP Fiori Launchpad App Manager [page 132]).

Procedure

1. Start transaction `LPD_CUST`.
2. Open an existing launchpad.
3. On the *Change Launchpad Role* screen, choose an application inside the launchpad.

4. Make the required changes to the parameters depending on the application type:

- **SAPUI5 Application**

Parameter	Description
Link Text	Enter the application title.
Description	Enter a text that explains the target application in more detail.
Application Type	From the dropdown, choose *URL*.
URL	Enter the URL of an SAPUI5 application. Example: `/sap/bc/ui5_ui5/my_ui5_app`
System Alias	Enter the alias of the target system. For more information on how to configure a system alias, see Configuring Remote Systems [page 308].
Application Alias	Enter the matching purpose of your application. Enter a value that is unique in the scope of this launchpad instance. Do not use dashes ("-").
Target Application Parameters	Enter parameters that should be forwarded to the application. Example: **`sap-theme=sap_corbu`** > i Note > Do not use parameter names and values that include the special characters **%** and **!** If your applications have different names for the parameter that is passed between them, you define a parameter mapping.
Additional Information	Enter a key value pair with the following syntax: **`SAPUI5.Component=<YOUR_APP_COMPONENT_NAME>.`** (case-sensitive) Example: **`SAPUI5.Component=mycompany.myorg.myui5app`**

- **Web Dynpro ABAP Application**

Parameter	Description
Link Text	Enter the application title.

Parameter	Description
Description	Enter a description text for your application.
Application Type	From the dropdown, choose *Web Dynpro ABAP*.
Application	Enter the ID of a Web Dynpro ABAP application.
System Alias	Enter the alias of the target system.
Application Alias	Enter the matching purpose of your application. Enter a value that is unique in the scope of this launchpad instance. Do not use dashes ("-").
Target Application Parameters	Enter parameters that should be forwarded to the application. Example: `sap-theme=sap_corbu` > i Note > Do not use parameter names and values that include the special characters **%** and **!** If your applications have different names for the parameter that is passed between them, you define a parameter mapping.
Configuration	Enter the ID of a Web Dynpro ABAP configuration.

- SAP GUI Transaction

Parameter	Description
Link Text	Enter the application title.
Description	Enter a description text for your application.
Application Type	From the dropdown, choose *Transaction*.
Transaction Code	Enter a transaction code. Example: **SU01** for *User Maintenance*
System Alias	Enter the alias of the target system.
Application Alias	Enter the matching purpose of your application. Enter a value that is unique in the scope of this launchpad instance. Do not use dashes ("-").

Parameter	Description
Target Application Parameters	Enter parameters that should be forwarded to the application. Example: `sap-theme=sap_corbu` **i Note** Do not use parameter names and values that include the special characters **%** and **!** If your applications have different names for the parameter that is passed between them, you define a parameter mapping.

- **SuccessFactors Applications or Any Web Application**

Parameter	Description
Link Text	Enter the application title.
Description	Enter a description text for your application.
Application Type	From the dropdown, choose *URL*.
URL	Enter the URL of a SuccessFactors application or of any web application.
System Alias	Enter the alias of the target system.
Application Alias	Enter the matching purpose of your application. Enter a value that is unique in the scope of this launchpad instance. Do not use dashes ("-").
Target Application Parameters	Enter parameters that should be forwarded to the application. **i Note** Do not use parameter names and values that include the special characters **%** and **!** If your applications have different names for the parameter that is passed between them, you define a parameter mapping.

For more information on launchpad parameters, see ▶ *http://help.sap.com/nw75* ▶ *Application Help: SAP Library* ▶ *UI Technologies in SAP NetWeaver* ▶ *Web Dynpro ABAP and Floorplan Manager* ▶ *Launchpads* ▶ *Working with Launchpads at Design Time* ▶ *Application Types* ▶.

For more information on parameter mapping, see ▶ *http://help.sap.com/nw75* ❯ *Application Help: SAP Library* ❯ *UI Technologies in SAP NetWeaver* ❯ *Web Dynpro ABAP and Floorplan Manager* ❯ *Launchpads* ❯ *Working with Launchpads at Design Time* ❯ *Mapping Parameters in a Launchpad* ◀ .

Related Information

Setting Up Navigation [page 127]
Configuring Target Mappings [page 217]
Overwriting the System Alias for the Target Application [page 528]
Configuring Remote Systems in SM59

1.3.4.6.4.3.3.5 Keeping SAPUI5 Apps Alive

You can implement an option for SAPUI5 apps that keeps the app alive so that the state is maintained when the user navigates between applications.

Context

By default, the state of an app is not kept when the user navigates away. For SAPUI5 apps, you can define a parameter in the target mapping that keeps the app alive when users navigate to another app. When the user navigates back to the app, it opens in the same state in which the user left it.

Add a Parameter in Target Mapping

1. Open the Launchpad Designer.
2. In the target mapping configuration page for the app, add the following parameter:
   - Name: `sap-keep-alive`
   - Value: `true`

Functions for Component Lifecycle Management

You can use the following optional functions. They should be implemented as part of the `Component.js` file:

- `active()`: Trigger this function when the application is up and running. You can use, for example, to enable the application to open in letterbox mode.

  > ≡, Sample Code
  >
  > ```
  > active: function () {
  > sap.ushell.services.AppConfiguration.setApplicationFullWidth(true);
  > },
  > ```

- `suspend ()`: Trigger this function when navigating away from an application that is marked as 'keep alive'. Common use cases: to stop connections, for event handling, or for rendering processes.

- `restore ()`: Trigger this function when navigating back to an application that was kept alive. For example, to restore a connection.

1.3.4.6.4.3.4 Copying Tiles or Target Mappings

You can create references to tiles or target mappings defined in another catalog.

Context

Copying Tiles

You can copy tiles from one catalog to another catalog. The copied tile is created as a reference tile in the catalog. To copy a tile, proceed as follows:

1. Drag and drop the tile into the ("Create Reference" zone). A *Select Catalog* dialog box appears with a list of catalogs.
2. Choose the desired catalog where the tile is to be copied and choose the *OK* button.
   The copied tile is created as a reference tile in the desired catalog. You can distinguish between the reference tiles and the other standard tiles as the reference tile is displayed with a blue-colored dotted border.

Copying Target Mappings

To copy a target mapping and create a reference target mapping, proceed as follows:

Procedure

1. Choose an entry from the target mapping table. The action buttons *Configure*, *Create Reference*, and *Delete* appear at the footer of the screen.
2. Choose the *Create Reference* button. A *Select Catalog* dialog box appears with a list of catalogs.
3. Choose the desired catalog where the referenced target mapping is to be added and choose the *OK* button.

   A reference target mapping is added to the desired catalog. A (tick) icon appears in the *Reference* column of the referenced target mapping in the target mapping table.

   > i Note
   >
   > - You can copy a tile or target mapping from and add them to catalogs of type catalog page only.
   > - You cannot copy a tile or target mapping within the same catalog.
   > - When you configure a referenced tile or target mapping, it is decoupled from the original tile or target mapping.
   > - The referenced tile or target mapping does not contain any translated texts, once it is decoupled from the original. The translation is lost and it is necessary to translate the content again.

Related Information

1.3.4.6.4.3.5 Searching for Tiles or Target Mappings

A catalog can consist of numerous tiles and target mappings, and it is difficult to search for a relevant tile or target mapping from a large catalog.

Context

To address this issue, a search or filter option is provided in the catalog screen. You can search or filter for either tiles or target mappings based on the parameters on the configuration screen. The tiles or target mappings that contain the search query is displayed as the search results.

Searching for Target Mappings

To search for target mappings, you can use any of the following parameters as text in the search query. The parameters are as follows:

- **Semantic Object**
- **Action**
- **Role**
- **Instance**
- **Application alias**
- **Application Id**
- **Target mapping ID**

Searching for Tiles

To search for tiles, you can use all the parameters available on the configuration screen as the search query.

> **i Note**
>
> The search or filter of the tile or target mapping is restricted to the current catalog in which the search is performed.

1.3.4.6.4.4 Creating Transport Requests for User Changes

Context

You can launch the launchpad designer in two scopes, the customizing and the configuration scopes.

Customizing Scope	Configuration Scope
You have to create a customizing request.	You have to create a workbench request.
The user's actions such as creating or deleting catalogs or groups, adding tiles to or removing tiles from catalogs or groups are captured under the customizing request.	The user's actions such as creating or deleting catalogs or groups, adding tiles to or removing tiles from catalogs or groups are captured under the workbench request.
i Note If you are using the launchpad designer in customizing scope, you see the client information in **blue** on the header.	i Note If you are using the launchpad designer in configuration scope, you see the client information in **red** on the header.

Prerequisites

- You have created a customizing request and a workbench request for your user

  For more information, see Creating Customizing and Workbench Request [page 236]
- You have defined the default customizing request and workbench request and default package in your user parameters as follows:
  - In configuration scope, you have maintained a default workbench request and default package.
  - In customizing scope, you have maintained a default customizing request.

> i Note
>
> The settings can be made either from the backend or from the launchpad designer. Since the user can have many customizing or workbench requests, the user has to select one default customizing or workbench request under which the user's actions are captured.

For more information on how to define the default transport request, see
- Setting Default Transport Request Information from Back-End [page 237]
- Setting Default Transport Request Information from Launchpad Designer [page 236]

> i Note
>
> The changes are not recorded under the customizing request as the *Change w/o automatic recording, no transports allowed* option is enabled by default for all SAP systems with client 111. This default setting can be viewed using the transaction SCC4.

1.3.4.6.4.4.1 Creating Customizing and Workbench Request

To make sure your changes can be transported, you have to create a transport request.

Prerequisites

You have access to transaction SE01 to create transport requests. The transport request can be divided into customizing and workbench requests.

Context

Procedure

1. Go to transaction SE01.
2. Choose *Request* from the menu tab.
3. Choose *Create* from the submenu.

   A *Create Request* dialog box appears. You can choose either customizing request or workbench request.
4. Choose *OK*.

Depending on your selection, either a customizing request or a workbench request is created.

1.3.4.6.4.4.2 Setting Default Transport Request Information from Launchpad Designer

Context

You define the default transport request information in two scopes, the configuration and the customizing scopes, as follows:

Configuration Scope

1. Launch the launchpad designer in the configuration scope.
2. Choose the *Settings* button.

   The *Assign Transport Request* dialog box appears.
3. Select the workbench request from the *Workbench Request* dropdown list.
4. Enter the default package name in the *Package Name* field.
5. Choose *OK*.

You have saved the selected workbench request and entered a package name in the user parameters as your default values.

> i Note
>
> You can create a local object in the configuration scope by starting the name with Z, for example, ZServiceCatalog. In that case, setting the workbench request and the package is not required

Customizing Scope

1. Launch the launchpad designer in the customizing scope.
2. Choose the *Settings* button.

   The *Assign Transport Request* dialog box appears.
3. You can choose either the *Local Object* or any customizing request from the *Customizing Request* dropdown list.

   > i Note
   >
   > • If you choose *Local object*, none of the actions the user performs are captured under any request. This option is provided when the user creates any test objects in which the user actions do not need to be captured.
   > • If you choose a customizing request, all the user actions are captured under the selected request.

4. Choose *OK*.

You have saved the selected customizing request in the target parameters as your default values.

> i Note
>
> The settings made in the launchpad designer are visible in the back end user parameters.

1.3.4.6.4.4.3 Setting Default Transport Request Information from Back-End

Context

1. Go to transaction SU2.
2. Choose the *Parameters* tab on the *Maintain User Profile* screen.
3. Maintain the following parameters:

Parameters	Description
/UI2/WDC_DEVCLASS	Package name, which contains the page.

Parameters	Description
/UI2/WD_TRKORR_CONF	Workbench request to which the workbench transport for the page has to be submitted.
/UI2/WD_TRKORR_CUST	Customizing request to which the customizing transport for the page has to be submitted.

You have saved the selected customizing request in the target parameters as your default values.

> **i Note**
>
> Transport request and package set in user parameters are visible in launchpad designer settings.

1.3.4.7 Setting Up Business Catalogs

Business catalogs contain a collection of tiles and target mappings relevant for a business role.

The content of the business catalog is a subset of the content of the technical catalog. This subset reflects the requirements of a specific business user.

1.3.4.7.1 Setting Up Business Catalogs with the Launchpad Content Manager

Administrators use the SAP Fiori launchpad content manager to browse launchpad content and to set up business catalogs according to their needs.

System Requirements

For more information on the system requirements for using the launchpad content manager, see SAP Note 2772270 .

Feature Scope

The launchpad content manager complements the launchpad app manager. It offers the following features:

Feature	Description
Exploring Launchpad Content [page 241]	For example, you can explore available tile/target mapping combinations or search for tiles and target mappings across all available catalogs.

Feature	Description
Maintaining Catalogs in the Launchpad Content Manager [page 247]	Create, copy, rename or delete catalogs. Add or remove reference tiles and target mappings to/from the catalogs (also in a mass operation).
Maintaining Roles in the Launchpad Content Manager [page 254]	Check the role assignment for catalogs, spaces and groups. Copy roles and add or remove catalogs, spaces and groups.
Displaying the Service Activation Status for Apps [page 258]	Check if the OData and Internet Communication Framework (ICF) services relevant to run SAPUI5 and Web Dynpro apps from the launchpad are activated.
Displaying Issues with Launchpad Content [page 261]	The launchpad content manager displays content issues detected during catalog loading and matching of tiles and target mappings.

The following is **not** supported in the launchpad content manager:

- Changing cross-client catalogs in the customizing scope.
  This is supported in the launchpad designer but not recommended because it can lead to inconsistencies.
- Creating, changing or deleting original tiles and target mappings.
  Please use the launchpad app manager instead.
- Managing custom tiles (displaying information about custom tiles and adding them to catalogs).
  Please use the launchpad designer instead.

You can open selected items in other design-time tools:

- You can open a selected catalog in the launchpad designer or launchpad app manager depending on where the original was created.
- You can open a selected group in the launchpad designer.
- You can open a selected space in the *Manage Launchpad Spaces* app.
- You can open a selected role in *Role Maintenance* (transaction PFCG).

See Tools for Setting Up Launchpad Content [page 112].

UI Overview

In the launchpad content manager, tiles and target mappings in the same catalog which refer to the same intent and target application are automatically matched and displayed in one row. See Tile/Target Mapping Combinations [page 243]. Depending on your focus, you can display them in the following tabs:

- *Catalogs*:
  The upper table displays all catalogs available in the ABAP system to which you are logged on. When you select a catalog, all tiles and target mappings assigned to it are displayed in the lower table. You can assign additional reference tiles and target mappings to the selected catalog or remove reference tiles and target mappings that are not needed. In addition, you can create, delete, copy and rename catalogs.
- *Tiles/Target Mappings*:
  The upper table displays available tile/target mapping combinations. Selecting an entry shows the catalogs to which the selected tiles and target mappings are assigned in the lower table. You can add a selected combination of tile and target mapping as references to another catalog or remove reference tiles and target mappings from a selected catalog.

> **i** Note
>
> You can adjust the displayed list content, for example by adding or removing columns or setting filters. See
> Working with SAP List Viewer (ALV)

Related Information

Best Practices and Typical Scenarios for Setting Up Launchpad Content [page 114]
Tools for Setting Up Launchpad Content [page 112]

1.3.4.7.1.1 Running the Launchpad Content Manager

The launchpad content manager allows you to make system-wide or client-specific changes.

Prerequisites

The launchpad content manager uses the user-independent cache to load persisted content. Therefore, the synchronization of the user-independent cache should be scheduled periodically to ensure consistent data for catalogs and groups. See Scheduling the Synchronization of User-Independent Cache [page 28].

The launchpad content manager only loads persisted content when the session is started. During a running session, there is no further check whether the displayed entities are still up to date. This can lead to data inconsistencies when changes are made in another session of the launchpad content manager or in the launchpad designer.

Transactions for the Launchpad Content Manager

System-wide changes are stored in the configuration scope and client-specific changes are stored in the customizing scope. Depending on the scope of your changes, use one of the following transactions:

Transaction Code	Description
/UI2/FLPCM_CONF	Allows you to make system-wide changes (configuration scope).
	SAP delivers predefined technical catalogs in the configuration scope.
	ⓘ Note The launchpad content manager allows you to create business catalogs in the configuration scope, which enables the cross-client usage of these business catalogs. However, this is only recommended in case no client-specific content is required.
/UI2/FLPCM_CUST	Allows you to make changes for the current client (customizing scope). These changes supersede configuration settings.
	Displays content available on the current client. In addition, cross-client content is displayed but cannot be edited here. You can filter the *Scope* column for "CUST" to display only client-specific content.
	ⓘ Note If you want to change cross-client content in the customizing scope, you can use the launchpad designer. However, this is not recommended since it can lead to outdated catalogs in the customizing scope. See Scope-Related Catalog Issues [page 261].

End users can adjust content defined by the administrator and save their preferences in the personalization scope. These settings supersede customizing and configuration settings.

Related Information

Scopes for Adapting Launchpad Content [page 111]

1.3.4.7.1.2 Exploring Launchpad Content

The launchpad content manager helps you to explore existing content that you can use as the basis for your own catalogs.

You can adjust the displayed list content by adding more columns. Select a row and choose *Details* in the toolbar above the table to get all available information about a tile/target mapping combination or a catalog.

For example, you can get information to the following questions:

- Which tiles and target mappings in a given catalog match according to their intent (combination of semantic object and action) and target application?
- Which matching tiles and target mappings are assigned to a given catalog?
- To which catalogs is a tile/target mapping combination assigned?
- Is the displayed tile or target mapping an original or a reference? In which catalog were original tiles and target mappings created?

Related Information

1.3.4.7.1.2.1 Searching for Launchpad Content

The launchpad content manager offers a search across all catalogs or tile/target mapping combinations available in the system in which you are logged on.

Use the *Search Catalogs* and the *Search Tiles/Target Mappings* field to search across all visible columns of the upper table of the respective tab. The search result list displays all items where at least one column is filled with a text that contains the string you entered.

> **i Note**
>
> You can perform a wildcard search by adding the asterisk (*) symbol anywhere in the search string. For example, entering `my*cata*g*` would return `mycatalog`, `my_catalog56`, `my_brand_new_cata_good`, but not `this_is_mycatalog` as a search result. If you don't specify any wildcards, the system automatically adds an asterisk before and after your search input.

You can use the standard SAP List Viewer (ALV) search and filter options to further restrict your search. See Functions in the List.

Example

You enter the string `sales order` in the *Search Tiles/Target Mappings* field. The search result table displays all tile/target mapping combinations where at least one visible column is filled with a text that contains the string you entered (for example, one item with the tile title `Manage Sales Orders` and one item with the tile subtitle `sales order_1`). You add a filter for column *Application Type* to only display tile/target mapping combinations that were defined for SAPUI5 apps. Select a tile/target mapping combination and the catalogs to which it is assigned are displayed in the lower table. You can now select a catalog in the lower table and choose *Catalog View* to find out which other tiles and target mappings are assigned to this catalog.

1.3.4.7.1.2.2 Tile/Target Mapping Combinations

The launchpad content manager automatically maps matching tiles and target mappings and displays them in the same row.

To make an app available to launchpad end users, administrators configure a tile to define its visual representation on the launchpad home page, and a target mapping to define the navigation. See About Launchpad Content [page 103].

To create or configure original tiles and target mappings, you use the launchpad designer or the launchpad app manager. The launchpad content manager allows you to reuse existing tiles and target mappings by creating references to them.

The launchpad content manager displays matching tiles and target mappings as a combination in the same row if both of the following applies:

- They have the same intent.
- They exist in the same catalog (as a reference or as an original).

At runtime, however, tiles and target mappings can also be displayed as a combination if they only have the same intent. They do not necessarily have to exist in the same catalog.

The image below visualizes the relation between tiles and target mappings:
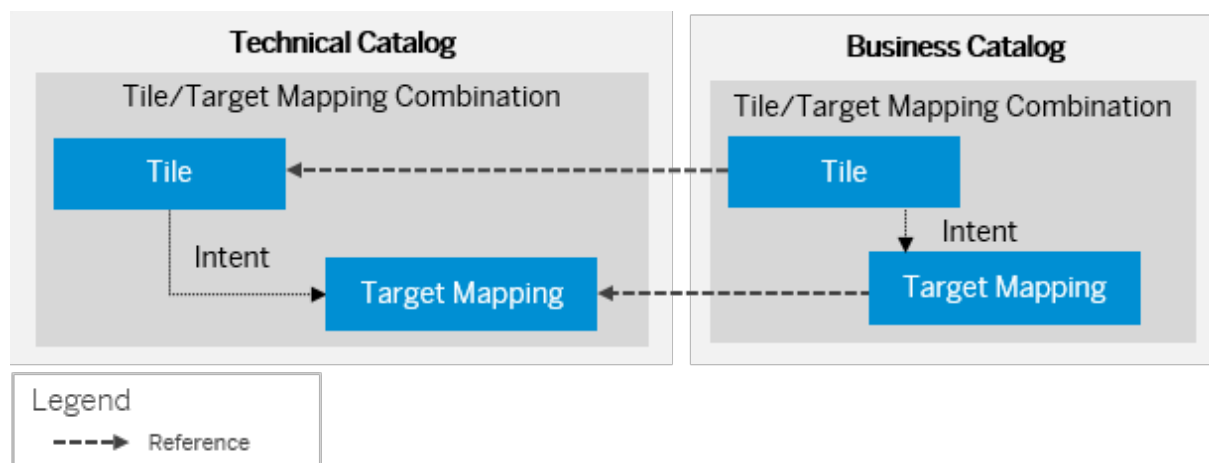
The *Tile/Target Mapping Matching* field in the list of tiles and target mappings indicates whether a given row displays a tile/target mapping combination or a single tile or target mapping without matching partner:

Value	Description
Tile + TM	The entry is a tile/target mapping combination.
	If several tiles have the same intent as a single target mapping and vice versa, a new row is created for each combination.
Tile Only	There is no matching target mapping in the same catalog.
	For example, when an application of type URL should be launched, no target mapping is needed.

Value	Description
TM Only	There is no matching tile in the same catalog.
	For example, when an app should not be launched from the launchpad home page but from another app (app-to-app navigation), no tile is needed.

The *Reference Detail* field in the list of catalogs in the *Tiles/Target Mappings* tab indicates whether the tile/target mapping was originally defined in a catalog or only exists as a reference. See References and their Originals [page 244].

1.3.4.7.1.2.3 References and their Originals

The launchpad content manager allows you to find out in which catalog a tile or target mapping was originally defined.

In the launchpad content manager, you can copy a catalog to create a new catalog with references to the tiles and target mappings in the original catalog. Alternatively, you can create a catalog from scratch and add references to tiles and target mappings to it.

To create or configure original tiles and target mappings, you use the launchpad app manager or the launchpad designer. Tile and target mapping can have different original catalogs.

To get information on the originals for a tile/target mapping combination and the catalog in which they were defined, you have the following options:

- In the list of catalogs on the *Tiles/Target Mappings* tab or in the list of tiles/target mappings on the *Catalogs* tab, check the *Reference Detail* field to get an overview of the catalog assignments and whether the selected tile/target mapping combination was created in the catalog or is only a reference:

Value	Description
Original (Tile + TM)	Tile and target mapping were originally defined in this catalog.
Original (Tile Only)	The tile was originally defined in this catalog. There is no matching target mapping in this catalog.
Original (TM Only)	The target mapping was originally defined in this catalog. There is no matching tile in this catalog.
Reference (Tile + TM)	Tile and target mapping only exist as a reference in this catalog. The originals are located in another catalog.
Reference (Tile Only)	The tile only exists as a reference in this catalog. The original tile is located in another catalog. There is no matching target mapping in this catalog.
Reference (TM Only)	The target mapping only exists as a reference in this catalog. The original target mapping is located in another catalog. There is no matching tile in this catalog.

Value	Description
Mixed (Tile Original, TM Reference)	The tile was originally defined in this catalog. The target mapping only exists as a reference in this catalog. The original target mapping is located in another catalog.
Mixed (Tile Reference, TM Original)	The target mapping was originally defined in this catalog. The tile only exists as a reference in this catalog. The original tile is located in another catalog.

- In the list of tiles/target mappings, check the fields *Original Tile ID* and *Original Target Mapping ID* to find the IDs of the originals.
- In the list of tiles/target mappings, check the fields *Original Tile Catalog ID* and *Original Target Mapping Catalog ID* to find the catalogs in which the originals are located.
- In the list of tiles/target mappings, check the field *Original Tile Repository* to find out with which tool the original tile was created and where it is stored:

Value	Description
Web Dynpro	The original tile was created in the launchpad designer and is stored in the Web Dynpro repository.
Replicated	The original tile was created in the backend using the launchpad app manager and replicated to the front-end server.

- In the list of tiles/target mappings, check the field *Original Target Mapping Repository* to find out with which tool the original target mapping was created and where it is stored:

Value	Description
Web Dynpro	The original target mapping was created in the launchpad designer and is stored in the Web Dynpro repository.
Replicated	The original target mapping was created in the backend using the launchpad app manager and replicated to the front-end server.
LPD_CUST	The original target mapping was created with transaction LPD_CUST. i Note This approach is obsolete. See SAP Note 2614740 .

If the original tile or target mapping from which a reference tile or target mapping was created is deleted, the reference is broken. In the launchpad content manager, this is indicated as a content error in the *Status* column of the tile/target mapping list. See Issues with Tiles and Target Mappings [page 263].

1.3.4.7.1.2.4 Where-Used Checks for Tiles

The launchpad content manager allows you to find out to which page or group a tile or target mapping was assigned.

Use Cases

With the where-used check you can do the following:

- Display pages and groups that contain the selected tile or target mapping
- Navigate to the SAP Fiori launchpad designer or to the *Manage Launchpad Pages* app to adjust a page or group (e.g., deleting a tile)

Checking the role assignments

If an end user cannot see a certain tile in the launchpad, it may be due to a lack of role assignments. With the where-used check, the admin can see to which pages, groups, or business catalogs the tile was assigned. The admin can then simply check if this page or group was also assigned to the business role that the user needs to view the tile on their launchpad home page.

Checking the references

If an end user gets the message `Cannot load tile` in the launchpad, it may be due to a broken reference between the group tile and the business catalog tile. See Issues with Tiles, Groups or Catalogs [page 632] for more details.

A tile or target mapping cannot be removed from a catalog if it is still referenced in a group or a page. The where-used check can be used to see where this tile or target mapping is still referenced in order to remove it. See Adding and Removing Catalog Content [page 250].

Display Usages In Pages

Use the dropwdown button *Display Usages in Pages* in the lower table of the *Catalogs* tab to find out to which pages a tile or target mapping was assigned to.

A pop-up appears and shows a list of pages containing the selected tile or target mapping. You have the option to choose *Open in Manage Launchpad Pages*. It will directly navigate you to the *Manage Launchpad Pages* app to make adjustments if needed.

In the pop-up you can switch between the views *All Usages of Originals* and *Direct Usages*. Tiles and target mappings can be maintained in technical catalogs (originals) and also be referenced in business catalogs (references). Both originals and references can be assigned to a page.

All Usages of Originals

Use this view to check which page needs to be assigned to a business user so that the selected tile and target mapping combination is visible in the launchpad.

Pages are listed in this table if one of the following applies:

- The selected tile or target mapping (original or reference) is assigned to the page
- The selected tile or target mapping is an original and has a reference assigned to the page
- The selected tile or target mapping refers to an original which has another reference assigned to the page

Direct Usages

Use this view to check if you can, for example, delete a tile and target mapping combination without breaking references. If the selected tile or target mapping is used in a page in this table, it can't be removed from the catalog in the launchpad content manager.

Pages are listed in this table if the following applies:

- The selected tile or target mapping (original or reference) is assigned to the page

Display Usages in Groups

Use the dropwdown button *Display Usages in Groups* in the lower table of the *Catalogs* tab to find out to which groups a tile (and links) was assigned to.

A new window appears and shows a list of groups containing the selected tile. You have the option to choose *Open in Designer*. It will directly navigate you to the launchpad content designer to make adjustments if needed.

1.3.4.7.1.3 Maintaining Catalogs in the Launchpad Content Manager

Administrators use the launchpad content manager to set up catalogs according to their needs.

Related Information

Searching for Launchpad Content [page 242]
Creating Catalogs [page 248]
Copying Catalogs [page 249]
Adding and Removing Catalog Content [page 250]
Renaming Catalogs [page 252]
Deleting Catalogs [page 252]
Transporting Catalogs [page 253]

1.3.4.7.1.3.1 Read-Only Catalogs

Some catalogs cannot be edited in the launchpad content manager.

A marked checkbox in the *Read-Only* column of the catalog list indicates that the launchpad content manager does not allow you to change the catalog. Here is how you can work with read-only catalogs in the launchpad content manager:

Why the Catalog is Read-Only	What you Can Do
The catalog was created in the backend using the launchpad app manager and replicated to the front-end server. i Note A marked checkbox in the *Backend Catalog* column indicates that the catalog was created in the backend and replicated to the front-end server.	Create a new catalog with references to the tiles and target mappings in the backend catalog.
The catalog is cross-client, i.e. stored on the configuration scope. You launched the client-specific launchpad content manager (customizing scope).	To change the cross-client catalog, launch transaction `/ui2/flpcm_conf`. To make changes for the current client only, stay in transaction `/ui2/flpcm_cust` and copy the catalog.
You are not logged on in the original language of the catalog.	Re-start the launchpad content manager in the original language of the catalog.
The catalog ID starts with "SAP" and the system is not an SAP system.	Create a new catalog with references to the tiles and target mappings in the SAP catalog.

1.3.4.7.1.3.2 Creating Catalogs

You can set up your content from scratch by creating a new catalog and adding content to it.

Procedure

1. In the *Catalogs* tab, choose *Create*.
2. Enter an ID and a title.

   i Note

   The ID must follow these conventions:
   - Max. 35 characters
   - Allowed characters: uppercase letters [A-Z], numbers [0-9], underscore [_]
   - May start with a reserved namespace (for example "/UI2/CATALOG_EXAMPLE")
     See Development in Reserved Namespaces.

Results

An empty catalog is created. To add content to the catalog, see Adding Reference Tiles and Target Mappings to a Catalog [page 250].

1.3.4.7.1.3.3 Copying Catalogs

You can create a copy of a catalog using the configuration or the customizing scope and then adapt it to your needs by changing the title or by adding or removing reference tiles and target mappings.

Context

When you copy a catalog, the copied catalog references the tiles and target mappings in the original catalog. Any changes to the tile or target mapping configuration in the original are reflected in the reference. You can change the catalog without breaking the reference of the tiles and target mappings. You can make a copy of a copied catalog.

> i Note
>
> You can only copy catalogs that do not contain content issues. See Issues with Tiles and Target Mappings [page 263].

Procedure

1. Search for the catalog you want to copy. You can use the *Search Catalogs* field to search across all visible columns. See Searching for Launchpad Content [page 242].
2. In the *Catalogs* tab, select the catalog to display its content.
3. Choose *Copy*.
4. Enter an ID and a title for the new catalog.

   > i Note
   >
   > The ID must follow these conventions:
   > - Max. 35 characters
   > - Allowed characters: uppercase letters [A-Z], numbers [0-9], underscore [_]
   > - May start with a reserved namespace (for example "/UI2/CATALOG_EXAMPLE")
   >   See Development in Reserved Namespaces.

Results

A new catalog containing references to the tiles and target mappings in the original catalog is created.

You can adjust the catalog by changing its title or by adding or removing reference tiles and target mappings.

Related Information

Adding and Removing Catalog Content [page 250]

1.3.4.7.1.3.4 Adding and Removing Catalog Content

In the launchpad content manager, you can add or remove reference tiles and reference target mappings to catalogs.

You can define whether only the tiles or only the target mappings should be added to or removed from the catalog.

> **i Note**
>
> Adding and removing content is disabled for read-only catalogs. See Read-Only Catalogs [page 248].

In the launchpad content manager, you cannot create or delete original tiles and target mappings. Use the launchpad designer for this action. See Adding or Removing Tiles or Target Mappings.

Related Information

References and their Originals [page 244]

1.3.4.7.1.3.4.1 Adding Reference Tiles and Target Mappings to a Catalog

> **i Note**
>
> You can only add tiles and target mappings that do not contain issues. See Issues with Tiles and Target Mappings [page 263].

To add reference tiles and target mappings to a catalog, you can first search for a catalog and then select the content you want to add:

1. In the *Catalogs* tab, select the catalog to which you want to add reference tiles or target mappings.
2. In the toolbar of the lower table, choose *Add Tiles/Target Mappings* > *Add Tiles/TMs to Selected Catalog*.
3. In the screen that opens, select one or more rows from the list of available tile/target mapping combinations.
4. Choose one of the following buttons:
   - *Add Tile/TM Reference* to add the tile/target mapping combination
   - *Add Tile Reference* to only add the tile
   - *Add TM Reference* to only add the target mapping

To add reference tiles and target mappings to another catalog, you can perform the following steps:

1. In the *Catalogs* tab, select the catalog in which your desired tiles or target mappings are located in.
2. In the lower table, select one more tiles or target mappings.
3. In the toolbar of the lower table, choose *Add Tiles/Target Mappings* > *Add Selected Tiles/TMs to Other Catalog*.
4. In the screen that opens, select the catalog to which the tiles and target mappings should be added to.
5. Choose one of the following buttons:
   - *Add Tile/TM Reference* to add the tile/target mapping combination
   - *Add Tile Reference* to only add the tile
   - *Add TM Reference* to only add the target mapping

Alternatively, you can first search for the relevant tiles and target mappings and then select the catalog to which you want to add them:

1. In the *Tiles/Target Mappings* tab, select one or more rows from the upper table.
2. Choose *Add Reference to Catalog*.
3. In the screen that opens, select the catalog to which the tiles and target mappings should be added.
4. Choose one of the following buttons:
   - *Add Tile/TM Reference* to add the tile/target mapping combination
   - *Add Tile Reference* to only add the tile
   - *Add TM Reference* to only add the target mapping

1.3.4.7.1.3.4.2 Removing Reference Tiles and Target Mappings from a Catalog

To remove reference tiles and target mappings from a catalog, you can first search for a catalog and then select the content you want to remove:

1. In the *Catalogs* tab, select a row from the upper table. The lower table displays the content of the selected catalog.
2. In the lower table, select one or more rows and choose *Remove Tiles/Target Mappings*.

Alternatively, you can first search for the relevant tiles and target mappings and then select the catalog from which you want to remove them:

1. In the *Tiles / Target Mappings* tab, select one row from the upper table. The lower table displays the catalogs to which the selected tiles and target mappings are assigned.

2. In the lower table, select the relevant catalogs and choose *Remove*.

> **i Note**
>
> Tiles and target mappings that are still referenced in groups or pages can't be deleted. You need to first remove the references from the groups and pages before you delete them from the catalog. See Where-Used Checks for Tiles [page 246].

1.3.4.7.1.3.5 Renaming Catalogs

You can change the title of a catalog.

Context

> **i Note**
>
> This option is disabled for read-only catalogs. See Read-Only Catalogs [page 248].

Procedure

1. In the *Catalogs* tab, choose the catalog you want to rename. You can use the *Search Catalogs* field to search across all visible columns. See Searching for Launchpad Content [page 242].
2. Choose *Change Title*.
3. Adjust the title as needed.

1.3.4.7.1.3.6 Deleting Catalogs

In the launchpad content manager, you can delete catalogs that do not contain original tiles or target mappings.

Context

> **i Note**
>
> Catalogs can only be deleted if they do not contain any original tiles or target mappings. See References and their Originals [page 244].

Procedure

1. Search for the catalog you want to delete. You can use the *Search Catalogs* field to search across all visible columns. See Searching for Launchpad Content [page 242].

2. In the *Catalogs* tab, choose a catalog from the upper table.

3. Choose *Delete*.

1.3.4.7.1.3.7 Transporting Catalogs

Use the launchpad content manager to transport catalogs.

If you want to transport catalogs, make sure a relevant transport request is available:

Scope	Required Transport Request
Launchpad Content Manager: Client-Specific (Customizing) (transaction `/UI2/FLPCM_CUST`)	Customizing request
Launchpad Content Manager: Cross-Client (Configuration) (transaction `/UI2/FLPCM_CONF`)	Workbench request

Transports for Client-Specific Catalogs

The client-specific launchpad content manager (transaction `/UI2/FLPCM_CUST`) takes the settings in *Display View "Clients": Overview* (transaction SCC4) into account. These settings control whether objects can be changed and whether the changes can be transported.

Setting in SCC4	Description
Changes without automatic recording	The launchpad content manager does not display a prompt for customizing request when you make changes.
Automatic recording of changes	The launchpad content manager displays a prompt for customizing request when you make changes.
No changes are allowed	The launchpad content manager runs in read-only mode.
Changes w/o automatic recording, no transports allowed	The launchpad content manager allows you to make changes but not to transport them.

Manually triggering transports

You can use this feature to trigger the transport of changes when the option *Changes without automatic recording* is set for the client. This feature also allows you to transport catalogs independent of a change, for example, to make catalogs that were created locally in the launchpad designer available in other clients or systems.

The option to manually trigger transports is available if the following prerequisites are met:

- In transaction SCC4, the client role is set to *Customizing*.
- In transaction SCC4, the option *Changes w/o automatic recording, no transports allowed* is *not* set for your client.
- The catalog you want to transport is client-specific (customizing scope). This is indicated in the *Scope* column.

To trigger the transport, proceed as follows:

1. In the *Catalogs* tab, choose a catalog from the upper table.
2. In the toolbar above the upper table, choose *Transport*.
3. In the dialog box that appears, select the customizing request that should be used for the transport.

Transports for Client-Independent Catalogs

When you trigger a change in the client-independent launchpad content manager (transaction `/UI2/FLPCM_CONF`), a dialog box is displayed in which you define whether you want to save your changes locally or transport them to other systems. To transport your changes, select a package and a workbench request.

Related Information

Creating Customizing and Workbench Request [page 236]

1.3.4.7.1.4 Maintaining Roles in the Launchpad Content Manager

Check the role assignment for catalogs in the launchpad content manager.

> i Note
>
> Since roles are client-dependent, this feature is only available for the client-specific launchpad content manager (transaction `/UI2/FLPCM_CUST`).

Roles allow administrators to limit access to launchpad content to defined users.

Here's what you can do:

- Check which catalogs are assigned to a selected role.
  The *Roles* tab displays all roles that exist in the client you are logged on and that have catalogs assigned. Select a role in the upper table to display all catalogs assigned to it in the lower table.
  You can also drill down to see the tiles/target mappings assigned to each catalog by choosing the *Catalog View* button above the lower table.
- Check to which roles a selected catalog is assigned.
  In the *Catalogs* tab, select a catalog and make sure the *Show Usage in Roles* toggle button is switched on. The lower table displays all roles to which the selected catalog is assigned.

Select a role in the lower table and choose the *Role View* button to find out which other catalogs are assigned to the role.

- Assign a role to a catalog in the *Catalogs* view. You can only assign single roles to a catalog. If you choose a derived or a composite role, you will get a message that this kind of role type is not supported.
See Assigning or Removing Catalogs to/from Roles [page 257].

- Open a role in *Role Maintenance* (transaction **PFCG**) to change its configuration.
The launchpad content manager does not allow you to edit roles. To change a role (for example, add or remove catalogs), select the role in the launchpad content manager and choose *Open in PFCG*. A new window is opened displaying the role in the *Menu* tab of the *Role Maintenance*. Make sure to click the *Refresh Roles* button in the launchpad content manager to see any changes you made.
See Configure Roles for Catalogs [page 329].

- Check if all services required for the content assigned to the role are active.
See Displaying the Service Activation Status for Apps [page 258].

The launchpad content manager does not indicate whether a given role is a composite role or a single role:

- If a composite role only contains other roles, these roles are listed as single roles in the launchpad content manager. The composite role itself is not displayed.
- If a composite role only contains assigned catalogs, the role is listed as single role in the launchpad content manager.
- If a composite role contains both catalogs and roles with catalogs assigned, the composite role is listed as single role in the launchpad content manager.

> **i Note**
>
> SAP doesn't recommend composite roles that contain both directly assigned catalogs and roles with catalogs assigned to it. The end user can only see the catalogs that are assigned directly to this composite role. The catalogs assigned to single roles within a composite role are not visible in the launchpad.

Related Information

About Launchpad Content [page 103]

1.3.4.7.1.4.1 Copying Roles

You can create a copy of a role using the customizing scope and then adapt it to your needs by adding or removing catalogs, spaces or groups.

Context

Procedure

1. On the *Roles* tab, select the relevant role.

   You can use the *Search Roles* field to search across all visible columns. See Searching for Launchpad Content [page 242].

2. Choose *Copy Role*.

3. Enter an ID and a title for the new role.

   > **i** Note
   >
   > The ID must follow these conventions:
   > * Max. 35 characters
   > * Allowed characters: uppercase letters [A-Z], numbers [0-9], underscore [_]
   > * May start with a reserved namespace (for example "/UI2/CATALOG_EXAMPLE")
   >   See Development in Reserved Namespaces.

Results

A new role with the same assignments of catalogs, groups and spaces as the original is created.

You can adjust the role by adding or removing catalogs, groups and spaces.

Related Information

Adding and Removing Catalogs, Groups and Spaces to/from Roles [page 256]

1.3.4.7.1.4.2 Adding and Removing Catalogs, Groups and Spaces to/from Roles

1. On the *Roles* tab, select the relevant role.
   You can use the *Search Roles* field to search across all visible columns. See Searching for Launchpad Content [page 242].
2. In the toolbar above the upper table, choose the right view mode:
   * Choose *Show Catalogs* if you want to add catalogs to a role or remove them from the role.
     The toolbar above the lower table will provide the options *Add Catalog* and *Remove Catalogs*.
   * Choose *Show Groups* if you want to add groups to a role or remove them from the role.
     The toolbar above the lower table will provide the options *Add Group* and *Remove Groups*.
   * Choose *Show Spaces* if you want to add spaces to a role or remove them from the role.
     The toolbar above the lower table will provide the options *Add Space* and *Remove Spaces*.

3. To add a catalog, group or space, click the respective button in the toolbar above the lower table and provide the ID of the item you want to add. You can use the value help for this.

4. To remove a catalog, group or space, select the item that should be removed in the table and click the respective button in the toolbar above the lower table.

Related Information

Configuring Roles for Catalogs, Spaces and Groups [page 328]

1.3.4.7.1.4.3 Assigning or Removing Catalogs to/from Roles

To assign catalogs to a role in the launchpad content manager, perform the following steps:

1. On the *Catalogs* tab, select the relevant catalog.
   You can use the *Search Catalogs* field to search across all visible columns. See Searching for Launchpad Content [page 242].

2. To assign a catalog to a role, choose *Assign Role* in the toolbar above the lower table and provide the ID of the role you want to add. You can use the value help for this.

> **i Note**
>
> You can only assign single roles to a catalog. If you choose a derived role or a composite role, you will get a message that this kind of role type is not supported.

3. To remove a catalog from a role, select the item that should be removed in the table and click *Remove Role* in the toolbar above the lower table.

Related Information

Maintaining Roles in the Launchpad Content Manager [page 254]

1.3.4.7.1.5 Displaying the Service Activation Status for Apps

Check the activation status of the OData and Internet Communication Framework (ICF) services relevant for SAPUI5 and Web Dynpro applications in the launchpad.

Context

SAPUI5 and Web Dynpro applications as well as dynamic tiles need services to retrieve data from the back-end system.

The launchpad content manager helps you to find out if the relevant services are active:

- For SAPUI5 apps:
  Are the required OData services active?
  Are the required ICF services active (service URLs starting with `/sap/bc/ui5_ui5/`)?
- For Web Dynpro apps:
  Are the required ICF services active (service URLs starting with `/sap/bc/webdynpro/`)?
- For dynamic tiles:
  Are the required OData services active?

The launchpad content manager checks app-specific services for tiles and target mappings and their related target applications. Generic services are not checked.

Checks are not supported for the following:

- Apps running on a remote system
- Custom tiles (including SAP Smart Business)
- OData V4 services
- Tiles/target mappings with content issues

How to Trigger the Check

You can run the check on the level of tiles/target mappings, catalogs and roles or globally for all items displayed in the launchpad content manager.

You trigger the check from the *Check Services* menu in the upper toolbar of the *Tiles/Target Mappings*, *Catalogs* or *Roles* tab.

> **i Note**
>
> Since roles are client-dependent, the check on role level is only available for the client-specific launchpad content manager (transaction `/UI2/FLPCM_CUST`).

Depending on the granularitiy of your check, choose one of the following options:

Option	Description
Check Services	• Displays a combined status of OData and ICF services. • Displays the service activation status for tiles/target mappings, catalogs or roles and their related target applications.
Check and Show Services	• Displays which services are used by the selected launchpad content on separate tabs for OData and ICF services. • Displays the activation status for each service. • Allows you to navigate to the relevant transactions to activate the services.
Check Services for All Objects	• Displays a combined status of OData and ICF services. • Displays the service activation status for all items displayed in the launchpad content manager.

What to Do with the Check Result

The *Service Activation Status* column displays the check result.

Status	Details	What to Do
⚪⚪🟩 All services are active	All services relevant for the selected launchpad content are active.	No actions needed.
⚪🔺⚪ Check incomplete	At least one service is inactive and/or the check for at least one service could not be performed.	Drill down to the level of tile/target mapping combination to find out the status on this level.

Status	Details	What to Do
🔴⚪⚪ Not all services are active	At least one service relevant for the selected launchpad content is inactive.	1. (Optional) Drill down to the level of tile/target mapping combination to find out the status on this level. 2. Select the affected entries and choose *Check and Show Services*. 3. If services with inactive status are displayed on the *OData Services* tab, choose *Activate and Maintain Services* to activate the service. See Activate OData Services for Individual Apps. 4. If services with inactive status are displayed on the *ICF Services* tab, choose *Open in SICF* to activate the service. See Activate ICF Services of Web Dynpro Apps and Activate ICF Services of SAPUI5 Application. ⓘ Note To automate the service activation, the following task lists are available: • SAP_FIORI_FCM_CONTENT_ACTIVATION See SAP Note 2813396🔗. • SAP_GATEWAY_ACTIVATE_ODATA_SRV • SAP_BASIS_ACTIVATE_ICF_NODES
⚪⚪⚪ No checks needed	No checks are required for any of the apps used in the selected launchpad content because: • The selected entry refers to an application type for which no checks are provided (for example, SAP GUI, WebClient UI application, or URL). For these application types, you do not need to activate app-specific services. • You triggered the check for an empty catalog.	No actions needed.

Status	Details	What to Do
Remote content	The selected item refers to Web Dyn-pro applications running on a remote system. Checks are only supported for local systems.	You have to manually check if the ICF services are active in the back-end system. See Activate ICF Services of Web Dyn-pro Apps.
Check failed	Possible reasons for failed checks are: • The tiles/target mappings contain content-related errors. See Issues with Tiles and Target Mappings [page 263]. • The check is not supported for the maintained service URL.	1. Check the *Tile/Target Mapping Status* column of the affected tile/target mapping combination. 2. If the tile/target mapping combination does not contain any content-related errors, choose *Check and Show Services*. The *Additional Information* column on the *OData Services* and *ICF Services* tabs provides more information on why the check failed.

1.3.4.7.1.6 Displaying Issues with Launchpad Content

The launchpad content manager displays issues with launchpad content.

The following issues are displayed:

- The catalog state is different on the configuration and the customizing scope.
- References to tiles and target mappings cannot be resolved.
- There are issues with the configuration of tile and target mapping.

For further analysis of content issues, use the tools described under Launchpad Support Tools [page 375].

Related Information

Scope-Related Catalog Issues [page 261]
Issues with Tiles and Target Mappings [page 263]

1.3.4.7.1.6.1 Scope-Related Catalog Issues

The launchpad content manager indicates if the state of a catalog differs on the configuration and the customizing scope.

The launchpad designer allows you to make client-specific changes to cross-client catalogs (for example, changing the catalog title or adding tiles). In this case, a copy of the cross-client catalog is created in the

customizing scope. This copy has the same ID as the original catalog in the configuration scope. If the original catalog contains original tiles/target mappings, the copied catalog includes these objects as originals with the same ID. The copied catalog in customizing is now decoupled from the catalog in the configuration scope. Any changes made in the configuration scope (except changes of texts in tiles or target mappings) are no longer reflected in the customizing scope.

> **i Note**
>
> To prevent these inconsistencies, cross-client catalogs cannot be changed in the client-specific launchpad content manager. We recommend you to change the cross-client catalog in the configuration scope (transaction /ui2/flpcm_conf) or to copy the catalog in the customizing scope (transaction /ui2/flpcm_cust) and make changes for the current client only. See Read-Only Catalogs [page 248].

How to Display the Issue

1. Launch the client-specific launchpad content manager (transaction `/ui2/flpcm_cust`).
2. Open the *Catalogs* tab.
3. Filter the *Scope* column for *Customization*.
4. Check the value of the *Status in Current Client* column:

Value	Description
Original	The catalog was created in the current client and does not exist in the configuration scope.
Changed	The catalog was created in the configuration scope and changed in the current client using the launchpad designer, which means that it is now decoupled from the original catalog on the configuration scope. If the original catalog is now changed in the configuration scope, these changes will not be reflected in the current client.
Outdated	The catalog was created in the configuration scope and changed in the current client using the launchpad designer, which means that it is now decoupled from the original catalog on the configuration scope. The original catalog was then changed in the configuration scope. The catalog on the current client is outdated because the changes in the configuration scope are no longer reflected. If the catalog is now changed in the customizing scope, its status will be set to "Changed" again.

How to Deal with the Issue

If a catalog is outdated, you have the following options:

- Update the catalog to the latest changes of the configuration scope. Changes made in the customizing scope will be lost.
  - If the catalog does not contain any original tiles or target mappings, you can use the launchpad content manager to delete the client-specific catalog in the customizing scope.
    See Deleting Catalogs [page 252].
  - In the launchpad designer, you can reset the outdated catalog. This also works if the outdated catalog contains originals.

See

- Continue working with the existing changes of the customization scope. In this case, you do not receive the latest changes provided in the configuration scope.

1.3.4.7.1.6.2 Issues with Tiles and Target Mappings

The launchpad content manager indicates broken references and issues with the configuration of tiles and target mappings.

To display the issues, proceed as follows:

1. In the list of tiles and target mappings, filter the *Status* column for warnings and errors.
2. Check the *Title / Subtitle / Information* column to get a first hint about the issue:
   - {Reference Lost to Backend Catalog}
   - {Reference Lost}
   - {Configuration Error}
3. To get further information on the issue, open the context menu for a selected row and choose *Status Details*.

The launchpad content manager detects the following issues with tiles and target mappings:

Issue	Category	Status	Description	How to Deal With the Issue
Reference Lost to Backend Catalog	Tile/target mapping reference issue	Warning	The selected tile/target mapping refers to an original tile/target mapping which was defined in the backend using the launchpad app manager. The reference is broken because the original tile/target mapping or the catalog in which the originals were defined do not exist on the front-end server.	If the referenced content is still relevant, proceed as follows: 1. Identify the catalog in which the original tile/target mapping was defined using the *Original Tile Catalog ID* and *Original Target Mapping Catalog ID* columns. 2. Replicate the catalog to the front-end server. See Replicating Back-End Catalogs [page 173].

Issue	Category	Status	Description	How to Deal With the Issue
Reference Lost	Tile/target mapping reference issue	Error	The selected tile/target mapping refers to an original tile/target mapping which was defined in another catalog. The reference is broken because the original tile/target mapping or the catalog in which the originals were defined do not exist in the system or client.	If the referenced content is still relevant, proceed as follows: 1. Identify the catalog in which the original tile/target mapping was defined using the *Original Tile Catalog ID* and *Original Target Mapping Catalog ID* columns. 2. Check whether the original catalog and tile/target mapping definition can still be found in another system or whether it was accidentally deleted. Make sure to check both the customizing and the configuration scope. 3. If you can find the catalog in another system, transport it to the system in which you found the broken reference. 4. If this does not help, remove the broken reference and create a new one.
Configuration Error: Tile Configuration Does not Exist	Tile/target mapping configuration issue	Error	An instance of the tile was created in the catalog, but the tile configuration form was not filled out.	Check the tile configuration in the launchpad designer and maintain it correctly. See Configuring Tiles [page 207].
Configuration Error: Target Mapping Configuration Does not Exist	Tile/target mapping configuration issue	Error	An instance of the target mapping was created in the catalog, but the target mapping configuration form was not filled out.	Check the target mapping configuration in the launchpad designer and maintain it correctly. See Configuring Target Mappings [page 217].
Configuration Error: Invalid Tile Configuration	Tile/target mapping configuration issue	Error	The tile configuration contains errors.	Check the tile configuration in the launchpad designer and maintain it correctly. See Configuring Tiles [page 207].
Configuration Error: Invalid Tile/Target Mapping Configuration	Tile/target mapping configuration issue	Error	The target mapping configuration contains errors.	Check the target mapping configuration in the launchpad designer and maintain it correctly. See Configuring Target Mappings [page 217].

Issue	Category	Status	Description	How to Deal With the Issue
Configuration Error: Custom tile data could not be read	Tile/target mapping configuration issue	Error	The configuration of the custom tile could not be extracted. This leads to incomplete information about this item.	Open the catalog in the launchpad designer to display information on the custom tile.
Configuration Error: LPD_CUST data could not be read	Tile/target mapping configuration issue	Error	The LPD_CUST entry referenced by the tile/target mapping configuration in the launchpad designer was not found.	• In the launchpad designer, check the configuration of the target mapping for application type *SAP Fiori App using LPD_CUST*. See Configuring Target Mappings [page 217]. • In transaction LPD_CUST, check the respective settings. See Changing LPD_CUST Entries for Navigation Targets [page 228].
App descriptor not found	Tile/target mapping configuration issue	Warning	The app descriptor for the SAPUI5 component ID was not found by the SAPUI5 application index.	Depending on the root cause, proceed as follows: • No SAPUI5 Fiori app with the referenced ID exists: In the launchpad designer, check the entry of the ID field of the tile/target mapping configuration. See Configuring Target Mappings [page 217]. • A SAPUI5 Fiori app with this ID exists in the SAPUI5 ABAP repository, but the app index is not set up correctly Ensure the app index is set up correctly. See . • A SAPUI5 Fiori app exists outside the SAPUI5 ABAP repository. An app descriptor exists but for this case is not returned by the app index search. • Known case: SAPUI5 Fiori apps provided by /ui2/ushell library Open the target mapping configuration in the launchpad designer and check the value of the *URL* field. See Configuring Target Mappings [page 217].

Related Information

Displaying Issues with Launchpad Content [page 261]

1.3.4.7.1.7 Adjusting Launchpad Content after an Upgrade

Maintain obsolete or deprecated launchpad content in the launchpad content manager using transactions.

Context

> **i Note**
>
> Upgrade support with transaction codes is only supported in an embedded scenario.

After an upgrade from a lower on-premise version to a higher one, it might be necessary to adjust your launchpad content due to some changes. Applications can become obsolete or deprecated and you might need to add a successor.

Transactions and their status allow you to find out if an app is still in use or if it needs to be replaced. Transactions are assigned to launchpad app descriptor items belonging to a SAP Fiori app. SAP assigns the transaction in the launchpad app manager and maintains the status in *Maintain Authorization Default Values* (transaction SU22) .

For more information on transactions, see Transactions [page 109].

In the launchpad content manager, you can check the transaction status and, if required, add new tiles and target mappings to your catalog using the associated successor transactions.

Supported app types are:

- SAP GUI transactions
- SAPUI5 Fiori applications
- URL apps
- Web Dynpro applications
- WebClientUI applications
- Tile Only

Checking the Status of Launchpad Content

The fields *Transaction* and *Transaction Status* are displayed by default in the *Tile/ Target Mappings* tab and show the status of the relevant content. In the tabs *Catalogs* and *Roles*, you see an aggregated status of the associated transactions.

The states of a transaction can be the following:

Transaction Status	Description	Action
Deprecated	This status is a precursor of the status *Obsolete*. The app is not being developed anymore and is planned to be removed in the near future. The app can still be used.	When the status was set to *Deprecated*, this usually means that launchpad content with an associated successor transaction is in place. We encourage you to use the successor and avoid using the deprecated content, as it will be removed in an upcoming version.
Obsolete	The app does not meet important functional requirements and should therefore no longer be used.	Remove the launchpad content with the obsolete transaction and add launchpad content with the associated successor transaction instead. See Adjusting Launchpad Content with Associated Successor Transactions to your Business Catalog [page 267].
Remote Content	The status of remote content (system alias is not mapped to local) cannot be determined.	You have to manually check in the backend system if the app is up-to-date.

If the status field is **empty**, the launchpad content is either up-to-date or no checks are needed, as e.g, the app type is not supported.

Adjusting Launchpad Content with Associated Successor Transactions to your Business Catalog

You can view and directly add your launchpad content with the associated successor transactions in the launchpad content manager. In order to do that, you need to perform following actions:

1. Open the launchpad content manager (transaction /UI2/FLPCM_CUST) and go to the *Catalogs* tab.
2. In the upper table, select the relevant catalog in which the obsolete or deprecated launchpad content is assigned to.
3. In the lower table, you can see a list of tiles and target mappings and their transaction status. Click ▶ *Other Functions* ▶ *Show/ Add Successors* ▶.
4. If new content with an associated successor transaction is available, a new window is opened. Select the relevant tiles and target mappings and click *Add Tile/ TM Reference*.
5. Remove the deprecated or obsolete tiles and target mappings from your catalog. See Adding and Removing Catalog Content [page 250].

Next Steps

After you have adjusted your launchpad content in the launchpad content manager, you also need to:

1. Depending on whether you use spaces/pages or groups, you need to do the following:
   - Add your new tiles to your page and remove the obsolete or deprecated ones. See Editing a Page [page 283].

- Add your new tiles to your group and remove the obsolete or deprecated ones. If you're using unlocked groups and the end user has personalized the group, we recommend to reset the personalization. See Adding Tiles to and Removing Tiles from Group [page 307] and Delete Personalization Data [page 355].

2. Activate the corresponding OData and ICF services of the newly added launchpad content. See Displaying the Service Activation Status for Apps [page 258].

3. Adjust the corresponding roles and authorizations using the transactions:

   - *Profile Generator: Upgrade and First Installation* (transaction SU25). See Checking for Changes in Authorizations After Upgrades.

   - *Role Maintenance* (transaction PFCG). See Maintaining Roles in the Launchpad Content Manager [page 254].

1.3.4.8 Setting Up Launchpad Layout and Structure

There are two options for structuring apps in the launchpad.

- Spaces layout (recommended)
  You have one or more spaces that contain one or more pages. The pages show apps clustered in different sections. The spaces mode offers more flexibility to influence the launchpad layout for specific user groups. Pages are assigned to users via spaces that are assigned to business roles. The business role defines which users see a specific page. If you enable spaces for your users and define specific pages for them, you can reach a better fit. By defining pages with meaningful sections, you can define in which order the apps are sorted on the page. You can e.g. sort the most used apps in the top-level section and then create separate sections for apps that belong together. Users can personalize the pages by adding and removing apps.

- Classic home page layout:
  There is one page that contains apps clustered in groups. A group consists of tiles (and links) representing a subset of apps. Administrators assign tiles from one or more catalogs to a group and groups and catalogs to a role. Users that have this role assigned, can view the group on their launchpad home page. Users can personalize their home page by adding or removing apps from the out-of-the-box groups or self-defined groups.

Related Information

Managing Launchpad Spaces and Pages [page 269]
Managing Groups [page 304]

1.3.4.8.1 Managing Launchpad Spaces and Pages

Learn how to manage the launchpad spaces and pages as well as create and edit pages and spaces.

The following sections describe how to create and edit your own spaces, how to define specific pages for different user groups and how to edit existing pages. The following figure shows how the layout with spaces and pages looks and how the different elements work together.

Spaces layout and related elements
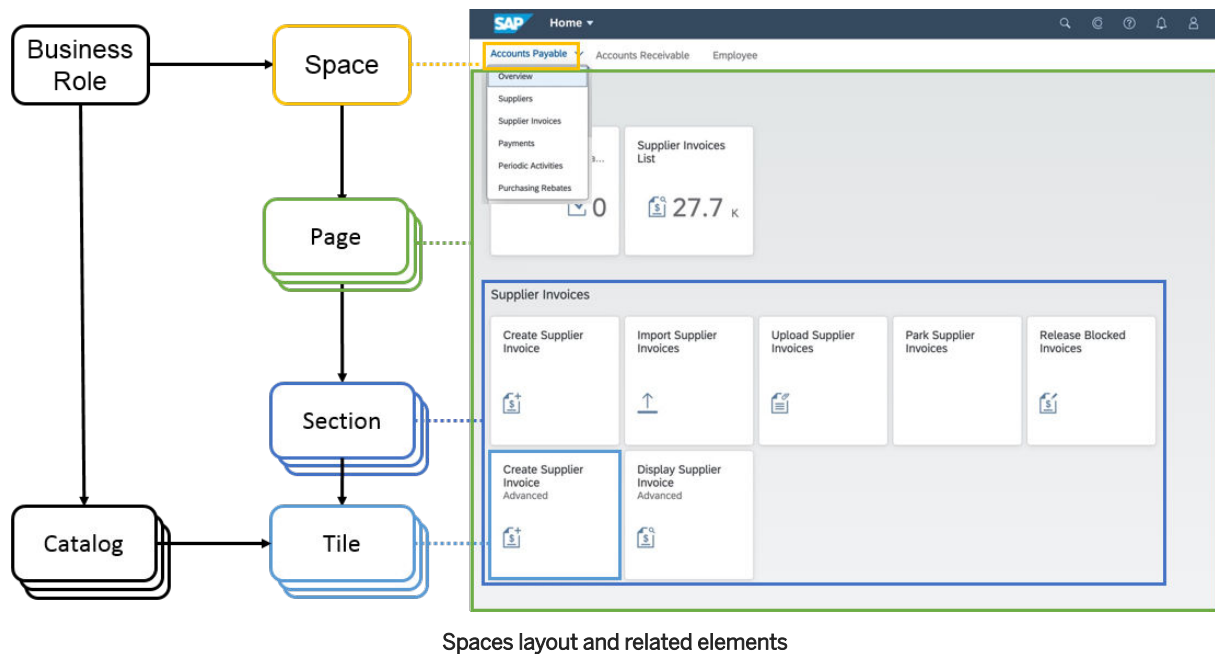
Advantages of the launchpad spaces and pages

The spaces mode was developed to offer more flexibility to influence the launchpad layout for specific user groups. Pages are assigned to users via spaces that are assigned to business roles. The business role defines which users see a specific space. If you enable spaces for your users and define specific spaces and pages for them you can reach a better fit. By defining pages with meaningful sections you can define in which order the apps are sorted on the page. You can e.g. sort the most used apps in the top-level section and then create separate sections for apps that belong together. SAP delivers predefined spaces and pages that you can use to start (see Working with Predefined Spaces and Pages [page 275]).

Tutorial

For a quick overview of the new features you can use the tutorial https://education.hana.ondemand.com/education/wa/s4/index.html?show=slide!SL_4A6B54A67A747B93. The five-minute-tour shows you how to create a space and page and to enable the spaces for your users.

Next steps

First, check out the Tools and Prerequisites for Managing Spaces and Pages [page 270]. If everything is set up, read How to Create and Assign Spaces and Pages [page 271].

1.3.4.8.1.1 Tools and Prerequisites for Managing Spaces and Pages

Find out what you need to set up spaces and pages in the SAP Fiori launchpad.

There are several tools you use and several prerequisites that have to be met to create and manage spaces and pages:

- Catalogs have to be defined already. See e.g. Setting Up Technical Catalogs [page 132] or Setting Up Business Catalogs with the Launchpad Content Manager [page 238] for detailed information.
- Role administration with transaction PFCG: Add the defined space to a role to make it available for the users with this role. You need special rights (e.g. security administration rights) to do this. See Configure Roles for Spaces [page 337] for details.
- The business catalog SAP_BASIS_BC_UI_FLD is assigned to you. It contains the following apps.
  - Manage Launchpad Spaces app: Here you create and manage spaces and you define the page assignment. You can also check to which business roles a space is assigned. This app has two main views: the spaces overview and the space details. See Manage Launchpad Spaces [page 291] for more information.
  - Manage Launchpad Pages app: Here you e.g. create new pages, define the page content and edit existing pages. This app has two main views: the pages overview and the page details. See Manage Launchpad Pages [page 278] for more information.
  - Create Launchpad Pages from Groups app: Here you can create pages based on existing groups. See Create Launchpad Pages from Groups [page 276] for more information.
- The authorization role SAP_FLP_ADMIN is assigned to you.
- The following OData services have to be activated. This can be done either with the task list SAP_FIORI_FOUNDATION_S4 (as described here) or manually (as described in Activating SAP Gateway OData Services [page 23] and Configuring ICF Nodes [page 24]).
  - FDM_SPACE_REPOSITORY_CUST_SRV: Read and write access to the Manage Launchpad Spaces app
  - FDM_PAGE_REPOSITORY_CUST_SRV: Read and write access to the Manage Launchpad Pages app
  - FDM_PAGE_RUNTIME_SRV: Enables users to view spaces and pages in the launchpad
  - FDM_TRANSPORT_SRV: Transport handling of spaces and pages. Also see Transport Information [page 301].
- Launchpad configuration parameters: Spaces and pages are enabled by default, but there are additional settings available. See Enabling Spaces [page 302] for more information.
- When you want to translate your own pages and spaces, you can use the standard SAP translation tools. See Translate Spaces and Pages [page 412] for more information.
- With the transaction /UI2/RSP_LIST, you can analyze spaces, pages, sections, and tiles assigned to specific roles. See Analyzing Space and Page Assignments [page 303] for more information.

1.3.4.8.1.2 Transition from the Home Page to Pages

There are several options to start working with pages and spaces.

When you have worked with the home page and now want to switch to spaces and pages, there are different ways:

1. Create new pages from scratch (see Scenario 2: Create and Assign Spaces and Pages in a Bottom-Up Process [page 274] and How to Create and Assign Spaces and Pages [page 271]).

You as an administrator create new pages. Then you create spaces, add pages to them and assign the spaces to your users by business roles.

2. Start with SAP-delivered spaces and pages. You can either work directly with SAP-delivered content (see Working with Predefined Spaces and Pages [page 275]) or copy the SAP-delivered spaces and pages to your system to be able to adapt them (see Scenario 1: Create Spaces and Pages Based on Predefined Content [page 273])

3. Create pages from existing groups (see Create Launchpad Pages from Groups [page 276]).

You as an administrator create pages with the group content defined for the home page. These pages can be edited. Then you create spaces, add pages to them and assign the spaces to your users by business roles.

Of course, it is possible to combine the options. In addition, users can import personalized information to My Home (see My Home [page 420]).

We recommend this, when a user has personalized content they want to keep. Each user imports apps from their personalized home page to their individual My Home.

> ℹ Note
>
> The content of the home page is not changed with any option. You can still switch between home page and spaces mode.

1.3.4.8.1.3 How to Create and Assign Spaces and Pages

To create a new space and page and make it available for a specific user group you need to perform different steps.

The following image shows an overview of the steps necessary to create and assign spaces.

Create a Page

Create a Space

Assign the Space to a Business Role

Assign Users to the Business Role

Edit Spaces and Pages

- Creating a Page [page 282]
- Creating a Space [page 295]
- Configure Roles for Spaces [page 337]
- https://help.sap.com/viewer/a630d57fc5004c6383e7a81efee7a8bb/latest/en-US/8980ad05330b4585ab96a8e09cef4688.html [https://help.sap.com/viewer/a630d57fc5004c6383e7a81efee7a8bb/latest/en-US/8980ad05330b4585ab96a8e09cef4688.html]
- Editing a Page [page 283]

> i Note
>
> There are several possible sequences to perform the different tasks, but keep in mind that you need to carry out all steps before the new page is visible for your users.
>
> Which sequence is best for you, depends on your system configuration and your authorization management. You need to have the rights to manage roles or work with someone who has it.
>
> Spaces also have to be enabled in your system. In most systems this is already done, if not, you can enable spaces yourself (see Enabling Spaces [page 302] for the different options).

Also see section Tools and Prerequisites for Managing Spaces and Pages [page 270] to inform you what is necessary to perform all steps.

Relations

The following image shows how the different objects work together.

Relations of spaces, pages, catalogs, roles and users
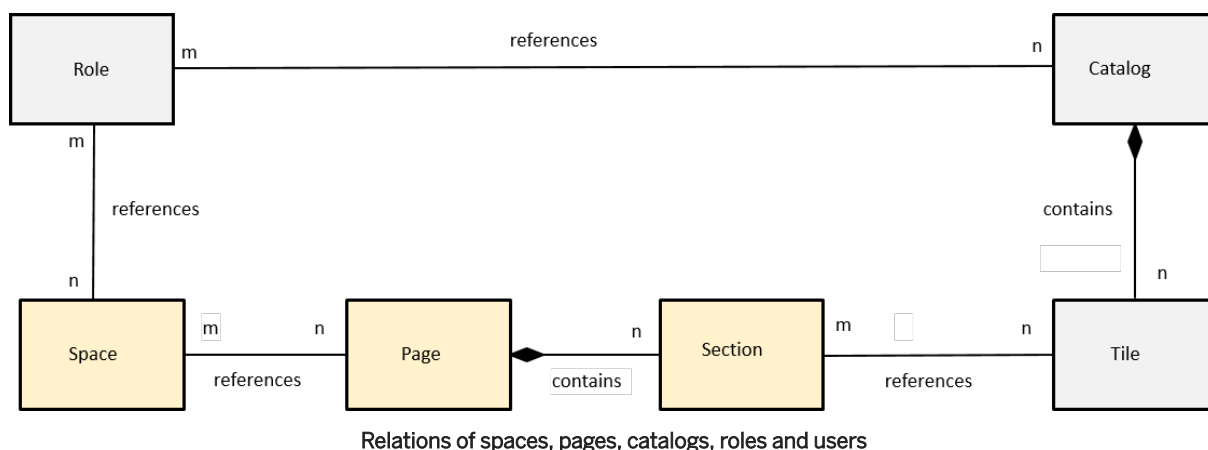
Scenarios

In general, you can always decide to create pages or spaces from scratch or start with SAP-delivered spaces or pages. What is recommended in your case, depends also on how role assignment is handled.

1. Scenario 1: Create Spaces and Pages Based on Predefined Content [page 273] shows a rather short way to define all necessary information working with SAP-delivered spaces. For this you need to have also the right to edit business roles or work closely with someone who does. When you have copied the SAP-delivered spaces and pages, you can adjust them according to the specific needs of your users.
2. Scenario 2: Create and Assign Spaces and Pages in a Bottom-Up Process [page 274] shows a process starting with the definition of a space and page and ending with the role assignment. This is recommended if you e.g. cannot assign spaces to roles yourself or if your pages need to be approved before release. This scenario is also applicable when you prefer a freestyle bottom-up approach.

1.3.4.8.1.3.1 Scenario 1: Create Spaces and Pages Based on Predefined Content

This is a rather short procedure to define everything you need for a space and page for a user group.

With the following steps you create a new space with one or more pages based on predefined content (for example, a space and page delivered by SAP). Then you assign it to a business role and thus make it available for your users. This is an overview for a process spanning over different apps and tools. The linked topics provide detailed information on the single steps.

1. Open the Manage Launchpad Spaces app (see Manage Launchpad Spaces [page 291]).
2. Go to the tab *Predefined* and search for an appropriate space (see Working with Predefined Spaces and Pages [page 275]). Click on *Copy* to create a copy of the space in your custom environment (see Copying a Space [page 298]).
3. Add the space information for the new space, select what to do with the assigned pages and confirm. The space details are displayed.
4. Now you need to establish a role context for the space and page, so that you can see all apps available for the users. This is done in transaction PFCG (you need special rights for this; see section Configure Roles for Spaces [page 337] for all steps necessary).

5. If not done before, the users have to be assigned to the business role. This is also done in transaction PFCG (see Assigning Users to Roles [page 339]).

6. When everything is assigned, you can edit the page in the Manage Launchpad Pages app. You now see all apps in the catalogs that are available for the users according to their business role. If necessary, you can add apps from additional catalogs (see Editing a Page [page 283]).

7. Enable spaces in the launchpad parameter settings (see Enabling Spaces [page 302]).

1.3.4.8.1.3.2 Scenario 2: Create and Assign Spaces and Pages in a Bottom-Up Process

If you want to create pages and spaces from scratch, perform the following steps.

This is an overview for a process spanning over different apps and tools. The linked topics provide detailed information on the single steps.

1. Open the Manage Launchpad Spaces app (see Manage Launchpad Spaces [page 291]).

2. Select *Create* and add the space information for the new space (see Creating a Space [page 295]).

3. Select the option *Also Create a Page*, enter the page information and click on *Create*. The space details are displayed. The *Pages* tab shows the newly created, empty page.

4. Select the page and you are redirected to the Manage Launchpad Pages app. Click on *Edit* to open the page in edit mode. If the page and space have to be approved before they can be assigned to users, make sure to add all apps that are useful for the user group that should use the page later. You need to add the appropriate business catalogs manually to access the apps. How this is done, is described in section Editing a Page [page 283]. Save the page.

5. Now you need to establish a role context for the space and page, so that you can see all apps available for the users. This is done in transaction PFCG (you need special rights for this). See Configure Roles for Spaces [page 337] for all steps necessary.

6. If not done before, the users have to be assigned to the business role (also done in transaction PFCG, see Assigning Users to Roles [page 339]).

7. When everything is assigned, you can edit the page in the Manage Launchpad Pages app again, if necessary. You now see all catalogs that are available for the users according to their business role. Use the *Preview* option to see how the page looks like.

8. Enable spaces in the launchpad parameter settings (see Enabling Spaces [page 302]).

> i Note
>
> When you can assign the spaces to roles yourself, it is easier to to this after step 3. Then you have access to the catalogs assigned to the role in the Manage Launchpad Pages app and can skip selecting all catalogs.

1.3.4.8.1.3.3 Creating Partner Spaces and Pages

Partners that have an ABAP namespace, can create spaces and pages that can be used as templates.

Partner-defined spaces and pages are handled the same way as SAP-delivered spaces and pages: They can be created and edited in a development system as normal spaces and pages. In the customer system, they are

shown in the *Predefined Spaces/Pages* lists and can't be edited. When you want to work with partner spaces or pages, you need to copy them first. See Working with Predefined Spaces and Pages [page 275] for detailed information.

Read more about partner namespaces in general in Development and Namespaces for Customers and Partners🔗 or Namespaces and Naming Conventions.

How to create a partner space or page

In general, you create partner spaces or pages the same way as a normal space or page. This is described in Creating a Page [page 282] and Creating a Space [page 295].

The only difference is that you have to enter the partner namespace as the first part of the ID, e.g. */partnernamespace/spacename*.

1.3.4.8.1.3.4 Working with Predefined Spaces and Pages

A space and page delivered by SAP or a partner can be used directly (when you work with role templates provided by SAP) or as a starting point to create your own space and page for a business role.

SAP provides spaces and pages for different SAP business roles. An SAP-delivered space contains one or more pages defined by SAP. The pages contain the apps that are relevant for the users with that specific role. Partners can deliver spaces and pages in the same way when they have their own namespace (see Creating Partner Spaces and Pages [page 274]).

The *Predefined* spaces overview tab in the Manage Launchpad Spaces app contains all spaces SAP or a partner provide. SAP uses the following name scheme: The space ID refers to the business area that the space was designed for, for example SAP_<AREA>_PG_<SUFFIX>. The space title reflects the work area that the space is designed for. You can use this information to identify which spaces might be interesting for your user group.

The Manage Launchpad Pages app shows a list of all predefined pages in the pages overview. Click on a specific page to see the page details, such as the apps and sections that are predefined. You can also open the page preview to see how it looks (also see Editing a Page [page 283]).

How to work with predefined spaces and pages

You can directly use SAP-delivered or partner-delivered spaces and pages when you work with roles that are based on role templates delivered by SAP. For this you need to assign the SAP-delivered space to a business role. This is a rather fast way to get started with spaces and pages. If you want to change a predefined space or page, you have to create a copy and use an ID with a customer namespace prefix. You can decide if you want to copy and assign the pages belonging to a space, too. When you work with your own catalogs, you can copy the predefined spaces or pages to your system and assign them to your business role. Then you can see how the SAP-delivered spaces and pages look.

The *Predefined* Spaces Overview tab contains all of the spaces from SAP that are available for your SAP product scope. Eligibility is checked for roles, spaces, pages, and apps so you will only see objects within your SAP product scope for the page creation.

Technically, an SAP-delivered space is assigned to a business role template. By this connection, either the SAP-delivered spaces can be assigned directly, or a space derived from an SAP-delivered space can be created for a business role based on the business role template. You can use an SAP-delivered space to easily create a customized space and page for a role.

> **i Note**
>
> When you work with your own business roles and catalogs, SAP recommends you create your own spaces and pages from scratch. Then assign the space and necessary business catalogs to your roles (see Scenario 2: Create and Assign Spaces and Pages in a Bottom-Up Process [page 274]). Then you see all catalogs with apps that are available for the users with the role, and you can assign all necessary apps to the pages.
>
> Assigned SAP-delivered spaces and pages are updated automatically, when SAP provides a new version. Note that users can personalize the SAP-delivered pages (when personalization is enabled by the customer administrator), the personalization is kept when an SAP-delivered space or page is updated by SAP.

Copying and customizing predefined spaces

When you want to start with an SAP-deliverd or partner space and then adjust it to your needs, you have to copy it first. Open the Manage Launchpad Spaces app and go to the tab *Predefined*. Copy an SAP-delivered space to your system and assign the copy to the role. Also see Scenario 1: Create Spaces and Pages Based on Predefined Content [page 273].

After the copy process you can then use the new space and pages "as is" or adapt them to the specific needs of your user group. There are several ways to adapt a page: You can see all pages that belong to a space in the Manage Launchpad Spaces app (*Pages* tab), or see all pages in your system in the Manage Launchpad Pages app. Select a page and click *Edit*.

Related Information

Editing a Page [page 283]
Manage Launchpad Spaces [page 291]
Manage Launchpad Pages [page 278]

1.3.4.8.1.4 Create Launchpad Pages from Groups

With this app you can create pages based on existing groups.

Key Features

This app provides the following key features:

- Create a page based on one or multiple groups
- Filter for group ID, group title, or role assignments
- View the assigned tiles of a group
- Manage your newly created page by using the direct link to the *Manage Launchpad Pages* app

Supported Device Types

- Desktop

Implementation Information

- Technical catalog: SAP_TC_FLP_COMMON
- Authorization role: SAP_FLP_ADMIN

Learn how to create pages from existing groups here: Creating a Launchpad Page from Groups [page 277].

1.3.4.8.1.4.1 Creating a Launchpad Page from Groups

Learn how to create a launchpad page from existing groups.

You can create a page based on existing groups. Those groups will be converted to the sections of the page. For each group that you select, a section with the same title and content will be created. You can adjust the title afterwards.

How to Create a Page from Groups

1. Open the *Create Launchpad Pages from Groups* app from the SAP Fiori launchpad.
2. In the *Group ID* search field, enter the group ID of the group that you want to convert to a page. You can also filter for *Group Title* and *Role*.
3. Choose *Go*.
   You can see the results in the left section called *Groups*. Each row shows a group. You can expand the row

   by choosing ❯ (*Expand Node*) to view the assigned tiles.

4. Now you want to add a group to *Page Sections* on the right. Choose ⊕ (*Add Line*) in the *Actions* column or simply drag and drop the group to the page section.

Repeat this step for each group that you would like to add to the page. You can also search for more groups as described in step 2. The current selection in the page section remains even if you search again.

To remove a group from the page section, choose *Remove All* or remove groups via drag and drop.

5. Choose *Create Page*.
6. In the *Create Page* dialog, enter an ID for the page. To learn more about the ID, see Creating a Page [page 282].
7. Enter a description for the page. It will be shown, e.g., in the *Manage Launchpad Spaces* app.
8. Give a short describing title for the page in the field *Title*. The title will be shown in the space menu.
9. If necessary, select a transport for the page. Start typing to receive a list of possible transports. See Transport Information [page 301] for additional information.
10. Click on *Save* to generate the page. A page with the related sections and tiles is created. To view the page, click on the button *Manage Launchpad Pages*. You will be redirected to the *Manage Launchpad Pages* app. Your newly created page is shown on top.

> **i Note**
>
> Keep in mind that a page always has to be assigned to a space to make it available to your users.

Next Steps

You can now continue with assigning your page to a space, or managing your page. See Manage Launchpad Pages [page 278] and Manage Launchpad Spaces [page 291].

1.3.4.8.1.5 Manage Launchpad Pages

With this app you can manage the launchpad pages, define the layout of pages and edit, copy or delete existing pages.

Key Features

This app provides the following key features:

- See all pages availabe in your launchpad.
- Define the layout of a page and select apps that should be displayed.
- Edit an existing page.
- Copy an SAP-delivered page or a customer-created page.
- Delete a customer-created page.

Supported Device Types
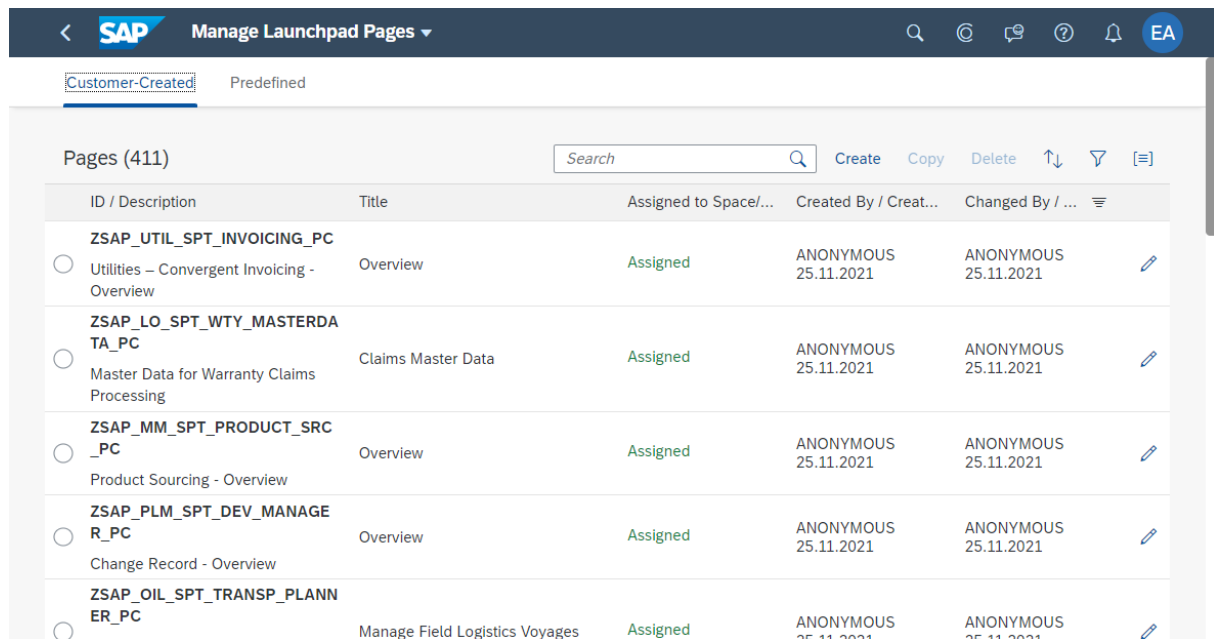
- Desktop

Implementation Information

- Business catalog: SAP_BASIS_BC_UI_FLD
- Business role: SAP_FLP_ADMIN

1.3.4.8.1.5.1 Pages Overview

When you open the Manage Launchpad Pages app, you see a list of all pages in your system.

By default, the pages are sorted by the *Changed On* date, the newest page is shown on top. The tab *Customer-Created* shows all pages you have created in your system. The tab *Predefined* shows all pages provided by SAP or partners (see Working with Predefined Spaces and Pages [page 275] for detailed information).

Manage Launchpad Pages

There are different options available to manage the pages, it also depends if you select a customer-created or predefined page. For customer-created pages, the following options are available:

- Search for a page.
- Sort, filter or group the list entries to see only pages that fit your criteria (see below for detailed information).
- Open an existing page to see the content in detail: Click on a page in the list.

- Copy an existing page (see Copying a Page [page 290]).
- Edit an existing page (see Editing a Page [page 283]).
- Open a page preview to see how the page looks like for the user (from page details view, also see Reusing Pages [page 289]).
- Delete a page (see Deleting a Page [page 291]).

> **i Note**
>
> For predefined pages the options to edit or to delete a page are not available. You can open the page details and the page preview to see how e.g. an SAP-delivered page looks like. But you need to copy predefined page to your system first, before you can actually work with it. Copying can be done here, in the page details, or in the Manage Launchpad Spaces app, where you can copy a predefined space together with its assigned pages.

Select a page and then select the option you want to perform in the header line or next to the page.

Adapting the Pages List

You can influence in which order the pages should be shown and you can group the pages by certain criteria. By using filters you can decide which pages should be shown in the list.

- To sort your pages according to values in one column, click on ↑↓ in the header bar. Select ascending or descending order in the area *Sort Order* and the column you want to use for sorting in *Sort By*.
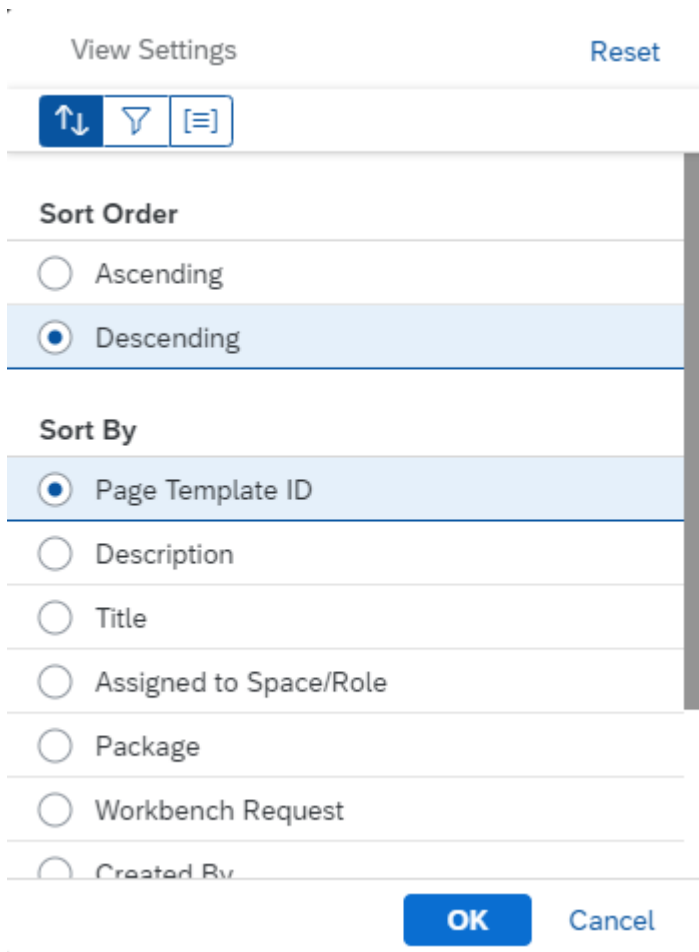
Define Your Sort Order

The selected sort order is indicated in the column header of the respective column in the page overview with the icons ≡ for ascending sorting, or ≡ for descending sorting.

- To filter the list to only see pages that match your defined criteria, click on ▽ . Click on a column you want to use for filtering, e.g. *Changed By*. Then select a value, e.g. your name, to see only pages that apply to the selected value. In our example, you will only see the pages that were changed by you as the last changer.

- You can group the pages according to the values in a specific column. Click on [≡] .

1.3.4.8.1.5.2 Page Details

The Page Details view is part of the Manage Launchpad Pages app. It shows all information for a page.

You can open this page by clicking on a page in the overview. You might also open the page in the *Pages* tab of the space details view in the Manage Launchpad Spaces app. Then you are redirected to this view (also see Details View for Spaces [page 294]).
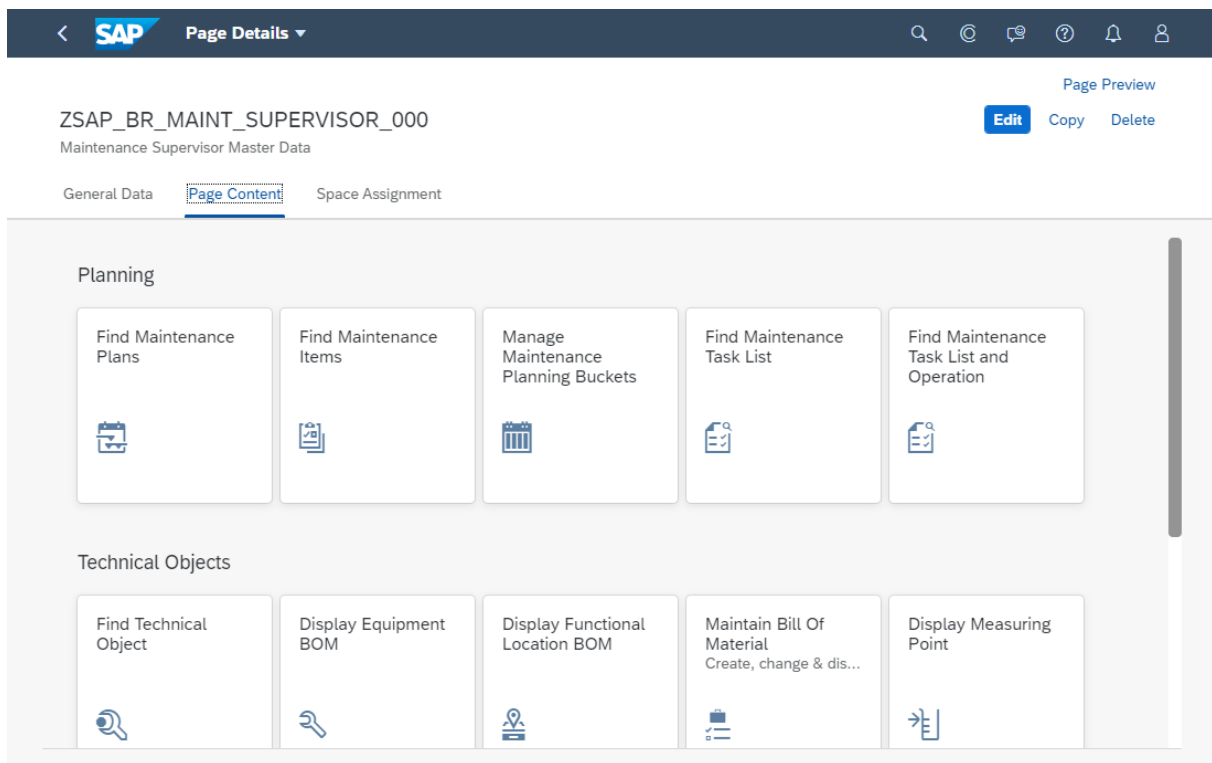
Example of a customer-created page

There are three tabs:

- General: Shows you all general page information, as e.g. the ID, title, information about who created or changed the space and when.

- Page Content: Shows the page layout with the sections and the assigned apps. When the page is a customer-created page, you can edit the page content and the page header, copy the page or delete it. You can also open a preview for the page and adapt this view for one or more roles (see Editing a Page [page 283]).
  When the page is predefined (by SAP or a partner), you can see the layout and you can open the page preview. You can't edit a predefined page or select a role context for it. To do this, you need to copy the page first. By this, you create your customer-specific version of the page that you can edit afterwards (see Copying a Page [page 290]).

- Space Assignment: Shows to which spaces this page is assigned to. To sort your spaces according to values in one column, click on ↑↓ in the header bar. Select ascending or descending order in the area *Sort Order* and the column you want to use for sorting in *Sort By*.
  When you click on a space in the list, you are redirected to the space details of the space in the Manage Launchpad Spaces app. With the button *Manage Launchpad Spaces* you open this app with the spaces overview.

1.3.4.8.1.5.3 Creating a Page

Learn how to create a page for a specific user group.

Here we describe how you create an empty page from scratch.

How to Create a Page

1. Select *Create* in the Manage Launchpad Pages app or select the option *Also Create a Page*, when creating a new space in the Manage Launchpad Spaces app.

2. In the *Create* dialog, enter an ID for the page. The ID of customer-defined pages must start with a Z, a Y (Y can be used for spaces designed by SAP partners without a namespace) or with a partner namespace. For a page in a partner namespace you enter the partner namespace as the first part of the ID, e.g. */partnernamespace/*.). The ID can only contain letters, 0-9 and _ or /. Other special characters are not allowed. The ID cannot be changed afterwards.

   New Page

   ID: *

   [   ]

   Description: *

   [Short description for the administrator]

   Title: *

   [Menu text for the user]

   Create   Cancel

3. Enter a description for the page. It will be shown e.g. in the Manage Launchpad Spaces app.

4. Give a short describing title for the page in the field *Title*. The title will be shown in the space menu.

5. If necessary, select a transport for the page. Start typing to receive a list of possible transports. See Transport Information [page 301] for additional information.

6. Click on *Create* to generate the page. An empty page is displayed in the edit page view. See Editing a Page [page 283] to learn how to add apps to your new page.

> **i Note**
>
> Keep in mind that a page always has to be assigned to a space to make it available to your users. If you have created a page from scratch, you need to assign it in the Manage Launchpad Spaces app. Read Details View for Spaces [page 294] for a description.

1.3.4.8.1.5.4 Editing a Page

Learn how to add apps to a new page and adapt an existing page.

You can open the edit page mode by clicking *Edit* in the page details or in the page overview. The figure shows an example page in edit mode.
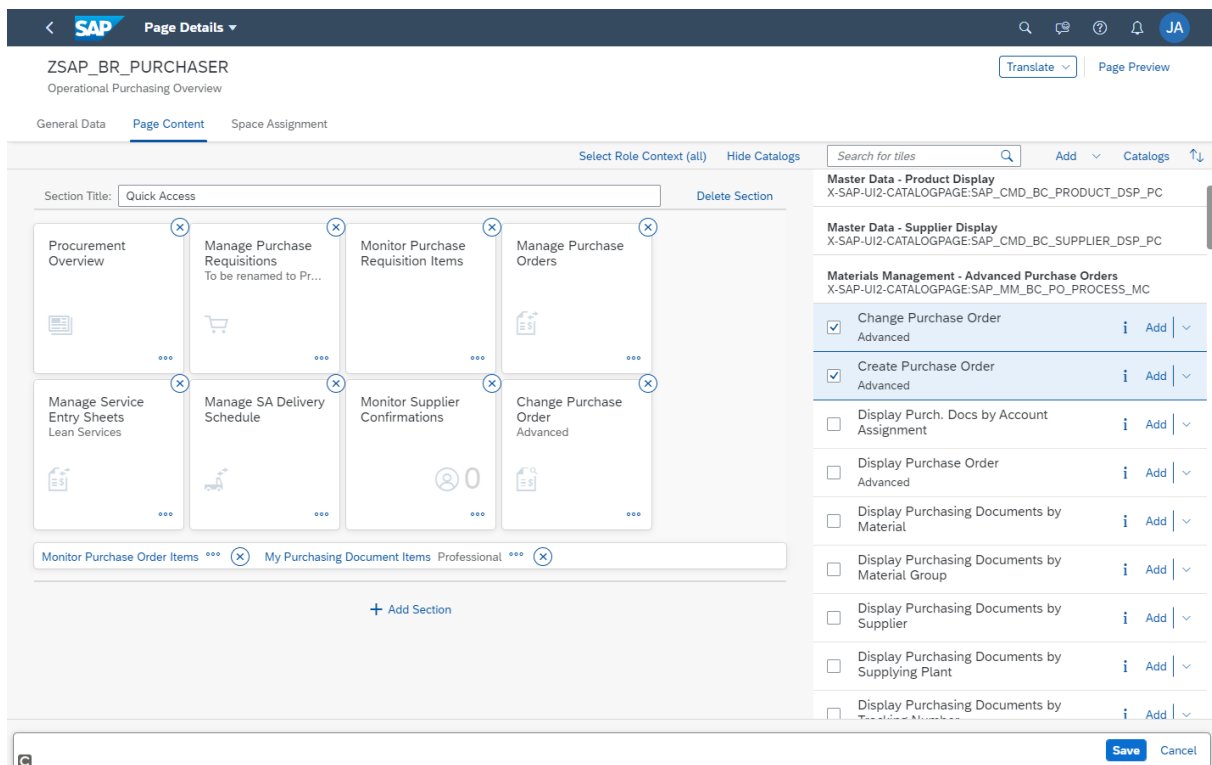
Editing a page

How To Edit the Page Header

You can change the page title or the page description if necessary. Go to the tab *General Data* and enter new values. Note that you cannot change the page ID.

How to Edit the Page Content

1. Open the Tab *Page Content*.
2. There is already one empty section in the page. Enter a meaningful name for it.
3. Add more sections to your page if useful.
4. Add apps from the tile catalogs to the sections. If the list of tile catalogs is not displayed, click on *Show Catalogs*. See section Tile Catalogs [page 286] to learn which tile catalogs and apps are shown here.
5. You can search for an app in the tile catalogs list or in the search field. You can also filter the list according to different criteria. See Tile Catalogs [page 286] for detailed information.
6. Click on *Add* to add it to one section or drag and drop the app to a section. You can select different tile sizes or select the link layout by clicking on the arrow next to *Add*. Select an option to add the app accordingly. When you don't select an option here, the app is added in the preferred tile size (usually as standard tile). Flat and flat wide tiles are added in a new row below the standard-sized tiles. Links are added in an area below the tiles in the selected section the same way as in the home page. See Tiles, Links and Tile Sizes [page 423] for more information on the different tile size options.
7. You can also select several apps by ticking the boxes in front of each app title. When you have selected all apps you want to assign to a section, click on *Add* in the catalog header. You can select a tile size or the

link layout by clicking on the arrow next to *Add* and selecting the appropriate entry. Select one or more sections in the list and click *Add*.

8. Save your changes.

> i Note
>
> Sometimes apps with the same title appear several times in the tile catalog. This happens if there are apps with e. g. different targets, e. g. on a desktop or on a mobile. An icon shows for which target the app is intended.
>
> When you edit a page that is personalized by a user, the personalizations are kept.
>
> When you work on a smaller device, e.g. a tablet, or zoom in, the tile catalogs list might be hidden. You can then switch between the page and the tile catalogs list with the buttons *Show Catalogs* and *Show Page*.

> → Tip
>
> We collected a list of helpful tips how to create a meaningful page in Best Practices for Managing Spaces and Pages [page 117].

Check for Possible Issues

If there are errors or inconsistencies on the page, you are notified. A message icon showing the number of problems is displayed in the footer. Click on the message icon in the footer to see more information on the issues.
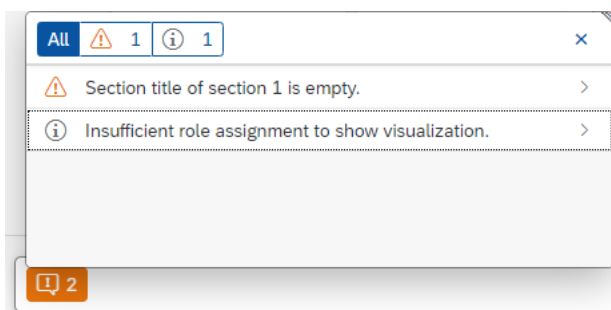
Dialog showing two messages for a page

There are three different kinds of messages that are also reflected in the icon color. The message icon color always presents the highest existent severity:

- Information: informs about a display issue, e.g. if you have not assigned the page to a space or the space to a role. The message icon will be displayed in gray.

- Warning: displayed for inconsistencies, e.g. when a section title is missing, as SAP recommends this to ensure a consistent user experience. You need to confirm the warning when you save a page. The message icon will be displayed in orange.

- Error: prevents you from saving a page. A message is displayed when you try to save. Errors may occur e.g., if you have selected a tile that cannot be displayed for the users in a page. The app then is displayed in a failed state with an x). For example, this can be a tile with an incorrect target mapping or a tile that is not available for the user (due to the role assignment or the business catalog assignment). The message icon will be displayed in red.

If a tile is involved, the tile information popover also shows information about the problem. It can be opened by clicking on the tile.

Show Preview

To see how a page actually looks like you can open the page you edit in a preview. Click on *Page Preview*. The preview is opened in a popover. If you select a specific role, the page preview shows how the page will look like for the users with this role (see Reusing Pages [page 289]).

1.3.4.8.1.5.4.1 Tile Catalogs

The catalogs show the apps that you can add to the page.

The catalog view is part of the *Edit Pages* view. A catalog contains a collection of apps. You select the apps here to add them to the page.

The catalog view has two tabs:

- Derived From Roles: Depending on the role assignment of the page you can see one or more catalogs (usually these are business catalogs, see Best Practices for Managing Catalogs [page 115]). If you have no role assignment yet, this list is empty.
- Manually Selected: This list shows all catalogs you have manually added for the page. In the beginning, this list is empty. You can add all catalogs you need to the list. You can select technical as well as business catalogs here.

> i Note
>
> The authorization to see and use a specific app and the corresponding tile is checked on catalog level, based on the tile intent. Therefore, users see only those tiles on a page that are referenced from a catalog that is part of a role the user is assigned to. In addition, the corresponding target mapping needs to be part of an assigned catalog. When you e.g. remove an assigned app from the business catalog you use, it will not be available for the user anymore.

If the catalog view is not shown, click on *Show Catalogs*. You can change the sort order of the catalogs with ↑↓ .

You can also filter the apps to show only apps with specific criteria by clicking ▽ . In the opening dialog, you can filter by device type for that the app is available, tile type and app type. Select an entry from the list, then enter the specific filter criteria. For example, select *Available for Device Type*, then select *Mobile* to see only apps suitable for mobile devices. You can combine several filter criteria. With *Reset* you delete all selected criteria.
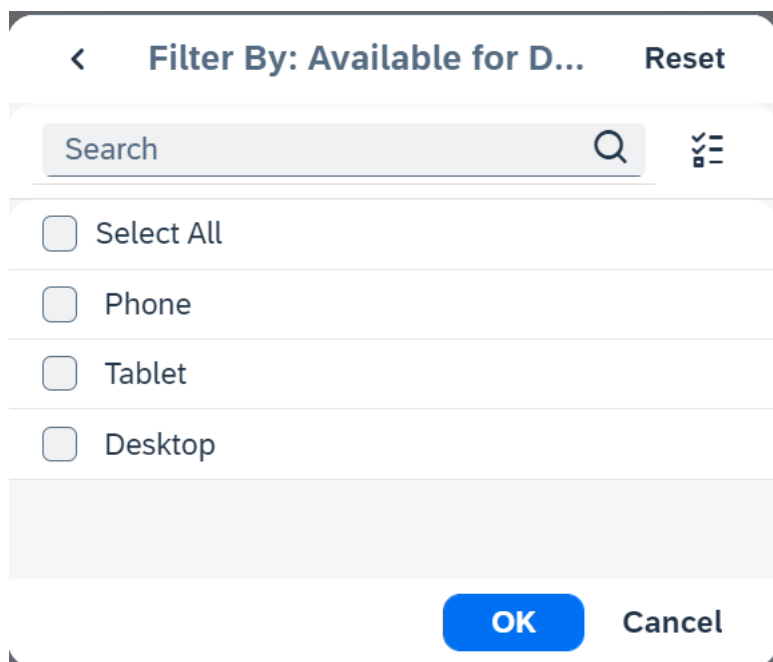
Each app in a catalog has an entry showing the app title and subtitle. When you click on the information icon, additional app information is displayed (also see Tile Information [page 288]).

Adding Catalogs

You can add all catalogs available in your system to the tab *Manually Selected*. So you can access all apps independent of the catalog's role assignment. You need to do this when the space is not assigned to a role and thus no catalogs derived from the role are available.

1. Click on *Catalogs* (or click on the *More* icon and select *Catalogs*).
2. In the *Select Catalogs* view, check one or more catalogs you want to use for the page design. You can search for any catalog.
3. Click on *Select*. The catalogs and the apps assigned to them are added to the *Manually Selected* tab.

> **i Note**
>
> When you add apps from catalogs that are not from a catalog assigned to the roles of the user, theses apps will not be shown to the user.

How to Add Apps to the Page

There are three options to select apps for the page:

- You can drag and drop an app from the catalog to a section.
- You can click on *Add* next to the app title. Select the section or sections to which you want to add the app in the popover.

- You can select several apps by marking the checkboxes in front of the apps. Then you add them all to one or more sections by clicking on *Add* in the tool bar above the catalog view. Select the section or sections to which you want to add the app in the popover.

1.3.4.8.1.5.4.2 Tile Information

For each app, detailed information is available.

You can open detailed app information for apps assigned to the page by clicking on the app (when you view a page in the page details) or by clicking on the overflow icon °°° when you are editing a page.

The pop-over shows detailed information e.g. for the semantic object that is used, the Fiori ID, the app detail or the tile type. You can see, in which technical and business catalogs the tile is contained. You can also see for

which devices the app is available: desktop, tablet and/or mobile. Here you can e.g. change the tile size of an assigned app or convert it to a link. Select a new entry in the field *Tile Size*. Preferred indicates the preferred tile size the tile was designed for.

> **i Note**
>
> Custom tiles might only be displayed in the standard tile size here (they show a *Custom* tag to identify them). In the actual page they will be displayed in the designed size, e.g. wide for news tiles.

Each app in a catalog has an entry showing the app title and subtitle. When you click on the information icon, additional app information is displayed. The pop-over shows nearly the same information as described above.

1.3.4.8.1.5.5 Assigning a Transport to a Page

You can assign transports to the page e.g. when creating or deleting a page or in the page details.

When you create or delete a page, you can enter a transport. Depending on your system configuration, a transport can be mandatory. See Transport Information [page 301] for more information.

You can see the existing transport assignments and add a new transport also in the page details view of the Manage Launchpad Pages app.

- Open the space in the page details view and go to the *Transport* tab. When the page is already assigned to one or more transports, the transports are listed in the list *Open Transports*.
- To assign the page to a transport, click on *Assign*. Enter an existing transport in the dialog. When you start typing, a list of available transports will be shown.

1.3.4.8.1.5.6 Reusing Pages

When you want to reuse a page, you can use the role context feature to see which apps a page contains for specific user groups.

In general, a page can be assigned to different spaces and so be used for different user groups. Reusing pages usually works best for user groups with a similar profile, so that the app availability is similar. Nevertheless, users with different business roles might see different apps on their page. The app availability for users depends on two factors: the user's role(s) and the catalog-to-role assignment.

If you work with business catalogs, note that different business catalogs may contain the same app from the technical catalog. The apps are identified by their technical catalog ID (original tile ID; also see About Launchpad Content [page 103]). Thus, it is possible that users with different business catalogs (assigned to them by the role) see the same app. It is not necessary to add the app multiple times from different business catalogs.

You can check how the pages look for your different user groups with the option *Select Role Context* in the Manage Launchpad Pages app. By selecting one or more roles for a page in edit mode or the preview, you can see which apps a page contains for specific user groups.

Selecting the role context

Open the page in the Manage Launchpad Pages app and click on *Edit*. In the edit mode you may see catalogs from different business roles for your page, if you have assigned the space that contains the page to several business roles. The assignment can be done in transaction PFCG, see Configure Roles for Spaces [page 337]. In addition, you might have added more catalogs manually (see Tile Catalogs [page 286]) and have selected apps from those catalogs. Apps from manually selected catalogs are always shown in a dimmed grey and have a warning because they have no role assigned.

You can filter the tile catalogs list, the edit view and the page preview to one or more specific roles. Then you see only the apps available for this specific role(s). Click on *Select Role Context* and select the roles. The button text shows the number of roles you have selected. In addition, the catalog view header indicates that you have filtered the tile catalogs list.

The preview will show all apps that are available for the selected roles according to the catalog assignment. By this you can check how the page will be displayed for a user with one or more roles. If the page contains apps from manually selected catalogs that are not assigned to the selected role or roles, you need to deselect all roles in *Select Role Context* to see them. The preview then shows all apps independent of the role assignment.

1.3.4.8.1.5.7 Copying a Page

You can copy an existing customer-created page, a page from a partner or an SAP-delivered page and then edit the new page.

Select the page you want to copy in the Manage Launchpad Pages app, either in the overview or in the page details view. You can find the pages from SAP on the tab *Predefined* in the overview. Pages from SAP or partners have to be copied to your customer system before you can use or edit them. Read Working with Predefined Spaces and Pages [page 275] first for a detailed description how to work with the feature.

Click on *Copy* and a dialog is shown where you enter a new page ID, a description, a title and, if necessary, a transport for the new page. The page ID has to start with Z_ or Y_, as you can only add pages these namespaces. Z should be used for customer-created objects, while Y should be used for SAP-partner objects.

Example of a Copy Page Dialog

Note that you need to assign the page to the space of the user group to make it available for the users.

1.3.4.8.1.5.8 Deleting a Page

You can delete a page that is no longer needed.

In the Manage Launchpad Pages app, you can delete an existing customer-created page. To delete a page, select it and click on *Delete* in the pages overview or in the page details. If necessary, enter a transport for the process (see Transport Information [page 301]). Confirm the warning message. The page will be deleted from the system.

You should check that you don't delete the only page in an assigned space, or assign a different page to this space when you delete the only page. You see all assigned pages in the Overview of Launchpad Spaces [page 292]. If an assigned space doesn't contain a page anymore, users might see an empty space in their launchpad.

1.3.4.8.1.6 Manage Launchpad Spaces

With this app you can create and manage spaces for the SAP Fiori launchpad. With spaces you can design the launchpad layout. Spaces are organizational units that combine several pages in a menu that is shown in

the navigation bar of the launchpad. The pages then contain the apps for the end users. The app provides an overview showing all spaces available in your launchpad and a detailed view for the spaces.

Key Features

You can use this app to:

- Manage existing spaces
- Create new spaces
- See and sort the pages that belong to the space
- Add or remove pages to or from a space
- See to which role or roles the space is assigned to
- Edit space title and description

Supported Device Types

- Desktop

Implementation Information

- Business catalog: SAP_BASIS_BC_UI_FLD
- Business role: SAP_FLP_ADMIN

1.3.4.8.1.6.1 Overview of Launchpad Spaces

The Manage Launchpad Spaces Overview displays all available spaces and offers options to manage your spaces.

There are two different lists available: the customer-created spaces and the predefined spaces. You can switch between the two lists with the tabs over the list. By default, the space changed as last one is shown on top of the list. The following figure shows an example of a list of customer-created spaces.

Spaces overview in the Manage Launchpad Spaces app

- The Predefined list contains all spaces from SAP or partners (see Working with Predefined Spaces and Pages [page 275] for detailed information). As the predefined content cannot be changed, you can open the space details by clicking on a space in the list, but you are not allowed to modify these spaces.
- The Customer-Created list contains all spaces you have created yourself. You can create a new space here, copy or delete an existing space. Click on a space in the list to go to the space details view. Or click on the pencil icon to directly open the space in the edit mode. See Details View for Spaces [page 294] for more information.

In both lists, you can search for a space, filter the list, or sort the list according to your own sort criteria.

Adapting the spaces lists

You can influence in which order the spaces should be shown and you can group the spaces by certain criteria. By using filters you can decide which spaces should be shown in the list.

- To sort your spaces according to values in one column, click on ↑↓ in the header bar. Select ascending or descending order in the area *Sort Order* and the column you want to use for sorting in *Sort By*.
  The selected sort order is indicated in the column header of the respective column in the page overview with the icons ≡ for ascending sorting, or ≡ for descending sorting.
- To filter the list to see only spaces that match your defined criteria, click on ▽ . Click on a column you want to use for filtering, e.g. *Changed By*. Then select a value, e.g. your name, to see only spaces that apply to the selected value. In our example, you will only see the spaces that were changed by you as the last changer.

- You can group the spaces according to the values in a specific column. Click on $[\equiv]$ and select a value.

1.3.4.8.1.6.2 Details View for Spaces

The details view shows all information on spaces and the pages assigned to the spaces.

The view has four different tabs:

- General: shows space information as e.g. space ID, title and creation information
- Pages: shows all pages that are assigned to the space
- Roles: shows all roles the space is assigned to
- Transport: shows the transport or transports the space is assigned to

Details of a Space with Two Pages

Note that you can only edit information for a customer-created space. When you have opened an SAP-delivered space you can only view the information or copy the space to create a customer space based on it.

Viewing general space information

In the *General* tab you can see all general space information, as e.g. the title, information about who created or changed the space and when. You can edit the title and the description. The title will be shown in the navigation bar of the launchpad. Note that you cannot edit the space ID.

Viewing the page assignment

In the *Pages* tab you can see and add or remove the pages that are assigned to the space. In addition, you can define the order the pages are later shown in the space menu in the navigation bar of the launchpad. The page visibility defines if a page is displayed in the launchpad.

Viewing roles

The *Roles* tab shows the business roles the space is assigned to.

Viewing transport assignment

The *Transport* tab shows the open transports the space is assigned to. You can assign the space to a transport or change the transport assignment.

Related Information

Editing a Space [page 297]

1.3.4.8.1.6.3 Creating a Space

Learn how to create a launchpad space.

A space is an organizational unit that contains one or more pages. It can be assigned to one or more business roles. The pages contain the apps the user works with. A space is shown in the navigation bar of the launchpad. When you click on it, the first page of the space is shown. When you click on the arrow next to the space title, a menu with the pages assigned to this space is shown. Spaces and the associated pages are assigned to users according to their roles. So, the space and role relation determine, which apps the users will see in their launchpad.

You can create a space based on a predefined space with the copy option (see Copying a Space [page 298]) or you can create a new and empty space as described in the following section.

How to create a new space

1. In the Manage Launchpad Spaces app, click on *Create*.

Create Space

Space ID: *

Z_BR_Customer_Service

Space Description: *

For users with role Customer Service

Space Title: *

Customer Service

☑ Also create a page

Page ID: *

Z_BR_Customer_Service_Overview

Page Description: *

First Page for Customer Service

Page Title: *

Overview

Create Cancel

Dialog to create a space with a page

2. Enter a unique space ID, a description and a title for the space. All three entries are mandatory. The space ID has to start with a Z, a Y (Y can be used for spaces designed by SAP partners without a namespace) or with a partner namespace. For a page in a partner namespace you enter the partner namespace as the first part of the ID, e.g. */partnernamespace/*.. Note that you cannot change the space ID anymore. The space title will be shown in the navigation bar.

3. Depending on your system configuration, you need to enter a transport. See Transport Information [page 301] for detailed information.

4. You can decide, if you want to create a page for the new page now, too. Select *Also create a page* and enter a page ID, a description and a title for the page. All three entries are mandatory. The page ID must be unique, and it has to start with a Z or a Y (Y should only be used for pages designed by SAP partners). The title will be shown as a menu item in the space menu. See Manage Launchpad Pages [page 278] for detailed information.

5. Click on *Create* to create the new space and page, if applicable. The new space is displayed in the spaces detail view.

> i Note
>
> When you have created the space, you can add existing or new pages in the space details view. Don't forget to assign the space to the business role or roles it is designed for.

1.3.4.8.1.6.4 Editing a Space

You can edit general space information, assign new pages to a space or remove currently assigned pages in the Manage Launchpad Spaces app.

Editing the general space information

In the *General* tab you can see all general space information, as e.g. the title, information about who created or changed the space and when. You can edit the title and the description. The title will be shown in the navigation bar of the launchpad. Note that you cannot edit the space ID.

Editing the page assignment

In the *Pages* tab you can see and add or remove the pages that are assigned to the space. In addition, you can define the page order and the page visibility.

- Add or remove pages: The list on the right side shows you all pages available for the launchpad. You can select one or more pages and then click on *Add* or drag & drop them.
- Define the page order: The order here determines how the pages are shown in the space menu for your users. The first page is displayed, when a user clicks on the space name. The first page of the first space is shown, when a user clicks on *Home*. You can sort the pages with the arrow up and down keys or by drag & drop.
- Set page visiblity: You can decide to hide a page so that your users won't see it by clicking on *Set Hidden*. You can e. g. hide a page, when you are editing it and don't want to show it to your users yet.

> i Note
>
> When you work on a smaller device, e.g. a tablet, or zoom in, the list of selectable pages is hidden. You can then switch between the list of available pages and the list of assigned pages with the buttons *Show Available Pages* and *Show Selected Pages*.

1.3.4.8.1.6.5 Assigning a Transport to a Space

You can assign transports to the space e.g. when creating or deleting a space or in the space details.

When you create or delete a space, you can enter a transport. Depending on your system configuration, a transport can be mandatory. See Transport Information [page 301] for more information.

You can see the existing transport assignments and add a new transport also in the space details view of the Manage Launchpad Spaces app.

- Open the space in the space details view and go to the *Transport* tab. When the space is already assigned to one or more transports, the transports are listed in the list *Open Transports*.

- To assign the space to a transport, click on *Assign*. Enter an existing transport in the dialog. When you start typing, a list of available transports will be shown.

1.3.4.8.1.6.6 Copying a Space

Learn how to copy a predefined space, e.g. an SAP-delivered space, a space from a partner or an existing customer-created space.

When you want to use a predefined space for your users or when you want to create a new space that is similar to one of the existing spaces, you can copy the space. During the copy process you can decide how to handle the pages that are assigned to the space. Afterwards you can adapt the copied space to your needs. Please read Working with Predefined Spaces and Pages [page 275] first to understand how they work exactly.

1. Select the space you want to copy in the overview or in the space details view of the Manage Launchpad Spaces app.


Copy a space step 1

2. Enter a unique space ID, a description and a title for the new space. All three entries are mandatory. The space ID must be unique and has to start with a Z or a Y (Y should only be used for spaces designed by SAP partners). Note that you cannot change the space ID anymore. If necessary, enter a transport (see Transport Information [page 301] for more information). When you have entered all information, click on *Step 2*.

3. Select how to proceed with the pages that are currently assigned to the original space. You can decide to ignore them (*Ignore All*), you can assign the existing pages to the new space (*Assign All*) or you can copy the pages (*Copy All*). It is also possible to select different options for each page: Select an action for the

page in the list column *Page Action*. When you copy a predefined space, you can only copy or ignore the assigned pages; predefined pages cannot be assigned directly.
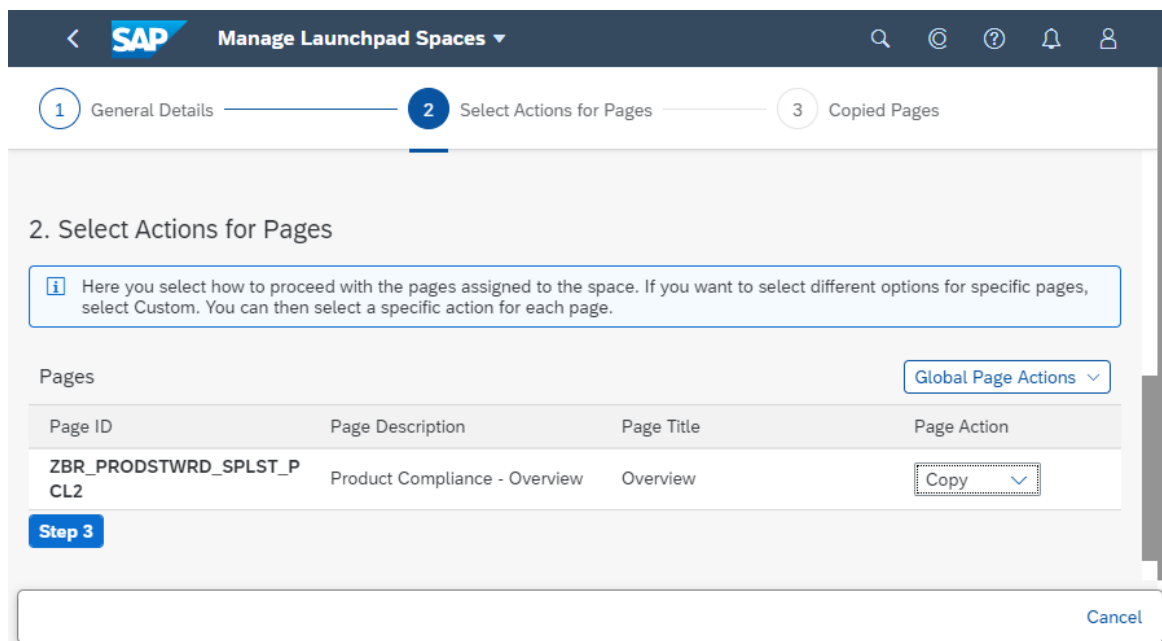
Copy a space step 2

4. If you have selected pages to be copied, you now need to enter a new page ID, description and title for each page. The page ID must be unique, and it has to start with a Z or a Y (Y should only be used for pages designed by SAP partners). When you are done, click on *Review* to see a preview of the new space with the selected pages.
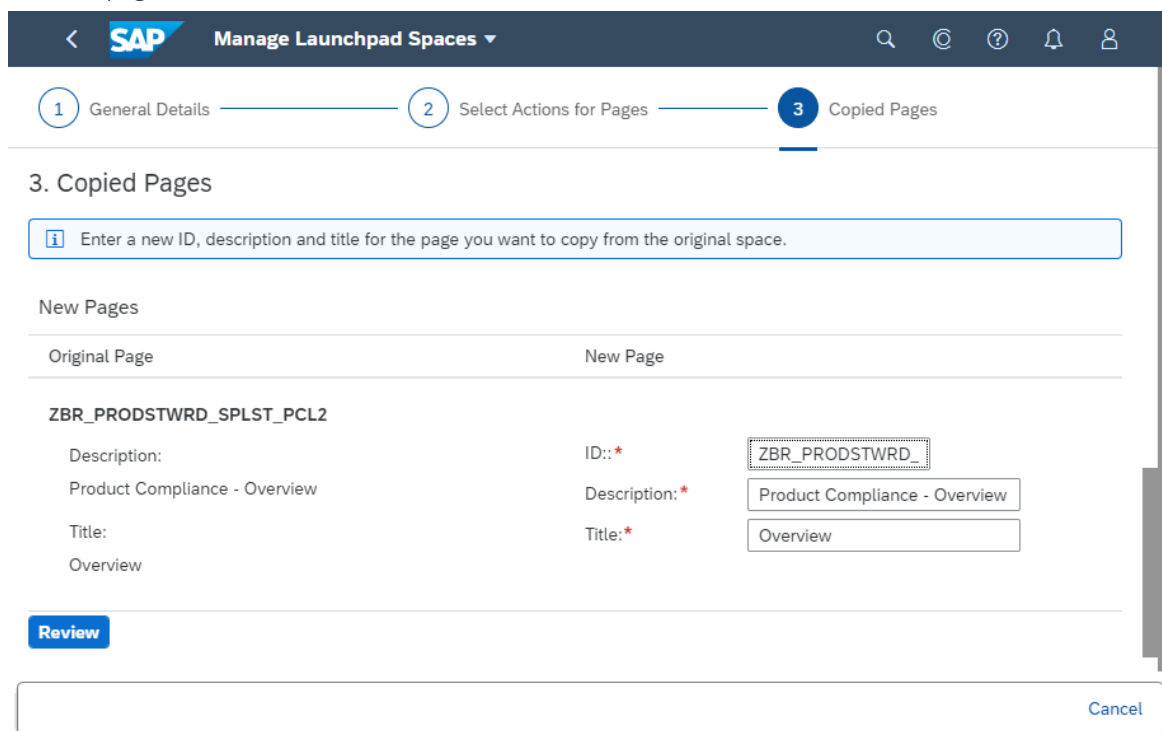
5. Click on *Edit* if you want to change the space details or the selected pages. When you are satisfied with the new space, click on *Save*. The space then is added to the customer-created spaces list.

> **i Note**
>
> You can add new or existing pages to the new space in the space details view. Don't forget to assign the new space to the business role or roles it is designed for (see Configure Roles for Spaces [page 337]).

1.3.4.8.1.6.7 Deleting a Space

If you no longer need a space, you can delete it.

In the Manage Launchpad Spaces app, you can delete an existing customer-created space. At first, make sure that the space is not assigned to a business role, as you cannot delete spaces that are still in use. By this, it will be prevented that users may not see any pages because no space is assigned to their business role.

You can remove the assignment to a business role in transaction PFCG (if you have the appropriate rights). See Configure Roles for Spaces [page 337] for more information.

To delete a space, select it and click on *Delete* in the spaces overview or in the spaces details. If necessary, enter a transport for the process (see Transport Information [page 301]). Then confirm the warning message and the selected space will be deleted from the system. If there are pages that are assigned to the space, they will be unassigned but not removed from the system.

1.3.4.8.1.6.8 Sorting the Spaces

You can influence the sort order of the spaces in the launchpad UI in different ways.

There are several ways to control in which order the spaces are displayed in the navigation bar of the launchpad user interface (see Spaces and Pages [page 419] for the user view).

First, you can decide, if the spaces should be sorted by space ID or by space title. This can be controlled with the launchpad configuration parameter SPACES_SORT_CRITERION. The default sort criterion is space title. Sorting by title gives you more flexibility as you may change the title later; the space ID can't be changed after you have created the space. How to set the parameter is explained in Setting Launchpad Parameters [page 63].

> **i Note**
>
> The spaces are sorted lexicographically in both cases and capitalization is considered. Spaces are sorted according the following criteria: Special characters (e.g "_", "-" and "/") are sorted first, then numbers, then small letters before capital letters (e.g. abc before aBC).

If this sorting is not sufficient, you can enhance it by adding individual priority values to selected spaces. You can add priority values from -100 to 100. Imagine this as as scale spanning from -100 to 100: The lower the value of the priority, the further left the space is shown in the navigation bar. The value indicates the position: -100 means that this space will be shown as first space. It is not necessary to assign priorities to all spaces. A space with no defined priority value will be handled as having priority 0. Spaces with the same sort priority will be sorted according to the space title or space ID (as defined with the parameter SPACES_SORT_CRITERION).

When you define priorities, remember that the individual users won't see all the spaces but only those spaces that are assigned to their user roles. So, it is probably not necessary to define values for all spaces and a strict

hierarchy, but rather sufficient to only define sort priority values for the most important and the least important spaces.

It is not possible to define different sort priorities for a space that is used for different roles. If you want to define different sort priorities in different contexts, you need to copy the space and then add a different priority to the new copied space.

Example, how spaces are sorted according to priorities

Space Title	Sort Priority	Position
ZFavorites	-100	1
Z_BR_Employee	-20	2
Z_BR_BILLING_CLERK	0	3
Z_BR_BILLING_SUPERVISOR	0	4
Z_BR_RERTURNS_REFUND_CLERK	50	5

How to enter a sort priority

You enter the sort priority values in the Manage Launchpad Spaces app in the tab *General*. Search the space you want to give a sort proirity value. Open the tab *General* and click on *Edit*. Enter a value from -100 to 100 in the field *Sort Priority*. Values may include up to three decimal numbers. Save your changes and process with the next space if necessary.

1.3.4.8.1.7 Transport Information

When you copy, delete or create a new space or page, you might need to enter a transport. When this is necessary depends on your system configuration.

There are three different transport modes:

1. `Off`: no transport. The *Transport* field should not be visible.
2. `Manual`: transport information is optional. The *Transport* field shows a blue frame, when you have not entered a transport. When you want to transport the page or space, you can select a transport, if the space or page should be transported to a different system.
3. `Automatic`: transport information is mandatory. The *Transport* label shows a * and the field shows a red frame, when you have not selected a transport. When you don't enter transport information, you can't create the space or page.

Which transport mode is active in a specific system, is defined by your administrator (transaction SCC4). If no transport is available to select, you can create a transport with transaction SE09 in the SAP GUI. In general, you can only assign a page or a space to a transport directly in the launchpad. If you want to change the transport assignment, this can only be done in transaction SE09.

The transport is a customizing transport. It uses the OData service FDM_TRANSPORT_SRV. The service is pre-registered but has to be activated. This is covered by a task list (see Configuration Using Task Lists). You can also activate the service manually as described in Activating SAP Gateway OData Services [page 23] and

Configuring ICF Nodes [page 24]. If the service is not activated but the system setup is `Manual` or `Automatic`, the transport component is not shown. An error occurs during creation, saving or deletion telling you that the transport information is missing.

> ### i Note
>
> Note that the space or page is not locked after it was assigned to a transport. A space or page can be assigned to several transports. In addition, spaces, pages and roles that are connected can be assigned to different transports. When you select different transports, it is necessary that all transports are released, so that all three objects are availabe in the target system. Otherwise the pages and spaces might not be shown correctly.
>
> You can check the transport assignment of a space in the *Transport* tab of the Space Details view in the Manage Launchpad Spaces app (see Assigning a Transport to a Space [page 297]). The transport assignment of a page is shown in the *Transport* tab of the Page Details view in the Manage Launchpad Pages app (see Editing a Page [page 283]).

1.3.4.8.1.8 Enabling Spaces

There are two parameters and one user setting that influence if the launchpad uses spaces or the home page for displaying the apps.

The layout of the page area can be changed. You can either show the classic home page layout where all apps are shown on one page in different groups. Or you can enable the spaces. The spaces contain a page that shows apps for a specific user group defined by the business role. (How to define a page is described in section Editing a Page [page 283]).

- If you want to enable spaces and pages for your users, set the parameter `SPACES` to true. False means that the home page is show as standard setting. Note that this setting is applied to all users. You should have created and assigned spaces and pages for all business roles you have assigned to your users, before you switch to spaces.
  If you only have defined a few pages so far, you should not yet enable the spaces mode for all users with the parameter `SPACES`. Instead, you can set the parameter `SPACES_ENABLE_USER` to true and ask those users with roles for that pages are available, to switch to the spaces mode themselves. They can then start working with pages.
- If your users should be able to switch between the spaces and the classic home page, set the parameter `SPACES_ENABLE_USER` to true. Users then can decide in the user settings, which layout they prefer. See Managing Your Settings [page 509] for more information. Please note that the user settings overwrite the custom settings you define here.

How to set parameters is explained in section Setting Parameters in SAP Fiori Customizing [page 63].

Examples

There are four possible configurations:

- If all users should see the spaces layout and be able to switch between the layouts, set `SPACES` and `SPACES_ENABLE_USER` to true.

- If all users should only see the spaces layout, set SPACES to true, and set SPACES_ENABLE_USER to false.
- If all users should see the home page by default but be able to switch between the layouts, set SPACES to false, and set SPACES_ENABLE_USER to true.
- If you want to deactivate the spaces layout completely for now, set SPACES and SPACES_ENABLE_USER to false. Then the home page is always displayed.

When you have enabled spaces, you can also enable My Home for your users with the parameter SPACES_MYHOME.

1.3.4.8.1.9 Analyzing Space and Page Assignments

With this report you can create an aggregated list of the available content in the SAP Fiori launchpad.

The report provides an overview of your launchpad content in a simple table view. It's based on business role level and shows assigned spaces, pages, sections, and tiles.

Key Features

The following features are available:

• Display a list of selected business roles and spaces and their assigned content.

• Export a spreadsheet for further offline analysis.

• Adjust the list by adding or removing columns.

Procedure

1. Start the transaction /UI2/RSP_LIST.
2. Enter the role ID in the *Roles* field. You can enter multiple role IDs.
   You can also filter for spaces. Simply enter the space ID in the *Spaces* field. You will get a list of all roles where the space is assigned.
3. You can optionally mark the checkbox *Include Roles without Assigned Spaces* to also display roles that have no space assigned.
4. Choose *Execute*.
   Now you see a list of your content assigned to the selected roles or spaces.

Next Steps

You can now analyze the content and also download the list for further analysis.

Related Information

[Working with the SAP List Viewer (ALV)]

1.3.4.8.2   Managing Groups

This section provides information about managing groups in the launchpad designer.

> **i Note**
>
> Creating new home pages based on groups is no longer recommended. Instead, use the "Manage Launchpad Spaces" and "Manage Launchpad Pages" apps to define the launchpad layout and structure. See Managing Launchpad Spaces and Pages [page 269].
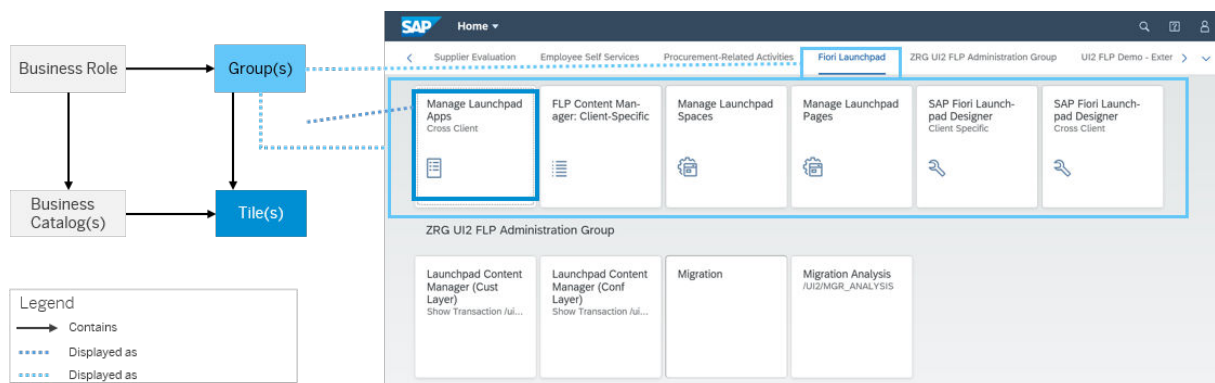
Prerequisites

You have finished the initial setup of the launchpad designer.

See Initial Setup of the Launchpad Designer [page 186].

What is a group?

The following graphic shows the launchpad layout based on groups:

A group consists of tiles (and links) representing a subset of apps. Administrators assign tiles from one or more catalogs to a group and groups and catalogs to a role. Users that have this role assigned, can view the group on their launchpad home page. Users can personalize their home page by adding or removing apps from the out-of-the-box groups or self-defined groups.

> **i Note**
>
> The authorization to see and use a specific app and the corresponding tile (or link) is checked on catalog level, based on the tile intent. Therefore, users see only those tiles and links in a group that are referenced

> from a catalog that is part of a role the user is assigned to. In addition, the corresponding target mapping needs to be part of an assigned catalog.
>
> For tiles and links that do not use intent-based navigation (e.g. static App Launcher Tiles with URL), no authorization check is done. Therefore, they are always displayed to the user, independent of the source catalog assignment.

Related Information

1.3.4.8.2.1 Creating or Deleting Groups

Context

A group contains predefined content that you can view on the page. You can add tiles to groups from catalogs. You can specify that a group cannot be personalized by the end user.

> i Note
>
> Personalization of a group includes the ability to delete or move tiles, and to add tiles to the group.

You cannot create target mappings in groups. You have to assign the groups to an authorization role in the back end to which the users can be assigned.

Creating a Group

Groups appear both in the left panel and on the content area of the screen. In the left panel, the groups appear with the number of the tiles in each group.

1. Choose the *Create Group* plus icon at the bottom of the left panel.

    A *Create Group* dialog box appears.
2. In the *Title* field, enter the title for the group.
3. In the *ID* field, enter a unique id for the group.
4. To enable users to personalize this group, select the *Enable users to personalize their group* check box.
    To disable users from personalizing this group, deselect this check box.

5. Choose *Save*.

Deleting a Group

1. Choose a group from the group list in the left panel.
2. Drag and drop the selected group in the trash zone.

   A *Confirmation* dialog box appears.
3. Choose *OK*.

Related Information

Personalizing the Home Page [page 501]

1.3.4.8.2.2 Editing Groups

You can edit the properties of a group.

Procedure

1. Choose the group to edit.
2. Choose the *Edit Group* pencil icon from the header.
   The *Edit Group* dialog box appears.
3. In the *Title* field, you can edit the title of the group.
4. To enable users to personalize this group, select the *Enable users to personalize their group* check box. To disable users from personalizing this group, deselect this check box.

   Personalization of a group includes the ability to delete or move tiles, and to add tiles to the group.

   Note that you do not fully control the lifecycle of personalized groups. For example, if an end user has personalized a group before you as an administrator made changes to it, your changes will no longer be reflected to the end user. See Issues with Tiles, Groups or Catalogs [page 632].
5. Choose *Save*.

Related Information

Personalizing the Home Page [page 501]

1.3.4.8.2.3 Adding Tiles to and Removing Tiles from Group

Context

For more information on configuring a tile, see Configuring Tiles [page 207].

Tiles can also be rendered as links in the SAP Fiori launchpad. This is helpful if a group contains a large number of tiles. You have the following options to add tiles to a group:

- Add them as tiles
- Add them as links
- Move existing tiles from the *Show as Tiles* to the *Show as Links* area and vice versa
  You can use drag and drop to move the tiles.

> **i Note**
>
> Tiles as links can only be added if you select action tiles (static and dynamic app launcher tiles).

> **i Note**
>
> An empty predefined group is still shown in the launchpad unless it has been locked by the administrator.

You can add tiles or links to a group or remove them from a group as follows:

Adding Tiles or Links

You can view all of the groups and tiles associated with each group on the panel to the left.

1. Select the group for which you want to add a tile.

   The tile or tiles associated with the selected group appear in the content area on the screen.
2. Select the *Add Tile* icon.

   The *Add Tile to Group* `<Name of the Selected Group>` screen appears.
   By default, the first catalog is selected in the *Catalogs* search.
3. In the *Catalogs* search, you can either enter any catalog name or press F4 to select a catalog from the list.

   > **i Note**
   >
   > The *Catalogs* search allows you to filter catalogs.

   Tile or tiles associated with the selected catalog appear on the screen.
4. Select the *Add Tile* button on the tile you want to add to the group.

   `Tile added successfully` or `Link added successfully` message appears on the screen.

   > **i Note**
   >
   > You can add multiple tiles from a catalog to a group.

5. Choose the *Back* button.

    The tile is added to the group.

Removing Tiles or Links

1. Choose the tile that you want to remove or delete from the selected group.
2. Drag and drop the selected tile into the trash zone.

> **i Note**
>
> If you're using unlocked groups and the end user has personalized the group, we recommend to reset the personalization. See Delete Personalization Data [page 355].

    The selected tile is removed from the group.

1.3.4.9   Integrating Remote Content

For applications located in the backend (for example, applications based on Web Dynpro ABAP or SAP GUI), you need to define remote function call (RFC) destinations from the ABAP front-end server to the ABAP back-end system(s) to enable integration in the launchpad.

Related Information

Configuring Remote Systems [page 308]
Setting Up Technical Catalogs with the SAP Fiori Launchpad App Manager [page 132]
Integrating Applications from SAP Easy Access Menu [page 312]
Configuring Multiple Back-End Systems [page 320]

1.3.4.9.1   Configuring Remote Systems

In SAP Fiori launchpad, the system that a user initially connects to, and the system where applications assigned to the user are running, the so-called remote system, may be different. This section explains how you as an administrator specify those remote systems.

In the launchpad, remote targets are used in the following cases:

-   When navigating with tiles that point to SAP GUI transactions or Web Dynpro ABAP applications.
    Since these application types are located in a back-end system, you need to define remote function call (RFC) destinations from the ABAP front-end server to the ABAP back-end system(s).
-   When passing data origin for SAP Fiori applications via sap-system parameters.
    In this scenario, the application requires information from a remote system. The data origin points to the location of data used by the target application and indicates the location from where the data shall be loaded. For SAPUI5 applications, this is a back-end system implementing OData services.

RFC Connections

Connection Types and Naming Conventions

You configure connections to remote systems in transaction SM59. The following connection types are relevant for the launchpad:

ABAP Connection (type 3)	This type is required to support navigation using RFC protocol or native application embedding. For example, you need this connection type when you integrate the launchpad in SAP Business Client and launch SAP GUI transactions using embedded SAP GUI for Windows. Create the corresponding connection using the following naming convention: **`<system alias>_RFC`**
HTTP(S) Connection to ABAP System (type H)	This type is required to support HTTP-based communication, e.g. all Web-browser-based application technologies such as Web Dynpro ABAP and SAP GUI for HTML. Create the corresponding connection using the following naming convention: **`<system alias>_HTTP`** **`<system alias>_HTTPS`**

> **i Note**
>
> Transaction SM59 does not allow you to create several destinations with the same name even if the connection types are different. Therefore, we recommend you to create entries with the same base name and an extension to distinguish between the different connection types (_RFC, _HTTP, _HTTPS). Using this approach, it is possible to reference multiple connections by passing the base name (**`<system alias>`**) only.

Defining RFC Connections in SM59

- ABAP Connection Definition
- HTTP Connection Definition

> **i Note**
>
> On the *Technical Settings* tab, make sure that the *Path Prefix* field is empty.

System Aliases

Referencing System Aliases

You specify a system alias in the target mapping to define application or data origin:

- For application types *Transaction*, *Web Dynpro* and *URL* you define the system alias directly in the target mapping in the SAP Fiori launchpad designer.
  The parameter *System Alias* in the target mapping configuration indicates that the application shall call the remote system that is specified.
  See Configuring Target Mappings [page 217].

- For application type *SAP Fiori App using LPD_CUST* you define the system alias in Launchpad Customizing (transaction `LPD_CUST`) on the frontend server.
  The parameter *System Alias* in the *Change Launchpad Role* screen indicates that the application shall call the remote system that is specified.
  See Changing LPD_CUST Entries for Navigation Targets [page 228].

The system alias references the RFC destinations which have been defined according to the naming conventions mentioned above.

You can use the URL parameter `sap-system` that is passed as part of the intent to overwrite the system alias parameter defined in LPD_CUST or launchpad designer.

See Overwriting the System Alias for the Target Application [page 528].

System Alias Mapping

To reduce maintenance effort if you have several system aliases pointing to the same physical source system, you can define a mapping for the system aliases.

See Map System Aliases to RFC Connections [page 17].

Related Information

Configuring Remote Systems in SM59
Establish Connections between Front-End Server and Backend [page 15]
Remote Systems (SAP Business Client)

1.3.4.9.1.1 Establish a Connection from Back-End System to Front-End Server

Each back-end system connected to the launchpad needs to know the location of the SAP Fiori front-end server on which the launchpad runs.

Procedure

> **i** Note
>
> You can also use the task list `SAP_SAP2GATEWAY_TRUSTED_CONFIG` for establishing the connection in an automated way.

Perform the following steps on the relevant back-end systems:

1. Decide on a unique logical system alias identifying the SAP Fiori front-end server (e.g. `FESCLNT700`).

2. Create an RFC destination of connection type 3 (ABAP connection).

   This connection type is used by *Role Maintenance* (transaction `PFCG`) to determine which authorizations are required to run the back-end applications referenced by the launchpad. See Assign Business Catalogs to Roles [page 330].

   a. In Customizing, choose ▶ *SAP NetWeaver* ❯ *Gateway* ❯ *OData Channel* ❯ *Connection Settings* ❯ *SAP NetWeaver Gateway to SAP System* ❯ and launch the IMG activity *Manage RFC Destinations* or directly launch transaction SM59.

   b. Select the *ABAP Connections* node and choose *Create*.

   c. In the *RFC Destination* field, enter a name following the naming convention `<Logical System Alias>_RFC` (for example **`FESCLNT700_RFC`**).

   d. Enter a description.

   e. On the *Technical Settings* tab, select the load balancing status and provide the required information:

      • Yes: Specify target system, message server, and group.

      • No: Specify target host and instance number.

   f. On the *Logon & Security* tab, enter the client of the front-end server system to which you want to connect and mark the *Current User*

3. Create an RFC destination of connection type H (HTTP(S) connection).

   This connection type is used to call the SAP Fiori launchpad from a back-end application.

   a. In Customizing, choose ▶ *SAP NetWeaver* ❯ *Gateway* ❯ *OData Channel* ❯ *Connection Settings* ❯ *SAP NetWeaver Gateway to SAP System* ❯ and launch the IMG activity *Manage RFC Destinations* or directly launch transaction SM59.

   b. Select the *HTTP Connections to ABAP System* node and choose *Create*.

   c. In the *RFC Destination* field, enter a name following the naming convention `<Logical System Alias>_HTTPS` (for example **`FESCLNT700_HTTPS`**).

   d. Enter a description.

   e. On the *Technical Settings* tab, enter the following:

      • *Host* field: Enter the HTTPS host.

      • *Port* field: Enter the HTTPS port.

      • *Path Prefix* field: Enter the path to SAP Fiori launchpad (for example **`/sap/bc/ui2/flp`**). See Launching the Launchpad [page 101] for initial installation or check the path of the actual FLP URL in the browser when updating existing FLP installation.

   f. On the *Logon & Security* tab, mark the *Trusted Relationship* checkbox and provide the following:

      • *Client* field: Enter the client of the front-end server system to which you want to connect.

      • Mark the *Current User* checkbox.

      • Under *Security Options*, set *SSL* to *Active* and *SSL Certificate* to *Default SSL Client*.

4. Add the system alias **`FIORI`** to the view `/UI2/VC_SYSALIAS`.

   a. Start *Maintain Table Views* (transaction `SM30`).

   b. In the *Table/View* field, enter **`/UI2/VC_SYSALIAS`**.

   c. Choose *Edit* and enter the alias **`FIORI`**.

5. In the view /UI2/V_ALIASMAP, map the system alias created in the previous step to the RFC destinations pointing to the front-end server.

   a. Start *Maintain Table Views* (transaction `SM30`).

   b. In the *Table/View* field, enter **/UI2/V_ALIASMAP**.

   c. Choose ▌ *Edit* ❭ *New Entries* ▌.

   d. Provide the following:

Field	Description
Client	To create a client-dependent mapping, enter the relevant clients. Leave this field blank for a client-independent mapping.
Source	Enter **FIORI** (see step 4).
Target System Alias	Enter the RFC destinations created in step 2 and 3, leaving out the suffix "_RFC" and "_HTTPS".

   For more information, see Map System Aliases to RFC Connections [page 17].

6. Register the front-end server location as safe host in the HTTP_WHITELIST table.

   For more information, go to https://help.sap.com/viewer/p/SAP_NETWEAVER, select your release and search for "Using a Whitelist for Clickjacking Framing Protection".

Related Information

Connecting SAP Gateway to Back-End System (Trusted RFC)

1.3.4.9.2   Integrating Applications from SAP Easy Access Menu

You can provide users with the option to select SAP GUI and Web Dynpro ABAP applications retrieved from the SAP Easy Access menu in the ABAP backend and add them to the launchpad.

Prerequisites

You are assigned to the role `SAP_FLP_ADMIN`. See Configuring Roles with Launchpad Start Authorizations [page 327].

Context

The SAP Easy Access menu provides access to SAP GUI and Web Dynpro ABAP applications. There are two types of menus:

| User menu | • Contains only those items that the user needs to perform daily tasks |
| | • Is assigned by an administrator using role maintenance in transaction PFCG. |

| SAP menu | • Contains the complete set of functions offered by an SAP system |
| | • Is role-independent and therefore displays the same content for all users |

> **i Note**
>
> The SAP Easy Access menu is not supported on phones.

> **→ Tip**
>
> You can hide the *SAP Menu* and *User Menu* in the app finder. For more information, see Launchpad Configuration Parameters [page 29], escpecially the entry for the Easy Access menu.

Here is a high-level overview of the configuration steps you need to perform if you want to offer apps from SAP Easy Access menu to launchpad users:

- Define the list of SAP systems for which SAP Easy Access menu entries should be retrieved.
  During launchpad runtime, a dialog with the list of defined systems will be displayed. End users select a system to get the SAP Easy Access menu entries displayed as tiles on the launchpad.
- Establish connections between front-end server and relevant back-end systems.
- Activate OData services to retrieve the relevant menu entries from the backend system.
- Enable that, during launchpad runtime, users can navigate from the launchpad to the backend applications.

You have two options to enable the SAP Easy Access menu:

| Role-free | Enable the feature for all users in the selected client or system. |
| Role-based | Enable the feature for all users assigned to a specific role. |

The following graphic provides an overview of the required steps:
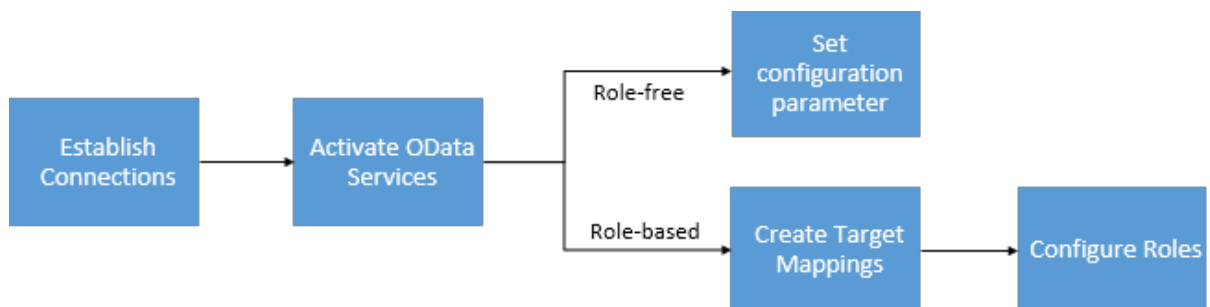
Perform the following steps on the SAP Fiori front-end server:

Procedure

1. Establish connections to back-end systems.

   a. Decide on a unique logical system alias identifying the back-end system in the target client (e.g. `FINASIA` or `FINCLNT100`).

      This alias will be used at runtime and definition time to identify the respective backend.

   b. Create RFC destinations pointing to the relevant back-end systems.

      1. In Customizing, choose ▶ *SAP NetWeaver* ❯ *Gateway* ❯ *OData Channel* ❯ *Connection Settings* ❯ *SAP NetWeaver Gateway to SAP System* ◀ and launch the IMG activity *Manage RFC Destinations* or directly launch transaction SM59.

      2. For each logical system alias, create the following RFC destinations:

         • Connection type 3: ABAP connection
           This connection type is used to extract SAP Easy Access menu entries from the back-end server to the front-end server.
           Use the following naming convention: `<Logical System Alias>_RFC`

         • Connection type H: HTTP(S) connection
           This connection type is used during the launchpad runtime to start the Web Dynpro applications and SAP GUI transactions from the launchpad.
           Use the following naming convention: `<Logical System Alias>_HTTP` or `<Logical System Alias>_HTTPS`
           We recommend to use an HTTPS connection.

         Note that you also have to create RFC destinations for the local system (`LOCAL_RFC` and `LOCAL_HTTPS`).

   c. Create or select an SAP system alias for each backend system to which you want to connect.

      1. In Customizing, choose ▶ *SAP NetWeaver* ❯ *Gateway* ❯ *OData Channel* ❯ *Configuration* ❯ *Connection Settings* ❯ *Connection SAP NetWeaver* ❯ *SAP NetWeaver Gateway to SAP System* ◀ and launch the IMG activity *Manage SAP System Aliases*.

      2. In the *SAP System Alias* field, enter the logical system alias defined in step 1.a.

      3. In the *RFC Destination* field, enter the name of the RFC destination of connection type 3 (ABAP connection) created in step 1.b.

      The RFC destination for the SAP system alias `LOCAL` should be set to `NONE`. In many systems, this setting is already preconfigured.

   d. For client-dependent setups: Add an entry in view `/UI2/V_ALIASMAP` to map the logical system alias to the destination name prefix. See Map System Aliases to RFC Connections [page 17].

      This step is not required for the simple client-independent setups, where the logical system alias and the destination name prefix are identical.

2. Activate OData services.

   a. In Customizing, choose ▶ *SAP NetWeaver* ❯ *Gateway* ❯ *OData Channel* ❯ *Administration* ❯ *General Settings* ❯ *Activate and Maintain Services* ◀ (transaction `/IWFND/MAINT_SERVICE`) and activate the following OData services:

`/UI2/EASY_ACCESS_MENU`	Retrieves the menu entries of the SAP menu.
`/UI2/USER_MENU`	Retrieves the menu entries of the user menu.

b. To add a service, choose ⊞.

c. Select the system alias of the backend system (see step 1), then select a service from the table above.

> **i Note**
>
> In an SAP Fiori hub deployment, the two services should not use the processing mode **co-deployed**, as this mode connects to the local system without using any alias or RFC destination.

For information about activating OData services, see the SAP Gateway Developer Guide at ▶ *http:// help.sap.com/netweaver* ❯ *SAP Gateway* ❯ *Developer's Guide* ❯ *OData Channel* ❯ *Basic Features* ❯ *Service LifeCycle* ❯ *Activate and Maintain Services* ◀.

3. Depending on your use case, enable SAP Easy Access menu for a user role or for the whole system/client:

- For the role-independent enablement, proceed as follows:

  1. In *SAP Reference IMG* (transaction `SPRO`), choose ▶ *SAP NetWeaver* ❯ *UI Technologies* ❯ *SAP Fiori* ❯ *Configuring SAP Fiori Launchpad* ◀.
  2. Launch the customizing activity for cross-client or client-specific settings.
     See Setting Parameters in SAP Fiori Customizing [page 63].
  3. Under *FLP Configuration*, add the parameter `FLP_EAM_ALIASES`.
  4. In the *Property Value* field, enter a comma-separated list of system aliases as defined in SAP Gateway customizing (see step 1c).
     For example: **LOCAL,FINASIA**.
     The specified aliases will be used to retrieve entries from the SAP Easy Access menu in the related back-end systems.

- For the role-dependent enablement, proceed as follows:

  1. To allow navigation from the launchpad to SAP GUI transactions and Web Dynpro ABAP applications, for each back-end system, you create one target mapping for transactions and one for Web Dynpro applications.
     See Creating Target Mappings for the SAP Easy Access Menu [page 316].
  2. Configure authorization roles.
     - Make sure the catalog containing the target mappings created in the previous step is assigned to a PFCG user role. See Assign Business Catalogs to Roles [page 330].
     - Make sure the users for whom you want to offer SAP Easy Access menu entries are assigned to the same role as the catalog. See Assigning Users to Roles [page 339].

Results

If the users for which you enabled this feature open the app finder in the launchpad, all SAP GUI and Web Dynpro ABAP applications integrated in an SAP GUI user menu or SAP menu are retrieved from a selected back-end system.

The users can add the applications retrieved from the SAP Easy Access menu to the launchpad homepage. See Adding Apps to the Home Page [page 504].

> i Note
>
> To ensure fast loading times of SAP menu apps in the app finder, we recommend you to schedule the report `/UI2/EAM_BUILD_CACHE` periodically. See Scheduling Caching of SAP Menu Entries [page 27].

Related Information

Issues with SAP Easy Access Menu [page 649]

1.3.4.9.2.1 Creating Target Mappings for the SAP Easy Access Menu

To provide SAP Easy Access menu entries in users' home page personalization and enable navigation to the respective transactions and Web Dynpro applications, you need to create two target mappings for each back-end system.

For each back-end system, you need to create two target mappings:

- A target mapping for transactions
- A target mapping for Web Dynpro applications

For more information, see Creating or Deleting Tiles or Target Mappings [page 203] and Configuring Target Mappings [page 217].

Target Mapping for Transactions

Configure this target mapping as follows:

Field	Description
Semantic Object	`Shell` This value needs to be written exactly like this.
Action	`startGUI` This value needs to be written exactly like this.
Application Type	`Transaction`

Field	Description
Title	Enter a title, for example `BE1 Client 100`.
	The title that you enter in this field will be displayed to end users when they select a system from which they can then select SAP User Access menu entries.
	→ Tip
	We recommend that you use the same title as in the target mapping for Web Dynpro applications that you create for the same back-end system.
	In case both target mappings are specified, the title in the `Shell-startGUI` target mapping will be preferred over the title in the `Shell-startWDA` target mapping.
	The title in the Shell-startGUI target mapping will be preferred over the title in the Shell-startWDA target mapping, in case both the target mappings are specified
Transaction	Enter a dummy value.
	This field must not be empty, but its value will be ignored.
System Alias	This field **must** be empty.
Device Types	Make sure that *Desktop* is selected, and *Tablet* and *Phone* are deselected.
Parameters	Enter the parameters as described in the table below.
Allow additional parameters	Make sure that this option is selected.

Enter the following parameters:

Parameters

Name	Mandatory	Value	Is Regular Expression	Default Value	Target Name
`sap-system`	Active	Enter the system alias as defined in SAP Gateway customizing (❚▶ *SAP NetWeaver* ❭ *Gateway* ❭ *OData Channel* ❭ *Configuration* ❭ *Connection Settings* ❭ *Connection SAP NetWeaver* ❭ *SAP NetWeaver Gateway to SAP System* ❭ *Manage SAP System Aliases* ❭). Example: `BE1CLNT100`	Inactive	Empty	Empty
`sap-ui2-tcode`	Active	`.*` Enter it exactly like this (a dot followed by a star).	Active	Empty	Empty

Target Mapping for Web Dynpro Applications

Configure this target mapping as follows:

Field	Description
Semantic Object	`Shell` This value needs to be written exactly like this.
Action	`startWDA` This value needs to be written exactly like this.

Field	Description
Application Type	`Web Dynpro`
Title	Enter a title, for example **`BE1 Client 100`**.
	The title that you enter in this field will be displayed to end users when they select a system from which they can then select SAP User Access menu entries.
	→ Tip
	We recommend that you use the same title as in the target mapping for transactions that you create for the same back-end system.
	In case both target mappings are specified, the title in the `Shell-startGUI` target mapping will be preferred over the title in the `Shell-startWDA` target mapping.
Application	Enter a dummy value.
	This field must not be empty, but its value will be ignored.
Configuration	Leave this field empty. Its value will be ignored.
System Alias	This field **must** be empty.
Device Types	Make sure that *Desktop* is selected, and *Tablet* and *Phone* are deselected.
Parameters	Enter the parameters as described in the table below.
Allow additional parameters	Make sure that this option is selected.

Enter the following parameters:

Parameters

Name	Mandatory	Value	Is Regular Expression	Default Value	Target Name
`sap-system`	Active	Enter the system alias as defined in SAP Gateway customizing (*SAP NetWeaver* > *Gateway* > *OData Channel* > *Configuration* > *Connection Settings* > *Connection SAP NetWeaver* > *SAP NetWeaver Gateway to SAP System* > *Manage SAP System Aliases*). Example: `BE1CLNT100`	Inactive	Empty	Empty
`sap-ui2-wd-app-id`	Active	`.*` Enter it exactly like this (a dot followed by a star).	Active	Empty	Empty

Related Information

[Adding Apps to the Home Page [page 504]](#)

1.3.4.9.3 Configuring Multiple Back-End Systems

In a system landscape with multiple back-end systems, users may need to access the several instances of the same application with data originating from two or more back-end systems, from the same launchpad.

This documentation describes how to configure an instance of an application with data from one back-end system, and another instance of the same application with data from another back-end system. It does not cover multi-origin scenarios where one application collects data from multiple back-end systems.

This can be configured in different ways, depending on the application type:

- Configuring SAPUI5 Apps from Multiple Back-End Systems [page 321]
- Configuring Other Apps from Multiple Back-End Systems [page 322]

Related Information

Example for Configuring Multiple Back-End Systems [page 324]

1.3.4.9.3.1 Configuring SAPUI5 Apps from Multiple Back-End Systems

In a system landscape with multiple back-end systems behind one front-end server, users may need to access several instances of the same SAPUI5 application with data originating from two or more back-end systems.

This documentation describes how to configure an instance of an application with data from one back-end system, and another instance of the same application with data from another back-end system. It does not cover multi-origin scenarios where one application collects data from multiple back-end systems.

The following figure shows a system landscape where a user accesses an SAPUI5 application that may originate from either of two back-end systems.
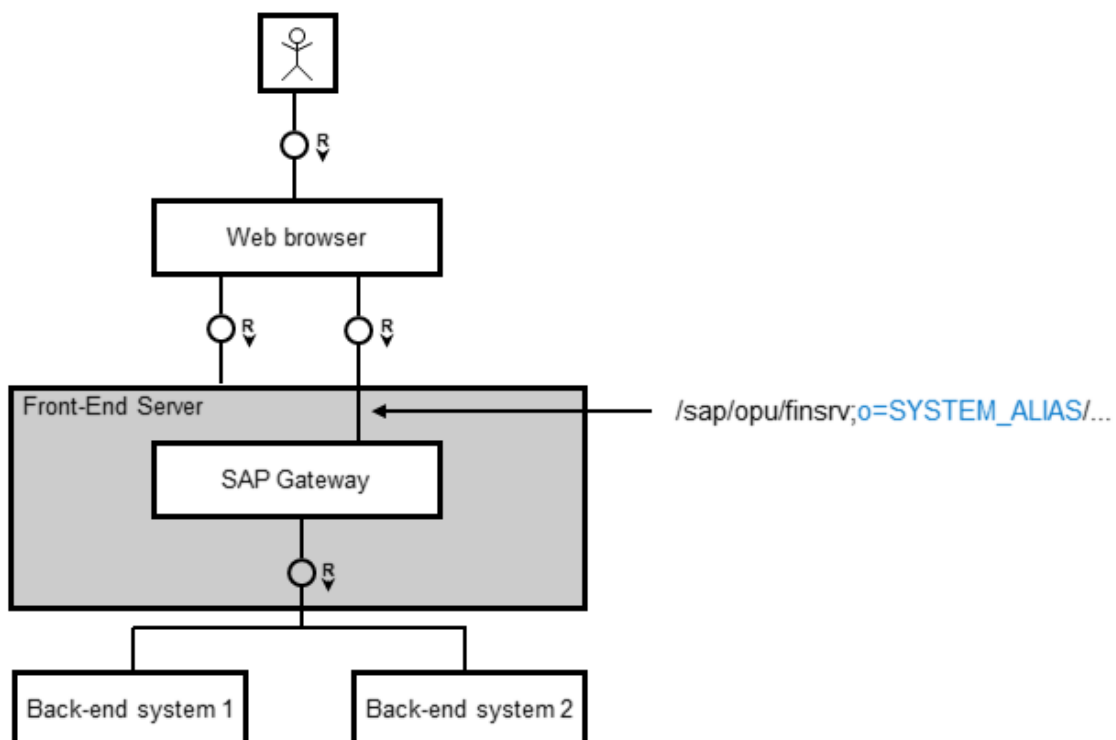
To achieve this, the OData URLs need to be distinct. SAP Gateway segment parameters (o=`<SYSTEM_ALIAS>`) are used to differentiate the requests.

If the SAPUI5 application is enabled for multiple back-end systems, you can use the `sap-system` parameter in the URL to specify from which back-end system you want to get the data. You can add the `sap-system` parameter to the URL by providing it in the tile configuration. For more information, see Configuring Multiple Back-End Systems Using the sap-system Parameter [page 326].

1.3.4.9.3.2 Configuring Other Apps from Multiple Back-End Systems

In a system landscape with multiple back-end systems, users may need to access several instances of the same Web Dynpro, SAP GUI or WebClient UI application or transaction originating from different back-end systems.

The following figure shows a system landscape where a user accesses an application or a transaction that may originate from either of two back-end systems.
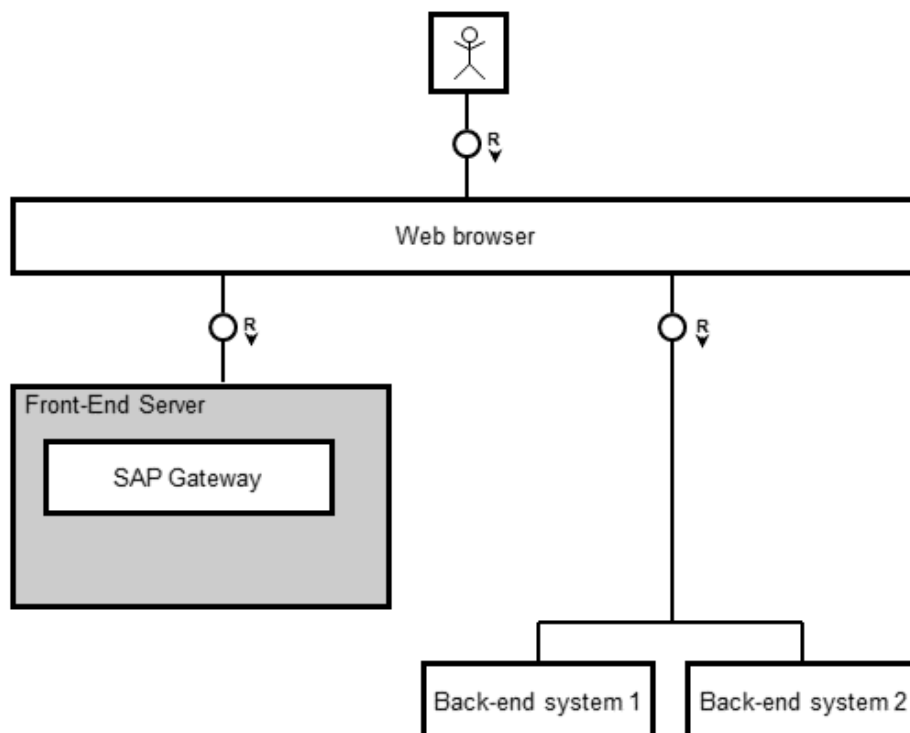
You can provide the SAP system alias at either of the following locations:

- In a target mapping (see Configuring Target Mappings [page 217]; this doesn't work for WebClient UI apps, use the launchpad app manager [page 132] instead)

SAP Fiori Launchpad
SAP Fiori Launchpad

- In transaction `LPD_CUST` (see ; this is not available for for WebClient UI apps)
- In the `sap-system` parameter in the URL

  If you provide the `sap-system` parameter in the URL, this overrides any SAP system settings in the target mapping or in transaction `LPD_CUST`. You can add the `sap-system` parameter to the URL by providing it in the tile configuration (see)

> **i Note**
>
> If you provide a system alias in transaction `LPD_CUST`, in the *System Alias* field in a target mapping, or as a `sap-system` parameter in the URL or in a target mapping, this system alias must match the name of a system alias maintained in `/UI2/VC_ALIASMAP` or in transaction `SM59`, following the naming conventions for suffixes as described in .
>
> If you add the `sap-system` parameter to the URL or target mapping, and the transaction navigates to the application, you must also maintain a corresponding SAP Gateway alias for all OData Services used by these apps.
>
> As a general guideline it is strongly recommended to maintain SAP Gateway and `/UI2/VC_ALIASMAP` aliases consistently when operating both SAPUI5 and other applications in a multi-backend scenario.

1.3.4.9.3.3 Example for Configuring Multiple Back-End Systems

SAP Fiori Launchpad

Process Receivables
Europe
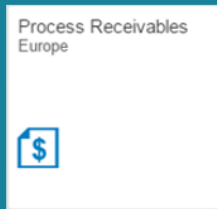
How to configure two tiles referring to two instances of the same application that get their data from two different back-end systems

Process Receivables
Asia-Pacific

Tile Configuration

Create two tiles with the same intent.

Provide the system aliases of the two back-end systems.

Semantic Object:	Customer
Action:	processReceivables
Parameters:	sap-system=FINEUR

Semantic Object:	Customer
Action:	processReceivables
Parameters:	sap-system=FINAPAC

Target Mapping

Create two target mappings for the same intent, referring to the same application.

Intent

| Semantic Object: | Customer |
| Action: | processReceivables |

General

| Information: | |
| Device Types: | ☑ Desktop ☑ |

| Parameters: | Name | Mandatory | Value |
| | sap-system | ☑ | FINEUR |

Intent

| Semantic Object: | Customer |
| Action: | processReceivables |

General

| Information: | |
| Device Types: | ☑ Desktop ☑ |

| Parameters: | Name | Mandatory | Value |
| | sap-system | ☑ | FINAPAC |

Create a filter for the respective target system.

SAP Gateway

In OData requests posted by the application, SAP Gateway adds the system alias in a segment parameter

/sap/opu/finsrv;o=FINEUR/...

/sap/opu/finsrv;o=FINAPAC/...

SAP Gateway

Map the system aliases to RFC destinations

Change View "Manage SAP System Aliases": Overview

New Entries

Manage SAP System Aliases

SAP System Alias	Description	Local GW	For Local App	RFC Destination	Software Version	System ID	Client
FINEUR	FIN Europe	☑	☐	BE1CLNT500	DEFAULT	BE1	500

1.3.4.9.3.4 Configuring Multiple Back-End Systems Using the sap-system Parameter

Procedure

1. In the launchpad designer, create two or more tiles.

   In the tile configuration, provide the parameter **sap-system=<SYSTEM_ALIAS>** in the *Parameters* field.

   The URL fragment for navigating to the application will then look like this:

   `#SemanticObject-action?sap-system=<SYSTEM_ALIAS>`

2. In the launchpad designer, create target mappings for the semantic object and action of your tiles.

   - If you want to enable users to access your application on all systems, you only need to create one target mapping for all tiles created in step 1.
   - If you want to prevent navigation to a system for users who do not have any authorizations for this application on this system, we recommend to create a filter in the target mapping. In the *Parameters* table of the target mapping configuration, enter the following settings:

Field	Value
Name	**sap-system**
Mandatory	Active
Value	Enter the system alias of the system for which you want to use this target mapping.

3. Assign the catalogs containing the tiles and target mappings to roles.

4. In Customizing for SAP Gateway, make sure that the SAP system aliases are maintained and assigned to OData services.

   In *SAP Reference IMG* (transaction SPRO), select the following paths:

   ▶ *SAP NetWeaver* 〉 *Gateway* 〉 *OData Channel* 〉 *Configuration* 〉 *Connection Settings* 〉 *SAP NetWeaver Gateway to SAP System* 〉 *Manage SAP System Aliases* ◗

   ▶ *SAP NetWeaver* 〉 *Gateway* 〉 *OData Channel* 〉 *Administration* 〉 *General Settings* 〉 *Assign SAP System Aliases to OData Service* ◗

1.3.5 Setting Up Authorization Roles

Authorizations to run the launchpad and the related design-time tools and to access launchpad content are assigned to users based on roles.

Related Information

1.3.5.1 Configuring Roles with Launchpad Start Authorizations

Users need authorization roles to run the SAP Fiori launchpad (as an end user) and the design-time tools (as an administrator). As a role administrator, you assign the necessary authorization roles and adjust them according to your needs.

Prerequisites

Procedure

1. In *Role Maintenance* (transaction `PFCG`), copy the roles `SAP_FLP_ADMIN` and `SAP_FLP_USER` to your customer namespace.
2. Add additional authorization default entries in the copied roles for the *TADIR Service*. On the *Menu* tab, choose and select *Authorization Default*.
3. In the *Service* dialog that opens, proceed as follows and repeat these steps for each service:
   a. Select *TADIR Service* and specify the following values:
      - *Program ID:*R3TR
      - *Object Type:* IWSG
      - *Object Name:* Use the value help to select the correct object name. The value help lists the technical service names for all the objects that you activated in the customizing activity *Service Maintenance of SAP NetWeaver Gateway*.
   b. The external service names in SAP Gateway are as follows:

- ZSAP_FLP_ADMIN
  - ZINTEROP_0001
  - ZPAGE_BUILDER_PERS_0001
  - ZPAGE_BUILDER_CUST_0001
  - ZPAGE_BUILDER_CONF_0001
  - ZTRANSPORT_0001
- ZSAP_FLP_USER
  - ZINTEROP_0001
  - ZPAGE_BUILDER_PERS_0001

4. On the *Authorizations* tab, generate the authorization profiles. Choose *Change Authorization Data* and generate the authorization objects.

   To ensure that authorization profiles are generated correctly, use *Upgrade Tool for Profile Generator* (transaction SU25) to copy the default authorization values by SAP to your customer namespace.

5. Assign end users of the SAP Fiori launchpad to the user role and assign administrators of the SAP Fiori launchpad designer to the admin role.

Related Information

Activating SAP Gateway OData Services [page 23]
Assigning Users to Roles [page 339]
Configuring Roles for Catalogs, Spaces and Groups [page 328]

1.3.5.2 Configuring Roles for Catalogs, Spaces and Groups

Role administrators make business catalogs, pages and spaces available on the end user's entry page by assigning business catalogs, pages, and spaces to an authorization role to which users can be assigned. Users logging on to the launchpad see all assigned spaces and pages on their home page, and when users open the catalog section, they can access all tiles in the assigned catalogs.

Spaces and pages provide the predefined content a user sees when first accessing the home page. Catalogs define the set of all tiles that users can use to personalize the home page.

> **i Note**
>
> In the past, business catalogs were defined in groups. Groups have been deprecated and replaced by spaces and pages. See Spaces and Pages [page 419].

Related Information

Assign Business Catalogs to Roles [page 330]
Configure Roles for Groups [page 338]

1.3.5.2.1 Configure Roles for Catalogs

Role administrators make tile catalogs available to end users by configuring roles for tile catalogs and assigning the roles to users.

Catalogs define the set of tiles that users can use to personalize the launchpad home page. For more information on creating and configuring catalogs, see Setting Up Navigation [page 127].

You use authorization roles to provide end users with access to the applications referenced by tiles in catalogs. If a user is assigned to the role that is configured for the catalog, the user can browse through the catalog, choose tiles from this catalog, and add them to the launchpad home page.

To assign a catalog created in the launchpad designer (Fiori launchpad catalog) to a role, you create a new menu entry for the tile catalog in the PFCG role menu. The system automatically reads all target mappings that fit to the selected catalog and the current system and collects the applications the target mappings refer to.

The following types are taken into account when reading authorization default entries:

- SAP NetWeaver Gateway OData services used in SAPUI5 applications
- Web Dynpro ABAP applications
- Transactions

All relevant authorization defaults for applications are added in the role menu as depending nodes of the catalog. Using this information, you maintain authorization data and generate a profile for the role. The authorization default values will be added as standard authorizations as soon as the merging procedure is performed. Standard authorizations contain empty fields which must be completed in order to have a complete set of authorizations.

The target applications referenced by a catalog can reside on a front-end or a back-end server. You have the following options to assign a catalog to a role and determine authorization objects:

- Local scenario:
  A specific catalog residing on the front-end server contains applications that reside on the same server. In this case, you can locally assign tiles and read the required authorizations.
- Remote scenario:
  A specific catalog residing on the front-end server contains applications residing on a different server (another front-end server or a back-end server). In this case, you remotely assign tiles and read the required authorizations.

Update authorization defaults:

When applications are added to or removed from a catalog, the system can retrieve this delta and you can, if necessary, update the authorizations for the role to which the catalog is assigned.

> i Note
>
> If you assign HANA Catalogs, Fiori Catalog Pages, or Remote Catalogs, you assign catalogs to roles without reading authorization default entries.
>
> For more information, see Assign Tile Catalogs to Roles Without Reading Authorizations [page 335].

Related Information

1.3.5.2.1.1 Assign Business Catalogs to Roles

To make business catalogs available to end users, you create a new menu entry for the business catalog in the PFCG role menu and determine which authorizations are required to run the applications referenced by the catalog.

Prerequisites

You have installed one of the following:

- SAP_BASIS 7.40 SP07 with SAP Note 2031538📝
- SAP_BASIS 7.40 SP11
- SAP_BASIS 7.50 or a higher SAP_BASIS release version

Context

You can assign business catalogs to a role to determine which content the launchpad user can access at runtime. When you add a business catalog to a certain role, you have the option to select *Include Applications* to ensure that the authorization defaults of the assigned applications are included as well.

Authorization defaults for the following application types are retrieved, given they are located in the system used for role maintenance:

- SAPUI5 applications with the following object types:

Object Type	Service Type	Service Type in Data Source Descriptor	Example
IWSG, IWSV	OData v.2	OData	/sap/opu/odata/sap/service_name;v=3/
G4BA	OData v.4	OData	/sap/opu/odata4/nsp/service_group/repository/nsp2/service/0001/
HTTP	HTTP Service	http	/sap/bc/http/sap/http_service/

Object Type	Service Type	Service Type in Data Source Descriptor	Example
SAPC	ABAP Push Channel (APC)	WebSocket	/sap/bc/apc/sap/apc_service/
INA1	Information Access (InA)	INA	/sap/bc/ina/ina1/sap/ext_ina_service

- SAP GUI transactions (object type TRAN)
- Web Dynpro ABAP applications with the following object types:
  - WDCA
  - WDYA

You can manage which services or applications you want to have displayed in *Role Maintenance* (transaction PFCG). You can do that by either enabling or disabling the following settings:

- PFCG_INCLUDE_IWSG
- PFCG_INCLUDE_IWSV
- PFCG_INCLUDE_G4BA
- PFCG_INCLUDE_HTTP
- PFCG_INCLUDE_SAPC
- PFCG_INCLUDE_INA1
- PFCG_INCLUDE_TRAN_GUI
- PFCG_INCLUDE_TRAN_UI5
- PFCG_INCLUDE_WDCA
- PFCG_INCLUDE_WDYA
- PFCG_INCLUDE_TRAN_TILE_ONLY
- PFCG_INCLUDE_TRAN_URL
- PFCG_INCLUDE_TRAN_URLT
- PFCG_INCLUDE_TRAN_WCF
- PFCG_INCLUDE_TRAN_WDA

For more information on the settings, see Launchpad Configuration Parameters [page 29].

For more information on the data sources or the descriptor for applications, see Descriptor for Applications, Components, and Libraries.

> **i Note**
>
> The system can only retrieve the authorization defaults automatically if the corresponding SAPUI5 app has an application descriptor. For more information on the descriptor for applications, see Descriptor for Applications, Components, and Libraries.
>
> If the application does not have an application descriptor, you manually enter the relevant data services as authorization default in a role in transaction PFCG.

Procedure

1. In *Role Maintenance* (transaction PFCG), enter the name of an existing role or create a new one.

   All users with this role are granted access to the catalog assigned to the role.

2. Choose the *Menu* tab.

3. Choose ![icon] and select ▌ *SAP Fiori Launchpad* ❯ *Launchpad Catalog* ▌.

4. In the dialog box that appears, select *Fiori Launchpad Catalogs* as catalog provider.

5. Depending on your scenario, proceed as follows:

   * To maintain roles on the front-end server, select the *Local Front-End Server* radio button.
   * To maintain roles on the back-end system:
     * Select the *Remote Front-End Server* radio button.
     * Provide one of the following to establish a connection to the front-end server:
       * RFC variable
       * RFC destination
     * Provide your logon data for the front-end system.

6. Enter the ID of a business catalog.

   You can use the value help for this.

7. Make sure the *Include Applications* checkbox is marked. This way you ensure that the authorization defaults of the assigned applications are included as well.

   This checkbox is only displayed if you have selected *Local Front-End Server*. For *Remote Front-End Server* the function is automatically active.

8. Confirm your selection.

Results

The business catalog you assigned to the role is available in the role menu. In the *Role Menu* tree, authorization defaults for applications used in the catalog are listed as nodes under the catalog.

> **i Note**
>
> If an application used in the assigned catalog is not available in the ABAP system when an administrator determines authorization defaults, a message is written to the application log.
>
> For more information, see Logging and Tracing for Launchpad and Launchpad Designer [page 401].

In the *Details* view of the node, you can change and display the node.

You can view the content of a catalog in the respective design-time tool (SAP Fiori launchpad content manager or SAP Fiori launchpad app manager) by choosing *Execute* from the context menu of the business catalog in the role menu.

To make sure the catalogs linked to the role are visible for the end user, the *Hide Menu from NWBC* checkbox in the *Menu Options* dialog box should not be selected.

Next Steps

You use the default authorization entries to maintain authorization data and generate a profile for the role.

The following reports support mass maintenance operations:

- *Create Back-end Roles for SAP Fiori Launchpad* (`PRGN_CREATE_FIORI_BACKENDROLE`): You can use this report to create a back-end role for selected SAP Fiori front-end roles and to copy the relevant front-end role menu. If the back-end role already exists, the role menu of the back-end role is deleted and replaced by the role menu of the front-end role. For more information, see the report documentation.
- *Create Frontend Roles for SAP Fiori Launchpad* (`PRGN_CREATE_FIORI_FRONTENDROLE`): You can use this report to perform the mass maintenance of menu options from selected SAP Fiori business catalogs. The assignment of the SAP Fiori business catalogs to roles takes place by means of a table that you either create or upload manually or generate automatically by means of catalog selection. For more information, see the report documentation.

Related Information

1.3.5.2.1.2 Assign SAP Cloud Platform Catalogs to Roles

To automatically determine which authorizations are required to run back-end content from the SAP Fiori launchpad on SAP Cloud Platform, you create a new menu entry for the SAP Cloud Platform catalogs in the PFCG role menu.

Prerequisites

- You have admin access to the systems:
  - In the ABAP back-end system: You are assigned the admin role SAP_FLP_ADMIN. See Configuring Roles with Launchpad Start Authorizations [page 327].
  - In SAP Cloud Platform: You are assigned to the cockpit Administrator role. For more information, see Account Member Roles.
- The SAP Cloud Platform Portal Service is enabled. To verify, open the SAP Cloud Platform cockpit, and go to *Services*.
- A connection is established between the ABAP system and SAP Cloud Platform. For more information, see Establish connection between ABAP and CP.

Context

You want to allow end users to launch SAPUI5 applications (which use OData services), Web Dynpro ABAP applications, or SAP GUI transactions from the SAP Fiori launchpad running on SAP Cloud Platform.

To automatically determine which authorizations are required to run these applications, the *Role Maintenance* (transaction PFCG) in the ABAP back-end system, needs to retrieve the catalogs referencing these applications.

To enable the Role Maintenance in ABAP to retrieve the catalogs created in the launchpad configuration cockpit in SAP Cloud Platform, you establish a connection between ABAP and SAP Cloud Platform.

Procedure

1. In your ABAP back-end system, launch *Role Maintenance* (transaction PFCG).
2. Provide the role name and choose *Change*.

   If no role exists, create a role by choosing *Create Single Role*.

   All users with this role are granted access to the catalog assigned to the role.
3. Choose the *Menu* tab.
4. Choose and select *SAP Fiori Tile Catalog*.
5. In the dialog box that appears, select *Fiori Launchpad CP Catalogs* as catalog provider.
6. Enter the following:

   - RFC destination for CP token
   - RFC destination for CP catalog
   - ID of the CP catalog

   You can use the value help for this.
7. Confirm your selection.

Results

The tile catalog you assigned to the role is available in the role menu. In the Role Menu tree, authorization defaults for applications used in the catalog are listed as nodes under the catalog.

Next Steps

Use the default authorization entries to maintain authorization data and generate a profile for the role.

1.3.5.2.1.3 Assign Tile Catalogs to Roles Without Reading Authorizations

Role administrators assign tile catalogs to a user's authorization role.

Context

A specific catalog residing on the front-end server contains applications that reside on the same server. In this case, you can locally assign catalogs to an authorization role.

Procedure

1. In *Role Maintenance* (transaction **PFCG**), provide the role name and choose *Change*.

    If no role exists, create a role by choosing *Create Single Role*.

    All users with this role are granted access to the catalog assigned to the role.
2. Choose the *Menu* tab.
3. Choose ⬛ and select *SAP Fiori Tile Catalog*.
4. In the dialog box that appears, select one of the following catalog providers:
    a. *Fiori Launchpad Catalogs*

    Choose this catalog provider if you want to assign one of the catalogs you created in the launchpad designer.
    b. *Fiori Catalog Pages*

    Choose this catalog provider if you want to use catalogs that you created in the context of SAP Fiori launchpad, the predecessor of the current release version called SAP Fiori launchpad. These catalogs can only be displayed by the home page of SAP Fiori launchpad.
    c. *Remote Catalogs*

    Choose this catalog provider if you want to use tiles from a catalog on a remote server (in the context of SAP Fiori, a server that is not the front-end server).

    > i Note
    >
    > The catalog provider *HANA Catalogs* is deprecated. Please use the catalog provider *Remote Catalogs* instead.

5. Depending on the type of catalog provider you select, proceed as follows:
    a. If you select *Fiori Launchpad Catalogs*, proceed as follows:

        1. Select the *Local Front-End Server* radio button.
        2. Enter a catalog ID.
            The catalog ID is the name that you provided when creating the catalog in the launchpad designer. You can use the value help for this.

3. Unmark the *Include Applications* checkbox.
    b. If you select *Fiori Catalog Pages* or *Remote Catalogs*, enter a catalog ID.

Results

The tile catalog you assigned to the role is available in the role menu.

In the *Details* view of the node, you can change and display the node.

To make sure the catalogs linked to the role are visible for the end user, the *Hide Menu from NWBC* checkbox in the *Menu Options* dialog box should not be selected.

You can view the content of a *Fiori Launchpad Catalog* in the SAP Fiori launchpad designer by choosing *Execute* from the context menu of the tile catalog in the role menu.

Related Information

Configure Roles for Catalogs [page 329]
Accessing Remote Catalogs [page 193]

1.3.5.2.1.4 Update Authorization Defaults for Tile Catalogs

When the catalog content is changed, you need to update the authorizations to which the catalog is assigned.

Context

When applications (tiles/target mappings) are added to or removed from a catalog, the set of authorizations required for launchpad users to run the applications might change. The system can automatically retrieve the delta between applications in a launchpad catalog and applications in the PFCG role menu.

You use program *PRGN_COMPARE_ROLE_MENU* for a mass comparison of roles. To check and, if necessary, update the status of a single catalog, you directly open the *Details* dialog box for a catalog in the PFCG role menu. In the latter case, you skip step 1 to 6 in the following procedure.

Procedure

1. Launch the program *PRGN_COMPARE_ROLE_MENU*.
2. (Optional) You can restrict your search by providing a role name in the *Role* field.
3. In the *Node Type* field, select *SAP Fiori Tile Catalog*.

4. Choose a display mode:

   - Overview: Displays a status for the relevant role or roles indicating whether an update is required or not.
   - Detail: Displays a list of all the applications included in the catalogs that are assigned to the selected role. The *Comparison* column indicates whether an application was deleted, added or not changed.

5. Choose *Execute*.

6. Depending on the selected display mode, proceed as follows:

   - If you selected the Overview view, doubleclick on a role that has status *To be adjusted*.
   - If you selected the Details view, doubleclick on a table row which does not have status unchanged in the *Comparison* column.

   The *Role Maintenance* (transaction PFCG) is launched.

7. Make sure you are in edit mode.

8. In the *Role Menu* tree, select a tile catalog.

9. In the context menu, choose *Details*.

   The *Assign Tile Catalog* dialog box appears. The *Applications in Catalog* section displays the applications that are currently used in the catalog. The *Comparison* column indicates whether authorizations required for these applications need to be added or deleted from the role you selected.

10. To allow the suggested changes, choose *Continue*.

    The authorizations are updated in the PFCG role menu.

11. On the *Authorizations* tab, update the authorization data and regenerate profiles.

Related Information

Assign Business Catalogs to Roles [page 330]

1.3.5.2.2   Configure Roles for Spaces

Assign spaces to roles to make them available for users.

After you have copied a SAP-delivered space or created a new space with one or more pages, you need to assign this space to the appropriate business role. Only then, the pages will be available for the users with the business role. The assignment is done in the ABAP system.

1. In *Role Maintenance* (transaction PFCG), provide the role name and choose *Change*.
   If no role exists, create a role by choosing *Create Single Role*. All users with this role are granted access to the space assigned to the role.

2. Choose the *Menu* tab.

3. Choose ![icon] and select ⏵ *SAP Fiori Launchpad* ⏵ *Launchpad Space* ⏵.

4. Enter a space ID. You can use the value help for this.
   The space ID is the name that you provided when creating the space in the *Manage Launchpad Spaces* app.
   See Manage Launchpad Spaces [page 291].

5. Save your changes.

Related Information

Managing Launchpad Spaces and Pages [page 269]

1.3.5.2.3　Configure Roles for Groups

Role administrators assign tile groups to a user's authorization role.

Context

To assign a tile group to the user's role (end user and catalog administrator) in *Role Maintenance* (transaction PFCG), you add a node of type *Tile Group* to one of the user's roles. Proceed as follows:

Procedure

1. In *Role Maintenance* (transaction **PFCG**), provide the role name and choose *Change*.

   If no role exists, create a role by choosing *Create Single Role*. All users with this role are granted access to the tile group assigned to the role.
2. Choose the *Menu* tab.
3. Choose ▦ and select *SAP Fiori Tile Group*.
4. To assign a tile group to a role, enter a group ID.

   The group ID is the name that you provided when creating the tile group in the launchpad designer.

   You can use the value help for this.

Results

The tile group you assigned to the role is available in the role menu. You can view the content of the tile group in the SAP Fiori launchpad designer by choosing *Execute* from the context menu of the group in the role menu.

In the SAP Fiori launchpad, tile groups that are assigned to one of the user's roles are displayed on the user's home page. An end user can personalize the home page by adding or removing apps to pre-delivered groups or self-defined groups.

1.3.5.3 Assigning Users to Roles

A role administrator makes catalogs, spaces and groups available to users by assigning them to an authorization role.

As a role administrator, you assign users of the SAP Fiori launchpad (end-users) and of the design-time tools for the launchpad (administrators) to authorization roles in *User Maintenance* (transaction SU01) or in *Role Maintenance* (transaction PFCG).

Prerequisites

- Activate the services in SAP Gateway and call each service once before you assign users to roles.
- If you want to assign a reference role to a user, you need to make sure the role assignments of the reference user are also taken into account by the SAP Fiori launchpad. In table SSM_CUST, set parameter **REFUS_NWBC** to **YES**. See SAP Note 2540430 .

Administrator Role

The launchpad administrator must have the authorization role SAP_FLP_ADMIN assigned. With this role, the administrator can run the design-time tools for the launchpad (e.g. launchpad content manager).

User Role

The launchpad user must have the authorization role SAP_FLP_USER assigned. With this role, the user can run the SAP Fiori launchpad on the *Personalization* level and has authorization to execute the following OData services:

- /UI2/PAGE_BUILDER_PERS
- /UI2/INTEROP
- /UI2/LAUNCHPAD

For more information, see SAP Library for SAP NetWeaver SAP Help Portal at http://help.sap.com/ nw_platform. Select a release. Under *Application Help*, open *Function-Oriented View* and choose ▶ *Security* ▶ *Identity Management* ▶ *User and Role Administration of Application Server ABAP* ▶ *Administration of Users and Roles* ▶ *User Administration* ▶ *User Administration Functions* ▶ *Assigning Roles* ▶.

Related Information

Activating SAP Gateway OData Services [page 23]

1.3.6 Integrating the Launchpad and Launchpad Content with Other UI Clients

The SAP Fiori launchpad can be combined with other SAP UI clients to provide a single point of access to all SAP content.

Here is an overview of the possible types of integration:

Integration Scenario	Description
Launchpads on SAP BTP as central entry point with ABAP platform as content provider	Integrate launchpad content into the launchpads on SAP BTP. See Exposing Launchpad Content to SAP Business Technology Platform [page 340].
SAP Enterprise Portal as central entry point with ABAP platform as content provider	Integrate launchpad content into the launchpad on Portal. See Exposing Launchpad Content to SAP Enterprise Portal [page 350].
SAP Business Client as central entry point with ABAP platform as content provider	Run the launchpad embedded in the SAP Business Client or enhance the classical SAP Business Client navigation structure in PFCG by singular SAP Fiori apps or the launchpad as an entry page. See .
Launchpad on ABAP platform as central entry point with SAP GUI as content provider	Integrate Web Dynpro and SAP GUI applications retrieved from SAP Easy Access menu entries in SAP GUI into the launchpad. See Integrating Applications from SAP Easy Access Menu [page 312].
Launchpad on ABAP platform as central entry point with SAP BTP as content provider	Integrate apps built on SAP BTP into the launchpad on ABAP Platform to extend the capabilities of your SAP S/4HANA system. See Integrating SAP BTP Extension Apps (Beta) [page 351].

1.3.6.1 Exposing Launchpad Content to SAP Business Technology Platform

To integrate your SAP S/4HANA launchpad content into SAP Business Technology Platform (SAP BTP), you select the content to be exposed in the ABAP system. On SAP Business Technology Platform, you then configure the destinations for SAP S/4HANA as a content provider.

Services that can integrate SAP S/4HANA content are:

- `SAP Build Work Zone, standard edition`
- `SAP Build Work Zone, advanced edition`
- `SAP Cloud Portal service`

Overview

Integrating SAP S/4HANA apps into a launchpad on SAP Business Technology Platform, requires to perform various tasks using different tools.

The following table provides an overview of the end-to-end process, including the tools that are involved:

Tool	Task	More Information
ABAP System	In an embedded deployment, before exposing the content, the ABAP system administrator needs to set the value of the customizing parameter `EXPOSURE_SYSTEM_ALIASES_MODE` to `CLEAR`. For more information, see Launchpad Configuration Parameters [page 29]. In the content exposure tool, select and expose the roles you want to integrate in the launchpad on SAP Business Technology Platform.	Manage Launchpad Content for Exposure [page 343]
SAP BTP cockpit	In the SAP BTP cockpit, configure a design-time destination and one or more runtime destinations.	Configure Destinations
SAP Build Work Zone, standard edition	In the Provider Manager, define the content provider and add roles to your subaccount.	Manage Content Providers (On Premise) Add Specific Roles to Your Subaccount Manually
SAP Build Work Zone, standard edition	In the Site Settings editor, assign one or more roles to sites.	Assign Roles to a Site
SAP BTP cockpit	In the SAP BTP cockpit, under *Trust Configuration*, assign role collections to users.	Assigning Role Collections to Users or User Groups

For more information on the complete end-to-end configuration steps, see Federation of Remote Content Providers .

In addition, you can do the tutorial Enhance Your SAP Launchpad Site with Federated SAP S/4HANA Content in the SAP Tutorial Navigator.

Related Information

1.3.6.1.1 Checklist for Successful Content Exposure

Certain boundary conditions have to be met in order for the content exposure to be successful.

To ensure that the selected launchpad content can be exposed to the launchpads on SAP BTP, check the following:

- Make sure the prerequisites listed under Manage Launchpad Content for Exposure [page 343] are met.
- Make sure the role selection was saved at least once.
- Some features are not yet supported for content exposure. Please check the list of restrictions. See Restrictions - Federation. For example, custom tile types are not supported for content exposure and are therefore not exposed.
- Make sure the following boundary conditions are met for the entities you want to expose:
  - The entity IDs (i.e. the ID of roles, spaces, pages, groups, tiles and target mappings) must be compatible with the CDM schema:
    - Maximum length: 70 characters
    - The following characters for entity IDs are allowed: ([a-zA-Z0-9_]) as well as :@+~.#- /
    - The following characters for role IDs are allowed: ([a-zA-Z0-9_]) as well as :@+~.#- /&()=,<>?

      > **i Note**
      >
      > Unlike for role IDs in transaction PFCG, a semicolon [;] cannot be used for role IDs in the content exposure.

  - Tile/target mapping combinations referenced in groups or pages are only exposed when they are part of an exposed catalog or when there is another exposed catalog that contains the same original tiles and target mappings.
  - The business catalogs you want to expose should not have any critical content issues (e.g. reference lost). Check your content in the launchpad content manager (transaction /UI2/FLPCM_CUST):
    1. On the *Roles* tab, search for the role you want to expose.
    2. In the upper toolbar, choose *Show Tiles/Target Mappings*.
    3. Check the *Status* column of the lower table displaying the tiles/target mappings.
    4. To get further information on the issue, open the context menu for a selected row and choose *Status Details*.
    5. If the launchpad content manager displays any entries with status "Error", make sure you fix them before exposure. See Displaying Issues with Launchpad Content [page 261].

- Tile and target mapping should be grouped together in the same catalog when the tile is intended to launch the app referenced by the target mapping. If tile and target mapping reside in different catalogs, the navigation from the tile won't work.
  - Tiles should have inambiguous navigation targets within one catalog. If this is not the case, duplicate tiles are exposed to ensure no content is lost.
  - Matching tiles and target mappings need to have the same original language. If this is not the case, the tiles/target mappings are not exposed.
- To ensure that the exposed content is up-to-date, content change notification should be in place. See Manage Content Change Notifications [page 348].

Related Information

Exposing Launchpad Content to SAP Business Technology Platform [page 340]
Launchpad Content Exposure Troubleshooting [page 651]

1.3.6.1.2 Manage Launchpad Content for Exposure

Administrators can expose SAP Fiori launchpad content to SAP BTP by choosing specific roles. The related content, such as groups, catalogs, pages, or spaces can then be consumed on SAP BTP.

To display, edit, or expose roles you need to launch transaction `/UI2/CDM3_EXP_SCOPE` (*Launchpad Content Exposure to SAP BTP*).

Prerequisites

- The report uses the user-independent cache to load persisted content. Therefore, the synchronization of the user-independent cache should be scheduled periodically to ensure consistent data for catalogs and groups. See Scheduling the Synchronization of User-Independent Cache [page 28].
- In an embedded deployment, before exposing the content, the ABAP system administrator needs to set the value of the customizing parameter `EXPOSURE_SYSTEM_ALIASES_MODE` to `CLEAR`. For more information, see Launchpad Configuration Parameters [page 29].
- Make sure that the ICF service `/sap/bc/ui2/cdm3` is active. For more information, see SICF Services [page 24].
- The following authorization roles are required:
  - `SAP_FLP_ADMIN` - with this role, the administrator can expose content to the launchpads on SAP BTP.
  - `SAP_FLP_EXP_USER` - with this role, the end user can view the exposed content.
    For more information, see Assigning Users to Roles [page 339].

Further prerequisites related to SAP BTP can be found here: Federation of Remote Content Providers.

Boundary Conditions

Certain boundary conditions have to be met in order for the content exposure to be successful. To ensure that the selected launchpad content can be exposed, check the following: Checklist for Successful Content Exposure [page 342].

Restrictions

There are some restrictions related to content exposure to SAP BTP. See Restrictions - Federation.

Launchpad Settings

There are launchpad settings relevant to the exposure. The parameters all start with the prefix EXPOSURE_. To find out more about the settings, go to Launchpad Configuration Parameters [page 29] and search for the relevant parameter in the list.

Feature Scope

The report offers the following features:

Feature	Description
▶ Goto ▶ Maintain Areas ▶	Create an area to bundle your set of roles or role selections. You can define your own role set and have a better and manageable overview of roles you want to expose.
	See Maintain and View Areas for the Content Exposure [page 347].
▶ Goto ▶ View All Areas ▶	View the select options of all defined areas. You can filter by a specific area if you need to.
	See Maintain and View Areas for the Content Exposure [page 347].
Save Selected Roles	Save your defined selection first before exposing it to SAP BTP.
	You can preview your selected content and see if there are any content issues before you expose it.

Feature	Description
Preview Area Content	Preview only your defined area content.
	The roles and their related content are displayed, such as groups, catalogs, apps, URL templates, and more.
	Click a node in the tree to view the content in JSON format in the window on the right side. You also have the possibility to directly open the log and identify content issues before the exposure. This is helpful to fix potential issues before exposing your content.
	If you are satisfied with your selection, you can go back and save and expose your content. Note that all roles in all defined areas will be exposed.
Preview Content	Preview the content of all roles in all areas.
	The roles and their related content are displayed, such as groups, catalogs, apps, URL templates, and more.
	Click a node in the tree to view the content in JSON format in the window on the right side. You also have the possibility to directly open the log and identify content issues before the exposure. This is helpful to fix potential issues before exposing your content.
	If you are satisfied with your selection, you can go back and save and expose your content.
Expose Content	Expose your defined content to the file repository.
	i Note The role selection must be saved first before exposing. Otherwise you will expose the last **saved** role selection.
	i Note You can only expose single roles, however, related composite and derived roles will be exposed as well: • You need to expose all single roles that are included in a composite or derived role. • The name of the composite or derived role is passed to SAP BTP as `"relatedRoleAssignments"` and can then be assigned to the users. See Federation of Remote Content Providers.

Feature	Description
View Exposed Content	View a history of successfully exposed versions. In the window on the left side you see a list of the versions and their related information. When you click on a row, a list of the respective content is shown in the window on the right side. In the drop-down menu above, you can choose whether you want to see changes of the full version (*Full Version Content*) or the delta changes to the previous version (*Changes in Version*).
View Log	View the application logs of the exposed content to analyze the error and warning messages and get more information about them. For more information on troubleshooting, see Launchpad Content Exposure Troubleshooting [page 651].
Content Change Notifications	Enable automatic content change notifications. Administrators on SAP BTP are notified when new content is available. See Manage Content Change Notifications [page 348].

Procedure

Expose your SAP Fiori launchpad content:

1. Launch transaction `/UI2/CDM3_EXP_SCOPE`.
2. Choose which roles you want to provide to SAP BTP. You have the following options:
   - *All roles with launchpad content* - By default, all roles are exposed.
   - *Selected roles in areas (all roles in all areas will be exposed)* - Select specific roles or bundle several roles in a specific area. See Maintain and View Areas for the Content Exposure [page 347].

   > i Note
   >
   > If you choose to expose further roles after the exposure, you would need to expose **all** desired roles again.
   >
   > **Example**: You expose roles **A** and **B**. After the exposure, you want to expose role **C** as well. With the next exposure, you must select roles **A**, **B**, and **C**.

   > i Note
   >
   > Only single roles can be selected for exposure. However, derived roles and composite roles that are related to an exposed single role are exposed as well.

3. Save your defined roles by clicking on *Save Selected Roles*. The role selection must be saved first before exposing. Otherwise you will expose the **last** saved role selection.
4. Optional: Click *Preview Content* or *Preview Area Content* to see a preview of the selected content. This might help you to see if you have selected all your required content and if the content contains any errors or warnings that you might need to fix before exposing.

5. Check whether your content contains any errors by either clicking *View Log* or ▶ *Preview Content* ❯ *Display Log* ◀. Note that only content without errors can be exposed.
6. Expose your launchpad content:
    - Schedule a job to expose your content on a regular basis. See Schedule Content Exposure [page 348].
    - Click *Expose Content* to manually trigger the exposure.

Results

After you have exposed your launchpad content, the roles with the related content will be imported to the file repository and are ready to be retrieved by the administrator.

You can enable content change notifications, so that new content is automatically fetched by SAP BTP. See Manage Content Change Notifications [page 348].

If you don't enable automatic content change notifications, the launchpad administrator needs to perform additional steps before the content is visible on SAP BTP. See Federation of Remote Content Providers.

Related Information

SAP Tutorial Navigator: Enhance Your SAP Launchpad Site with Federated SAP S/4HANA Content

1.3.6.1.2.1 Maintain and View Areas for the Content Exposure

Create and expose areas in the content exposure.

Context

Areas allow you to select specific roles or bundle several roles. Administrators can define their own role set and thus have a better overview of existing roles.

A dedicated area preview allows the administrator to view the exposed role content of the selected area.

Maintain and View Areas

Create an area:

1. Launch transaction `/UI2/CDM3_EXP_SCOPE` *(Manage Launchpad Content for Exposure)*.
2. Choose ▶ *Goto* ❯ *Maintain Areas* ◀.

3. Change to the edit mode and choose *New Entries*.

4. Choose an ID and a name for your area and click *Save*.

Add role selections to your area:

1. On the main screen, choose the radio button *Select roles in areas (all roles in all areas will be exposed)*.

2. Choose your area with the help of the button 🔧(Select area).

3. In the *Roles* field, choose your roles or role sets that you want to expose. You can choose ➡ (Multiple selection) to define and view the values and ranges.

4. If you are satisfied with your selection, expose your content as described in Manage Launchpad Content for Exposure [page 343].
   All roles from **all** areas with launchpad content will be exposed. It does not matter which area is currently displayed in the selection of the report. The content of all areas is combined and exposed together. Excluded roles that are included in another area will still be exposed, as they are required in the context of the other area.

5. To clear your role selection for your area, choose 🗑 (Clear role selection) next to the *Roles* field.

6. To view the select options of all areas, you can get an overview in ▶ *Goto* ❯ *View All Areas* ❯. You can filter by a specific area if you need to.

1.3.6.1.3 Schedule Content Exposure

Administrators can create a job schedule for their content exposure to the launchpads on SAP BTP instead of manually triggering it in `/UI2/CDM3_EXP_SCOPE`.

Procedure

To schedule the exposure, you do the following steps:

1. Launch *Define Job* (transaction `SM36`).

2. Create a job schedule for `/UI2/CDM3_EXP_PUBLISH` by defining how often you want your content to be exposed. The content that will be exposed is the one you defined in transaction `/UI2/CDM3_EXP_SCOPE`. If you choose to define a periodic job, we recommend you to set the minimal interval to 1 hour.

More information on how to schedule a job can be found here: Creating a job schedule in SAP applications.

1.3.6.1.4 Manage Content Change Notifications

Manage content change notifications for launchpads on SAP BTP.

You can notify launchpads on SAP BTP when new, changed, or removed content is available. If you don't want to retrieve launchpad content manually, you can enable an automatic synchronization. New content is automatically fetched and visible in the launchpads on SAP BTP.

To enable content change notifications, the system uses callback URLs that are provided by each content provider on SAP BTP. These callback URLs are called when the content has changed.

How to Enable Content Change Notifications

Perform the following steps to set up your content change notification:

1. Set up an RFC destination to the host on SAP BTP side:

1. Launch the transaction SM59 (Configuration of RFC Connections) or in the report /UI2/CDM3_EXP_CCN select ▌ *Go to* ❭ *Maintain RFC dest.* ❭.
2. Click *Create*.
3. Enter a name in the *RFC Destination* field.
4. Choose **Type G (HTTP Connections to External Server)** in the *Connection Type* field.
5. Under the tab *Technical Settings*, enter the host of the callback URL on SAP BTP side. For more information, see Set Up Automatic Updates for ABAP-Based Content Providers.

> **i** Note
>
> Don't use the *Check Connection* test as it can't be used for content change notifications.

6. Under *Logon & Security* make sure that *SSL* is set to active.

2. Register your RFC destination

1. Launch the transaction /UI2/CDM3_EXP_CCNREG or launch the report /UI2/CDM3_EXP_CCN and select ▌ *Go to* ❭ *Register Targets* ❭.
2. Click *New Entries* to enter your previously created RFC destination.
3. Enter your previously created RFC destination under **RFC Destination**.
4. Enter a name for the target system under **Target Identifier**.
5. Save your changes.

After you have registered your callback URL, the table of the report /UI2/CDM3_EXP_CCN is empty at first. The callback URL that you have just set up will be displayed after SAP BTP fetches the content again.

> **i** Note
>
> If the registered callback URL is not displayed in the table, try to fetch the content manually and check if you have executed the steps above correctly.

It is possible to set up **automatic updates** so that the content of the provider is updated automatically every time there are changes to the exposed content. Notification is only sent when there is a new exposure with changes, so to have this fully automated, customers would need to schedule regular exposure runs for the report /UI2/CDM3_EXP_PUBLISH which would then trigger the notification about the change to be sent and so the content channel to be updated. See Schedule Content Exposure [page 348].

Managing Your Content Change Notifications

Once you have enabled the content change notification, you can manage it in the exposure tool :

1. Launch the transaction `/UI2/CDM3_EXP_SCOPE`.
2. Click *Content Change Notifications* to open the report `/UI2/CDM3_EXP_CCN`.

A table is displayed with a list of the enabled content consumers:

Column	Description
Callback Status	Displays the status of the last callback: • ✔ (Ok): The automatic content synchronization is working correctly and will be called when new content is available. • ✘ (Incomplete): The automatic content synchronization is not working properly. A manual synchronization of the content provider on SAP BTP might be required.
Callback Host	Displays the callback host that was entered when creating the RFC destination.
Consumer ID	Displays the ID of the consumer. This can be either the provider ID that has been set in the destination on SAP BTP or a hash value.
Last HTTP Code	Displays the HTTP return code of the last call.
Last Message	Displays the message of the last call.
Retry Count	Displays the number of retry attempts of a failed callback.
Changed On	Displays the date of the last call.
Changed At	Displays the time of the last call.

You can send content change notifications manually by choosing *Send*.

To disable the content change notification again, select the callback URL and choose *Delete*.

1.3.6.2 Exposing Launchpad Content to SAP Enterprise Portal

You can integrate launchpad catalogs and groups into the SAP Fiori launchpad on Portal.

It is possible to restrict access to specific SAP Fiori catalogs and SAP Fiori groups to ensure that only those tiles for which the end user has access rights are displayed on the home page.

For more information on how to enable access to launchpad content for SAP Fiori launchpad on Portal, see Remote Content in the Portal documentation.

Related Information

1.3.6.3 Integrating SAP BTP Extension Apps (Beta)

Integrate SAPUI5 Fiori apps built on SAP BTP into the launchpad on ABAP platform to extend the scope provided by SAP S/4HANA.

> i Note
>
> This feature is currently available as beta version. It is not part of the officially delivered scope that SAP is going to guarantee for future releases – means it may be changed by SAP at any time for any reason without notice. This feature is NOT FOR PRODUCTION USE. You may not demonstrate, test, examine, evaluate or otherwise use this feature in a live operating environment or with data that has not been sufficiently backed up.
>
> This feature is only supported for customers using SAP Cloud Portal service.
>
> For more information on the current limitations, see SAP Note 2974154 .

Overview

To integrate SAP BTP extension apps into a launchpad on the ABAP platform, you need to develop a launchpad module to expose these apps.

Custom applications that are developed and deployed in SAP BTP are blocked when running in an iFrame in an on-premise launchpad because the on-premise domain is different. To integrate these apps into an on-premise launchpad, you need to configure the list of on-premise domains in which they're allowed to run. You configure these domains in the Portal site.

To finish the setup, you need to perform configuration steps in the ABAP system.

The app type **SAPUI5 Fiori App on SAP BTP (Beta)** available in the launchpad app manager allows you to open apps running on SAP BTP in-place in the launchpad on ABAP platform. This means that the app is opened in the same browser window or tab as the launchpad - hence not disrupting the user experience.

End-to-End Process

Integrating SAP BTP extension apps into a launchpad on the ABAP platform, requires to perform various tasks by different personas using different tools.

The following table provides an overview of the end-to-end process, including the personas and tools that are involved:

Persona	Tool/System	Task	More Information
Developer	SAP Web IDE/SAP Business Application Studio	Develop extension apps using the launchpad module and deploy them to SAP BTP.	Developing a Launchpad Module
Administrator	SAP BTP	Configure single sign-on by configuring each system to trust an identity provider.	Configuring Single Sign-On for Extension Apps Embedded in SAP Fiori Launchpad
Administrator	SAP BTP	Enable iFraming by configuring the app running on SAP BTP to trust the domain of the launchpad on ABAP platform or by ensuring that both run under the same domain.	IFrames and Domain Security
Administrator	SAP BTP cockpit	Assign a key user to a role collection that contains the *Domain_Access_Manager* role template.	Assign a Key User to the Domain_Access_Manager Role Template
Key User	SAP Cloud Portal (Runtime)	Configure the allowed on-premise domain(s) in the *Domain Access Manager*.	Configure the Allowed On-Premise Domain(s) in the Domain Access Manager
Administrator	SAP BTP	Configure principal propagation. This step is only required if the extension app uses OData services to get data from a back-end system.	Configuring Principal Propagation
Administrator	Prerequisite for ABAP system configuration	Obtain the URL that launches the cloud application, to get its domain and application ID. You will need this information when configuring the ABAP system.	Configure the Integration of SAP BTP Extension Apps in the ABAP System [page 353]
Administrator	ABAP System	Configure the ABAP system by establishing connections and setting up the launchpad content.	Configure the Integration of SAP BTP Extension Apps in the ABAP System [page 353]

Related Information

Integrating SAP Business Technology Platform Apps into the ABAP Platform

1.3.6.3.1 Configure the Integration of SAP BTP Extension Apps in the ABAP System

To finish the setup required to integrate SAPUI5 Fiori apps built on SAP BTP into the launchpad on ABAP platform, you need to perform several configuration steps in the ABAP system.

Prerequisites

You need the domain and application ID of the SAP BTP app you want to integrate. You can get them out of the URL that launches the SAP BTP app directly.

The URL has the following format: `https://<domain>:<port>/<path>?<parameters>#<intent>?sap-ui-app-id-hint=<app_id>`

Context

For an overview of the end-to-end process, see Integrating SAP BTP Extension Apps (Beta) [page 351].

The following graphic provides an overview of the configuration in the ABAP system:
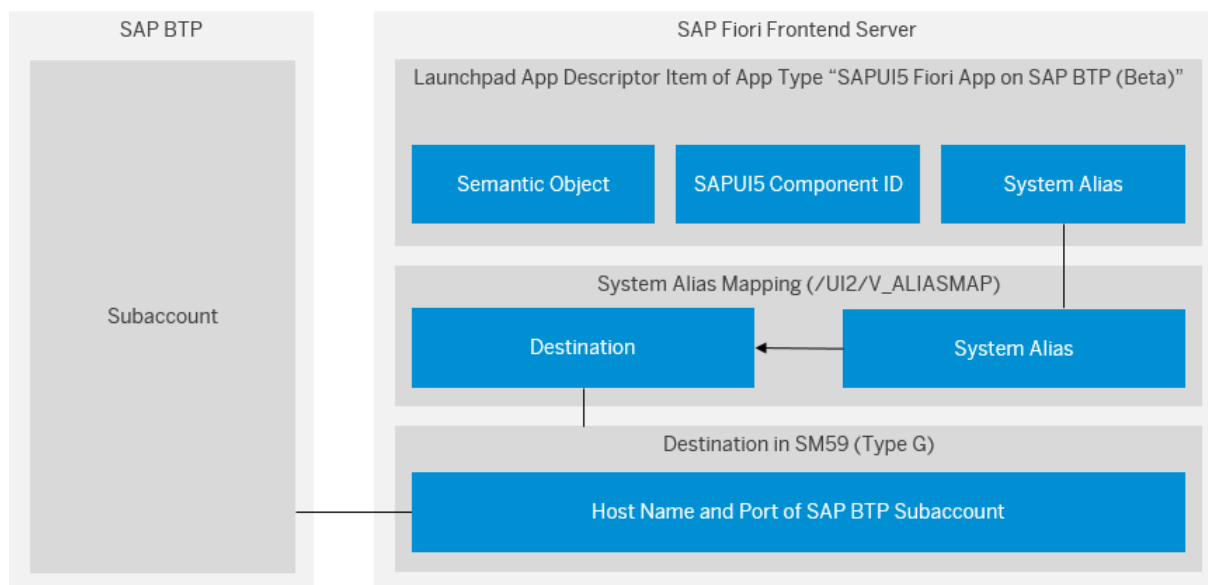
Procedure

1. Establish a connection between the ABAP platform and SAP BTP.

   1. In *Configuration of RFC Connections* (transaction SM59), create a destination of type G (*HTTP Connections to External Server*).

The destination is used to store technical connection data. These data will be used by the client (the launchpad user's browser) to connect to the SAP BTP subaccount.

1. In Customizing, choose ▮▶ *SAP NetWeaver* ❯ *UI Technologies* ❯ *SAP Fiori* ❯ *Initial Setup* ❯ *Customer System Aliases* ▮ and launch the IMG activity *Manage RFC Destinations* or directly launch transaction SM59.

2. Select the *HTTP Connections to External Server* node (type G) and choose *Create*.

3. In the *RFC Destination* field, enter a name ending with **_HTTPS**.

4. On the *Technical Settings* tab, enter the following:

    - *Host* field: Enter the domain of your SAP BTP subaccount. It ends with **cfapps.sap.hana.ondemand.com**.

    - *Port* field: As long as the default port for https (443) is used, you do not need to enter anything here.

5. On the *Logon & Security* tab, under *Security Options*, set *SSL* to *Active*.

> **i Note**
>
> The connection test in transaction SM59 only checks the server-to-server connection from the AS ABAP to the entered host name. This might fail due to the network configuration.
>
> During the runtime of the SAP Fiori launchpad, the connection data will only be used for client-to-server connection (user's browser to the entered host name). Therefore, the connection test in SM59 is not important here.

2. Map the connections.
   Typically the destination has a rather technical ID. For having a more business-oriented ID, you can define a system alias. See Map System Aliases to RFC Connections [page 17].

2. Maintain the catalogs, pages/spaces and roles required to launch the app in the launchpad.

   1. In the launchpad app manager, add a launchpad app descriptor item with the app type **SAPUI5 Fiori App on SAP BTP (Beta)** to a new or existing technical catalog.

      1. Maintain at least the following fields:

         - *Semantic Object*

         - *Action*

         - *SAPUI5 Component ID*: Enter the app ID taken out of your URL mentioned above.

         - *System Alias*: You can either enter the system alias created in the system alias mapping (see step 2 under *Establish Connections*) or the destination name created in SM59 (see step 1 under *Establish Connections*).

         See Maintaining Launchpad App Descriptor Items [page 148].

      2. Add a static tile to the launchpad app descriptor item.
         See Static Tiles [page 157].

   2. To finish the setup, maintain pages/spaces and roles following the recommendations under Advanced Scenario - Adapt SAP Template Content and Add Your Own Content [page 123].

Related Information

Integrating SAP BTP Extension Apps (Beta) [page 351]
Integrating SAP Business Technology Platform Apps into the ABAP Platform - SAP BTP Guide

1.3.7 Operations

This section provides a starting point for administrators in procedures for the optimal operation and use of the launchpad. It also includes an overview of performance concepts and troubleshooting information.

Related Information

1.3.7.1   Data Administration

This section provides a starting point for managing SAP Fiori launchpad and maintaining and running it optimally.

Related Information

1.3.7.1.1   Delete Personalization Data

You can delete personalization data for one, multiple or all users in your system.

Prerequisites

If you want to delete authorization data for other users, you need authorization for object `/UI2/PERS`. See .

Procedure

1. In the Implementation Guide (IMG), choose ▌ *SAP NetWeaver* ❯ *UI Technologies* ❯ *SAP Fiori* ❯ *Data Administration* ❯ *Delete Personalization Data* ❯ or directly start transaction `/UI2/FLP_DEL_PERS`.

2. Under *Scope Selection*, select which personalization data should be deleted:

Option	Description
Classic Home Page Personalization	Deletes the classic home page personalization and user-specific caching data for the page builder and start-up service. The following options are available: • *Complete*: Deletes the entire home page personalization. • *Selected Groups*: Deletes only the personalization of the selected groups. See Personalizing the Home Page [page 501].
Page Personalization	Deletes the page personalization. The following options are available: • *Sections*: Deletes the personalization of sections in pages. See Managing Launchpad Spaces and Pages [page 269] and Setting Up Launchpad Layout and Structure [page 268].
User Settings	Deletes user settings (for example, for theme, content spacing and user activities). See Managing Your Settings [page 509].
App-Specific Personalization	Deletes personalization data saved by the SAP Fiori apps (for example, filter criteria for lists).

3. Under *User Selection*, select one or multiple users. To delete personalization data for all users in the system, enter an asterisk (*).

4. (Optional) Mark the *Test Mode* checkbox for a test run in which no data will be deleted.

5. Choose *Execute*.

6. The result screen displays which users were affected and how many entries were deleted from each selected repository.

1.3.7.1.2 Migrate Personalization Data

You can transport home page personalization data and application personalization data to another system using the SAP transport tools. A report allows you to collect the personalization data for a specific user, a set of users, or all users, and writes these personalization data to transport requests.

Prerequisites

You need to create a workbench request and a customizing request in the client from where you want to migrate personalization data. For this reason, the system needs to allow the creation of workbench requests as well as customizing requests in the same client.

Context

As personalization data is client-specific, make sure to import it to the desired client in the target system. Also, make sure to disable client cascading in the transport control. For more information, see the documentation on the SAP transport tools.

> ! Restriction
>
> App variants cannot be migrated using the procedure described below.

Procedure

1. In transaction SA38, execute the program /UIF/MIGRATE_FES_PERSO.
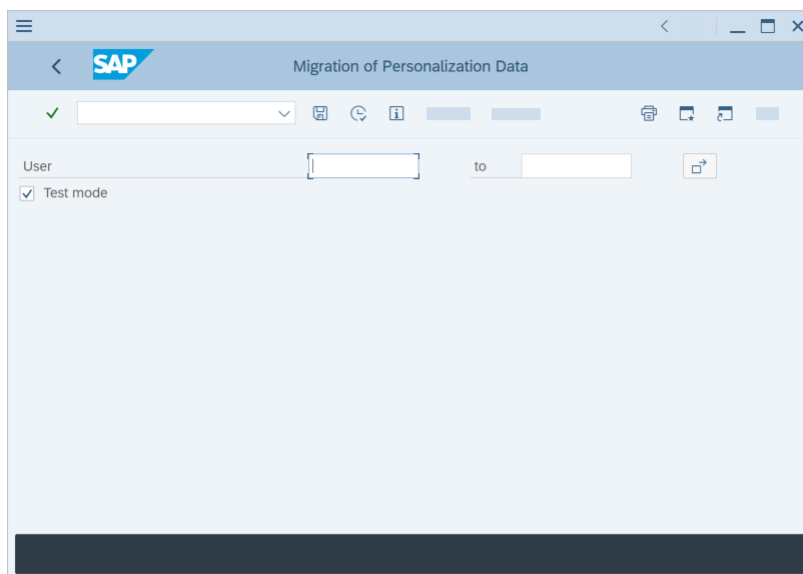
2. Specify the users whose personalization data you want to migrate.

    You can specify a single user name, a list of user names, or patterns for user names. If you specify something, only the user-specific personalization for the specified users will be collected.

    If you want to collect user-specific personalization data for all users, leave the *User* field empty.

3. Make sure that *Test mode* is active, and execute the program (F8).

    In test mode, the program does not write any records into transport requests. It just displays the number of transport entries that can be recorded into a workbench request, as well as the number of transport entries that can be recorded into a customizing request.
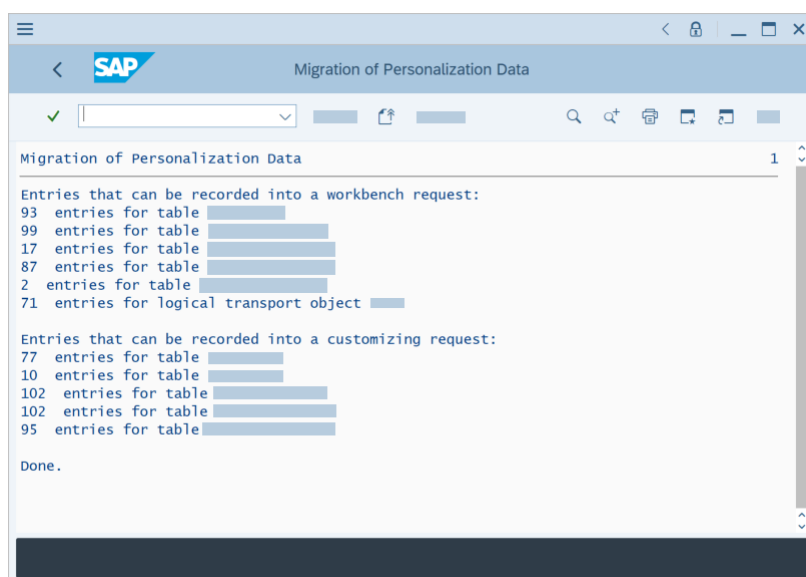
4. If there are transport entries for workbench requests, create a workbench request in a separate session.

5. If there are transport entries for customizing requests, also create a customizing request.

6. In the session where you have run the program /UIF/MIGRATE_FES_PERSO, navigate back to the selection screen (F3), and switch off test mode.

    Two additional input fields are displayed:

    *   *Workbench request*
    *   *Customizing request*
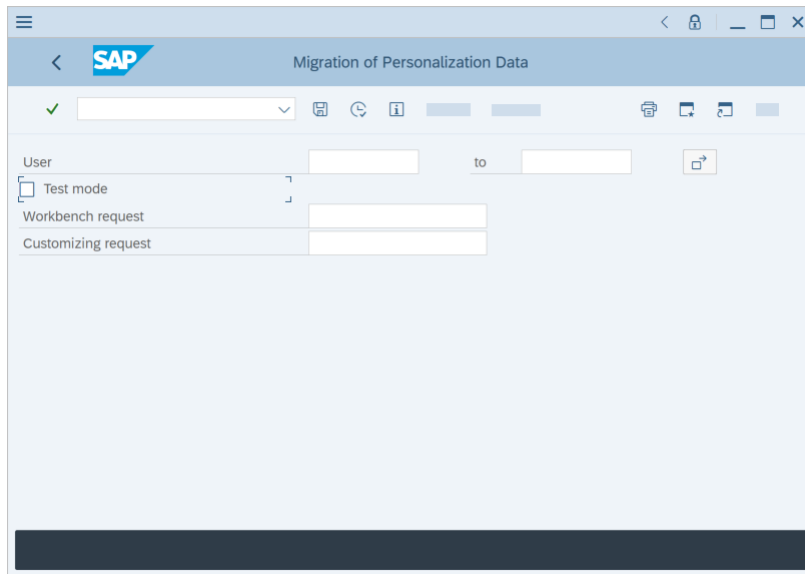
7. If there are transport entries for workbench requests, enter the ID of the workbench request that you have created into the corresponding field.

   Input help is available - press the F4 key.

8. If there are transport entries for customizing requests, enter the ID of the customizing request that you have created into the corresponding field.

9. Execute the program again (F8).

   The transport entries are now written to the transport requests that you specified.

10. Use the standard transport tools (for example transaction SE09) to release the transports.

   Some SAP Fiori launchpad user settings, like the personalization of the theme, may already exist in the target system. To overwrite these settings in the target system, select the option *Overwrite originals* in the transport options. If this option is deselected, old settings that already exist in the target system will be kept.

Related Information

SAP Note 2789848

1.3.7.1.3   Cleanup of Expired Personalization

You can run a background job to clean up expired personalization.

Personalization containers have a validity. Once a container has expired, you can no longer use it. We recommend you delete containers periodically by running a background job. A Customizing activity is available that allows you to create a background job to clean up user-specific data periodically after its validity has expired.

For more information, see Customizing for SAP NetWeaver, and choose ▶ *UI Technologies* > *SAP Fiori* > *Data Administration* > *Create Cleanup Jobs* ◀.

1.3.7.1.4 Cleanup of Expired Application State

You can schedule a cleanup job to periodically clean up application state records that have not been accessed for a long time.

Prerequisites

You are assigned to authorization role `SAP_FLP_ADMIN`. See Configuring Roles with Launchpad Start Authorizations [page 327].

Context

The SAP Fiori launchpad uses application states to shorten long URLs and to enable users to navigate back in the browser history.

An application state is described by a list of parameters that define a specific point in an application to which a user has navigated. For example, these parameters may include filter values or application URL parameters.

The list of parameters defining an application state is stored on the front-end server, together with a unique key. To address an application state, the launchpad includes this unique key in the URL that is displayed in the browser's address bar. This makes it possible to have short URLs that reference huge numbers of parameters containing huge numbers of characters.

If a link containing an application state is shared or bookmarked, it can only be restored if the application state is still available on the server.

To avoid that too much storage space is consumed, you should regularly clean up application states using the program *Cleanup Layered Repository Files* (transaction `/UIF/CLEAN_LREP`). Since the last access and the total number of accesses are monitored, you can use these criteria to clean up application states which have never been restored or which have not been restored for a long time, for example within the last year.

> i Note
>
> We recommend that you create a background job to run the cleanup program daily. For more information, see Customizing for SAP NetWeaver, and choose ▶ *UI Technologies* > *SAP Fiori* > *Data Administration* > *Create Cleanup Jobs* ◀.

Procedure

1. Start *Cleanup Layered Repository Files* (transaction `/UIF/CLEAN_LREP`) in the ABAP system.
2. Enter the following data:

Option	Description
Maximum number of times read	You can select the maximum number of read accesses that needs to be met for the application state to be deleted. Default value: **1**
Minimum number of days not accessed	You can select the minimum number of days since the last access that needs to be met for the application state to be deleted. Default value: **5**

The default selection cleans up records which have been accessed at most once, but have not been accessed within the last 5 days.

3. Deselect the *Test mode* checkbox.
4. Choose *Execute*.

Results

A list with the application state records that were deleted is shown.

A summary can be found in the *Application Log: Display Logs* (transaction `SLG1`) using object `/UI2/BE` (*UI2 Backend Enablement: Application log*).

1.3.7.1.5 Consistency Checks for the Layered Repository

SAPUI5 flexibility services use a consistent layering concept to store UI changes as semantic delta information.

To improve performance, all changes done in the `VENDOR`, `CUSTOMER_BASE` and `CUSTOMER` layers are consolidated in a so-called `LOAD` layer. This speeds up the loading of apps at runtime, as the effects of the changes on different layers have been pre-calculated.

The report `/UIF/CHECK_LOAD_4_CONS_BG` checks the consistency of the `LOAD` layer and triggers the regeneration of inconsistent entries. This report is executed by different batch jobs.

By default, the batch job `/UIF/LREP_LOAD_CONSISTENCY_CHECK` is created for the user who first accesses the layered repository after the shared-memory instance has expired or has been invalidated.

If you want to have these jobs created for a specific user, you can set a user by executing report `/UIF/SET_LOAD_CHECK_BTC_USER`. You can only set a user who has the authorization to release jobs and to modify the jobs of other users. For more information, see SAP note 1623250 .

As the job `/UIF/LREP_LOAD_CONSISTENCY_CHECK` is client-dependent, you need to set a user for the batch job in each client separately.

If you no longer want to have a specific user for the creation of the batch job `/UIF/LREP_LOAD_CONSISTENCY_CHECK` you can delete this user assignment by executing report `/UIF/DEL_LOAD_CHECK_BTC_USER`.

For more detailed information, see SAP note 2524948 .

1.3.7.2 Performance

The performance and stability of SAP Fiori launchpad and SAP Fiori apps depends on cache management.

An OData service call can request resources that are so large it can take many seconds or even minutes to return them to the consumer. Moreover, scrolling to the next page causes the data to be read from the database again before loading that page. Users do not want to wait a long time for the system to respond. In such a case it makes sense to use a caching mechanism to store requested resources. We differentiate between two cache types:

- Server-Side Cache [page 362]
- Client-Side Cache [page 364]

Related Information

1.3.7.2.1 Server-Side Cache

The server-side cache stores resources requested by the client on the server. In the context of SAP Fiori launchpad, this is the front-end server.

UI2 Cache

UI2 caches are relevant for all services with the prefix /UI2/ (for example, /UI2/PAGE_BUILDER, /UI2/INTEROP).

Cache Validity

The resources remain in the cache until an event in the system occurs, upon which they are invalidated. Typical events are:

- Change to a Web Dynpro configuration: This affects SAP Fiori launchpad artifacts such as groups, tiles, and catalogs.
- Change to an authorization role
- Change in the navigation repository (transaction `LPD_CUST`)
- Change to a SAPUI5 app

Manual invalidation of UI2 caches should not be required. See SAP Note 2362875.

Related Information

1.3.7.2.1.1 Analyzing Cache Statistics

Analyze the performance of OData services, and quickly find out why the UI2 cache was invalidated, with the cache statistics utility (transaction `/UI2/CSU`).

Cache Quality Statistics

Cache quality is an indicator of how often data is read from the UI2 cache. Please note that the hit rate of the browser cache is not taken into account. Therefore, a low cache quality indicated by the cache statistics utility does not necessarily indicate performance issues related to UI2 OData services.

The *Overview* section of the cache statistics utility displays the overall cache quality for the whole client. In the *Cache Quality* tab, you can also calculate the cache quality for a specific service by selecting the service and choosing *Cache Quality* in the toolbar.

Cache Invalidation Statistics

There are different scenarios in which the UI2 cache is invalidated. See Server-Side Cache [page 362]. If the UI2 cache is invalidated more often than expected, you can trace the invalidation events and log the call stack that lead to the invalidation and the data that was deleted from the cache.

Activating and Deactivating the Trace

To activate the trace, choose *Activate Cache Invalidation Trace* in the toolbar and enter the desired number of hours in the popup that opens. When the specified time is over, the trace is deactivated automatically. To deactivate the trace, select *Deactivate Cache Invalidation Trace*.

Note that traces are written independent of the client. If you switch on the trace, cache invalidation events occurring in all clients of the system will be logged.

Analyzing Trace Results

The trace results are displayed in the *Invalidation Statistics* tab, and show the following details:

* Client: The client in which the invalidation event occurred. Note that it is possible that data was also invalidated in other clients.
* User: The user who was logged on when the invalidation event occurred. This is not necessarily the user whose data has been invalidated from the cache. It is possible that data was also invalidated for other users.
* Timestamp: The date and time of the invalidation event. Note that several rows can belong to a single invalidation event. You can find more details in the *Action* column.
* Call Stack Hash: The hash of the call stack which lead to the invalidation. You can click the hash to open a popup with more details.
* Records Invalidated: Counts the number of records that have been deleted from the cache. You can click the number to open a popup with more details.
* Action: Indicates when an invalidation event starts and ends by displaying **ACTION_START** and **ACTION_END**.

Deleting Traces

To delete trace results, in the toolbar choose *Delete cache statistics* and specify the date up to which the traces will be deleted. Make sure that the *Delete Cache Invalidation Statistics* checkbox is selected.

> i Note
>
> The cache statistics utility writes to the application log (object `/UI2/BE`, subobject `/UI2/CSU`). Choose *Display Log* to navigate from the cache statistics utility to transaction `/UI2/LOG`. See Display Logs for User Interface Services [page 402].

Related Information

Server-Side Cache [page 362]

1.3.7.2.2   Client-Side Cache

The client-side cache typically resides in a browser.

The cache stores the requested resources. If certain conditions are met, subsequent requests read the resources from the cache. This improves the performance considerably as roundtrips to the front-end server

are avoided. An example of such a condition is if the maximum retention period in the cache has not been reached (HTTP response header cache-control: `<max-age>`).

Cache Busting

The SAP Fiori launchpad infrastructure uses the technique of **URL expiry** (**cache busting**). The server typically assigns the resource an expiry date in the far future to avoid conditional HTTP requests, which require a roundtrip to the server.

The cache expiration is controlled by cache buster tokens, which are automatically updated when changes within the SAP Fiori stack or configuration requiring an invalidation of the cache occur. A roundtrip is only required when the cache buster token is updated.

The following image shows the cache validity with cache buster tokens:
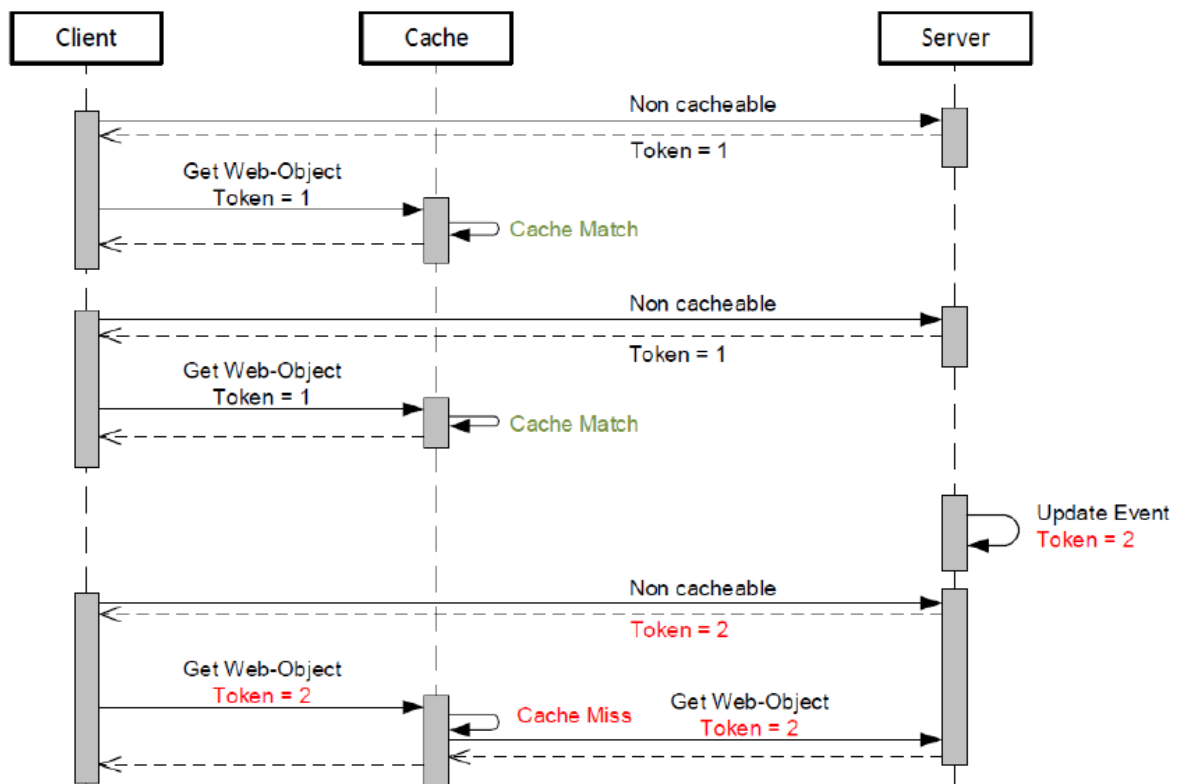
Related Information

Cache Buster for Target Mappings and UI2 Services [page 366]
Cache Buster for SAP Fiori Launchpad and SAP Fiori Apps [page 367]
Cache Buster for OData Metadata of SAP Fiori Apps [page 370]

1.3.7.2.2.1 Cache Buster for Target Mappings and UI2 Services

Target mappings are stored in the client cache. When navigating between applications or when filtering tiles according to the device types on the home page, the client cache stores the resources that are retrieved from it when the user requests them again.

The target mappings are user-specific and are based on the roles assigned to the user in PFCG.

Data remains in the browser cache until it is deleted by the user or cleaned up by the browser when its expiry date is reached. To avoid data caching, use the private or incognito browsing mode.

Cache Buster for Target Mappings

A cache buster token for target mappings is used to make sure that data is read from the server rather than from the client when settings for target mappings were changed. Parameter sap-cache-id is added to the URL to store the cache buster token.

A new cache buster token for target mappings is, for example, calculated if one of the following changes:

- Target mapping configurations in the launchpad designer or in the *Launchpad Customizing* (transaction LPD_CUST) if the role of the user sending the request is assigned the catalog which comprises the changed target mapping.
- SAPUI5 application components
- Settings in the *Configuration of RFC Connections* (transaction SM59)
- System alias mappings in table /UI2/SYSALIASMAP

If one of these settings changes, the client-side cache is invalidated and the target mapping is read from the front-end server. Otherwise, the target mapping request is read from the client cache, which significantly improves the performance.

Changes to *Launchpad Customizing* (transaction LPD_CUST) affecting a user's target mappings not only result in the generation of a new cache buster token (leading to a client-side cache invalidation) but also in the invalidation of server-side cache.

> **i Note**
>
> To make sure the cache busting mechanism works correctly for target mappings defined in *Launchpad Customizing* (transaction LPD_CUST) we recommend you implement SAP Note 2243928 .
>
> If you do not implement this note, changes in the *Launchpad Customizing* may not be reflected in the launchpad. To fix this inconsistency, you need to manually invalidate the cache using report /UI2/ DELETE_CACHE.

Cache Buster for UI2 Services

The `sap-cache-id` token is used as well for the personalization service (`/UI2/INTEROP/PersContainers`), the Page Builder service (`/UI2/PAGE_BUILDER_PERS`) and the startup service (`/UI2/START_UP/`).

Related Information

1.3.7.2.2.2 Cache Buster for SAP Fiori Launchpad and SAP Fiori Apps

Web browsers store static resources, like JavaScript files, stylesheets, and images in the browser cache. When these resources are changed on the server in a software upgrade, you want the browser to load the new resources from the server rather than from the cache, without having to manually clear the browser cache.

Cache buster techniques cause web browsers to load content from the server rather than from the browser cache when new resources are available on the server.

The cache buster for the SAP Fiori launchpad and SAP Fiori apps is based on versioned URLs. Cache buster tokens are added to the URLs of the resources. After a software update, URLs containing new tokens are used, which means that the browser considers these URLs as initial requests and loads the new resources from the server.

The advantage of this approach is that resources in the cache are only reloaded when a new version is available. Rather than simply limiting the time a resource may remain in the browser cache, the system invalidates the cache only when resources are actually updated on the server.

> i Note
>
> The SAP Fiori launchpad uses a cache buster implementation which is different from the SAPUI5 cache buster.

Cache Buster Behavior depending on Resource Type

Cache buster tokens are added to the following types of resources:

Type	Description
SAPUI5 library resources	The build timestamp of the last deployed SAPUI5 version is added to the URLs of these resources.
SAP Fiori apps and SAP Fiori libraries which are deployed in the SAPUI5 repositories	The SAP Fiori launchpad can generate specific cache buster tokens for individual SAP Fiori apps and SAP Fiori libraries. The advantage of this approach is that the browser only reloads those applications that have been updated on the server.
	To identify updated applications, the SAP Fiori launchpad uses the SAPUI5 application index on the ABAP front-end server. The SAPUI5 application index itself is updated by the ABAP program *Calculation of SAPUI5 Application Index for SAPUI5 Repositories* (`/UI5/APP_INDEX_CALCULATE`). We recommend that you schedule this program to run at regular intervals. For more information, see .
Other SAPUI5 application resources	These resources include classic SAPUI5 applications which are not implemented as SAPUI5 components or which are no application libraries. In the context of SAP Fiori, such applications contain implementations for SAP Fiori launchpad tiles.
	For these application resources, the SAP Fiori launchpad uses a global cache buster token, which is not application-specific and needs to be updated manually. If the SAPUI5 application index is incomplete, the missing applications also get the global cache buster token.
	You can manually invalidate the cache for such resources by running the ABAP program `/UI2/INVALIDATE_CLIENT_CACHES`. For more information, see Invalidating Client Caches [page 369].

URL Parameters for the SAP Fiori Launchpad

The following URL parameter is available:

Parameter	Description
`sap-ushell-nocb`	To deactivate cache busting in your web browser, set this parameter to "true".
	This is only needed for troubleshooting purposes and only affects the SAPUI5 library resources.

Restrictions

No cache buster tokens are added to URLs of SAPUI5 demo kit apps. Developers, testers, and administrators who use these resources must manually clear the browser cache.

SAP Gateway Caches

You may also want to clear SAP Gateway caches, for example if you did any changes to OData services. For more information, see the SAP Gateway documentation at ▶ *http://help.sap.com/nwgateway* ▶ *Application Help* ▶ *SAP Gateway Developer Guide* ▶ *OData Channel* ▶ *Basic Features* ▶ *Service Life-Cycle* ▶ *Metadata Cache* ▶

Related Information

Customizing the Launchpad URL [page 22]
Invalidating Client Caches [page 369]
Cache Buster for OData Metadata of SAP Fiori Apps [page 370]

1.3.7.2.2.2.1 Invalidating Client Caches

Since the introduction of application-specific cache busting of SAPUI5 resources, you only need to run the program `/UI2/INVALIDATE_CLIENT_CACHES` in exceptional cases.

Prerequisites

To invalidate other users' caches, you need to have the authorization role `SAP_UI2_ADMIN_700` assigned. For more information, see .

Context

You only need to run this program in exceptional cases if you need to invalidate specific resources. These resources include data originating from OData service requests of the launchpad itself (not from SAP Fiori apps), like target mappings and personalization data, as well as a few very specific application resources that use the global cache buster token. For more information on the latter, see Cache Buster for SAP Fiori Launchpad and SAP Fiori Apps [page 367].

Procedure

1. Start transaction `SE38`.
2. Execute the program `/UI2/INVALIDATE_CLIENT_CACHES`.
3. Select the users for whom you want to invalidate the cache:

Option	Description
For all users	Invalidate the cache for all users (including the current user).
For user	Invalidate the cache for a specific user by specifying a user name.

4. Choose .

Results

The system invalidates the client caches.

Every cache invalidation is logged in the *Application Log* (transaction `SLG1`), using object `/UI2/BE` and subobject `/UI2/CLEANUP_CACHES`.

1.3.7.2.2.3 Cache Buster for OData Metadata of SAP Fiori Apps

To ensure fast loading times for SAP Fiori apps started from the launchpad, OData metadata and annotations of SAP Fiori apps are cached on the web browser.

Context

Besides static UI resources like JavaScript files, stylesheets, and images, SAP Fiori apps consist of OData metadata (metadata and annotations documents).

This metadata is stored in the backend and cached on the client using cache buster techniques. The cache buster makes sure that, when the OData metadata is changed on the server, the browser loads the new resources from the server rather than from the cache, without the need to manually clear the browser cache.

The SAP Fiori front-end server manages and persists cache buster tokens to cache OData metadata. During the startup of the launchpad, the cache buster tokens are provided to the client to use the correct representation from the underlying web caches.

Activating the Cache Buster for OData Metadata

To enable the caching and to ensure that the cache buster tokens of the corresponding OData providers are up-to-date, the report `/UI5/UPD_ODATA_METADATA_CACHE` has to be executed periodically. See Scheduling Update of OData Metadata Caching [page 26].

The report reads the SAPUI5 apps from the SAPUI5 app repository, constructs the metadata and annotation URLs, and dispatches them to both local and remote SAP Gateway hubs. The SAP Gateway hub returns the cache buster tokens. These tokens are processed according to defined rules. The result of the execution is stored in the application log, (object `/UI5/APPIDX`, subobject `BC_COLLECT`). The UI2 application log (object `/UI2/BE`, subobject `/UI2/BC_COLLECTOR`) displays which OData providers were contacted and whether a cache invalidation took place. See Display Logs for User Interface Services [page 402].

> **i Note**
>
> In a system landscape with multiple SAP Gateway hubs, you also need to maintain the OData providers which are relevant for the token collection. See Enabling Caching Across Multiple SAP Gateway Hubs [page 371].

Related Information

Client-Side Cache [page 364]
Cache Buster for SAP Fiori Launchpad and SAP Fiori Apps [page 367]
Issues with OData Metadata Caching [page 643]

1.3.7.2.2.3.1 Enabling Caching Across Multiple SAP Gateway Hubs

In a system landscape with multiple SAP Gateway hubs, you need to perform configuration steps to activate the client-side caching of OData metadata.

Context

In the recommended SAP Fiori system landscape, SAP Gateway is deployed on the SAP Fiori front-end server. See http://help.sap.com/fiori_implementation ▶ *Configuration Information* ❯ *<Select your SAP NetWeaver release>* ❯ Setup of SAP Fiori System Landscape. For this local SAP Gateway scenario, the configuration steps described below are not required.

For some use cases, it is required to connect multiple back-end systems acting as OData providers to the front-end server via multiple SAP Gateway hubs. The following graphic illustrates such a system landscape:
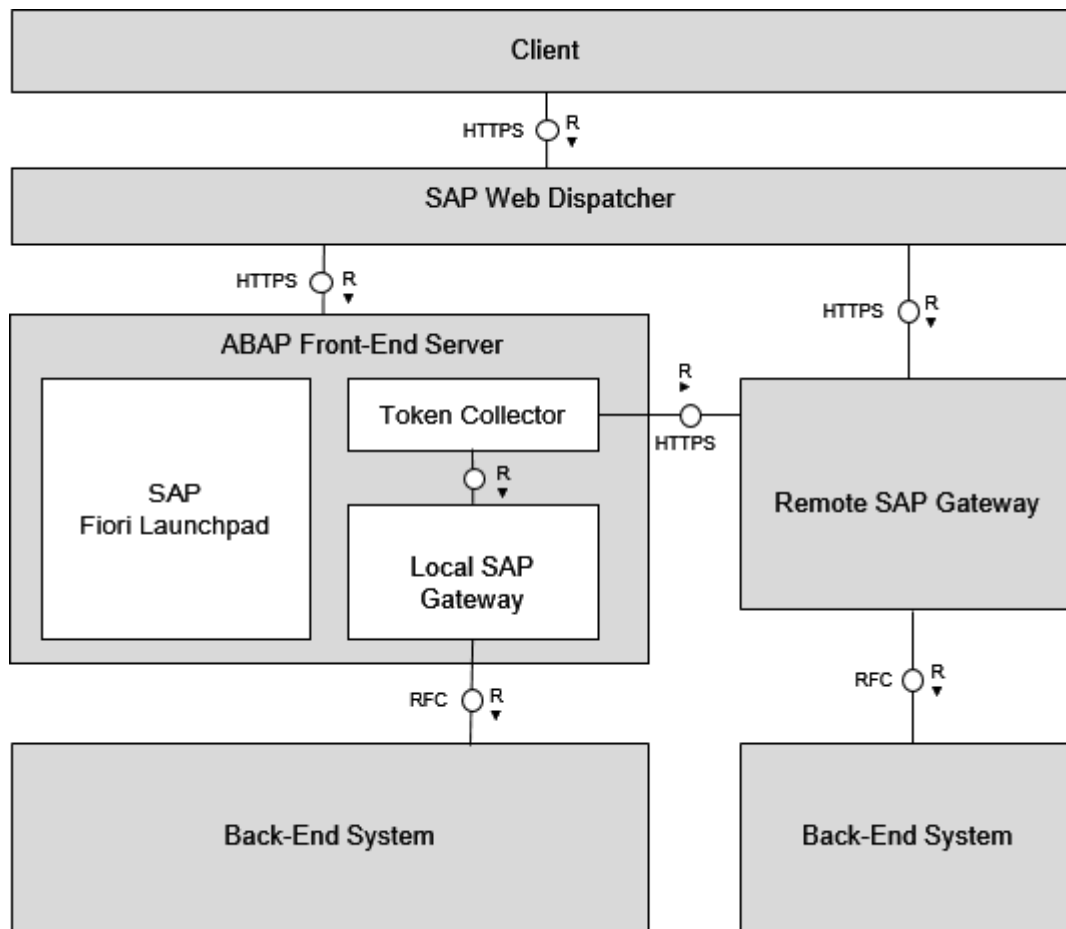
For this system landscape, several configuration steps are required to activate the cache buster for OData metadata. The front-end server needs to know which SAP Gateway hubs are deployed in the system landscape to be able to communicate with them. SAP Gateway hub services are called to collect the cache buster tokens for OData metadata from the connected back-end systems. Therefore, you first need to define an alias for each SAP Gateway hub. Then you define which OData providers are relevant for the collection of change tokens.

Procedure

1. Establish a connection from the front-end server to the remote SAP Gateway hubs:
   a. Start *Configuration of RFC Connections* (transaction SM59).
   b. Create an HTTP(S) connection of type H (*HTTP Connections to ABAP System*).

   You can also use the mapping table to reuse an existing entry. See Map System Aliases to RFC Connections [page 17].
2. Define the relevant OData providers for the token collection:
   a. Start *Maintain Table Views* (transaction SM30).
   b. Open the view /UI2/V_ALIAS.

c. In the *Alias* field, enter the RFC destination defined in step 1.
d. In the *Alias Type* field, choose *Alias for Backend Context*.
e. Mark the *Active* checkbox.

> **i Note**
>
> For local hub scenarios, you can deactivate the cache buster for OData metadata by entering **LOCAL** in the *Alias* field and removing the checkmark for the *Active* checkbox.

Results

When the report `/UI5/UPD_ODATA_METADATA_CACHE` is executed, it contacts the OData providers you defined in the view `/UI2/V_ALIAS`, retrieves the change token for each resource and updates the tokens if required. A log summary indicates which OData providers were contacted and whether a cache invalidation took place.

Related Information

Cache Buster for OData Metadata of SAP Fiori Apps [page 370]
Scheduling Update of OData Metadata Caching [page 26]

1.3.7.2.3 Performance Statistics for OData Calls

You can access SAP performance statistics from the SAP NetWeaver Gateway framework for each OData request that is submitted by the launchpad..

Performance statistics for the SAP Fiori launchpad are supported for SAP NetWeaver Gateway SP09 and higher.

To activate performance statistics, add the following query parameter to the SAP Fiori launchpad URL:

`sap-statistics=true`

Alternatively, you can also choose Ctrl + Shift + Alt + P , and activate the setting *Enable SAP-statistics for oData calls*.

When this query parameter is set, an `sap-statistics` HTTP request header is added to OData calls, and SAP NetWeaver Gateway sets an sap-statistics header in the HTTP response. This header contains SAP performance statistics from the SAP NetWeaver Gateway framework.

For more information on SAP performance statistics, see the SAP NetWeaver Gateway documentation on SAP Help Portal at *http://help.sap.com* > *Technology* > *SAP NetWeaver Platform* > *SAP NetWeaver Gateway Foundation* > *SAP NetWeaver Gateway Foundation Technical Operations Guide* > *Supportability* > *SAP Performance Statistics* .

1.3.7.2.4 Improving Navigation Performance for SAP GUI Applications

You can improve the performance when navigating to applications based on SAP GUI for HTML in the SAP Fiori launchpad.

Prerequisites

This feature requires a minimum kernel version for the Netweaver ABAP application server that hosts the SAP GUI application. For more information, see SAP note 2657777 .

Context

The launchpad uses so-called application containers for embedding apps with different UI technologies. When users navigate from one app to another, the default behavior of the launchpad is to remove the container, and then create a new container containing the new app. This is called a stateless container.

To improve the performance when navigating to an SAP GUI app, you can activate the stateful application container for SAP GUI apps. After you have done this, when navigating from an SAP GUI app to another app, the container that contains the SAP GUI app will be kept alive. As a result, any subsequent navigation to an SAP GUI app on the same back-end system is faster.

To activate the stateful navigation container for applications based on SAP GUI for HTML, proceed as described below.

Procedure

1. In *SAP Reference IMG* (transaction SPRO), choose ▶ *SAP NetWeaver* ❯ *UI Technologies* ❯ *SAP Fiori* ❯ *Configuring SAP Fiori Launchpad* ❯ *Client-Specific Settings* ▶.
2. In *FLP Configuration*, set the parameter *NAVIGATION_GUI_STATEFUL_CONTAINER* to **true**.

Related Information

Setting Parameters in SAP Fiori Customizing [page 63]
Launchpad Configuration Parameters [page 29]

1.3.8 Launchpad Support Tools

Several tools are available to support you in analyzing errors occurring in the launchpad or launchpad designer.

Category	Tool	Description	More Information
Content checks	Launchpad content manager (transactions /UI2/FLPCM_CONF and /UI2/FLPCM_CUST)	Display issues with launchpad content. The following issues are displayed: • The catalog state is different on the configuration and the customizing scope. • References to tiles and target mappings cannot be resolved. • There are issues with the configuration of tile and target mapping.	Displaying Issues with Launchpad Content [page 261]
	Fiori Launchpad Checks (transaction /UI2/FLC)	Check the consistency of delivered and customized content created for use with SAP Fiori launchpad.	Launchpad Content Checks [page 378]
	Orphaned Catalogs and Groups (transaction /UI2/FLC1)	Check launchpad content in the customizing scope for orphaned objects and clean up the content.	Checking Launchpad Content for Orphaned Catalogs and Groups [page 377]
	Fiori Launchpad Intent Analysis (transaction /UI2/FLIA)	Check the assignment of a given intent (semantic object - action pair) to a user or find out which target mapping belongs to a selected intent.	Analyzing Intents for SAP Fiori Launchpad [page 387]
	Fiori Launchpad Texts (transaction /UI2/FLT)	Analyze SAP Fiori launchpad texts maintained in the SAP Fiori launchpad designer.	Analyzing Launchpad Texts [page 389]
	SAP Fiori launchpad content aggregator (transaction /UI2/FLPCA)	Get an aggregated overview of all the launchpad content maintained in the SAP Fiori launchpad content manager (customizing scope).	Analyzing Launchpad Content [page 392]
Health checks	Task list /UI2/FLP_HEALTH_CHECKS	Check whether the system configuration required to run the SAP Fiori launchpad is in place.	Health Checks Via Task List /UI2/FLP_HEALTH_CHECKS [page 398]

Category	Tool	Description	More Information
	Fiori System Alias Check (transaction /UI2/FSAC)	Check the consistency of system aliases configured for the launchpad.	System Alias Checks [page 399]
Logging and tracing	UI2 log (transaction /UI2/LOG) and UI2 trace	Display messages for OData services and utilities (e.g. reports) of the user interface services.	Logging and Tracing for Launchpad and Launchpad Designer [page 401]
Performance	Cache Statistics Utility (transaction /UI2/CSU)	Check how often data is read from the cache or find out why the cache was invalidated.	Analyzing Cache Statistics [page 363]
Client-side configuration	SAPUI5 diagnostics window	View the SAP Fiori launchpad configuration on a specific client.	Displaying Client-Side Configuration (Deprecated) [page 406]
Configuration and authorization	App Support tool	Display configuration or authorization issues for a specific app in the SAP Fiori launchpad.	App Support [page 393]

Related Information

Troubleshooting [page 631]

1.3.8.1 Checking and Analyzing Content

Check and analyze errors that occur in the launchpad or in the launchpad content manager.

Related Information

Checking Launchpad Content for Orphaned Catalogs and Groups [page 377]
Launchpad Content Checks [page 378]
Analyzing Intents for SAP Fiori Launchpad [page 387]
Analyzing Launchpad Texts [page 389]
Analyzing Launchpad Content [page 392]

1.3.8.1.1 Checking Launchpad Content for Orphaned Catalogs and Groups

The program *Orphaned Catalogs and Groups* (transaction `/UI2/FLC1`) checks launchpad content in the customizing scope for orphaned objects and offers tools to clean up the content.

The following scenario can lead to orphaned catalogs and groups:

1. Catalog/group is created in the configuration scope.
2. Tiles referenced by the catalog/group are modified in the customizing scope.
3. Catalog/group is deleted in the configuration scope.

As a result, tile texts may be displayed incorrectly or translated texts become invisible.

To clean up orphaned catalogs or groups, you use the program *Orphaned Catalogs and Groups* (transaction `/UI2/FLC1`).

The following table lists the problems reported by the program and explains the necessary action by the user:

Problem Type	Action

Status: ⚠️
 	
Catalog/group in the customizing scope has a reference to a catalog/group in the configuration scope but the referenced content does not exist.
	• Repair the catalog/group:
	  1. Open the SAP Fiori launchpad designer in the customizing scope.
	  2. Select a catalog/group and open the *Edit Catalog* or *Edit Group* dialog box.
	  3. Choose *Save*.
	• Delete the customizing for the catalog/group:
	  1. In the program `/UI2/FLC1` select the link in the *Page Configuration ID* or *Property Bags Configuration ID* column.
	    Web Dynpro application `WD_ANALYZE_CONFIG_USER` is launched.
	  2. In the *List of Component Personalizations* table, choose the line corresponding to the orphaned catalog/group.
	  3. In the table toolbar, choose *Delete*.

Status: 💣 When you tried to delete a catalog/group in the customizing scope, a part of it was not deleted and prevents that a new catalog/group can be created with the same ID. ⓘ Note This issue was fixed with UI add-on 2.0 SP00.	Delete the customizing for the catalog/group: 1. In the program `/UI2/FLC1` select the link in the *Page Configuration ID* or *Property Bags Configuration ID* column. Web Dynpro application `WD_ANALYZE_CONFIG_USER` is launched. 2. In the *List of Component Personalizations* table, choose the line corresponding to the orphaned catalog/group. 3. In the table toolbar, choose *Delete*.

Related Information

1.3.8.1.2 Launchpad Content Checks

This report is available to check the consistency of delivered and customized content created for use with SAP Fiori launchpad.

The report *Fiori Launchpad Checks* (transaction `/UI2/FLC`) provides an overview of all existing content in a given area based on development packages and exposes consistency errors in these packages. It allows you to check to which authorization role the content is assigned and if a given user is assigned to this role. You can filter for content assigned to a given user.

The data retrieved by the report can be used for troubleshooting as well as for data analysis and monitoring.

Troubleshooting

Correct standard issues with launchpad content

The tool analyses the current configuration for standard problems that appear during content preparation and helps you to fix errors and warnings effectively.

The table below lists the checks performed and the problems reported by the report and explains the necessary action by the user:

Check	Test ID and Problem Type	Action
Catalog checks (CAT*)	• CAT01 (Error): Catalog does not have software component assigned • CAT02 (Error): Catalog does not have package assigned • CAT03 (Warning): Catalog does not have tiles (empty)	To be fixed in the launchpad app manager or in the launchpad designer

Check	Test ID and Problem Type	Action
Group checks (GRP*)	• GRP01 (Error): Group does not have software component assigned • GRP02 (Error): Group does not have package assigned • GRP03 (Warning): Group does not have tiles (empty) • GRP04 (Error): Tile in group has an error • GRP05 (Errror): Tile in the corresponding catalog was a reference to an original but the original tile was deleted. Therefore, the reference is broken.	To be fixed in launchpad designer
Target mapping checks (TM*)	• TM01 (Error): Semantic object used in target mapping does not exist • TM02 (Error): Action is not defined for target mapping • TM03 (Warning): Target mapping action does not start with lower case • TM04 (Error): Launchpad (transaction `LPD_CUST`) used in target mapping does not exist in this client • TM05 (Error): When specifying the navigation target in the target mapping definition for SAPUI5-based SAP Fiori apps, the SAPUI5 component was not entered correctly. • TM06: (Error): Target mapping was a reference to an original but the original target mapping was deleted. Therefore, the reference is broken.	To be fixed in the launchpad app manager or in the launchpad designer
	TM07: Issues regarding the system alias that is referenced in the target mapping configuration For the list of checks, see System Alias Checks [page 399].	To be fixed using the IMG activities *Manage SAP System Aliases* or *Manage RFC Destinations*.

Check	Test ID and Problem Type	Action
Tile checks (AP*)	• AP01 (Error): Semantic object targeted by tile does not exist • AP02 (Error): Tile has empty action • AP03 (Warning): Action used in tile does not start with lower case • AP04 (Error): Tile does not have valid target mapping for object or action within the same catalog • AP05 (Error): Tile was a reference to an original but the original tile was deleted. Therefore, the reference is broken. • AP06 (Warning): Refresh interval is too small. Consider setting the interval to >300 seconds. The checks are performed for the following tile types: • Static tiles • Dynamic tiles • SAP Smart Business tiles	To be fixed in the launchpad app manager or in the launchpad designer
	• AP07 (Error): ICF service does not exist • AP08 (Error) : ICF service is inactive	To activate the ICF services, see Activate ICF Services of Web Dynpro Apps and Activate ICF Services of SAPUI5 Application.
ABAP launchpad checks (LP*)	• LP01 (Error): Unspecified error in launchpad (transaction `LPD_CUST`) • LP02 (Error): Additional info field of launchpad definition must contain 'SAPUI5.Component=' (type URL) • LP03 (Error): System alias in launchpad definition is not empty (type URL) • LP04 (Error): System alias in launchpad definition is empty (type non-URL) • LP05 (Error): Target app parameters field is empty for launchpad definition with additional information equal "SAPUI5.Component=factsheet" • LP06 (Warning): Encoding of target app parameter is invalid	To be fixed in transaction (`LPD_CUST`)

Correct issues with translated launchpad texts

An end user of the launchpad has reported an issue with a translated text in a tile, catalog, or group title. You can use *Fiori Launchpad Checks* to easily find the relevant text in the translation system and correct it.

See Translating Launchpad Texts [page 408].

Data Analysis and Monitoring

You want to analyze launchpad content and improve the content lifecycle.

For example, you can analyze the following questions:

- Are the naming conventions for catalogs, groups and tiles applied consistently?
- Are the settings for device types applied correctly?
- Are there any tiles or target mappings which are not used in roles?
  If yes, you should either enhance the roles or delete the tiles and target mappings that are no longer required.
- How many catalogs are available for each product area?
- Which applications are used as navigation targets in a given launchpad?
  For example, how many Web Dynpro ABAP applications are referenced as opposed to SAPUI5 applications?
- Which application types are referenced from a given catalog?

The content check report retrieves the following types of information:

Test ID	Description
INFO_AP	Provides information on static app launcher tiles
INFO_DAP	Provides information on dynamic app launcher tiles
INFO_CR	Provides information on catalogs
INFO_GR	Provides information on groups
INFO_GR_AP	Provides information on tiles in groups
INFO_TM	Provides information on target mappings

Related Information

Checking Launchpad Content [page 382]
Setting Up Technical Catalogs with the SAP Fiori Launchpad App Manager [page 132]
Updating Content Created with the Launchpad Designer [page 182]

1.3.8.1.2.1 Checking Launchpad Content

This section describes how you use the launchpad content check report.

Procedure

1. Start *Fiori Launchpad Checks* (transaction `/UI2/FLC`) in the back-end ABAP system.
2. Select an *Adaptation Layer*. You have the following options:

   a. Select *Configuration* to check content in the configuration scope (client-independent).

   b. Select *Customizing (layered)* if you want to check content in the customizing scope (client-dependent), and to verify that *Configuration* content is not overwritten by customized content.

   c. Select *Customizing (only)* if you want to check content in the customizing scope (client-dependent).

   See Scopes for Adapting Launchpad Content [page 111].
3. Optional: Enter criteria to filter the results list:

   - ID of a catalog for which you want to review the content
   - ID of a group for which you want to review the content
   - Development package for which you want to review the content
     Select the *Include Subpackages* checkbox if you use a main package or a structure package and want a full content overview.
4. Optional: Enter a user name to display only the content that is assigned to this user.

   - Select the *Restrict to Assigned Roles* checkbox to display only the content for which the specified user has an assigned authorization role.
   - If you do not select the *Restrict to Assigned Roles* checkbox, all content will be shown. If the content is assigned to the specified user, an indicator in column *PFCG Role Assigned* is displayed.
5. Run the report. The result of the check is displayed.

   The default result table lists all content for the selected packages. In addition, the errors or warnings for content that is invalid with additional troubleshooting information is displayed. For an error, the result table can list two entries for the same target mapping or tile:

   - An entry that provides the information about the existence of the object found with all the details.
   - An entry that lists the error. An explanation of the error is available in the *Message* field.

   Messages containing information have the message type `INFO` and the test ID `INFO*`.

   In the result screen, you have the following options:

   - Use a predefined layout to view the results, for example, to display only the launchpad errors or only the catalog content view.
   - Use the search or filter functions from ALV to locate a specific problem.
   - Use the ALV export function to save the table entries in a local file.
     You can then optionally add data to the data retrieved by the report and create a pivot table for advanced data analysis.
   - Select a row and choose *Details* to display only cells containing information.

Related Information

1.3.8.1.2.2 Determining Catalog and Authotization Role for Tile ID

You want to determine which catalog and authorization role a tile ID belongs to.

A tile is part of a catalog. An authorization role is used to assign the tile to a user and to the catalog to which the tile belongs.

To determine catalog and authorization role a given tile ID belongs to, proceed as follows:

1. Identify the tile ID in SAP Fiori launchpad or in SAP Fiori launchpad designer.
2. Use the launchpad content check report to determine the catalog ID and authorization roles for the identified tile ID. Filter the result list according to the tile ID.

Related Information

1.3.8.1.2.2.1 Determining Tile ID in SAP Fiori Launchpad

You want to determine the tile ID in SAP Fiori launchpad.

Context

To determine the tile ID in SAP Fiori launchpad, proceed as follows:

Procedure

1. Open the SAP Fiori launchpad with the tile for which you want to determine the tile ID.
2. Open the SAPUI5 console by pressing `CTRL-SHIFT-ALT-S`.

   This is needed to enable debug log level in SAPUI5 controls.
3. Open the Google Chrome console by pressing `F12` and switch to the *Console* view.

   Optional: In Google Chrome, clear the content of the console for better visibility by choosing *Clear console log*.
4. Click the tile to navigate to an application. The tile ID is written to the console log.

   The example from the log below shows the tile ID (`<chipInstanceId>`):

   {"chipId":"X-SAP-UI2-PAGE:X-SAP-UI2-CATALOGPAGE:SAP_CRM_BC_SalesRep:00O2TO88JXF2O0MBML2VLFJYJ","chipInstanceId":"00O2TR8BR0IIDAK3ITH6IYT3D"}

   Copy the tile ID to the clipboard.

1.3.8.1.2.2.2 Determining Tile ID in SAP Fiori Launchpad Designer

You want to you determine the tile ID in SAP Fiori launchpad designer.

Context

To determine the tile ID in SAP Fiori launchpad designer, proceed as follows:

Procedure

1. In the catalog section, select a catalog from the catalog list on the left panel, and select the tile for which you want to determine the ID.

   The ID is displayed in the tile details. Double-click the tile ID to select it and copy it to the clipboard.

2. To determine the tile ID in the group section, select a group from the group list on the left panel and click

   the ⓘ icon in the tile. The catalog ID (catalog the tile belongs to) and the tile ID is displayed. Double-click the tile ID to select it and copy it to the clipboard.

1.3.8.1.2.2.3 Determining Catalog and Authorization Role for Tile ID

You want to determine which catalog and authorization role a tile ID belongs to.

Context

To determine which catalog and authorization role a tile ID belongs to, you use the content check report for SAP Fiori launchpad.

Procedure

1. Start *Fiori Launchpad Checks* (transaction `/UI2/FLC`) in the back-end ABAP system.
2. Select *Customization* as the *Adaptation Layer*.

   This ensures that system-wide and client-specific content is covered. To search content in any package, leave the *Package* field empty.

3. Mark the *Analyze Roles* checkbox.

4. Choose *Execute*.

5. In the toolbar above the result table, choose (*Find*) to search for the tile ID.

Results

The *PFCG Role* column displays the authorization roles the tile is assigned to (as part of the catalog the tile is assigned to).

Related Information

1.3.8.1.2.3 Determining Groups Assigned to a User

You want to determine which groups are assigned to a user.

Context

To determine which groups are assigned to a user, you use the content check report for SAP Fiori launchpad. Perform all steps on the front-end server.

Procedure

1. Start *Maintain User Profile* (transaction SU01).
2. Enter a user name and choose *Display*.
3. Choose the *Roles* tab and copy all roles.
4. Start *Fiori Launchpad Checks* (transaction /UI2/FLC).
5. In the start screen, choose *Execute*. The result list is displayed.
6. Filter the result list as follows:
    a. Define filter criteria for the authorization roles. Paste the roles you copied earlier from the clipboard.
    b. Then define filter criteria for the group ID for which a value is defined.
7. In the filtered result list, hide all columns except for *Group ID* and *Group Title*.
8. Optional: You can save the current table layout for later use.

9. Optional: You can export the result list for further analysis to Microsoft Excel.

Results

The result list displays all groups assigned to the user.

> **i Note**
>
> To determine the number of groups assigned to you by the administrator, you can also use the API *GetNavigationTree* for the server runtime for SAP NetWeaver user interface services in the browser:
>
> `http://<server>/ui2/nwbc/~api/GetNavigationTree`
>
> Search for *sap-ui2-group*. The number of hits is the number of groups. The second part of the tile ID indicates the group name.

Related Information

APIs to Export Detailed Information

1.3.8.1.3   Analyzing Intents for SAP Fiori Launchpad

This report is available to check the assignment of a given intent (semantic object - action pair) to a user in the context of SAP Fiori launchpad.

The report *Fiori Launchpad Intent Analysis* (`/UI2/FLIA`) provides an overview of all intents for a user. It allows you to check to which authorization role the intent is assigned and if a given user is assigned to this role.

You can filter the results from the report by device (form factor).

The check is system-dependent. Start the report in the system for which you want to verify the content.

Related Information

Determining Roles for an Intent [page 388]
Determining Target Mappings for an Intent [page 388]

1.3.8.1.3.1 Determining Roles for an Intent

Use the report *Fiori Launchpad Intent Analysis* to check the assignment of a given intent (semantic object - action pair) to a user in the context of SAP Fiori launchpad.

Procedure

1. Start *Fiori Launchpad Intent Analysis* (transaction `/UI2/FLIA`) in the back-end ABAP system.
2. Enter the name of the intent for which you want to review the authorization roles.
3. Enter the form factor, for example, *Phone*.
4. Optional: To check only the roles assigned to a specific user, select the *Restrict to assigned roles* checkbox and enter the user name in the *Analyze for User* field.
5. Optional: Enter the development package for which you want to review the data.
6. Optional: Select the *Include Subpackage* checkbox.

   Use this setting if you use a main package or a structure package and want a full content overview.
7. Select an *Adaptation Layer*. You have the following options:
   a. Select *Configuration* to check client-independent content.
   b. Select *Customization* if you want to check client-dependent content, and to verify that *Configuration* content is not overwritten by customized content.
8. Run the report. The result of the check is displayed.

   The default result table lists all semantic object - action pairs for the selected intent and user.
9. Optional: You can use the search or filter functions from ALV to locate a specific problem.

1.3.8.1.3.2 Determining Target Mappings for an Intent

Use the report *Fiori Launchpad Intent Analysis* to determine the target mapping for a given intent.

Context

When the target mapping for a given intent is not configured correctly, the launchpad will not navigate to the expected application when an end user clicks a tile. To correct this issue, you might first have to find out which target mapping belongs to the affected intent.

Procedure

1. Start *Fiori Launchpad Intent Analysis* (transaction `/UI2/FLIA`) in the back-end ABAP system.

2. Enter the name of the intent for which you want to determine the target mapping (e.g. SalesOrder-display).

3. Enter the form factor, for example, *Phone*.

4. Optional: Enter a user name.

5. Optional: Select the *Restrict to assigned roles* checkbox.

6. Optional: Enter the development package for which you want to review the data.

7. Optional: Select the *Include Subpackage* checkbox.

    Use this setting if you use a main package or a structure package and want a full content overview.

8. Select an *Adaptation Layer*. You have the following options:

    a. Select *Configuration* to check delivered content (client-independent).
    b. Select *Customization* if you want to check customized (client-dependent) content, and to verify that *Configuration* content is not overwritten by customized content.
    c. Select *Personalization* if you want to check personalized (user-specific) content. This is the default setting.

9. Run the report. The result of the check is displayed.

10. Copy the entry in the *Tile or TM ID* column.

11. Double-click the entry of the *ID of a Catalog* column.

    The SAP Fiori launchpad designer is launched, displaying the selected catalog.

12. In the launchpad designer, switch to *Target Mappings*.

13. In the search field, enter the target mapping ID you copied from the *Fiori Launchpad Intent Analysis* result list.

14. You can change the settings for the target mapping by choosing *Configure*.

1.3.8.1.4   Analyzing Launchpad Texts

Fiori Launchpad Texts (transaction `/UI2/FLT`) helps you to solve issues with SAP Fiori launchpad texts maintained in the SAP Fiori launchpad designer.

Context

The report *Fiori Launchpad Texts* provides the following features:

- Search for texts in catalogs, groups, tiles and target mappings and have them displayed in one column.
- Navigate to the objects containing the text definition (catalog, group, tile, target mapping) in the SAP Fiori launchpad designer.
- Display the configuration ID and the text ID required to identify the text in the translation system.

Procedure

1. Start *Fiori Launchpad Texts* (transaction `/UI2/FLT`) in the ABAP system.
2. Optional: Enter criteria to filter the results list:
   - Text string
   - Software component for which you want to review the texts
   - ID of a catalog or group for which you want to review the texts
   - Development package for which you want to review the texts
     Select the *Include Subpackages* checkbox if you use a main package or a structure package and want a full content overview.
3. Select an *Adaptation Layer*. You have the following options:
   a. Select *Configuration* to check delivered content (client-independent).
   b. Select *Customization* if you want to check customized (client-dependent) content, and to verify that *Configuration* content is not overwritten by customized content.
4. Optional: Enter a user name to display only the content that is assigned to this user.
   - Select the *Restrict to Assigned Roles* checkbox to display only the content for which the specified user has an assigned authorization role.
   - If you do not select the *Restrict to Assigned Roles* checkbox, all content will be shown. If the content is assigned to the specified user, an indicator in column *PFCG Role Assigned* is displayed.
5. Run the report. The result of the check is displayed:
   - *Text* column: displays all texts occurring in the launchpad.
   - *Text Type* column: indicates the text type.

CAT_TITLE	Title of a catalog
GRP_TITLE	Title of a group
GRPT_TITLE	Title of a tile that is referenced by a group
GRPT_SUBTITLE	Subtitle of a tile that is referenced by a group
GRPT_INFO	Information text of a tile that is referenced by a group
TILE_TITLE	Title of a tile
TILE_SUBTITLE	Subtitle of a tile
TILE_INFO	Information text of a tile
TM_TITLE	Title of a target mapping
TM_INFO	Information text of a target mapping

   - *Reference* column: allows you to navigate to the corresponding text definition if the object containing the text refers to another object (e.g. navigate from a group tile to the catalog containing the tile or from a reference tile to the original tile from which it was copied).

- *ID of a Catalog* and *Group ID* column: allow you to open the relevant catalog or group in the launchpad designer.
- *Catalog Title* and *Group Title*: allow you to navigate to a table displaying the configuration ID, the text ID, and the ID of the table in which the texts are stored.

Related Information

Translating Launchpad Texts [page 408]
Launchpad Content Checks [page 378]
Blog: Identifying Fiori Launchpad Texts

1.3.8.1.4.1 Determining IDs to Find Texts in the Translation System

To allow translators to find a launchpad text in the translation system and correct it, administrators determine the relevant text ID and the configuration ID.

Context

The following procedure is relevant for texts maintained in the launchpad designer (object type WDCC).

For more information on other object types, see Translating Launchpad Texts [page 408].

Procedure

1. In the system in which you run the SAP Fiori launchpad (on the front-end server for SAP Fiori launchpad), start *Fiori Launchpad Texts* (transaction `/UI2/FLT`).
2. In the *Text Filter* field, enter the text string you want to change.
3. (Optional) In the *Catalog ID* field, enter the ID of the catalog.
4. In the *Adaptation Layer* section, choose the relevant scope (customization or configuration).
5. Choose *Execute*.
6. Open the link of the relevant entry in the *Text* column.

   > **i Note**
   >
   > If the object containing the text refers to another object, choose the link icon in the *Reference* column to navigate to the object containing the text definition (e.g. navigate from a group tile to the catalog containing the tile or from a reference tile to original tile from which it was copied).

A table displaying the text ID and configuration ID is displayed, highlighting the relevant text.

7. Note the entry of the *Text ID* and *Configuration ID* fields.

Related Information

Translating Launchpad Texts [page 408]
Analyzing Launchpad Texts [page 389]
Scopes for Adapting Launchpad Content [page 111]

1.3.8.1.5   Analyzing Launchpad Content

The launchpad content aggregator provides an aggregated list of the content available in the launchpad content manager (customizing scope).

The report can be used for reporting or for offline analysis. It's based on role level and shows assigned catalogs, tiles and target mappings, and optionally, corresponding OData services, ICF services, and successor transactions.

The following features are available:

• Display an aggregated view of selected roles and their related content.

• Export a spreadsheet for further analysis.

• Adjust the list by adding or removing columns.

Procedure

1. Start the transaction `/UI2/FLPCA`.
2. In *Role Filter*, select your relevant roles to view their assigned content.
3. Mark the checkboxes *Display OData Services* or *Display ICF Services* if you want them to be displayed.
4. Mark the checkbox *Display Successor Transactions* to display an additional field with the successor information.

   > **i Note**
   >
   > Note that you can't click the checkbox with the other two checkboxes. You can either display the OData services and the ICF services or show the successor transactions.

5. Choose *Execute*.
   Now, you see an aggregated list of the content assigned to your selected roles.

Related Information

Setting Up Business Catalogs with the Launchpad Content Manager [page 238]
Exploring Launchpad Content [page 241]

1.3.8.2 Displaying Configuration and Authorization Issues

Display and analyze configuration or authorization errors for apps in your SAP Fiori launchpad.

Related Information

1.3.8.2.1 App Support

The App Support tool helps you to troubleshoot configuration and authorization errors for apps in the SAP Fiori launchpad.

The App Support tool is a plug-in tool for the SAP Fiori launchpad. It supports the following app types:

- SAPUI5 Fiori App
- Web Dynpro Application
- WebClientUI Application
- Transaction

With this app, you can troubleshoot configuration and authorization errors for apps in the SAP Fiori launchpad.

Key Features

This app provides the following key features:

Provided they have the corresponding permissions, end users can:

- check for general configuration errors
- check for app-related issues during runtime
- download logs and forward them to the admin user

In addition to the options mentioned above, the admin user is also allowed to:

- see the logs of other users

For more information on the required authorizations, see Setting Up App Support [page 396].

Launching App Support

Prerequisites

You need to configure the plug-in tool in your SAP Fiori launchpad. For more information, see Setting Up App Support [page 396]

Procedure

> **i Note**
>
> Please note that *App Support* is only visible if you launch an app in your launchpad. You won't see *App Support* in your user actions menu on the home page.

1. Launch your SAP Fiori launchpad.
2. Open the relevant app for which you want to see the launchpad checks.
3. In the user actions menu dropdown, select *App Support* to launch the window.

Feature Scope

When you open the tool, a window is presented to you with the following menus:

Feature	Description
App Information	You start in the *App Information* menu where you get more detailed information on the app, such as
	Overview
	• App Type
	• Transaction Code
	Technical Details
	• App Version
	• App ID
	• System
	• Client
	• Application Component ID

Feature	Description
Embedded Server *Front-End Server* *Back-End Server*	**i Note** Depending on your system setup, you either see a menu called *Embedded Server* or *Front-End Server* along with one or more *Back-End Server*. If no back-end details are maintained in the target mapping of the respective target application, the tool recognizes it as an embedded server. You get a list of app-specific configuration checks and their related status. If an error is displayed, you can click on *Manual Check* to directly jump into the corresponding configuration transaction. Checks that are made are, e.g., if ICF services and OData services are active or if the RFC connection to the back-end server is working.
Authorization Errors	The *Authorization Errors* menu is the equivalent to *Display Authorization Data* (transaction SU53). You can see which authorization checks have been performed and can immediately analyze authorization errors that occurred in a timely manner. The errors that are displayed in this menu are not app-specific. The authorization logs for the back-end server are only available for systems with SAP_BASIS NW 7.50 or with a higher SAP_BASIS release version. Implement SAP note 2871194.
Gateway Errors	The *Gateway Errors* menu is the equivalent to /IWNFD/ERROR_LOG or /IWBEP/ERROR_LOG. It shows errors from all users for a given client. See Display SAP Gateway Logs [page 405]. The errors that are displayed in this menu are not app-specific.
Runtime Errors	The *Runtime Errors* menu is the equivalent to *ABAP Runtime Errors* (transaction ST22). You get a list of the ABAP runtime errors that have occurred in the ABAP system. The errors that are displayed in this menu are not app-specific.

Feature	Description
Download Logs	You can download an excel with all the data that is displayed in the tool by clicking on *Download Logs*.
	Downloading your logs is useful if you encounter an error in your app and you want to contact your admin. You can directly forward the error logs which makes it easier and faster to solve the problem.

Related Information

1.3.8.2.2 Setting Up App Support

Configure the App Support plug-in to troubleshoot configuration and authorization errors of your application in the SAP Fiori launchpad.

Prerequisites

- SAPUI5 version must be 1.84 or above.
- The authorization logs for the back-end server are only available for systems with SAP_BASIS NW 7.50 or with a higher SAP_BASIS release version. Implement SAP note 2871194 .
- The following authorization object needs to be added to your role to be able to view and download logs:
  - S_FLP_AS
  In the authorization field SUI_ADEUS, you can control if users are allowed to see the logs from other users. The value CURRENT restricts you to see logs from other users. The value ALL allows you to see logs from other users.
- Most of the displayed logs have their own authorization checks. To be able to display the content of these logs, the following authorizations need to be maintained as well:
  - ADEFRONTENDERRORLOG
  - ADEFRONTENDRUNTIMELOGS
  - ADELOCALAUTHORIZATIONLOG
  - ADEREMOTEAUTHORIZATIONLOG
  - ADEREMOTEERRORLOG
  - ADEREMOTERUNTIMELOG
  For more information about authorization objects, see .

Procedure

1. Activate the OData Services

Make sure that the OData services are active.

1. Log in to your ABAP system.
2. Open the transaction *SICF* (Define Services).
3. Choose *Execute*.
4. Activate the services mentioned below (either by right-clicking the mouse and selecting *Activate Service* or choosing ▶ *Service/host* ❭ *Activate* ❭ from the menu). The relevant paths to be activated are as follows:
   - */default_host/sap/bc/bsp/sap/sui_flp_app_sup*
   - */default_host/sap/bc/ui5_ui5/sap/sui_flp_app_sup*

2. Add and activate SAP Gateway OData Service

In addition, you need to add and activate the OData service that is needed for the app to retrieve data.

1. In Customizing (transaction SPRO, choose ▶ *SAP NetWeaver* ❭ *Gateway* ❭ *OData Channel* ❭ *Administration* ❭ *General Settings* ❭ *Activate and Maintain Services* ❭ and add and activate the following OData service:
   - *sui_flp_app_sup_srv*

When adding the OData service, the system alias must be LOCAL.

3. Create and Configure your Plug-In

The plug-in can be configured in two ways:

- Option 1: Configure your plug-in in Customizing
- Option 2: Configure your plug-in via a business catalog

Option 1: Configure Your Plug-In in Customizing

The following authorization roles are required in addition to the ones mentioned in the prerequisites:

- SAP_FLP_ADMIN - is required to see the menu entry in the launchpad
- SAP_FLP_USER - is required for the end user to use the tool after it has been configured

The plug-in can be activated in client-specific settings:

Create a plug-in for App Support

1. Open transaction /UI2/FLP_CONF_DEF.
2. Choose ▶ *Define Launchpad Plugins* ❭ *New Entries* ❭. Create a new entry for your plug-in.
3. Maintain the following values for your plug-in:

FLP Plugin ID	APP_SUPPORT
Description	App Support Tool in the User Menu
UI5 Component ID	nw.core.flp.appdiagnostics

4. Save your changes.

Activate your plug-in

1. In Customizing (transaction **SPRO**), choose ▶ *SAP NetWeaver* ❯ *UI Technologies* ❯ *SAP Fiori* ❯ *SAP Fiori Launchpad Settings* ❯ *Change Client-Specific Settings* ◗ to open transaction `/UI2/FLP_CUS_CONF`.
2. Choose *Launchpad Plug-Ins*.
3. Select your plug-in and set the activity state to *Active*.
4. Save your changes.
   See Activating Plug-Ins on the ABAP Platform [page 606].

Option 2: Configure Your Plug-In Via Business Catalog

Alternatively, you can set up your plug-in by creating a custom catalog. The plug-in is delivered in the technical catalog `SAP_TC_FLP_COMMON`. After you created a business catalog, you need to assign it to the relevant authorization roles, in order to use it in your system:

Create a custom catalog

1. Open transaction `/UI2/FLPCM_CUST` to launch the launchpad content manager.
2. Search for the technical catalog `SAP_TC_FLP_COMMON` and click *Copy*.
   A copy of the technical catalog and its assigned content is created. Make sure that your business catalog is in your customer namespace.

Assign the authorization roles

1. In *Role Assignment* (transaction PFCG), assign the catalog to a role.
2. If you haven't already done this, assign the role to the respective users.
   The plug-in is now assigned to all users assigned to the respective authorization role. For more information, see Assigning Users to Roles [page 339].

Next Steps

1. Run your launchpad.
2. Open an application in your launchpad.
3. In the user actions menu, select *App Support* to launch the tool.

Related Information

App Support [page 393]

1.3.8.3 Health Checks Via Task List /UI2/ FLP_HEALTH_CHECKS

You use task list `/UI2/FLP_HEALTH_CHECKS` to check whether the system configuration required to run the SAP Fiori launchpad is in place.

The task list automatically performs the following configuration tasks in the system:

- Checks if the OData and ICF services required for the launchpad are active.
- Checks the the consistency of system aliases configured for the launchpad.
  These checks are also available in *Fiori System Alias Check* (transaction `/UI2/FSAC`). See System Alias Checks [page 399].

Prerequisites

Before you carry out a given automatic task list, the standard roles and permissions used for automated technical configuration activities must be assigned to your user.

For more information, see Standard Roles and Permissions.

Procedure

1. Start *Task Manager for Technical Configuration* (transaction `STC01`).
2. Specify the task list **`/UI2/FLP_HEALTH_CHECKS`**,
3. Press *F8* (*Generate Task List Run*).
4. Specify a transport request.

1.3.8.3.1   System Alias Checks

This report is available to check the consistency of system aliases configured for the launchpad.

Fiori System Alias Check (transaction `/UI2/FSAC`) displays consistency issues for the selected system alias or SM59 destination.

The table below lists the checks performed and the problems reported by the report and explains possible actions to fix the issue:

Status	Check	How to Fix the Issue
Error	System alias is not maintained	In *Manage SAP System Aliases*, check your entries and create missing entries if required.
Warning	No RFC destination is maintained for system alias	In transaction SM59, maintain a destination with connection type 3 (Connection to ABAP System) with the name `<system alias>_RFC`.

Status	Check	How to Fix the Issue
Info	No HTTP destination is maintained for system alias	In transaction SM59, maintain a destination with connection type H (HTTP Connection to ABAP System) with the name `<system alias>_HTTP`.
Warning	No HTTPS destination is maintained for system alias	In transaction SM59, maintain a destination with connection type H (HTTP Connection to ABAP System) with the name `<system alias>_HTTPS`.
Error	*Path Prefix* field should be empty for system alias	In transaction SM59, on the *Technical Settings* tab, make sure that the *Path Prefix* field is empty.
Error	SSL should be active for HTTPS connections for system alias	In transaction SM59, on the *Logon & Security* tab, make sure that SSL is set to active.
Warning	HTTP and HTTPS are using the same port for system alias	In transaction SM59, on the *Technical Settings* tab, make sure that different ports are maintained in the *Service No.* field for HTTP and HTTPS destination.
Warning	HTTP port for system alias should start with <port number>	In transaction SM59, on the *Technical Settings* tab, make sure that the correct port is maintained in the *Service No.* field.
Warning	HTTPS port for system alias should start with <port number>	In transaction SM59, on the *Technical Settings* tab, make sure that the correct port is maintained in the *Service No.* field.

The checks listed above are also available in *Fiori Launchpad Checks* (transaction `/UI2/FLC`) and *Fiori Launchpad Intent Analysis* (transaction `/UI2/FLIA`).

Related Information

Configuring Remote Systems [page 308]
Launchpad Content Checks [page 378]
Analyzing Intents for SAP Fiori Launchpad [page 387]

1.3.8.4 Logging and Tracing for Launchpad and Launchpad Designer

This section explains how message logging is handled for SAP Fiori launchpad and SAP Fiori launchpad designer.

The SAP Fiori system landscape comprises a front-end server and application servers. All errors occurring in the launchpad or launchpad designer are written into the UI services application log (object `/UI2/BE`) and into an SAP Gateway message container. All logs are written on the front-end server.

You have the following options to display logs:

- Display Logs in the Browser [page 401]
- Display Logs in the front-end server
  - Display Logs for User Interface Services [page 402]
  - Display SAP Gateway Logs [page 405]

1.3.8.4.1 Display Logs in the Browser

If an error occurs during processing, you can use the developer tools available for browsers to display the message.

Context

If no error occurs during processing, you can only display the messages in the back end for SAP Gateway SP07 or lower versions.

Procedure

1. Start the SAP Fiori launchpad.
2. In the browser, open the developer tools.

   > **i Note**
   >
   > For Internet Explorer, Google Chrome and Firefox, this is done by pressing the F12 key.

3. In the developer tools, switch to *Network* to display a network trace. All network requests sent by the client are displayed.
4. Select a request from the list. To display further details, for example headers that were sent or the response that was retrieved, switch to *Headers* or *Response*.
5. If there is an error like HTTP 404 (Page/file not found), select the entry and choose *Response*. For more details, switch to *Console*.

Related Information

1.3.8.4.2 Display Logs for User Interface Services

OData services, REST services and utilities (such as reports) write important information to the UI2 log.

Context

These logs can be displayed using transaction `/UI2/LOG`. It is a variant of the standard application log (transaction `SLG1`) that is easier to use as the screen contains fewer fields and some fields have default values. In addition, it provides the option to download the logs to a local file.

Procedure

1. In Customizing, choose ▌ *SAP Fiori* ❯ *Support Tools* ❯ *SAP UI2 Application Log* ❯.

   Transaction `/UI2/LOG` is launched.

   > **i Note**
   >
   > You can also access the UI2 logs using transaction `SLG1` and object `/UI2/BE`.

2. Enter a subobject:

Subobject	What is logged
/UI2/INTEROP	• Transfer personalization from Web Dynpro Configuration to database • Delete expired personalization • Errors in navigation link resolution • Cleanup of application states
/UI2/PFCG	An application used in the assigned catalog is not available in the ABAP system when an administrator determines authorization defaults in Role Maintenance (transaction PFCG). See Assign Business Catalogs to Roles [page 330].
/UI2/PAGE_BUILDER	Entities in the launchpad designer or in the launchpad home page are created, updated, or deleted.

Subobject	What is logged
/UI2/CLEANUP_CACHES	All server-side caches are invalidated.
/UI2/TRACE	When the trace is activated, steps executed by an OData or REST service provided by SAP NetWeaver User Interface services are logged. See Tracing for User Interface Services [page 404].
/UI2/SANITY_CHECKS	Messages from *Fiori System Alias Check* (transaction /UI2/FSAC). See System Alias Checks [page 399].
/UI2/BC_COLLECTOR	Shows which SAP Gateway hubs were contacted and whether an invalidation of the OData metadata cache took place. See Cache Buster for OData Metadata of SAP Fiori Apps [page 370].
/UI2/CSU	Cache statistics. See Analyzing Cache Statistics [page 363].

> i Note
>
> Alternatively, you can enter the *External ID*. This is a unique ID that is displayed in the error details in the browser.

3. Choose *Execute*.

   In the message list that is shown, you can double-click an entry to display the message text.

4. Depending on the error cause, you can choose ![icon] (*Details*) for further information:

   - Errors in navigation link resolution:
     You are directly navigated to the*Fiori Launchpad Intent Analysis* (transaction /UI2/FLIA). Run the report to determine the root cause of the failed link resolution. For more information, see Issues with Navigation [page 635].
   - Errors occuring in launchpad designer or launchpad home page for actions create/update/delete:
     A call stack is displayed and you can navigate to the message code.

5. Optional: Download the complete selection of logs as a local file.
   a. Choose *Download all logs*.
   b. Choose ▌ *List* ❭ *Save* ▌ and select a format.

   > i Note
   >
   > The UI2 log does not always return the required results for the SAP Fiori launchpad use case because there might be other errors occurring before the UI2 log can be written (for example, errors occurring because the user does not have the required authorization to call a service or because the service is inactive). In this case, you use one of the logs that are written in the SAP NetWeaver Gateway infrastructure. See Display SAP Gateway Logs [page 405].

Related Information

1.3.8.4.2.1 Tracing for User Interface Services

Use this trace to record specific steps executed by user interface services (e.g. the ones generated by the UI2 OData/REST requests).

Context

Trace files are mainly used by developers or support (engineers) to record and analyze the occurrence of certain events at runtime. Therefore, tracing is normally turned off. Traces can be switched on if a problem has occurred and a detailed analysis of a distinct aspect of the launchpad is necessary.

You can download the trace to a local file to, for example, attach it to an incident.

Procedure

1. To activate the trace, go to *Maintain User Profile* and set user parameter `/UI2/TRACE` to value `X`.
2. To display the trace file, launch transaction `/UI2/LOG` and specify subobject `/UI2/TRACE`.

    See .
3. Once you are done with your analysis, switch off the trace by removing the value of user parameter `/UI2/TRACE`.

    Writing to the trace does not affect the performance of a service but adds unnecessary information to the UI2 log.

Related Information

1.3.8.4.3　Display SAP Gateway Logs

Errors occurring in the launchpad or launchpad designer are written into an SAP Gateway message container.

You can use one of the SAP Gateway logs if you cannot analyse the error in the UI2 log (for example, for errors occurring because the user does not have the required authorization to call a service or because the service is inactive).

Application Log: Display Logs for the SAP Gateway Runtime (transaction `SLG1`)

1. Start *Application Log: Display Logs* (transaction `SLG1`) in the ABAP system.
2. Enter the following data:
   - *Object:* /IWBEP/
   - *Subobject:* RUNTIM
   - *External ID*
     This is the transaction ID displayed in the error details in the browser.
3. Choose *Execute*.

GW Backend Error Log (transaction `/IWBEP/ERROR_LOG`)

1. Start *GW Backend Error Log* (transaction `/IWBEP/ERROR_LOG`) in the ABAP system.
2. Choose *Re-Select*.
3. Enter the following data:
   - User ID
   - Date
   - Time interval
4. Choose *Execute*.

SAP Gateway Error Log (transaction `/IWFND/ERROR_LOG`)

1. Log on to the SAP Gateway server in case it is not installed in the back end.
   If SAP Gateway is installed in the back end, you can start the transaction directly in the back end.
2. Start *Gateway Error Log* (transaction `/IWFND/ERROR_LOG`) in the ABAP system.
3. Choose *Re-Select*.
4. Enter the following data:
   - User ID
   - Date
   - Time interval

5. Choose *Execute*.
6. Select an error message.
7. To display the application log of SAP Gateway, choose *Application Log*.

Related Information

Logging and Tracing for Launchpad and Launchpad Designer [page 401]

1.3.8.5 Displaying Client-Side Configuration (Deprecated)

You can quickly view the SAP Fiori launchpad configuration on a specific client.

Prerequisites

The SAP Fiori launchpad configuration is available in the SAPUI5 diagnostics window only when an SAP Fiori launchpad has been loaded.

Procedure

1. To display the diagnostics window, in the SAP Fiori launchpad, press `Ctrl` + `Shift` + `Alt` + `S`.
2. Open the *SAP Fiori Launchpad Configuration* section.

   You can search for parameter names.

   The parameters are displayed in a tree structure. You can filter the tree by clicking on the categories that are displayed below the search field. These categories correspond to the first-level entries in the tree.

> ### i Note
>
> Please note that this feature is deprecated and will be removed in a future version. SAP recommends using logging mechanisms instead (for details see Logging and Tracing for Launchpad and Launchpad Designer [page 401]).
>
> This tool displays the startup configuration. Any changes that occurred after startup are not reflected.

> This tool displays the complete startup configuration, including parameters that might be deprecated or that might be subject to change. For a list of supported parameters, see Launchpad Configuration Parameters [page 29].

1.3.9 Translating Launchpad Texts

The translation of texts appearing in the SAP Fiori launchpad (such as titles of tiles, spaces or groups) follows the standard ABAP translation process.

Text Type	Translation Process
Texts maintained in the launchpad designer (object type `WDCC`): texts in catalogs, tiles and groups	Translate Texts from Launchpad Designer (Customizing Scope) [page 408] Translate Texts from Launchpad Designer (Configuration Scope) [page 410]
Texts maintained in the launchpad app manager (object type `UIAD`): texts in catalogs and launchpad app descriptor items	Translate Texts from Launchpad App Manager [page 411]
Texts maintained in Manage Launchpad Spaces and Manage Launchpad Pages: space titles, page titles and section titles in pages	Translate Spaces and Pages [page 412]

Related Information

Launchpad Content Checks [page 378]
Analyzing Launchpad Texts [page 389]
Translate ABAP Texts in SAP Fiori

1.3.9.1   Translate Texts from Launchpad Designer (Customizing Scope)

You want to translate launchpad texts such as the titles of tiles, catalogs or groups in the customizing scope. This is possible using the standard SAP translation tools.

Context

The following procedure is relevant for objects of type `WDCC` in the customizing scope. You can determine the storage location using *Fiori Launchpad Checks* (transaction `/UI2/FLC`) or *Fiori Launchpad Texts* (transaction `/UI2/FLT`). Check the value of the *Object Type* field.

> ⚠ Caution
>
> To translate texts, only start SAP Fiori launchpad in the original language. Otherwise, you receive an error message when saving the configuration.

Procedure

1. To find the relevant texts in the translation system, determine the configuration ID and the text ID. See Determining IDs to Find Texts in the Translation System [page 391].

2. On the front-end server, start *Initial Screen: Translation Editor* (transaction SE63).

3. Choose ▶ *Translation* ❯ *ABAP Objects* ❯ *Short Texts* ❮.
   The *Object Type Selection (Object Groups)* dialog box appears.

4. Expand the *00 Meta Objects* node and select the meta object type *TABL Tables*.

   The *ABAP Short Texts: Tables (Meta)* screen appears.

5. Use the value help to search for **WDY_CONF_USERT2** in the *Object Name* field.

6. Select the required original and target languages.

   The language of the transported table entries is defined in the data dictionary as a property of the table. The texts you create will only be transported if the language is the same as defined in the data dictionary.

7. Choose *Edit*. Enter the following data in the table that appears:
   a. Enter *A* for the *User Scope* to change only the texts for objects in the Customizing scope.
   b. Enter **07** as the *Configuration Type* to limit the result set to texts relevant for the SAP Fiori launchpad.
   c. Enter the configuration ID you copied before (see step 1).

8. Choose *Execute*.

   This takes you to the short text editor. See Short Text Editor.

   To find the relevant text in the result list, search for the text ID you copied before (see step 1).

9. (Optional) After the translation is completed, an administrator needs to synchronize the user-independent cache to make the changes visible in the launchpad and in the design-time tools. To do this, call the transaction /UI2/SYNC_PBC with recommended log level 1. See Scheduling the Synchronization of User-Independent Cache [page 28].

Results

Your custom tiles, catalogs, or group titles are displayed with the intended text in SAP Fiori launchpad.

> ℹ Note
>
> For more information on how to transport launchpad texts, see SAP Note 980626 ➦.

Related Information

1.3.9.2 Translate Texts from Launchpad Designer (Configuration Scope)

You want to translate launchpad texts such as the titles of tiles, catalogs or groups in the configuration scope. This is possible using the standard SAP translation tools.

Context

The following procedure is relevant for objects of type `WDCC` in the configuration scope. You can determine the storage location using *Fiori Launchpad Checks* (transaction `/UI2/FLC`) or *Fiori Launchpad Texts* (transaction `/UI2/FLT`). Check the value of the *Object Type* field.

> ⚠ Caution
>
> To translate texts, only start SAP Fiori launchpad in the original language. Otherwise, you receive an error message when saving the configuration.

Procedure

1. To find the relevant texts in the translation system, determine the configuration ID and the text ID. See Determining IDs to Find Texts in the Translation System [page 391].

2. On the front-end server, start *Initial Screen: Translation Editor* (transaction `SE63`).

3. Choose ▌▶ *Translation* ❯ *ABAP Objects* ❯ *Transport Object* ▌.

4. In the *Transport Object* field, enter the following:

   `R3TR WDCC <CONFIG_ID>07`

5. Select the required source and target languages.

6. Choose *Edit*.

7. Expand the *<TLGS> Lockable Logical Objects (Short Texts)* node.

8. Select the entry with table `WDY_CONFIG_COMPT`.

   This takes you to the short text editor. See Short Text Editor.

   To find the relevant text in the result list, search for the text ID you copied before (see step 1).

9. (Optional) After the translation is completed, an administrator needs to synchronize the user-independent cache to make the changes visible in the launchpad and in the design-time tools. To do this, call the transaction /UI2/SYNC_PBC with recommended log level 1. See Scheduling the Synchronization of User-Independent Cache [page 28].

Results

> **i** Note
>
> For more information on how to transport launchpad texts, see SAP Note 980626 .

Related Information

Translation Tools for Translators (BC-DOC-TTL)
Internationalization and Translation (in the context of developing Web Dynpro ABAP applications)
Translate Texts from Launchpad Designer (Customizing Scope) [page 408]
Analyzing Launchpad Texts [page 389]

1.3.9.3 Translate Texts from Launchpad App Manager

You want to translate texts maintained in the launchpad app manager (such as the titles of tiles or catalogs). This is possible using the standard SAP translation tools.

> **i** Note
>
> The following procedure is relevant for texts maintained in the launchpad app manager (objects of type UIAD). You can determine the object type using *Fiori Launchpad Checks* (transaction /UI2/FLC) or *Fiori Launchpad Texts* (transaction /UI2/FLT). Check the value of the *Object Type* field.

Launching the Translation Editor from the Launchpad App Manager

1. Open the launchpad app manager. See Running the Launchpad App Manager [page 139].
2. Search for the relevant catalog or launchpad app descriptor item and select it.
   See Searching for Catalogs and Launchpad App Descriptor Items [page 140].
3. Choose *Translate* to open the selected object for translation in the short text editor (transaction SE63). Source and target language are already set.
   Choose *Translate with Language Selection* to open the selected object in *Translation: Transport Object* (transaction SE63). Select the required source and target languages.

4. After the translation is completed, an administrator needs to synchronize the user-independent cache to make the changes visible in the launchpad and in the design-time tools. To do this, call the transaction /UI2/SYNC_PBC with recommended log level 1. See Scheduling the Synchronization of User-Independent Cache [page 28].

Finding Objects in the Translation Editor

If you already know the ID of the object to be translated (catalog ID or launchpad app descriptor item ID), you can search for the object in the Translation Editor as follows:

1. On the front-end server, start *Initial Screen: Translation Editor* (transaction SE63).
2. Choose ▶ *Translation* ❯ *ABAP Objects* ❯ *Transport Object* ▶.
3. In the *Transport Object* field, enter one of the following:
   R3TR UIAC <CATALOG_ID>
   R3TR UIAD <LAUNCHPAD_APP_DESCRIPTOR_ITEM_ID>
4. Select the required source and target languages.
5. Choose *Edit*.
   This takes you to the short text editor. See Short Text Editor.

1.3.9.4 Translate Spaces and Pages

You can translate your spaces and pages with the SAP standard translation tools.

To translate the space title and description, the page titles, descriptions and the section titles in the pages, you need to translate the ID entries in transaction SE63.

Select ▶ *Translation* ❯ *ABAP Objects* ❯ *Short Texts* ▶. Expand *A5 User Interface Texts* and select UISC or UIPC:

- To translate spaces enter the object type UISC and then the space ID in the field *Object Name*.
- To translate pages enter the object type UIPC and then the page ID in the field *Object Name*.

When your source texts are in English, select *enUS* as source language and your desired target language. Otherwise select your source language.

See Translating Launchpad Texts [page 408] for more information on how to translate e.g. tile titles if necessary.

1.4 User Guide

This guide provides you, the end user, with information on how to optimally use the SAP Fiori launchpad.

This image is interactive and links to the main sections of the user guide. Hover over each area for a description. Click the highlighted areas for more information.

About the Launchpad	Use the Launchpad	Adapt Apps	Adapt the Launchpad
Get to know the UI elements and their functions	Learn how to perform tasks in the launchpad	Personalize your app or adapt the user interface for all users	Personalize your launchpad according to how you work
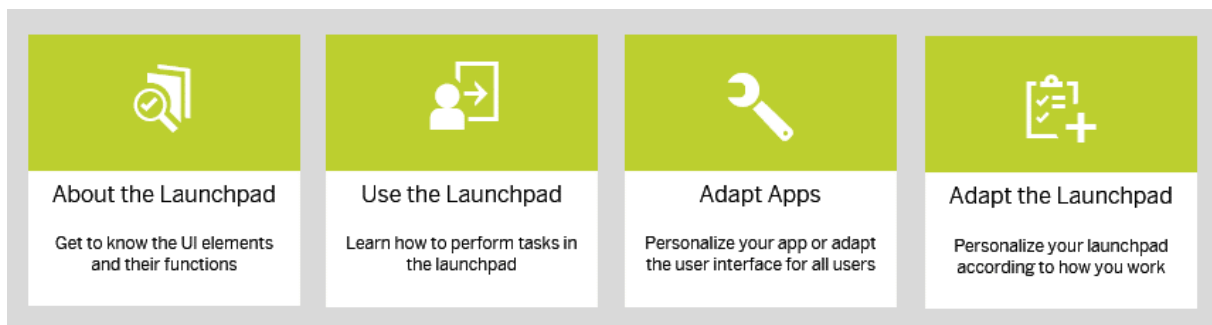
- About the Launchpad [page 413]
- Using the Launchpad [page 429]
- Personalizing and Adapting Apps [page 458]
- Personalizing the Launchpad [page 496]

Tutorial

For new users we recommend the SAP Fiori Launchpad tutorial. In this five-minute tutorial you learn how to work with the launchpad and get to know the most important features.

1.4.1 About the Launchpad

The SAP Fiori launchpad is a real time, role based, and personalized environment where all your SAP Fiori applications run. Navigation and search features ensure that you can quickly and effortlessly carry out your daily tasks.

When you first log on to the launchpad, the home page opens displaying predefined groups of apps configured by your administrator. You can personalize your launchpad by adding more groups and apps, removing those that you don't need, and reorganizing the groups according to the way you want to work.

Want to know more about the launchpad?

Hover over each area in the image for a description, click highlighted areas for more information.
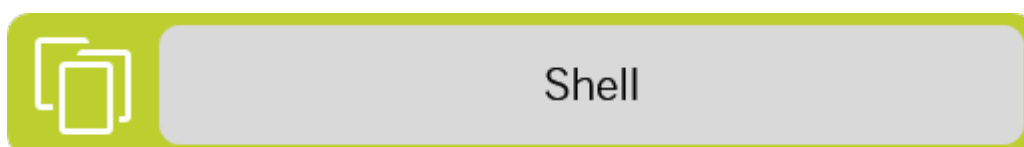
Shell
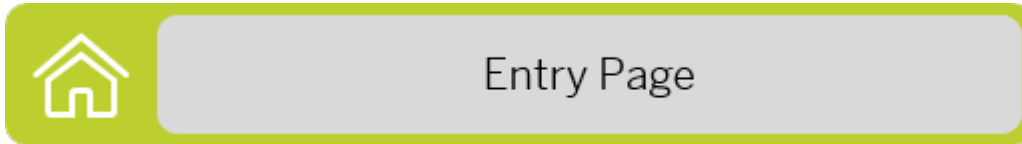
Entry Page
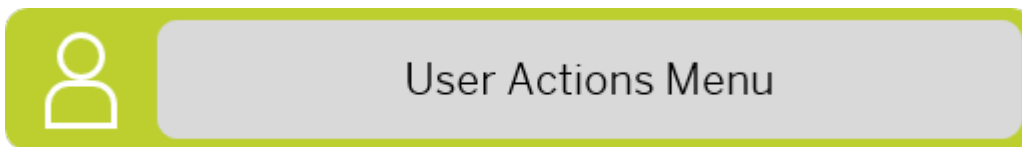
User Actions Menu
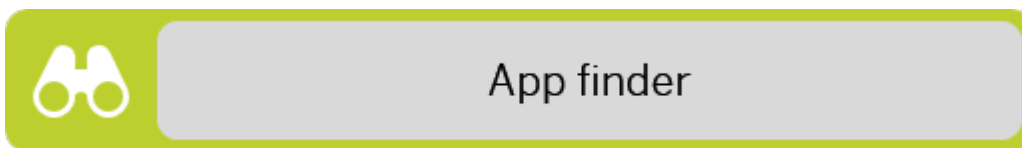
App finder

1.4.1.1   Shell and Shell Bar

When you launch apps from the SAP Fiori launchpad, the apps run within a shell. The shell is the outer frame of the SAP Fiori launchpad.

The shell contains a shell bar on top and footer areas with various buttons, icons, and a logo. A default footer is not provided out-of-the-box with SAP Fiori launchpad.

What will I see in the default shell bar?

The following table lists the icons available in the shell bar of the launchpad.

> **i Note**
>
> Some of the options described here and the buttons you see in the shell bar will vary according to the platform you are using and the features that are enabled. Your administrator may have moved some of the

> actions normally found in the user actions menu to the shell bar. The shell bar may also contain icons for custom plug-ins that have been developed by your own organization.
>
> The icons described in the table below may also change depending on where you are working in the launchpad. For example, you may see the back button when navigating in an app, but not when you are in the home page.

Icon	Name	Description
SAP	Company or brand logo	Displays the default SAP logo that you can replace with your own logo using the UI theme designer. If you click on it, you return to the home page, if you are not already on the home page. On a smaller screen this might be replaced by a home icon.
		If you work with spaces and pages, clicking on the logo will open the first page in your first space.
‹	Back	Navigates from an app back to the previous screen where you were working.
◎	Digital Assistant	Launches the digital assistant.
		See: Working with the Digital Assistant [page 441]
?	Help	Opens the in-app help. The help panel appears with help content or, additionally, further information on how to find help. You can also use F1 to open the help.
🔍	Search	Opens a global search user interface for finding apps and objects. On larger screens, the search field is directly shown centered in the shell bar.
		See: Searching for Business Objects and Apps [page 433] for more information.
✉	Contact Support	Allows you to contact the support directly from the launchpad.
		See: Contacting Support [page 442].
🔔	Notifications	Opens the Notifications window with the alerts, tasks, or requests that are related to your business role.
		The icon also shows a badge with the number of new notifications you have received.
		See: Working with Notifications [page 438]
○○○	More options	This icon is displayed when there is not enough space to show all icons in the shell bar. Click on it to see all hidden icons. If the notifications icon is hidden, this icon shows the number of new notifications.

Icon	Name	Description
(person icon) or **JP**	User Actions Menu	This icon usually show the initials of your user name. It opens the user actions menu.

> **i Note**
>
> This icon is replaced with your profile picture if SAP Jam is integrated in your environment and you have a picture defined in SAP Jam.

See: User Actions Menu [page 424]

In the middle of the shell bar, you'll see either *Home* (when you have the home page displayed) or the title of the current app you are using. You can click *Home* ∨ to directly open the All My Apps navigation option, or click the app's title to open the hierarchical navigation menu. The hierarchical navigation menu provides you with several quick actions depending on the context you are in, such as:

- Navigate within an app.
- See a history of your recent steps that have brought you to the current view.
- Access apps and objects that are related to the current app.
- Navigate back to the home page.
- Launch the All My Apps navigation option.

For more information, see Navigating Within and Between Apps [page 436].

1.4.1.2   Entry Page

The entry page is the heart of the SAP Fiori launchpad. It is where you open your SAP Fiori apps.

What will you see in the entry page?

The entry page is your starting point. It's the main area where you work and where tiles, representing the apps you work with, are clustered. When you first log on to the launchpad, you'll see the apps and groups or sections that have been preconfigured for you. How the apps are displayed here depends on your launchpad configuration. Depending on your system setup, there are two general options available:

- the spaces layout: you have one or more spaces, that contain one or more pages. The pages show your apps clustered in different sections (see Spaces and Pages [page 419]). In spaces mode, a personalized start page is available (depending on the system configuration). My Home [page 420] is a page, where you can add your preferred apps.
  The spaces layout was introduced to provide more flexibility regarding which apps are shown on a page. Your administrator can now assign pages defined for specific user groups delivered by SAP to you or they can create own user-group specific pages. By this, a page should show less apps that are more precisely adapted to your needs.
- the classic home page layout: there is one page that contains your apps clustered in groups (see Home Page [page 422]).

Pages and Sections (Groups)
Contain tiles and links that represent the apps in your launchpad.

Tiles
Represent individual apps and display information about your apps (such as title, KPI, and description of your app).

Links
An alternative visualization of apps. Useful if you need direct access to apps without having to display information about the app in a tile.
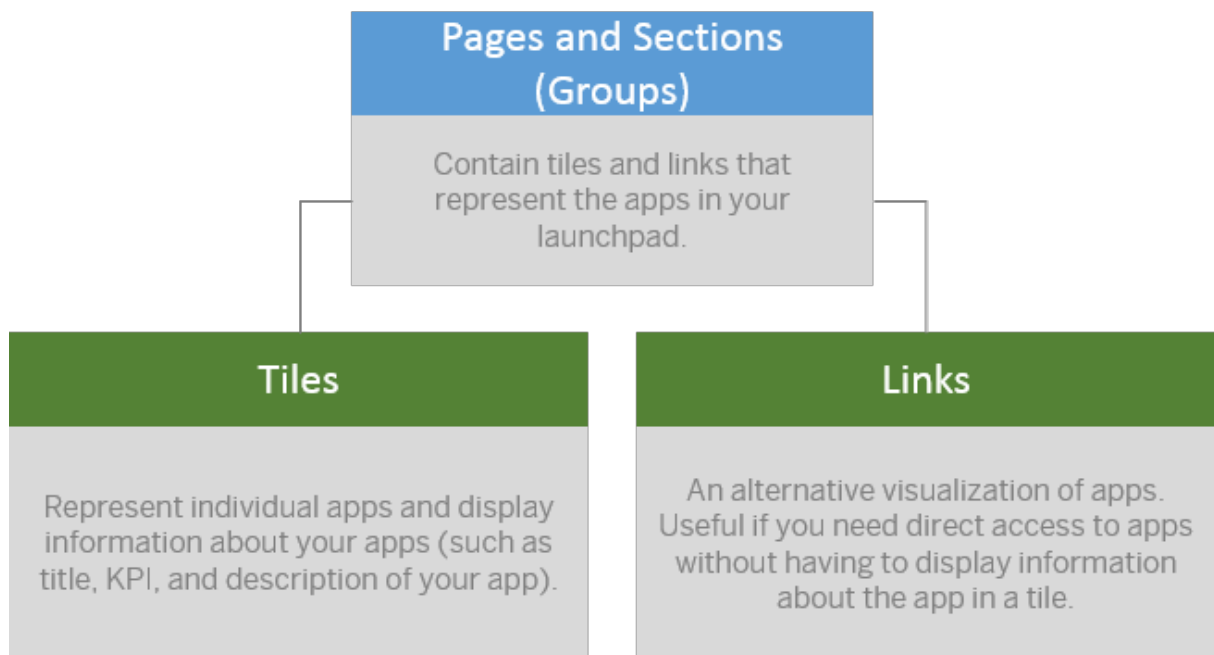
- Home Page [page 422]
- Home Page [page 422]
- Home Page [page 422]

What you see on the entry page may also change depending on the device that you are using. For example, if you are using a mobile device, you won't see apps on your launchpad that do not support mobile devices.

Differences between the display options

In general, the handling is the same, the new spaces layout offers most of the options you already know from the home page. Yet, there are some differences:

- The most important one is that the navigation bar on top shows the groups in the home page mode but the space name in the spaces mode. In the home page mode you go to a group on the home page when clicking on a name. In the spaces mode you go to a different page in another space if you have more than one assigned space.
- There are more tile sizes available in the spaces mode. A page in the spaces mode can display tiles in four different sizes and as links. Read for more information.
- There are some changes in the personalization options. See sections Personalizing the Home Page [page 501] and Personalizing a Page [page 496] for detailed information.

How to switch between the display options?

How the apps are displayed is set by your administrator. You may be able to decide which display you prefer if your administrator has enabled this option for you. Go to the *Settings* dialog and enable (or disable) the spaces mode. When you change the status and save your changes, the launchpad will be restarted. See Managing Your Settings [page 509] for more information.

Navigating in the entry pages

As mentioned above, the navigation is different in the two layout options.

- In the home page you click on an entry to go to the group of apps with that name.
- In the spaces mode there are different possibilities, depending how the spaces and pages are configured for you. When a space only has one page, you go to this page when you click on the space name. When the space contains several pages, a down arrow is displayed next to it. You open the first page when you click on the space name. Click on the arrow icon next to the space to open the menu with all pages. Click on a page name to open this page.

Open apps from the launchpad

Click a tile to open an app. You can also use the right-click menu to open an app in a new browser tab or window.

When you open an app in a new browser tab or window, it opens in **lean mode**:

- The *Notifications* and *Search* icons are not displayed in the shell header, and are only available in the main launchpad page.
- The *Back* button only appears after further in-place navigation.
- The *Home* entry is not displayed in the navigation hierarchy in shell navigation and *Related Apps*.

> **i Note**
>
> If some apps cannot be loaded, this might be related to a browser setting. If you work e.g. with a Safari browser and have enabled the option *Cross Site Tracking Prevention*, loading of an app may be prevented. Disable this setting to ensure that all apps can be loaded correctly.
>
> In the home page: If you see a tile with and error "Cannot load tile", you can click on it to see more information why the tile was not loaded. Clicking on *Show Details* opens more technical information. It should help your administrator finding the cause for this error.

Personalize your entry page to suit your working environment

There are many other personalization options available to help you adapt your launchpad to your working environment, such as dragging and dropping tiles or links to rearrange your home page or a page. In the

user actions menu, you can access a number of tools and useful links that allow you to personalize your launchpad. For example, depending on your role, you can access the app finder and choose from a wide array of ready-to-use apps and add them to your home page or a page.

For information about more personalization options, see Personalizing the Home Page [page 501] and Personalizing a Page [page 496].

1.4.1.2.1 Spaces and Pages

The launchpad page is the central part of the launchpad. Here you access your SAP Fiori apps.

A space is the unit that holds one or more pages. It is assigned to you based on your work profile (user role). You may see several spaces in your launchpad. The spaces are displayed in the navigation bar where you can switch between the different available spaces and pages.
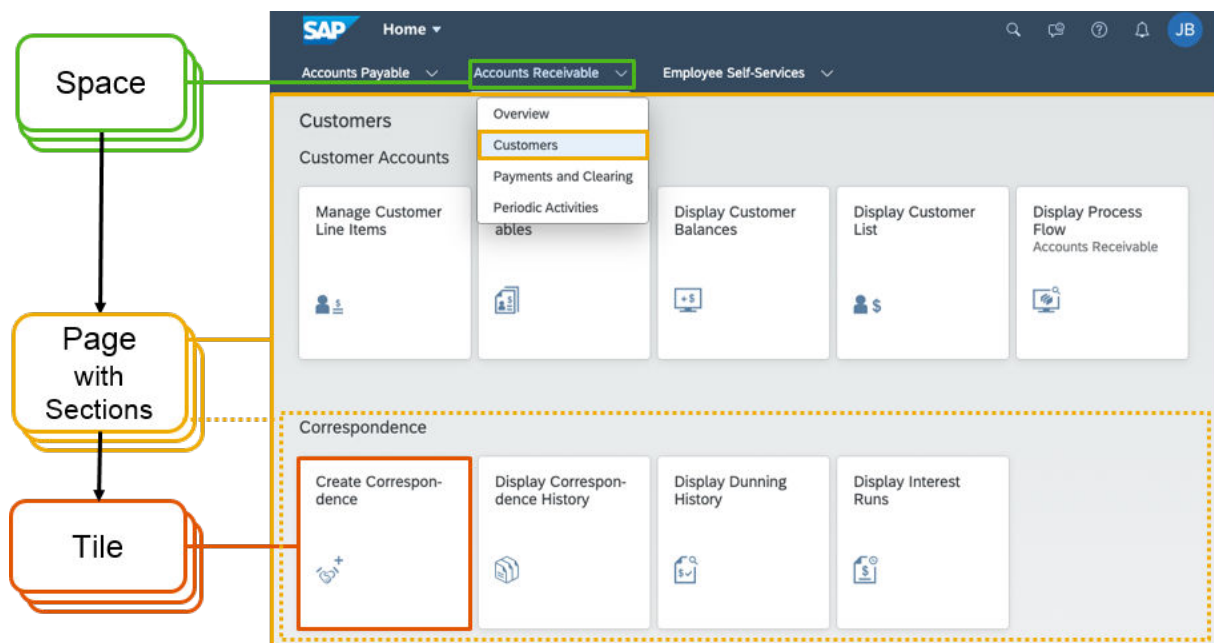
A schematic display of spaces and pages

A page is the part of the space that contains the apps as tiles and links clustered into different sections. It is shown in the main area of the launchpad. When you log on to the launchpad, you'll see the apps and sections that have been preconfigured for you. The page is your central starting point in the launchpad and should show you the apps that are relevant for the context of your daily work. Note that the pages do not show all apps you can access based on your role and the catalogs assigned to the roles, but only a selection your administrator made available. You can access all apps you are eligible to use by searching for them in the search field, in the app finder or in the All My Apps menu.

The navigation bar above the page shows the space name. If you have several spaces assigned you can click on the arrow next to a space to navigate to the page that belongs to the space.
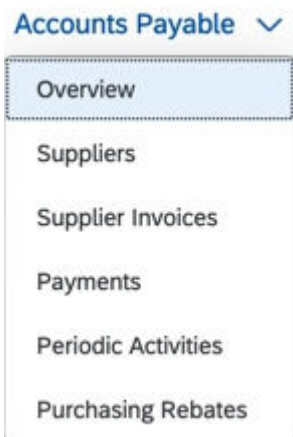
On top of the of a page a *Recently Added Apps* section may be shown. It is displayed if you added apps using the app finder or by bookmarking them and shows you the apps you just added to the launchpad page (see Adding Apps to a Page [page 498] for more information).

What you see on the page may also change depending on the device that you are using. For example, if you are using a mobile device, you won't see apps on your launchpad that do not support mobile devices.

Tutorial

For a quick overview on what's new or different with spaces and pages you can use the tutorial Working with and Personalizing Spaces and Pages .

1.4.1.2.2   My Home

My Home is your personal space and page, where you can add your preferred apps.

On My Home you can for example show the apps you use frequently, or apps that you quickly want to access but which are not on your assigned pages. You can also add bookmark tiles here.
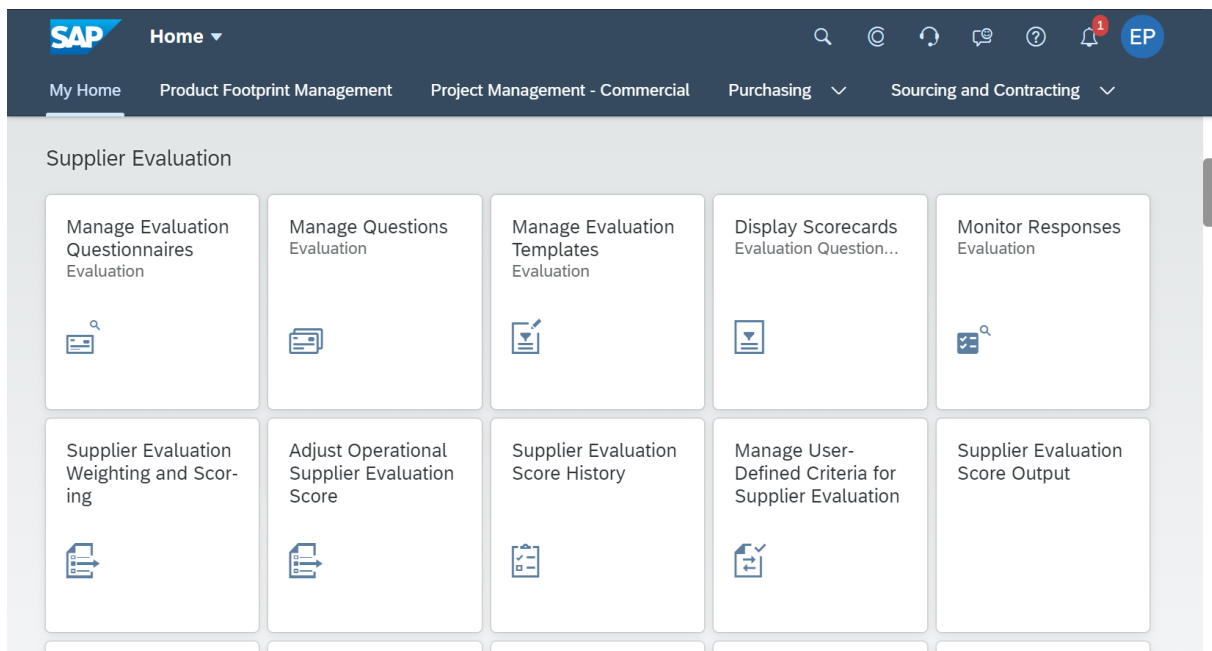
Example of My Home for a purchaser

My Home is only shown when you work with Spaces and Pages [page 419], and it can be disabled by your administrator. If you don't want to work with My Home, you can disable it yourself in the User Settings (see Managing Your Settings [page 509]). When My Home is active, it is always shown as the first space in the navigation bar. When you click on *Home* in the shell bar, you return to My Home.

When you first open My Home, it is empty. Depending on your prerequisites, you have three or four options to add your apps:

- Import personalized content from the classic home page. Read Importing Apps and Personalizations from the Classic Home Page to My Home [page 499] to learn when this option is available and how to use it.
- Edit My Home and add apps. This works the same way as editing any page and is described in Personalizing a Page [page 496].
- Select a page and add one or more apps from this page to My Home. Open the page, select *Edit Current Page* in the user actions menu. Then select the option *Add to My Home* in the tile context menu (∘∘∘).
- Adding apps from any page to My Home is also possible when personalization is disabled by your administrator. Go to the page, open the user actions menu and select *Add Tiles to My Home*. Then open the context menu of the app you want to add to My Home and select *Add to My Home*. See Personalizing a Page [page 496] for more information.
- Add apps with the bookmarking options (see Share Apps and Create Bookmarks for Apps [page 507])

> → Tip
>
> When all options are available, you can start by importing your personalized content from the classic home page and then add specific other apps. SAP suggests that you only import your most important apps to get a rather lean page.
>
> When you structure your page, focus on these most important apps as you probably have the apps for your daily work already assigned to your pages. Also read Best Practices for Managing Spaces and Pages [page 117] for tips on how to structure your page.

1.4.1.2.3 Home Page

The SAP Fiori launchpad home page is one of the entry pages for the launchpad - it's where you access your SAP Fiori apps.

What will I see on my home page?

When your administrator has set it, the home page is your starting point (see Entry Page [page 416] for more information on the available entry pages). It's the main area where you work and where tiles or links, representing the apps you work with, are clustered into groups. When you first log on to the launchpad, you'll see the apps and groups that have been preconfigured for you by your administrator. The launchpad is role-based so you see only those groups, tiles, and links that are assigned to your role.

On top of the groups the Featured Group is displayed if it is enabled by your administrator. This is a locked group showing two lists with your recent activities and frequently used apps. See section Working with Recent Activities and Frequently Used Apps [page 431] for information, what apps or activities are shown in the lists.

What you see on the home page may also change depending on the device that you are using. For example, if you are using a mobile device, you won't see apps on your launchpad that do not support mobile devices.

Navigate within the home page and to other areas in the launchpad

The groups by which your tiles and links are clustered are listed in the group selection bar at the top of the home page. Use this bar to quickly navigate to groups of apps.
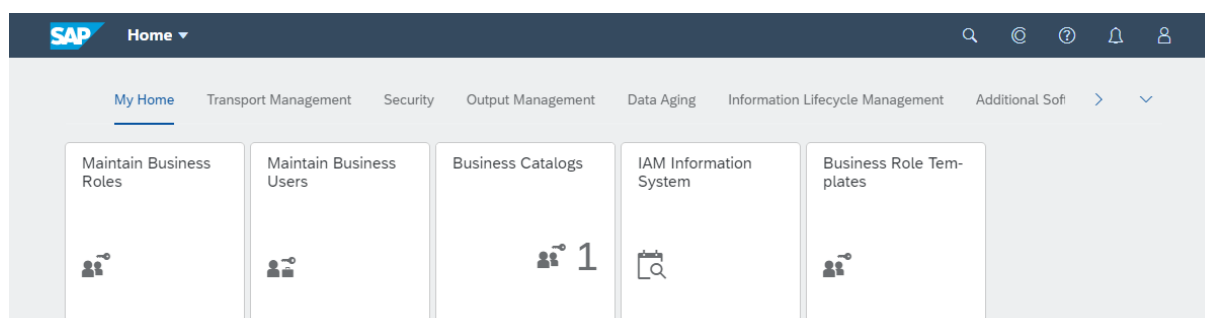
A sample home page of the SAP Fiori launchpad, showing the group selection bar just below the shell bar, and apps displayed as tiles that are assigned to the selected group

Depending on how the launchpad is configured by your administrator, you may see the content of all of your groups on the home page at once, or only one group at a time.

- If the former option is configured, the group selection bar behaves as an anchor bar; in other words, choosing a group in the bar scrolls the page to that group.
- If the latter option is configured, the group selection bar behaves as a tab bar; in other words, choosing a group displays the content of the selected group.

> **→ Tip**
>
> You might be able to switch between these two display options. For more information, see Managing Your Settings [page 509].

For information about navigation, see Navigating Within and Between Apps [page 436].

Personalize your home page to suit your working environment

There are many other personalization options available to help you adapt your launchpad to your working environment, such as dragging and dropping tiles or links to rearrange your home page. In the user actions menu, you can access a number of tools and useful links that allow you to personalize your launchpad. For example, depending on your role, you can access the app finder and choose from a wide array of ready-to-use apps and add them to your home page.

For information about more personalization options, see Personalizing the Home Page [page 501].

Related Information

Shell and Shell Bar [page 414]
User Actions Menu [page 424]
App Finder [page 427]

1.4.1.2.4   Tiles, Links and Tile Sizes

Tiles can be displayed in different sizes in the SAP Fiori launchpad.

In general, tiles are shown as square tiles or links. When you work with spaces and pages, two additional tile sizes are supported. By choosing a larger tile size, you e.g. display more information at a glance.

The following tile sizes are available:

- Tile: 2x2 square tile (supported on page and home page)
- Flat tile: 2x1 (supported only on a page)
- Flat wide tile: 4x1 (supported only on a page)
- Wide tile: 4x2 , e.g. a smart KPI tile (supported on page and home page)
- Custom tiles: defined by customers; have a specific tile size, not always available (supported on page and home page)
- Links: small tiles showing only the title and subtitle, e.g. for a compact layout (supported on page and home page)

My Apps		
Communication Systems	Communication Arrangements	Monitor Employee Imports
Maintain Communication Users	Manage Archiving Variants	Business Event Subscription
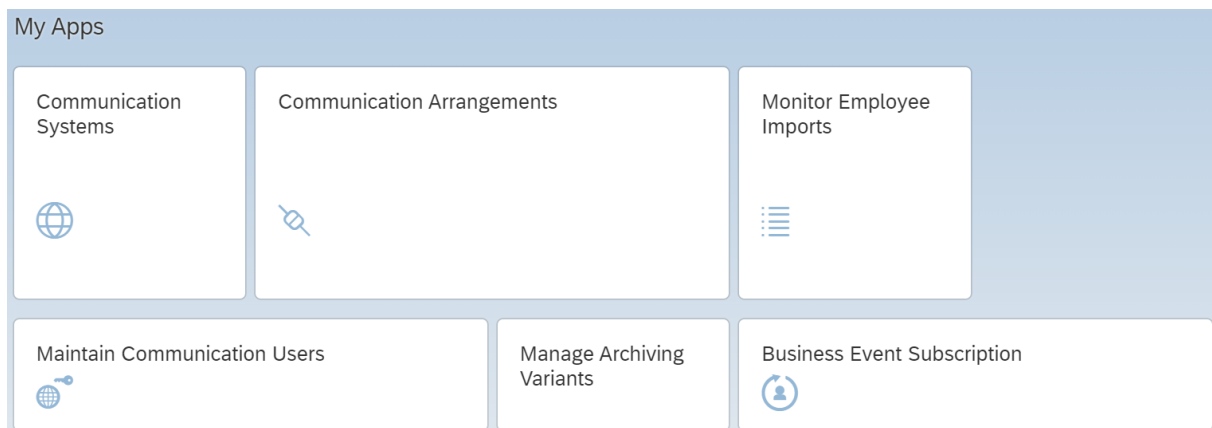
In general, there is a specific tile order within one section: Tiles of the same height are shown in the same row or several rows directly one below the other. Tiles in standard size are shown first, flat and flat wide tiles are shown below the standard-sized tiles. Links are always shown below the tiles.

When you move a tile to another row with tiles in a different size, this will change the tile size. How to change the tile size is described in Personalizing a Page [page 496] in detail.

1.4.1.3 User Actions Menu

The user actions menu offers user-related options e.g. to open apps, to change the launchpad layout or to contact support.

To open the user actions menu, click your profile picture, your initials (e.g. JP) or the 👤 *(User Actions Menu)* icon located at the top right corner of the shell header. Your profile picture appears only if SAP Jam integration is enabled and you have uploaded a picture in SAP Jam.

What can I do in the User Actions Menu?
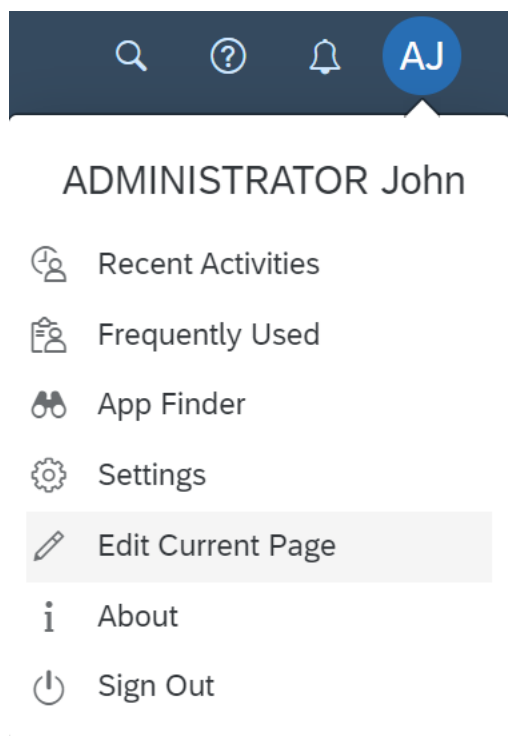
> **i Note**
>
> Some options listed here are available only if available on your platform or enabled by your administrator.
>
> Your administrator may also have moved some of the options from the user actions menu to the shell header.

The menu gives you direct access to the following features. At the top of the menu you can see your user name.

Feature	Description
Recent Activities	Displays a Quick Access dialog with an interactive history of the items you have searched for in Enterprise Search and worked with recently. The apps or objects are listed chronologically. When you have disabled the user activity tracking, this menu item is not shown. For more information, see Managing Your Settings. [page 509]
Frequently Used	Displays the apps you have searched for in Enterprise Search and used most frequently over the last 30 days. Only days working in the launchpad are counted. The apps are listed in order of usage frequency with the most used listed first. An app must be used at least twice to appear in the list. When you have disabled the user activity tracking, this menu item is not shown. For more information, see Managing Your Settings. [page 509]

Feature	Description
App Finder	A convenient tool for finding apps and adding them to your launchpad. For more information, see App Finder [page 427] and Adding Apps to the Home Page [page 504]. If you work with spaces and pages, the options are different. See Adding Apps to a Page [page 498] for detailed information.
Settings	A central area where you can view and maintain the settings for your launchpad. For more information, see Managing Your Settings [page 509].
Edit Home Page	Enables you to access the home page in edit mode so that you can organize your launchpad to suit your daily tasks. Once you have completed your editing, exit edit mode by clicking *Done* at the bottom of the home page or clicking *Exit Edit Mode* in the user actions menu. For more information, see Personalizing the Home Page [page 501].
Edit Current Page	**i Note** This option is only available if you work with spaces and pages, and if your administrator has enabled the personalization option. See Spaces and Pages [page 419] for details. Enables you to access the currently shown page in edit mode so that you can organize your launchpad to suit your daily tasks. Once you have completed your editing, exit edit mode by clicking *Done* at the bottom of the page or clicking *Exit Edit Mode* in the user actions menu. For more information, see Adding Apps to a Page [page 498].
Edit My Home	**i Note** This option is only available if you work with spaces and pages, and have My Home acitvated, and if your administrator has disabled the personalization option. When personalization is enabled, My home can be edited when displayed with the option *Edit Current Page*. Also see Spaces and Pages [page 419] for details. Enables you to access your personal My Home in edit mode so that you can organize My Home to suit your daily tasks. Once you have completed your editing, exit edit mode by clicking *Done* at the bottom of the page or clicking *Exit Edit Mode* in the user actions menu. For more information, see My Home [page 420] and Adding Apps to a Page [page 498].
Contact Support	Allows you to contact support directly from the launchpad. For more information, see Contacting Support [page 442].

Feature	Description
✏️ Personalize App	You can personalize an object page used in your app by adding, removing or rearranging sections. This is an experimental feature. For more information, see Personalizing Object Pages (Experimental) [page 494].
🔧 Adapt UI	With key user adaptation, you can change the user interface of SAP Fiori apps directly in the launchpad, intuitively and without having to write new code. For more information, see Adapting SAP Fiori UIs at Runtime - Key User Adaptation [page 459].
i Info	Display information for a specific app and the launchpad. For more information, see Information About App and Launchpad Version [page 443].
🕐 Input History Settings	Use this option to control what should happen with the data you entered in an app. When the history setting is enabled by your administrator, an app may save the last five values you have entered (only for supported apps). You can enable or disable the value history with the option *Track history of recently entered values*. You can also delete your history of saved values by clicking on *Clear*. Save your changes.
⏻ Sign Out	Use this option to end your session in the launchpad. For more information, see Signing In and Signing Out [page 431].
🩺 App Support	Use this option to troubleshoot configuration and authorization errors for apps in the launchpad. For more information, see App Support [page 393].

1.4.1.4   App Finder

The app finder is a convenient tool for finding apps, all in one place, and then adding them to your home page or to one of the pages.

The app finder lists all the apps available for your role that you may potentially use in your daily work. You can access the app finder from the User Actions Menu.

When you work with pages in the spaces mode, the handling is slightly different as you may have several pages assigned. We will show the differences below. See Entry Page [page 416] for more information.

What will I see when the app finder opens?

The app finder displays the following tabs where you can find all the apps available for your role:

Tab	Description
Catalog	A repository of all the apps that are assigned by your administrator to your role.
User Menu	A customized menu that includes folders of those SAP GUI transactions and Web Dynpro ABAP apps assigned to your role in a specific system.
SAP Menu	A standard menu delivered by SAP that includes folders of all SAP GUI transactions and Web Dynpro ABAP apps offered by a particular SAP system. This menu is not role-dependent and therefore displays the same content for all users.

The *SAP Menu* and the *User Menu* are also known as the SAP Easy Access menus.

When you open the app finder for the first time, the *Catalog* tab is in focus.

> **i Note**
>
> Depending on how the administrator has configured your launchpad, you may not see all of these tabs.
>
> If your administrator has disabled both the SAP Menu and the User Menu, then the tab control is completely hidden; the repository you see will be that of the catalog only (all the SAP Fiori apps that are assigned to your role).
>
> The SAP Easy Access menus are not supported on mobile devices.

What can I do with the apps in the app finder?

In the app finder, each app displays one of the following icons:

Icon	Description
📌 Add tile to groups	Indicates that the app has not yet been added to any group
📌 Add tile to additional groups or remove from a group.	Indicates that the app is already assigned to at least one group or page (if you work in spaces mode). You can click it to add the app to more groups or pages or to change the app's group assignment.
+ *Add* Add tile to a page	Indicates that the app has not yet been added to any page. Is displayed if you work in spaces mode.

Icon	Description
✔ *Add* Add tile to additional pages or remove from a page.	Indicates that the app is already assigned to at least one page if you work in spaces mode. You can click it to add the app to more pages or to change the app's page assignment.

The handling differs depending if you work with the home page or spaces and pages:

When you click these icons, a dialog box opens with a list of groups in your home page. You can select one of the groups or create a new group and add the app to it. For more information about using the app finder in the home page, see Adding Apps to the Home Page [page 504].

When you work in spaces mode, a list of the available spaces and the contained pages is shown. Select one or more pages. The app is added to a special section in the selected page: the *Recently Added Apps*. This section is shown as first section. It contains the apps you have added to the page with the app finder or as a bookmark. This is a locked section, so you can only drag apps from here to other sections but no apps to this section. When the section is empty, it will be deleted automatically. When you return to the page, you can drag the app to your preferred section. See Adding Apps to a Page [page 498].

1.4.2 Using the Launchpad

End users use the launchpad to launch their SAP Fiori apps. They can also add more apps to their home page, adapt the user interface, and change the look and feel of their launchpad to suit their work environment.

What can you do in the launchpad?

Hover over each area for details about what you can do in the launchpad, click highlighted areas for more information.

- Signing In and Signing Out [page 431]

Search for apps and business objects

- Searching for Business Objects and Apps [page 433]

Navigate between apps

- Navigating Within and Between Apps [page 436]

Work with notifications

- Working with Notifications [page 438]

Contact support

- Contacting Support [page 442]

Use launchpad accessibility features

- Usability Features in the Launchpad [page 445]

1.4.2.1 Signing In and Signing Out

When you want to work with the launchpad you have to be signed in.

When you open the launchpad URL, you are usually either signed in automatically via single-sign on or a sign-in screen is displayed. There you enter your user name and your password for the launchpad. You get user name and password from your adminstrator. Depending on your system configuration you can select other settings for start-up, e.g. the language.

After you have finished your work you should sign-off. Select *Sign Out* in the user actions menu. After sign-out a sign-out page or the start page is displayed.

Automatic Sign-Out when Inactive

For security reasons usually a time for automatic sign-out is set. Your adminstrator can define the time, when the sign-out should happen. When you have not been active in the system for this time, you are automatically signed out.

It can be configured that a few minutes before this timeout, a warning is displayed, informing you that you will be signed out soon. You can choose to stay signed in (this resets the timeout), or sign-out directly.

This warning is also shown, if you have switched to a different browser tab or minimzed the browser. The first time the warning should be shown in this case, you need to allow that the launchpad may show notifications in the browser. Note that this feature is not supported on mobile devices or in the Microsoft Internet Explorer.

1.4.2.2 Working with Recent Activities and Frequently Used Apps

Find your most used apps or open a recently used function in the Quick Access Dialog.

To access the lists, open the user actions menu and click *Recent Activities*. The Quick Access dialog shows you lists of your most used apps and the last functions you have used in the launchpad. Each list displays a maximum of 30 items. You can directly navigate to any of these apps or objects by clicking them.

> **i Note**
>
> This functionality is optional and may be disabled by your administrator. It is only available when you have enabled the track user activities options. For more information, see Managing Your Settings [page 509].

Recent Activities

You see an interactive history of the items you have searched for in Enterprise Search and worked with recently. The apps or objects are listed chronologically.

> **i Note**
>
> Apps and objects are only added to the *Recent Activities* list if you close them using the *Home* button or the back navigation button.

Frequently Used Apps

Select *Frequently Used* in the user actions menu or select the second tab in the Quick Access dialog. The dialog displays the apps you have searched for in Enterprise Search and opened or the apps used most frequently over the last 30 days. Only days working in the launchpad are counted. The apps are listed in order of usage frequency with the most used listed first. An app must be used at least twice to appear in the list.

> **i Note**
>
> You can clear the history or stop tracking your user activity at any time in the *Settings* dialog box.

You can also display cards in the Featured Group of the home page showing the recent activities or frequently used apps. The Featured Group needs to be enabled by your administrator. If you disable tracking your user activity, the cards and the respective menu items in the user actions menu will be hidden.

1.4.2.3 Searching for Business Objects and Apps

Search for business objects and apps, filter and save the search results.

> **i Note**
>
> To locate business objects using the search bar, Enterprise Search must be integrated and set up in your environment. Otherwise, the search returns apps that are located only in your home page and the catalog. The search result list contains all the business objects and apps you are authorized for. For example, you will find all the apps that you have access to via the business catalogs and business roles.

The search function is embedded in the shell bar. You start searching by clicking on 🔍 *Search* in the top right corner of the launchpad.

First you may filter what kind of results you want to see by choosing one of the shown options on the left of the search bar or selecting *All*.

Search Operators

For writing in the search bar you can use the following operators:

Operator	Example	Description
None	`shares warrants`	Finds results that contain both the word "shares" and the word "warrants".
`OR`	`shares OR warrants`	Finds results that contain either the word "shares" or the word "warrants".
`*`	`warr*`	Finds results containing words that start with "warr", for example "warrants", "warranty", and "warranted".
`-`	`shares - warrants`	Finds results that contain the word "shares" but not the word "warrants".
`""`	`"with best regards"`	Finds results that contain the exact phrase "with best regards".

If you can't find the expected results try again using *****, for example ***12345** or ***john***.

The search is case-insensitive.

Navigating in the Results

The result list is sorted by relevance automatically, which is calculated based on the following factors:

- Weights of attributes, for example, a hit in the surname is ranked higher than a hit in the first name.
- Number of hits within an object (proportional to the text length and frequency of other words).

Once the search results are shown, several icons in the top right appear, that enable you to work with the search results.

⤓ *Export Results*: By clicking on this icon you can download all of your search results (or the first 1000, if there are more than that) as a comma-seperated text file with an .xlsx extension which will open in the assigned application.

↑↓ *Sort*: You can choose the sort order to either be ascending or descending and also by which of the listed information you would like to sort (e.g.: as best match, bussiness partner role, phone number). The sort options will be processed as soon as you click *Okay*. Via *Reset* you will get to default settings.

↗ *Share*: You can save the search settings as tile or so called apps, which will then be pinned on your entry page or send them via e-mail. If you choose to ✦ *Save as Tile*, you can insert a *Title*, a *Subtitle*, a *Description* and assign it to a *Group*. This app will then be pinned as tile to your *Home Screen*. For more information see Adding Apps to the Home Page [page 504] or Personalizing the Home Page [page 501]. If you chose to send the search results via mail your e-mail application will open up and you only have to fill in the address of the recipient.

☰ ⊞ ⊚ *Choose Display*: The search results can be displayed as ☰ *List*, ⊞ *Table* or ⊚ *Map*.

For each result you can decide to show more details ⌄ or hide them ⌃ .

If your search results show employees, you can directly call or e-mail them by clicking on the dedicated information or click on the name to show their address on online map services.

Filtering Search Results

You can further filter the results shown by clicking ▽ *Show filters* in the top left. A panel on the left side of your search results opens, which shows different filter possibilities depending on your search request. You can scroll through this panel and select and deselect multiple filters.

- For some filter aspects you can toggle between ◕ *Display as: Pie Chart*, ⊨ *Display as: Bar Chart* and ☰ *Display as: List*.
- You reset all filters by clicking on ▽ₓ *Reset Filters* in the upper right part of the filter panel.

By clicking on the button *Show More*, you open a different view of the filter possibilities. The window that pops up contains the same icons and filters as the panel, but also additional filter features:

- *Define Conditions* lets you add one or more precise filter conditions by selecting *Equal to*, *Begins with*, *Ends with* or *Contains* and inserting letters or even whole words.

- You can add another condition by clicking on ✛ *Add* and delete an existing condition by clicking on
  ⊗ *Remove* next to the condition that you wish to delete.

Filtering by Company Names

Filters	🔽ₓ	Company Name	
Changed At			
City Name		Select Items	Define Conditions
Company Name	**5**	Begins with ⌄	a ⊗
Country		Equal to	Filter Condition ⊗
Created At		Begins with	
Postal Code		Ends with	
Street Name		Contains	

> **i Note**
>
> For this example the search results would show all *Company Names*, that start with an *a*.

Personalized Search

- By clicking on 👤 *Account* in the top right corner under ⚙ *Settings* you can decide what you want to happen with your search activity.
- Under 🖥 *User Activities* you can turn the tracking of your recent activities and frequently used apps on and off as well as clear the search history.
- Under 👥 *User Profiling* you can toggle between allowing personalized search or not.

> **i Note**
>
> ⚙ *Settings* also gives you Quick Access to 👥 *Recent Activities* and 👥 *Frequently Used*. For more information see Working with Recent Activities and Frequently Used Apps [page 431].

Other Features

You can also use the *App Finder* to find all your SAP Fiori, SAP GUI, and Web Dynpro ABAP apps, and then add them to your home page. For more information, see App Finder [page 427] and Adding Apps to the Home Page [page 504].

You can also use the *All My Apps* navigation option to quickly access your home page groups and apps assigned to catalogs, from anywhere in the launchpad. For more information, see Navigating Within and Between Apps [page 436].

You can personalize the launchpad, see .

1.4.2.4 Navigating Within and Between Apps

End users can easily navigate in the launchpad between apps and within apps.

SAP Fiori launchpad offers different navigation possibilities:

- Navigate within apps using hierarchy navigation.
- Navigate between apps from various points in the launchpad.

> i Note
>
> Some of these options may not be available to you. This varies according to the platform you are using and the features that are enabled.

Navigating within an app

Once you've launched an app you can navigate within the app itself using a convenient hierarchical navigation menu. To get to the navigation menu, simply click the app's title in the shell bar.

You will see a history of your recent steps in the context of the current app that lead you to the current view. The first entry in the hierarchical menu is always *Home*, and as you continue navigating to different views within the app, more entries are added to the top of the navigation path. When you work with spaces and pages, the page and space name on that you have opened the app are shown as second entry in the menu.
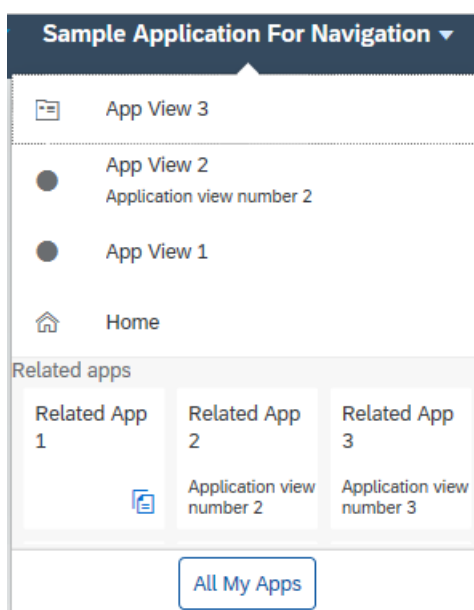
Navigating between apps

There are various ways that you can navigate between apps:

| Navigation option	More information
Related Apps	Located below the hierarchical navigation menu shown above.
	Navigate to apps and objects that are closely related to the current app.
All My Apps	Located in the hierarchical context menu of the shell bar when navigating within an app.
	Use the All My Apps navigation option to quickly access any of your role-based apps from anywhere in the launchpad, without having to open and scroll through the page. It is especially useful if you work with a large number of apps because it allows you to reach apps assigned to catalogs without first adding the apps to your home page or page if you work with spaces and pages.
	To access it:
	• When you are navigating within an app, click the app's title in the middle of the shell header, and then click *All My Apps* at the bottom of the hierarchical navigation menu.
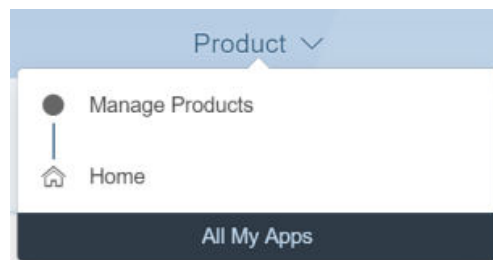
	• When you are in the home page, click *Home* ⌄ in the middle of the shell bar. Since there is no hierarchical navigation menu when you are in the home page, All My Apps is opened immediately.
	Click the structured entries to drill down into the content assigned to your home page or pages if you work with pages and spaces and catalogs. For example, the *Home Page Apps* entry provides you with quick access to your home page groups. Your assigned catalogs are listed below the *Home Page Apps* entry.

> ℹ **Note**
>
> Use the app finder instead of All My Apps in the following cases:
>
> • To access the SAP Easy Access menus (EAM) and custom tiles.
> • To access apps in hidden home page groups or hidden sections on a page.
> • To add apps to your home page or to a page in a space.
>
> Your administrator may disable access to either your home page apps or catalog content in the All My Apps navigation option. If, for example, catalog content is disabled, your home page groups are shown immediately without the first-level *Home Page Apps* entry.

Navigation option	More information
Recent Activities list	Located in the user actions menu.
	Click any app to navigate quickly to apps or objects that you recently worked with.
Frequently Used list	Located in the user actions menu.
	Click any app to navigate quickly to apps or objects that you use often.
‹ *(Back)*	Located on the left of the shell bar.
	Use it to navigate to the previous app (or object) that you opened.

1.4.2.5 Working with Notifications

SAP Fiori launchpad provides a Notifications window that lets you know about important tasks and requests requiring your timely action or knowledge. They allow you to view immediate updates on the latest and most important events that are related to your business role.

The launchpad consumes notifications from the notification channel. The notification channel is a backend component that aggregates notifications from the different notification providers configured in your environment. For some notification types you can select to receive notifications also on a mobile device or as e-mails.

> i Note
>
> This functionality is optional and must be set up by your administrator.

How do I view my notifications?

Click the ⏰ *(Notifications)* icon on the right-hand side of the shell bar.

Notifications icon in the shell bar and the Notifications window

→ Tip

The notifications icon in the shell bar also shows a badge with the number of new notifications you have received since you last opened the Notifications window. This number resets when you open the Notifications window.

If you receive new notifications while you have the Notifications window open, the badge on the notifications icon updates to show you how many new notifications you have received.

A popup alert notifies you when you receive a high-priority notification while you are working in an application or the app finder.

What can I see in the Notifications window?

The Notifications window lists all the notifications that are relevant to you and your business role. The type of notifications you receive also depends on which notification providers and channels are configured for you.In addition, you can choose yourself which notification types you want to see in your Notifications window.

- The content displayed in each notification depends on the type of notification and how the notification channel is configured. Typically a notification shows a short description, the name of the requester or sender, and a timestamp.
- Notifications that require your action display buttons that allow you to quickly process the notification.
- By default, your notifications are sorted by date from newest to oldest (newest at the top).
- Notifications are color-coded according to their level of priority (green=low priority, orange=medium priority, and red=high priority).
- Unread notifications are displayed in bold text.

What can I do in the Notifications window?

Sort the notifications	At the top of the Notifications window, choose *By Date*, *By Type*, or *By Priority*. → Tip Choose *By Date* again to toggle the sort order between ascending to descending, and back again.
Open the notification's corresponding application	Click on the notification. This opens the referenced business object in a SAP Fiori app that is configured for the notification.
Process or take action on a notification	Click the relevant action button displayed with each notification. If the action is successful, the notification is removed from the Notifications window. → Tip You can process all notifications of the same type together by first sorting the notifications by type. i Note In some cases, certain notifications are flagged in a way that they remain displayed in your Notifications window even after you have processed the notification.

| Close a notification | Click the notification's ✕ *(Close)* icon. |

> → Remember
>
> Closing a notification that is expecting an action from you will dismiss the notification without taking any form of action.

Which notification settings can I personalize?

In your *Settings*, you can personalize the following notification preferences:

- Choose the notification types you want to see in your Notifications window. For example, if you prefer not to see any leave requests in the notifications window, you can turn them off.
- Choose whether or not to be notified by a popup alert whenever you receive a high-priority notification.
- Set specific notification types to always be flagged as high priority notifications, regardless of the default priority level determined by the backend system.
- Specify which notification types trigger native notifications on your mobile device.
- Specify which notification types trigger an additional e-mail.

For more information, see Setting Notification Preferences [page 516].

1.4.2.6 Working with the Digital Assistant

SAP offers a contextual collaboration tool that sits on top of SAP Fiori-based apps, for example SAP CoPilot. Depending on which digital assistant is integrated, it is possible to gather and create artifacts such as notes, screenshots, and objects and place them in ad hoc collections, all in the context of the app that you are working in.

The digital assistant is triggered with the ◎ *(Digital Assistant)* icon in the SAP Fiori launchpad shell bar. The tooltip shows you which tool is integrated in your system.

For information about SAP CoPilot key functionalities, refer to the SAP CoPilot User Guide.

1.4.2.7 Working With SAP Mobile Cards

SAP Mobile Cards is an application that allows you to share content from SAP Fiori-based applications from your desktop computer's SAP Fiori launchpad to your Android phone or iPhone, which provides "any time from anywhere" access to the most up-to-date business data available.

Here's what needs to happen before you can use SAP Mobile Cards:

- You need to download the application from the Google Play Store (Android) or the Apple App Store (iOS).
- Your system administrator must complete the required back-end configuration.

- You or your system administrator must configure the SAP Mobile Cards application for use on your device. See "Registering the Application" in the User Guide (link below) for more information.

Once SAP Mobile Cards is configured, you can begin sharing content from your SAP Fiori launchpad to the application on your mobile device.

To add a card to the card deck:

1. Launch an application that supports SAP Mobile Cards. If you aren't sure which apps are supported, please contact your system administrator.
2. Select the content you want to add to the application on your mobile device.
3. In the top-right of the shell bar of the launchpad, select 👤, then select *Add to Mobile*. When you see "Successfully added Card", it means the content has been added to your mobile device.

> **i Note**
>
> To share content from Overview applications, select an overview app, then select *Add to Mobile*, then choose the content from the overview app that you want to share to your mobile device.

For more information about using SAP Mobile Cards, see SAP Mobile Cards for Android User Guide or SAP Mobile Cards for iOS User Guide.

1.4.2.8 Contacting Support

End users can contact support from the launchpad.

Prerequisites

To receive support, SAP Solution Manager must be part of the system landscape.

Context

When the feature for contacting support is active, technical information, such as navigation data, errors, user details, and server details, is collected while you work. If you choose to contact support, from the user actions menu or from an error message, you can enter a free text to describe the problem. The free text and the technical data are used to create a support ticket (for example an incident in SAP Solution Manager).

Procedure

1. In the user actions menu, select *Contact Support*; or in an error message, choose the *Contact Support* button.

2. Enter a short summary of your issue in the field *Subject* and add a more detailed description of the problem in the field *Description*.

   Choose *Show Technical Data* to view some of the technical data that will be appended to the ticket. You can see further installation information in the *About* dialog (for further information, see Information About App and Launchpad Version [page 443]).

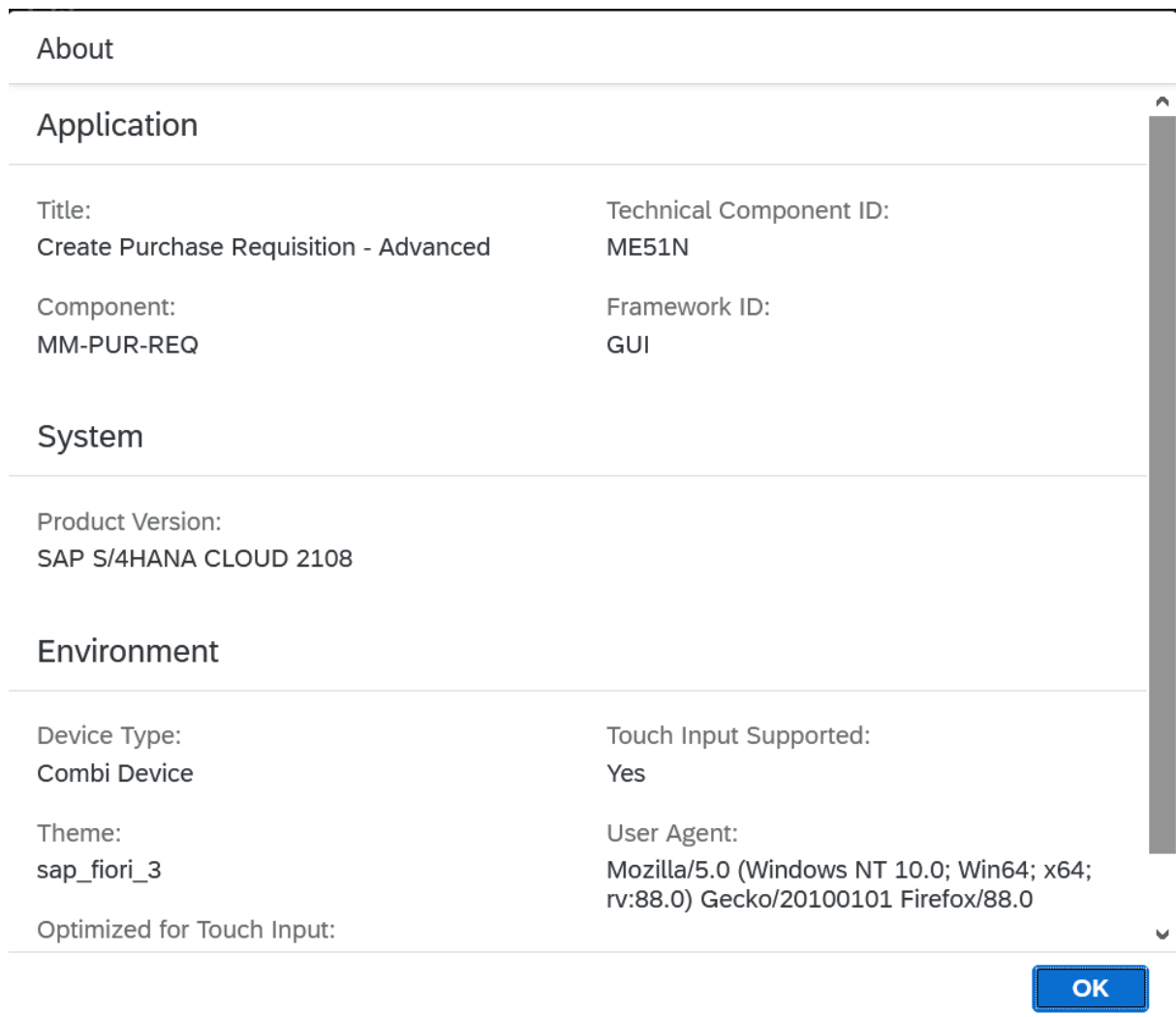3. Choose *Send* to create the support ticket (for example, an incident in SAP Solution Manager).

Related Information

User Actions Menu [page 424]

1.4.2.9 Information About App and Launchpad Version

Display information for a specific app and the launchpad.

Click on *About* in the user actions menu.

About

Application

Title:
Create Purchase Requisition - Advanced

Technical Component ID:
ME51N

Component:
MM-PUR-REQ

Framework ID:
GUI

System

Product Version:
SAP S/4HANA CLOUD 2108

Environment

Device Type:
Combi Device

Touch Input Supported:
Yes

Theme:
sap_fiori_3

User Agent:
Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:88.0) Gecko/20100101 Firefox/88.0

Optimized for Touch Input:

OK

About dialog with technical information about app and launchpad

You see a dialog showing technical information for the selected app, the system and the environment. The information is mostly used for support. The figure shows example information for the Create Purchase Requisition app. For other apps, different fields may be displayed. A field is not shown if no information is available.

Field	Description
Application	
Title	App title
Technical Component ID	Technical name of the opened app
Component	Application component (ACH) of the opened app
Framework ID	Framework ID on that the opened app is based on (e.g. SAPUI5, SAP GUI etc.)

Field	Description
Framework Version	Framework version on that the opened app is based on
Version	Version of the opened app
System	
Product Version	Version of the used product, e.g. the SAP NetWeaver version.
Product Name	Name of the used product (defined by SAP or customer)
System Role	Shows the system in which the launchpad runs (e.g. test or production, defined by SAP or customer)
Tenant Role	Shows the role of the tenant in the system in which the launchpad runs (e.g. production or test client, defined by SAP or customer).
Environment	
Device Type	Device type of the used device
Touch Input Supported	Indicates if the app supports touch input.
Optimized for Touch Input	Indicates if the app is optimized for touch input.
Theme	Shows which theme is currently selected for the launchpad.
User Agent	Information on the used browser.

1.4.2.10 Usability Features in the Launchpad

SAP Fiori launchpad includes usability features, such as keyboard-only navigation, screen reader support, and high contrast themes to assist you while working in the launchpad. Users seeking information about accessibility may also find some valuable information here.

> **i** Note
>
> Depending on how your administrator has configured your launchpad environment, some of the screens and features described in the following topics may not be available to you.

Related Information

1.4.2.10.1 Keyboard Navigation in the Home Page

To navigate in the launchpad using the keyboard instead of the mouse, you use a combination of the `Shift`, `F6`, and `Tab` keys.
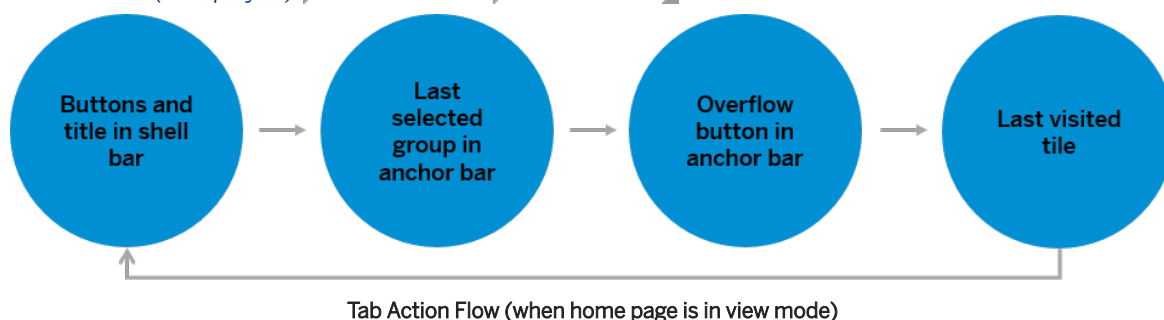
Navigation Flow

- The forward navigation of the `F6` flow is: ▶ *First clickable icon in shell bar* ❯ *Currently selected group in group selection bar on the home page* ❯ *Last visited tile* ❯ *Close button (only in edit mode)* ❯ *First clickable icon in shell bar* ❯

First clickable icon in shell bar → Currently selected group in anchor bar → Last visited tile →(Edit home page)→ Close button (edit mode only)

F6 Action Flow
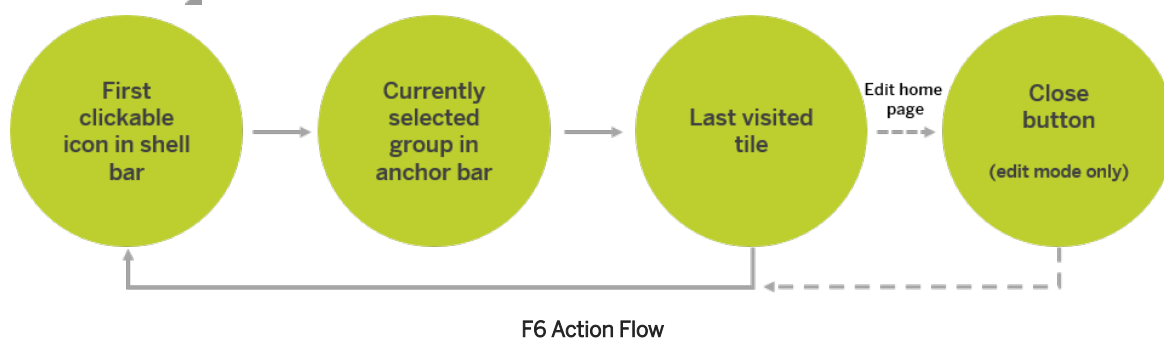
Pressing `Shift` + `F6` reverses this flow. The first clickable item can be for example the SAP logo, a company logo or the *Back* button.

-

- The forward navigation of the `Tab` flow is: ▶ *Browser fields (for example tab or navigation bar)* ❯ *First icon in shell bar* ❯ *Last selected group in the group selection bar on the home page* ❯ *Overflow icon in the group selection bar (if displayed)* ❯ *Last visited tile* ❯ *Browser field* ❯.

Buttons and title in shell bar → Last selected group in anchor bar → Overflow button in anchor bar → Last visited tile

Tab Action Flow (when home page is in view mode)

You may need to first press `Enter` or the `Spacebar` to open or display the selected user interface element.

The sections that follow describe how you navigate inside these launchpad elements with the keyboard.
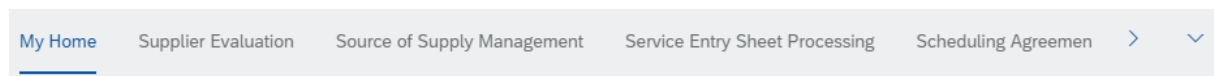
Navigating in the Home Page

The home page displays groups of tiles and links. When you first open your launchpad, the first tile on your home page is in focus. You can navigate between the tiles and groups using your keyboard as follows:

Keys	Behavior
`Left arrow`, Right arrow, `Up arrow` and `Down arrow`	Navigate between tiles.
`Home`	Navigate to the first tile in the same group.
`End`	Navigate to the last tile in the same group.
`Page down`	Navigate from any tile in a group to the first tile in the next group.
`Page up`	Navigate from any tile in a group to the first tile in the previous group.
`Home` and `Ctrl`	Navigate to the first tile in the first group.
`End` and `Ctrl`	Navigate to the last tile of the last group.
`Enter` or `Spacebar`	Launch the selected tile.

Navigating in the Group Selection Bar

Apps are clustered in groups on the home page. The bar at the top of the home page displays quick links to all of your groups. Depending on how your administrator has configured the bar, it may behave as an anchor bar or a tab bar.

| My Home | Supplier Evaluation | Source of Supply Management | Service Entry Sheet Processing | Scheduling Agreemen | ❯ | ⌄ |

Example of the group selection bar in the launchpad

Use your keyboard as follows to navigate within the group selection bar:

Keys	Behavior
`Enter` or `Spacebar`	Select the group in the group selection bar. When you press the key, the focus is on the first tile in the group.
`Tab` or `F6`	Navigate from the group selection bar to the tiles area. The focus is on the first tile in the group.

Navigation in the Home Page Edit Mode

When you have opened the Home Page in edit mode, you can also navigate with `F6` and `Tab`. The navigation with `F6` has the following flow:

▐▶ *Currently selected group in group selection bar on the home page* ❯ *Last visited tile* ❯ *First clickable icon in shell bar* ❯ *Currently selected group in group selection bar* ❯ *Last visited tile* ❯ *Close button* ❯ *First clickable icon in shell bar* ▐

It is the same as in the home page view with an additional step for the *Close* button (see image above).

The navigation with `Tab` has the following steps:

▐▶ *Browser tab* ❯ *Browser navigation bar* ❯ *First clickable icon in shell bar* ❯ *Next icon in shell bar* ❯ *Currently selected group in group selection bar on the home page* ❯ *Group Title* ❯ *Reset button or Delete button (if*

available) 〉 *First tile in group* 〉 *Next tile* 〉 *Add Group button* 〉 *Next group title etc.* 〉 *Close button* 〉 *Tab icon in browser header bar* 〉

Keyboard Shortcut: `F2` (only in a group title in edit mode): Enter new group title. `Enter` confirms the title change; `Esc` cancels the title change. Click `Tab` to confirm you action and move to the next element.

Related Information

Shell and Shell Bar [page 414]
Home Page [page 422]
User Actions Menu [page 424]
App Finder [page 427]

1.4.2.10.2 Keyboard Navigation with Spaces and Pages

To navigate in a launchpad page using the keyboard instead of the mouse, you use a combination of the Shift, F6, and Tab keys.
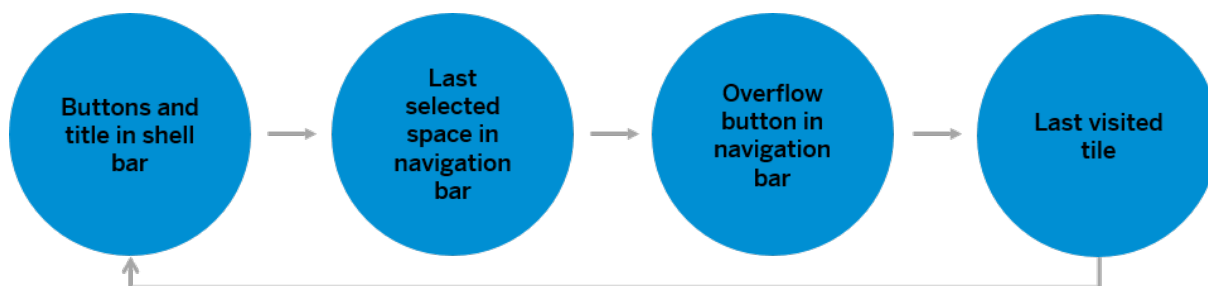
Navigation Flow

When you work with pages and spaces, navigation with `F6` is: 〉 *First clickable icon in shell bar* 〉 *Currently selected space in the navigation bar* 〉 *Last visited tile* 〉 *Close button (only in edit mode)* 〉 *First clickable icon in shell bar* 〉

Pressing `Shift` + `F6` reverses this flow.

When you work with spaces and pages, the `Tab` navigation flow is: 〉 *Shell bar icons and title* 〉 *Last selected space in the navigation bar* 〉 *Overflow icon in the navigation bar (if displayed)* 〉 *Last visited tile* 〉 *First icon in shell bar* 〉

Pressing `Shift` + `Tab` reverses this flow.

When you edit a page, `Tab` navigation also includes the first section header, the section title, and then the action buttons (*Show*, *Hide*, *Reset*, or *Delete*) after the overflow button in the navigation bar. Before going back to the shell bar icons, the focus is set on the *Close* button.

You may need to first press `Enter` or the `Spacebar` to open or display the selected user interface element.

The sections that follow describe how you navigate inside these launchpad elements with the keyboard.

Navigating in the Page

The page displays sections with tiles. When you first open your launchpad, the first tile on your first page is in focus. You can navigate between the tiles and sections using your keyboard as follows:

Keys	Behavior
`Left arrow` or `Right arrow`	Navigate between tiles in the same section.
`Home`	Navigate to the first tile in the same section.
`End`	Navigate to the last tile in the same section.
`Page down`	Navigate from any tile in a section to the first tile in the next section.
`Page up`	Navigate from any tile in a section to the first tile in the previous section.
`Home` and `Ctrl`	Navigate to the first tile in the first section.
`End` and `Ctrl`	Navigate to the last tile of the last section.
`Enter` or `Spacebar`	Launch the selected tile.

1.4.2.10.3 Keyboard Navigation in Notifications and App Finder

Learn how to navigate in the Notifications area and the app finder.

Navigating in the Notifications Area

Use your keyboard as follows to navigate and use the Notifications area (if it enabled in your system):

Keys	Behavior
`Tab`	Navigate between the elements in the Notifications window in the following order: 1. *By Date* sort option 2. First or last selected notification or notification group 3. *Close* button in the selected notification or notification group 4. Action buttons, such as *Approve* and *Deny*, for the selected notification or notification group
`Left arrow` or `Right arrow`	Navigate between the sort options (*By Date*, *By Type*, *By Priority*) in the header bar of the Notifications area.
`Enter` or `Spacebar`	Execute the selected sort option or press the selected button in a notification or notification group. When a notification is selected, pressing `Enter` opens the application related to the notification.
`Up arrow` or `Down arrow`	Navigate between the individual notifications or notification groups. → Tip To move to the next or previous item in the list, the focus must be on the whole notification item or group, not on one of its buttons.
`Esc`	Close the notification window.

Navigating in the App Finder

Use your keyboard as follows to navigate in and use the app finder:

Keys	Behavior
`F1`	Open the in-app help if it is enabled by your administrator. The help carousel appears with help content or, additionally, further information on how to find help.

Keys	Behavior
`F6`	Pressing `F6` enables you to navigate in the app finder *Catalog* tab in the following order: 1. *Catalog* tab 2. First item on the left 3. First tile in the first group Pressing `F6` enables you to navigate in the *User Menu* or *SAP Menu* tabs as follows: 1. *User Menu* or *SAP Menu* tabs 2. *Select System* dropdown list 3. First folder in the list on the left side of the screen 4. Title of first folder
`Tab`	Navigate from items in the left-hand column to the first tile. Navigate from an app to the ✧ *(Add tile)* icon to add an app to a group.
`Up arrow` or `Down arrow`	• Navigate between items in the left panel. • Navigate between apps.
`Page Up` or `Page Down`	Move to the first tile in the next group.
`End` or `Home`	Move to the first or last tile on the page.

1.4.2.10.4 Keyboard Shortcuts

You can use the following keyboard shortcuts in the launchpad.

Use `Ctrl` + `F1` to view a popup window that lists the available keyboard shortcuts to quickly navigate to different areas in the launchpad:

Keys	Behavior
`Alt` + `H`	Set the focus on either the logo or the home button.
`Alt` + `A`	Set the focus on the app finder menu entry.
`Ctrl` + `S`	Set the focus on the search bar in the app finder (works only when the app finder is available, already open and the catalog search is enabled by the administrator).
`Ctrl` + `Enter`	Save changes when editing the home page or the current page.
`Alt` + `F`	Set the focus on the search icon.

Keys	Behavior
`Alt` + `M`	Set the focus on the user actions menu icon.
`Alt` + `N`	Set the focus on the Notifications menu entry.
`Alt` + `S`	Set the focus on the *Settings* menu entry.
`Ctrl` + `Comma`	Display the *Settings* dialog box.
`Ctrl` + `Shift` + `F` (`Cmd` + `Shift` + `F` on Mac)	Set the focus on the search entry field.

1.4.2.10.5 Adapting the Home Page with the Keyboard

You can move tiles or edit the layout of the home page using the keyboard.

> **i Note**
>
> Remember that you can't edit locked groups.

Moving Tiles or Links in the Home Page

You can can move a tile or a link to a new position. Set the focus on the tile or link, then press `Ctrl` and hold it. Use the left or right arrow keys to move the tile or link to a new position in the group. Use the up or down arrow keys to move a tile or link to a different group. It is possible to change a tile into a link by moving it into an available link section. You can also convert a link into a tile.

Working in Edit Mode

You can edit the home page of your launchpad by choosing the *Edit Home Page* entry in the user actions menu. Once you have completed your editing, exit the edit mode by choosing *Close* or choosing *Exit Edit Mode* in the user actions menu.

Keys	Behavior
1. Click `Alt` + `M` to open the user actions menu. 2. Select the *Edit Home Page* entry in the user actions menu with the arrow up or down key. 3. Press `Enter` or `Spacebar`.	Switch the home page to edit mode.

The following actions apply only after you have switched to edit mode.

1. Use the `Up arrow` or `Down arrow` until you get to the *Add Group* button. 2. Press `Enter` or `Spacebar`.	Add a new group.

Keys	Behavior
1. In the group to which you want to add a new tile, select *Add Tile*. 2. Press `Enter`.	Add a new tile.
1. Get to the tile you want to move. 2. Press `Enter` to open a list of options. 3. Use the `Up arrow` or `Down arrow` to get to the *Move* option and press `Enter`. 4. Select the group to which you want to move the tile using the `Up arrow` and `Down arrow` keys and then press `Enter`.	Move tiles to another group.
1. Select the tile you want to move. 2. Use `Ctrl` + `Up arrow` or `Ctrl` + `Down arrow` to move the tile one row up or one row down. Or use `Ctrl` + `Left arrow` or `Ctrl` + `Right arrow` to move the tile horizontally one position at a time.	Move tiles within a group. > **i Note** > If the tile you are moving is at the beginning or or end of a group, it moves to the previous or next group.
1. Select the tile that you want to remove from a group. 2. Press `Delete`.	Remove tiles from a group.
1. Navigate to the header of the group. 2. Press `Tab` until you get to the *Delete* button. 3. Press `Spacebar` or `Enter` to delete.	Delete a group.
1. Select the title of the group. 2. Press `Tab` until the text of the group name is in focus. 3. Press `F2` to change the name.	Rename a group.
1. Select the title of the group. 2. Press `Tab` until you get to the *Reset* button. 3. Press `Spacebar` or `Enter` to reset the group.	Reset a group.
1. Select the title of the group. 2. Press `Tab` until you get to the *Show* or *Hide* buttons. 3. Press `Spacebar` or `Enter`.	Show or hide a group.

1.4.2.10.6 Adapting a Page with the Keyboard

You can move tiles or edit the layout of the selected page using the keyboard.

> **i Note**
>
> This option is only available if you work with pages and spaces (see Entry Page [page 416] for additional information). The spaces mode does not support all features known from the home page, e.g there are no links or locked groups and tiles cannot be added in Edit Mode, only with the app finder. If you work with the home page, see Adapting the Home Page with the Keyboard [page 452]

Moving Tiles in a Page

You can move a tile or link to a new position. Set the focus on the tile or link, then press `Ctrl` and hold it. Use the left or right arrow keys to move the tile or link to a new position in the section. Use the up or down arrow keys to move a tile or link to a different section.

Working in Edit Mode

You can edit the currently shown page of your launchpad by choosing the *Edit Current Page* entry in the user actions menu. Once you have completed your editing, exit the edit mode by choosing *Close* or choosing *Exit Edit Mode* in the user actions menu.

1. Click `Alt` + `M` to open the user actions menu. 2. Select the *Edit Current Page* entry in the user actions menu with the arrow up or down key. 3. Press `Enter` or `Spacebar`.	Switch the home page to edit mode.

The following actions apply only after you have switched to edit mode.

1. Use `Tab` until you get to the *Add Section* button. 2. Press `Enter` or `Spacebar`.	Add a new section.
1. Get to the tile you want to move. 2. Press `Enter` to open a list of options. 3. Use the `Up arrow` or `Down arrow` to get to the *Move* option and press `Enter`. 4. Select the group to which you want to move the tile using the `Up arrow` and `Down arrow` keys and then press `Enter`.	Move tiles to another section.
1. Select the tile you want to move. 2. Use `Ctrl` + `Left arrow` or `Ctrl` + `Right arrow` to move the tile horizontally one position at a time.	Move tiles within a section. > **i Note** > > If the tile you are moving is at the beginning or or end of a section, it moves to the previous or next section.

1. Select the tile that you want to remove from a section.	Remove tiles from a section.
2. Press `Delete`.	

1. Navigate to the header of the section.	Delete a section.
2. Press `Tab` until you get to the *Delete Section* button.	
3. Press `Spacebar` or `Enter` to delete.	

1. Select the title of the section.	Reset a section.
2. Press `Tab` until you get to the *Reset* button.	
3. Press `Spacebar` or `Enter` to reset the section.	

1. Select the title of the section.	Show or hide a section.
2. Press `Tab` until you get to the *Show* or *Hide* buttons.	
3. Press `Spacebar` or `Enter`.	

1.4.2.10.7 Screen Reader Support in the Launchpad

The screen reader reads out the contents of the user interface to you. Labels, headings, and descriptions help describe the contents and visual elements of an application.

Screen reader support in SAP Fiori launchpad is based on built-in keyboard navigation and roles, and the names and states of functional elements and areas. For example, in the launchpad the screen reader announces:

- Menu options and how many there are.
- Which tiles and which groups are in focus.

> ⚠ Caution
>
> Screen readers are 3rd party software not affiliated with or related to SAP products. SAP cannot grant that statements about specific behavior will always work with any combination of 3rd party software, or take responsibility for respective issues in 3rd party software. Therefore, the descriptions in this topic serve only as a means for orientation and basic information. SAP encourages all users to experiment with the most recent 3rd party software combinations that yield the best result for their needs.

General Information

> i Note
>
> Before operating the launchpad with a screen reader, we recommend that you familiarize yourself with launchpad concepts and structure.

Information about the launchpad areas and functional elements on a page is given in the following ways:

- When focusing on an element using built-in keyboard navigation
- With structural navigation using virtual navigation features of screen readers.

> **i Note**
>
> Virtual navigation is in-page navigation using special screen reader keys to read non-interactive page content such as text and information about page structure and contained elements.

Some page content, such as plain page text and images, can only be accessed using virtual navigation.

Orientation

Use virtual modes (special screen reader modes for reading non-interactive page content) and landmark listings (lists of areas that form the page structure) of screen readers to get an orientation of the hierarchical launchpad structure and the content of the different areas. Some areas are only available for virtual cursor investigation (using special screen reader modes for reading attributes of page content) when they are expanded using the respective launchpad buttons.

Identify and remember typical elements inside these areas (such as the first named button, the first element in a named list) to help you access them later when you use the built-in keyboard navigation features.

> **i Note**
>
> - Headings and their hierarchies (such as of tile groups and other regions) can be accessed using navigation by heading features in virtual modes (hierarchy information such as paragraph or region heading levels)
> - Some pages are divided in subsections (such as regions and forms). Their names and hierarchies typically show up in landmark listings of the page
> - Other elements show up in respective element listings (such as lists of buttons, lists of links)

Navigation and Interaction

We recommend that when using a screen reader you should navigate to and use functional elements in non-virtual modes. You may need to switch to virtual modes occasionally to access non-focusable page content.

There are three types of content in the different areas of the launchpad:

1. Focusable single elements such as buttons or input fields. Navigate between elements with the `TAB` key.
2. Focusable items in lists of elements such as the list of tile groups or list of tiles within a group. Navigate between elements with the arrow keys.
3. Non-focusable single elements such as tile group headings, images, or additional text. Access using virtual cursor functions.

There are also modal dialog boxes such as the *Settings* dialog box typically invoked as functions of respective buttons.

> **i Note**
>
> - List items in the launchpad are typically active. Their activation triggers changes as part of the respective page or opens up dialog boxes or new page areas.
> - Launchpad tiles are named active list items. Their activation triggers entire page changes by default.

Messages

Additional messages for users are announced in dialog boxes with respective titles and content. You may need to switch to virtual mode to access non-focusable content in these dialog boxes. Short informational messages that do not require user acknowledgment are announced automatically if the screen reader software can detect them.

Potential Issues

- There can be conflicts with the native keyboard bindings of screen readers and launchpad key bindings. In such cases, use the key bypass functions of your screen reader to trigger the respective launchpad key bindings.
- There may be unwanted automatic mode switching in screen readers while using the launchpad or apps. Make sure that you detect these automatic mode changes and switch to non-virtual modes for correct keyboard navigation and operation.
- Virtual modes of screen readers may not be able to detect modal dialog boxes and the reading focus may escape from the dialog box to the page background. If this happens, reopen the dialog box and listen to any indication of dialog box boundaries when navigating in virtual mode.
- Screen reader support in the launchpad is designed to work out of the box. Some important information for functions may only be available in descriptions and tooltips of elements. Make sure that the respective features of the screen readers are switched on, or that you can access them if needed.
- If you encounter issues with screen reader speech output, make sure that:
  - You are using an updated combination of browser and screen reader.
  - If needed, any special support for screen readers is switched on in your browser.
  - No personalization, customization, or extra scripts that could influence speech output are running.

Related Information

1.4.2.10.8 Themes

Various themes are predefined to assist users with various color and vision preferences.

Usually, you can choose between the SAP default theme family *Quartz* (which supports dark and light settings), the SAP theme family *Horizon* (experimental, supports dark and light settings in combination with high contrast), the *SAP Belize* themes in dark and light, or the high-contrast themes *High Contrast Black* and *High Contrast White*. In addition, your administrator may provide a custom theme, set a different default theme or restrict which themes are available for you.

Which themes are available, also depends on your personal system settings. When you have enabled the automatic dark mode detection in the launchpad and use a Horizon or Quartz theme, a combined theme for light and dark theme is displayed. It is called for example *SAP Horizon*. According to your color settings in your operating system, the light or the dark theme is used. For example, when you have selected a Horizon theme and your color settings in your operating system are set to "dark", the Horizon Evening theme is set. This also applies to the Horizon or Quartz High Contrast Dark and Light themes. For information about changing the theme in SAP Fiori launchpad and the dark mode settings, see Managing Your Settings [page 509].

Related Information

Managing Your Settings [page 509]

1.4.3 Personalizing and Adapting Apps

When running an app in the SAP Fiori launchpad, end users can personalize object pages and key users can adapt the user interface for all users of the app (for example, a team lead can add a field that's then available for all team members).

How can you personalize and adapt apps?

Hover over each area in the table for a description, click for more information.

End users: Personalize apps for yourself
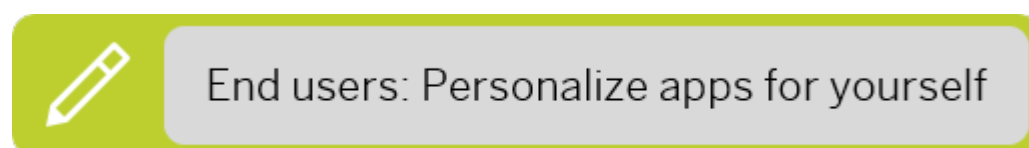
- Personalizing Apps [page 487]

Key users: Adapt app UIs for others

-

1.4.3.1 Adapting SAP Fiori UIs at Runtime - Key User Adaptation

With key user adaptation, you can change the user interface of SAP Fiori apps directly in the launchpad - intuitively and without having to write new code.

Adapting SAP Fiori UIs can be this simple

Support special business cases, speed up workflows, improve usability - there are many reasons why you may want to adapt the user interface (UI) of your apps. With key user adaptation, this couldn't be easier. It's highly intuitive and requires no technical or programming skills. Just use drag and drop or the context menu to make UI changes, for example rearrange fields or groups, remove them or add additional ones. Check out our overview video.

As opposed to personalization, the changes you make with key user adaptation apply to the entire app and therefore to all app users. You can even publish your UI changes as a separate app, a so-called app variant, and make this app variant available to other users.

Wait ... What if I delete something important or even break the app?

Don't worry! Key user adaptation comes with a built-in safety net. It informs you whenever you're about to do something that's worth double-checking, for example removing mandatory fields from the UI. You can also undo your changes and even discard them all and reset the UI to the default app.

> → Tip
>
> Want to play around with a demo app and check out how key user adaptation works? Just open the SAPUI5 Demo Kit. On the *Demo Apps* tab, choose one of the *SAPUI5 Flexibility Demo Apps*.

What are the requirements for using key user adaptation?

- You run your SAPUI5 app on a desktop computer or laptop.
- Your app is enabled for key user adaptation.
- Your administrator assigned you to the authorization role `SAP_UI_FLEX_KEY_USER` (see Enabling Key User Adaptation [page 75]).
- You use the new `FioriLaunchpad.html` page.

If in doubt, simply open your app and check if your user actions menu contains the *Adapt UI* option. Also check Something Isn't Working like It Should? [page 485].

How do I start key user adaptation in an app?

Simply go to the user actions menu and choose *Adapt UI*.

Related Information

1.4.3.1.1 Adapting the UI

When you are in the adaptation mode, you have many options to adapt the UI to your liking. You can also adapt and save certain views.

Prerequisites

You have started the UI adaptation mode by choosing *Adapt UI* in the user actions menu.

Caution: Don't mix systems or clients when you make UI changes

It's possible that you distinguish between test and productive environments in your organization. When you adapt the UI of an app, in this case it's important that you make the UI changes only in the test environment and then transport or copy them to your productive environment. For example:

- Adapt the UI in the test system **only** and transport the changes to the productive system.
- Adapt the UI in the test client **only** and copy the changes to the productive client.

Your productive system or client doesn't prevent you from making UI changes there, but it's important that you do not mix these scenarios for one app (that is, making changes in both the test and the productive environment). This would lead to severe UI adaptation conflicts.

If you don't have or don't use a test system or client, please do everything in the productive environment.

Overview

Header Bar
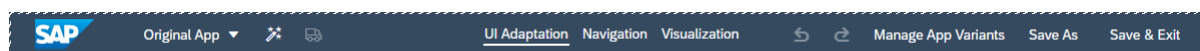
Version history	If the user interface of the current app has been adapted previously, you can switch to a recent version or to the original app. Just click on the title of the current version to expand the version history, then select a version from the dropdown list to display it. See Drafts and Versioning [page 476].
Activate New Version	Use ⚡ to activate the current draft or a selected version that is not currently active, so it becomes a new version.
Discard Draft	This option is only available when you display a draft that has not yet been activated. Use 🗑 to discard all UI changes that you did in draft mode. You can then start adapting the UI again, based on the last activated version.

Publish Version	This option is only available when you display an unpublished version. Use ⛟ to publish a version of a changed app or app variant to a target system. See Publishing UI Changes [page 479] and Publishing App Variants [page 484].
UI Adaptation and *Navigation* modes	When *UI Adaptation* is activated, you can edit the current screen on which you started the adaptation mode, but you cannot navigate using a button (for example, to another list item, screen, or dialog box triggered by the button). To interrupt the adaptation mode and reach these "hidden" elements, choose *Navigation*. Now you can navigate on the UI as usual, for example to reveal other areas. Please note: Any changes you make while you're in the *Navigation* mode (for example setting a filter) will **not** be saved later. So make sure you choose *UI Adaptation* again before you make further changes that you want to preserve.
Visualization mode	This highlights all or certain supported key user changes that have been made on the UI and are part of the current UI. When you click on a highlighted change, you see details about all changes made to this element, including a timestamp for when the changes were saved. When you're in this mode, you will also see a dropdown list in the header bar that lets you filter for certain change types. By default, you see *All Changes*. See Visualizing UI Changes [page 470].
Undo and *Redo*	Use ↶ to undo and ↷ to redo the changes you made during your current session. If you activate a new version, you cannot undo any changes that you did before activating the new version.
Manage App Variants	See Working with App Variants [page 482].
Save As	This saves your UI changes as a new app variant, which your administrator can then add as a new tile to your SAP Fiori launchpad home page. See App Variants: All You Need to Know [page 480].
Save & Exit	When you're done with your UI changes, use this option to exit the UI adaptation mode and return to your running app or app variant. Unsaved UI changes are automatically saved to the app or app variant from which you started the adaptation mode.

For the specific kinds of changes you can make on the UI, check the subpages of this chapter.

Security

UI adaptation should not be used as a means to fulfill security requirements. UI changes affect only the visibility of the respective elements on the UI. Example: If you remove a field, it doesn't show up on the UI any longer. However, no other security measures are applied that prevent a user from accessing the data associated with the removed field. If you want to to restrict access to certain data for security reasons, this has to be done in the back-end system via roles and authorizations.

Related Information

1.4.3.1.1.1 Making UI Changes (Object Pages)

When you're in the adaptation mode of an object page, you can edit all UI elements (such as fields, groups of fields, or sections) that get highlighted when you hover over or select them.

An object page is used to display and categorize all relevant information about an object.

> **i** Note
>
> Lots of change actions are available in UI adaptation. The possible actions depend on the type of UI element (fields, groups, sections, and so on) and the container that the UI element is part of.

Moving a UI Element

To move a UI element, just drag and drop it. When you drag a UI element, dashed boxes indicate where you can drop it. Drag the element to its new location (indicated by a space) and drop it.

You can also cut and paste UI elements: Right-click the UI element and choose *Cut* from the context menu. The element gets grayed out and a dashed box indicates where you can paste it. Right-click within the dashed box at the position where you want to paste the cut element and choose *Paste* from the context menu.

Removing a UI Element

Select a UI element it and press `Delete` or choose *Remove* from the context menu. The element is then removed from the UI, it's not permanently deleted from the app. You can add it again at any point within its original container. Exception: When you delete a group of fields, the grouping information (group title and order of fields) gets lost, but the fields are still available and can be added.

> ⚠ Caution
>
> Removing a field does not prevent users from accessing the data associated with the field. The field is merely removed from the UI and no other security measures are applied. If you want to implement security requirements to restrict access to certain data, this has to be done in the back-end system.

Adding a UI Element

Right-click the UI element container where you want to add an element and choose *Add* from the context menu (for example, *Add: Field*). Select the available elements that you want to add and choose *OK*. To add a new group, choose *Create: Group* from the context menu.

> **i Note**
>
> If you want to add a field, the list of available fields may vary depending on the context. This means that certain fields are only available for certain object types but not for others.

You can also create a custom field by choosing **+** in the dialogue for the available UI elements (SAP S/4HANA back end required). To use the custom field, you have to restart the app and the UI adaptation mode.

For more information about custom fields, go to the SAP S/4HANA product page, select your SAP S/4HANA version, and search for **Custom Fields and Logic**.

Renaming a UI Element

You can change editable text of a UI element. If you double-click it and the text (such as the title of a group of fields or a field label) gets highlighted, you can rename it. Just edit the text and press `Enter`. You can also right-click the element and choose *Rename* from the context menu.

If you rename a UI element, the element will be visible under the new name in any language version of the app that you use. Example: If you use an app in English and change the field name *Vendor* to *Supplier*, this field will also be visible as *Supplier* (in English) in the Japanese version of the app.

To translate the renamed UI element for other language versions, your administrator needs to set up translation as described here: Translating UI5 Flexibility Key User Adaptations [page 77].

Combining and Splitting Fields

You can combine up to three fields so that they're displayed in a single line. First, select a field. Press and hold `Ctrl` while selecting the other fields. Choose *Combine* from the context menu of one of the selected fields.

To split a combined field, right-click it and choose *Split* from the context menu.

Changing Settings of UI Elements

If a UI element supports it, you can change its settings by right-clicking it and choosing *Settings* from the context menu. For example, you can define the list of additional links that can be displayed in the popover of a link.

Resetting Changes

By choosing 🗑 *(Discard Draft)* in the header bar, you can discard all UI changes within the app and reset it to the last activated version. This cannot be undone.

If the developers of your app have enabled this, you can also reset the UI changes made in certain parts (containers such as certain tables or forms) of the UI without resetting the entire app. If this is enabled, you will see the *Reset Container* option when you right-click a container. This option only resets the controls within this container.

Related Information

Something Isn't Working like It Should? [page 485]
Publishing UI Changes [page 479]

1.4.3.1.1.2 Embedding Content (Object Pages)

You can embed content from external applications or websites to your object page app, for example, external videos or maps. The content is then added to an iFrame.

Prerequisites

You have started the UI adaptation mode by choosing *Adapt UI* in the user actions menu.

Process

Adding Content

You can add external content to a header or as a section as follows:

1. Right-click an area where you want to embed external content, for example, a section. If you can add external content there, you will see an *Embed Content...* context menu item. Depending on where you clicked, the menu item contains an *Embed Content: in Header* and/or *Embed Content: as Section* part.
2. In the *Define Embedded Content* dialog, specify the size of the new iFrame either as a percentage of the space that's available, in pixels, or in rem units.
3. Add the URL of the external content that you want to embed.
   This could either be an absolute (full) URL or - if the embedded content is running on the same host as your app - a relative (local) URL. With relative URLs, you start the URL with a forward slash and omit the host and port information. This option is useful if the embedded content has the same path on different systems and only the host name is different (for example if you have a quality system and a productive system).
   Example for an absolute URL:

   ```
   https://www.somewebsite.com/embed?more_URL_parameters
   ```

   Example for a relative URL:

   ```
   /images/image.png
   ```

   > i Note
   >
   > There are some things that you have to keep in mind when you add the URL. For more information, see Things to Watch Out For [page 466] below.

4. You can now edit the URL manually or use the available values to add additional parameters to the URL of the embedded content. The available values depend on the context of the area where you want to embed the content. You will only see values that are exposed in the relevant context.
5. Choose *Show Preview* to preview the result. If the preview doesn't show anything, check if the URL parameters are correct. See also Things to Watch Out For [page 466] below.
6. When you're done, choose *Save*.

> **i Note**
>
> If you embed the content to a section, this creates a new section with the default title *Embedded Content*.
> You can rename this section just like any other UI element.

Editing Embedded Content

If you want to change embedded content, right-click the content and choose *Update Embedded Content* to open the *Define Embedded Content* dialog.

Things to Watch out For

There are some things to keep in mind when you embed a URL:

- Some websites let you generate special embedding HTML content for you to copy and paste. This HTML content sometimes already contains iFrame information (`<iframe src="...">...</iframe>`). Since the iFrame will be generated in UI adaptation mode, this predefined iFrame information is not necessary. Only copy and paste the value of the `src` attribute when you embed the URL. Otherwise, the preview will show a 404 error.
  Example:

  ```
  <iframe src="https://www.somewebsite.com/embed?more_URL_parameters"
  width="600" height="450" frameborder="0" style="border:0;"
  allowfullscreen=""></iframe>
  ```

  You only need the highlighted part.
- Some technical restrictions of the target website or of the browser may prevent you from embedding it as an iFrame:
  - Some websites don't allow embedding as iFrames at all, for example, if they use CSP or X-Frame-Options. If you try to embed a URL of a website that restricts embedding, the preview will remain empty. If possible, contact the web server administrator of the target website to ask if there is a way around this restriction.
  - Some websites use a frame busting mechanism. If you embed such a URL, the target website will not be displayed in an iFrame. Instead, its content will try to take over the hosting parent, that is, your app. You may lose unsaved data.
  - If authentication is required for the website you want to embed, the hosted site may use a login window or require single sign-on. You can pass your current user information using parameters. If you want to authenticate the same user, both websites must go through the same identity provider (IDP).
  - The browser may enforce security checks that prevent the embedded content from being loaded. Contact your IT department to check if they can loosen these security checks within your organization.
  - Most browser show warnings or errors or will not load embedded content in case of "mixed content". Mixed content occurs if you embed HTTP content to an HTTPS website (or vice versa). This is not recommended.

    > **→ Tip**
    >
    > If the server of the target website supports both HTTP and HTTPS, you can omit the protocol in the URL, for example `//somewebsite.com` instead of `http(s)://somewebsite.com`. Your

> browser will then automatically first request the content with the HTTPS protocol, and if HTTPS is not available, it will request the content with HTTP.

- **Special Characters (&, ", {...})**
  Under certain circumstances, you need to encode or escape special characters.
  - **Embedding parameters**: When you add values from the embedding context to the URL as parameters, you might need to escape the values to generate a valid address. You can change the value description using an advanced syntax.
    Example: The current context provides the `DocumentName` field, which contains characters like `&` or `"`. These characters are usually forbidden in URLs. To use this value as a parameter, you can use expression binding. In this example, you could change the `{DocumentName}` parameter to `{= encodeURIComponent(${DocumentName})}`. For more information, see Expression Binding📄 in the SAPUI5 documentation.
  - **URLs with JSON payload**: If the embedded URL contains a JSON payload, the curly brackets that are part of the JSON need to be encoded (`{` becomes `%7B` and `}` becomes `%7D`).

    > ℹ Note
    >
    > Curly brackets are fine in binding contexts but they need to be encoded in other parts of the URL.

1.4.3.1.1.3 Making UI Changes (Overview Pages)

When you're in the UI adaptation mode of an overview page, you can edit all UI elements (such as cards, fields, or link lists) that get highlighted when you hover over or select them.

Introduction

An overview page (OVP) is a data-driven app type that provides all the information a user needs on a single page based on the user's specific domain or role.
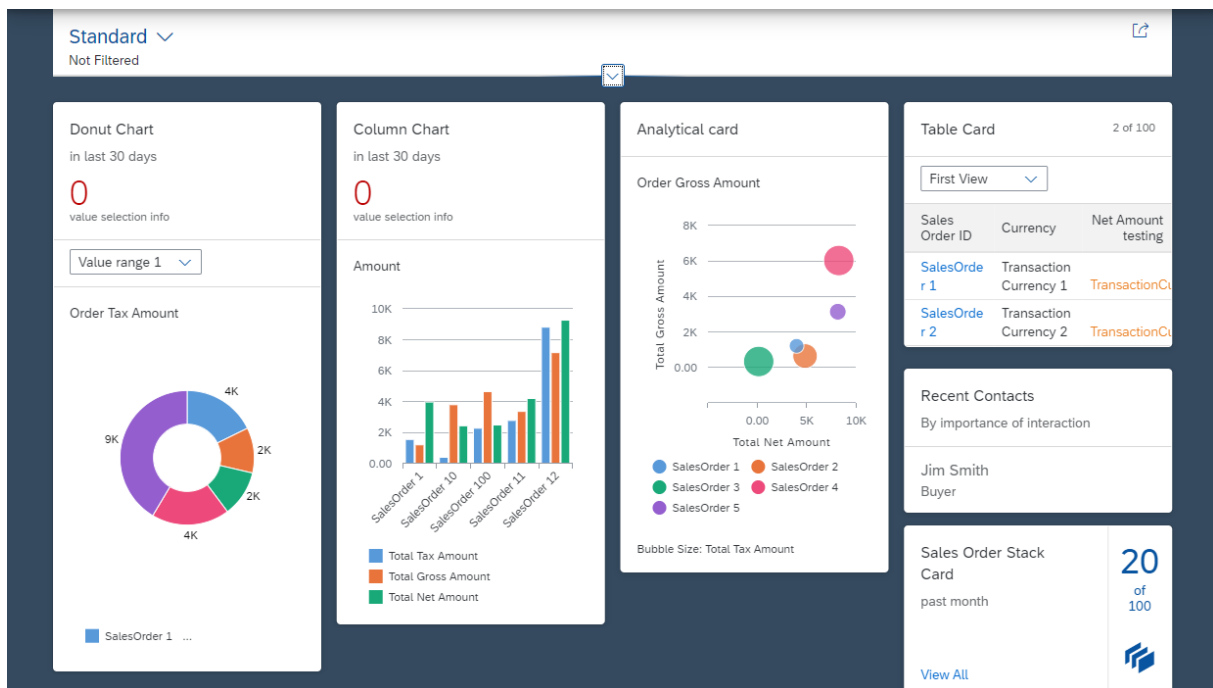
Adding a Card

You can add available cards that haven't been added to the UI yet or that have been removed. Right-click in the title area of an existing card and choose *Add: Card* from the context menu. Select the available card that you want to add and choose *OK*. The card is added to the UI.

Hiding a Card

You can hide existing cards that you don't want to be displayed initially. End users can still add cards themselves later. To hide a card, right-click in the title area of an existing card and choose *Hide: Card* from the context menu.

Moving a Card

To move a card, right-click in the title area of an existing card and choose *Cut* from the context menu. Then right-click in the title area of a different card and choose *Paste* from the context menu. The card is pasted above the other card.

> i Note
>
> End users without key user authorizations can also move cards for themselves by dragging and dropping them.

Editing a Card

Certain changes can be done directly on the overview page:

- To change UI texts, right-click on a text, choose *Rename*, and change the text.
- For table cards: To add a table column, right-click on an existing table column and choose *Add: Column*. Now, you can choose which columns you want to add.

General Card Settings

To make more in-depth changes to the card, right-click in the title area of an existing card and choose *Edit Card*. The changes that you can make depend on the type of card. If the *Card Settings* window has a dropdown list at the top, you can make various changes to the basic card attributes or to individual views of the card.

Depending on the card type, you can change the following basic card settings:

- Change the card title and subtitle.
- Activate/deactivate the dynamic subtitle. Whether a dynamic subtitle is available, depends on the app.
- Activate/deactivate the KPI header. If it's activated, you can add further KPI information to the card.
- Activate/deactivate resizing of the card (if the app uses the resizable layout). If resizing is activated, you can set the number of columns and rows for the card.
- For link list cards and contact cards: Change the display type of the card.
- For table cards: Change the settings for *View By*/*View Fields By*/*Navigate To*. The values that you can select for these elements depend on the app.
- For analytical cards and chart cards: Change the settings for *View By*/*Navigate To*/*Type of Chart*. The values that you can select for these elements depend on the app.

View Settings

If the *Card Settings* window has a dropdown list at the top, you can make changes to the views that are available for the card. Depending on the card type, you can change the following view settings:

- You can add new views to a card and make settings for each one.
- You can delete existing views for a card.
- You can set a view as the default view.

The settings that you can make for each view depend on the card type and are similar to the options described under *General Card Settings*.

When you've finished editing the card, choose *Save*. To reset the card to its default values, choose *Reset Card*.

Copying a Card

To copy a card, right-click in the title area of an existing card and choose *Copy Card* from the context menu. The card is cloned automatically and flagged as a *Copy* while you're in UI adaptation mode. You can then make individual changes to the copied or original card.

Creating a Link List Card

To create a link list card, right-click in the title area of an existing card and choose *Create Link List Card* from the context menu.

Possible settings: *Card Title*, *Card Subtitle*, and *Display Type* are the same as when you edit a card. With ✚ , you can add links. You can link to a URL or to a specific application that is available for your launchpad. For each link, you can make further settings (title, subtitle, image, etc.).

To delete a link, choose 🗑 .

Creating a KPI Card

> **i Note**
>
> It depends on your current system whether this option is available.

To create a KPI card, right-click in the title area of an existing card and choose *Create KPI Card* from the context menu.

Possible settings: *Card Title* and *Card Subtitle* are the same as when you edit a card. To set a certain KPI, select it from the list of available KPIs and choose *Save*.

Related Information

1.4.3.1.1.4 Visualizing UI Changes

The *Visualization* mode provides the key users with both an overview of all key user changes made to the application as well as detailed information for individual changes.

Prerequisites

- You have started the UI adaptation mode by choosing *Adapt UI* in the user actions menu.
- You have made at least one UI change.

Overview

When you're in *Visualization* mode, each UI control for which at least one change exists, is denoted with a blue dot as a change indicator. The more changes exist for this control, the darker the change indicator becomes, giving the key user an intuitive heat map that shows to which degree different parts of the app have been adapted.

Visualizing UI Changes

After you have chosen *Adapt UI*, choose *Visualization* in the header bar. When you hover over a control with a change indicator, you see a border around it. Clicking anywhere on such a control opens a popover window that shows each change made for this control with the following information:
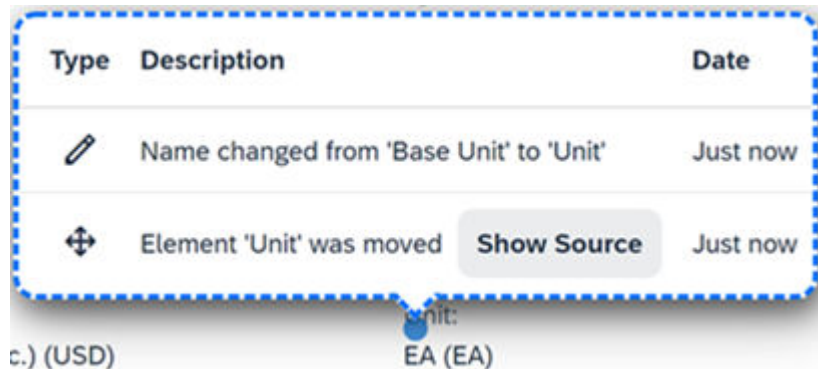
Type	Icon for the type of change, such as *rename* or *move*
Description	Provides more information about the change. Examples: For a renaming, the previous name of the element is shown; for a moved control, a button is available to show where the control was moved from, etc.
Date	The date of the change is shown in an easily readable format. If you want to know the exact time, hovering over the date shows it as a tooltip.

Filtering Changes

In *Visualization* mode, the header bar contains a dropdown menu on the right to filter the changes that are visualized.
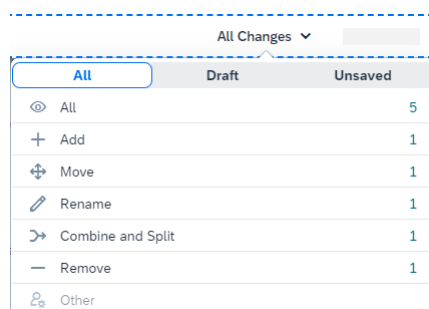
At the top of the dropdown menu, select which changes you want to view:

- *All*: All changes (default).
- *Draft*: Only changes that have not yet been activated.
- *Unsaved*: Only changes that you've made in the current session and not yet saved.

You can filter for the following types of changes:

- *All* (default)
- *Add*

- *Move*
- *Rename*
- *Combine and Split*
- *Remove*
- *Other*

In some cases, this dropdown shows a hint that a certain number of changes can't be visualized. There are several possible reasons for this:

- The non-visualized changes are part of a dialog or a popover that is currently not displayed.
- The non-visualized changes aren't supported by the change visualization feature yet, such as table or filter changes.
- The section, group, or form that contained the non-visualized change is no longer available (for example, because the key user has deleted it).

1.4.3.1.1.5 Creating and Adapting Views

With the UI adaptation mode, you can change existing views, create new views, and manage available views for other users in your organization.

Prerequisite

The app contains a view element, for example, a table with a view selector:
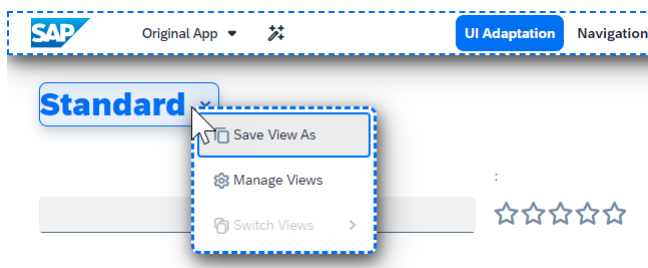
Create a New View

Essentially, any end user can create views for themselves. They can even make the views public for others if the administrator has allowed this. This is described here: Personalizing Apps [page 487]. As a key user, you can also create adapted views for other users and even share them for certain roles only. Here's how this works.

1. In the user actions menu, choose *Adapt UI*.
2. Click on the UI element that you want to change, for example, a table and choose 🏭 .
3. Make your preferred settings. The settings that you can make depend on the app and the UI element that you're adapting.

4. Right-click on the view selector and choose *Save View As*.

5. Enter a unique name for your view.

6. If you want the view settings to be applied automatically when a user selects the view, select *Apply Automatically*.

7. If you want the view to become the default view when a user opens the app, select *Set as Default*.

8. Adjust the visibility of the view:

   - Leave it as visible for all users if the view should be available for every app user in your organization.
   - Add roles if you want to restrict the visibility to certain business roles:
     1. Choose *Add Roles*.
     2. Select at least one role that should be able to select the view when they use the app.
     3. When you're done, choose *Select*.
     4. Back in the *Save View* dialog, you can remove certain or all roles again.

⚠ Caution

Restricting the visibility of a view merely hides the view on the UI for the respective users. It does not prevent users from accessing the data associated with the view. No other security measures are applied, so if you want to implement security requirements to restrict access to certain data, this has to be done in the back-end system.

9. Choose *Save*.

10. In the header bar, choose *Save & Exit* to save your UI changes as a draft, or choose ⚡ *(Activate New Version)* to save them as a new version.

As soon as you've published your UI changes (see Publishing UI Changes [page 479]), the end users can use the newly defined view. If you've selected certain roles for the view, it will be visible only to users with at least one of the selected roles.

ⅈ Note

If you as a key user **don't** have any of the assigned roles, you'll be able to see the view only in UI adaptation mode but not in your personal view selector.

Change an Existing View
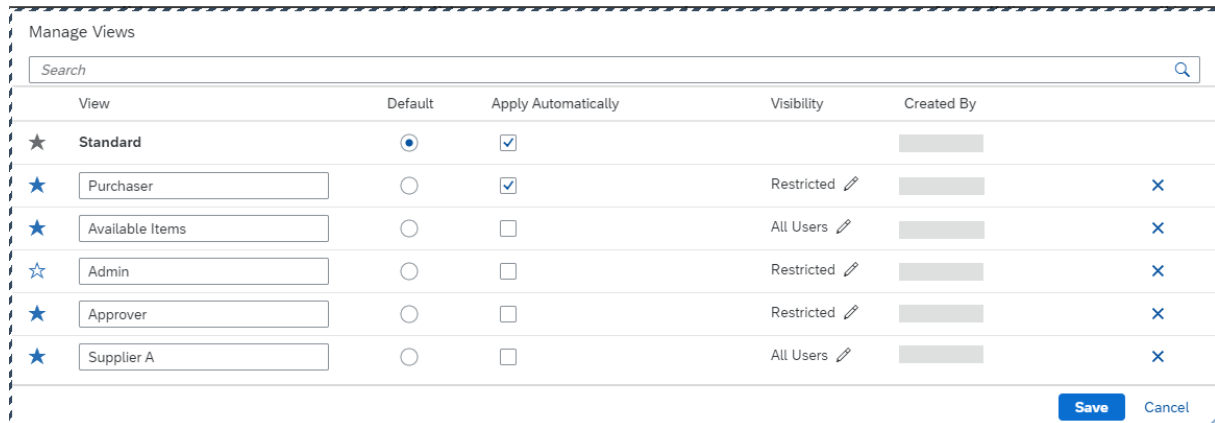
Here's how you can make changes to an existing view.

1. In the user actions menu, choose *Adapt UI*.

2. If you're not on the desired view, right-click on the view selector, choose *Switch Views*, and choose the view that you want to change.

3. Choose 🔧 and adapt the view as needed.

4. Right-click on the view selector and choose either *Save View* to overwrite the selected view or *Save View As* to save the view under a different name and optionally with different role assignments.

5. In the header bar, choose *Save & Exit* to save your UI changes as a draft, or choose ⚡ *(Activate New Version)* to save them as a new version.

As soon as you've published your UI changes (see Publishing UI Changes [page 479]), the changes are reflected for the end users.

Manage Views

This dialog lists the views that you or other key users have created **in UI adaptation** as well as **public** views created by end users in this system. It does not show your personal views. If you want to delete your **personal** views, see Manage Views [page 490].

Manage Views

In the *Manage Views* dialog, you can perform the following actions:

- To add a view to the view selector for end users or to remove it, click on the ★ *Favorite* icon.
- To rename a view, click in an input field and overwite the old name with a new name.
- To determine which view should be displayed when the end user enters the app, select *Default* button for the respective view.
- To determine whether the view settings should be applied immediately when an end user selects the view (or, for the default view, also when they enter the app), select *Apply Automatically*.
- To change the role assignment for a view, choose ✎ to open the *Select Roles* dialog, make your adjustments, and save them.
- To delete a view that you or another key user has created in UI adaptation or a **public** view that was created by an end user in this system, click on the ⊗ *Delete* icon.

As soon as you've published your UI changes (see Publishing UI Changes [page 479]), the changes are reflected for the end users.

Please note that the settings you make for favorites, *Apply Automatically*, and *Default* are not binding for your end users. They can change these settings for themselves as described under Manage Views [page 490].

Related Information

Publishing UI Changes [page 479]
Something Isn't Working like It Should? [page 485]

1.4.3.1.1.6 Keyboard Shortcuts

When adapting SAP Fiori UIs at runtime, the following key combinations improve accessibility and speed up your work.

Key combination	What it does
`Tab` `Shift` + `Tab`	Moves forwards or backwards in the tab chain to select elements on the user interface (UI)
Arrow keys	Navigates between groups and fields Using the `Left arrow` and `Right arrow` keys, you navigate between groups and between fields in a group. Using the `Down arrow` key, you navigate from a group to the first field in this group. Using the `Up arrow` key, you navigate from a field in a group to the group itself.
`Enter`	Selects an UI element
`Shift` + `F10`	Opens the context menu
`Ctrl` + `X` Mac: `Command` + `X`	Cuts a group or field
`Ctrl` + `V` Mac: `Command` + `V`	Pastes a cut group or field
`Ctrl` + `Z` Mac: `Command` + `Z`	Undo
`Ctrl` + `Y` Mac: `Command` + `Y`	Redo
`Esc`	Quits an editing function

1.4.3.1.1.7 Drafts and Versioning

When you as a key user adapt the user interface of an application, the adapted app is in a draft state. You can use drafts to create different versions of the app.

Save UI Changes as Draft

To save UI changes as a draft, click *Save & Exit*.

Your changes are saved but won't be visible to end users yet. Later on, you can continue your work by simply starting UI adaptation again.

In addition, as all key users share the same draft, you can also continue the work of other key users. Or you can have your work verified by other key users, all without affecting any end users.

Discard a Draft

This option is only available when you display a draft that has not yet been activated.

To discard a draft, click 🗑 *(Discard Draft)*.

You can then start adapting the UI again, based on the last activated version.

Activate a New Version

1. Click ⚝ *(Activate New Version)*.
2. Enter a title for the new version.

Every time you activate UI changes, a new version is created. You can see the title of this version in the version history dropdown list.

Display Available Versions

To display available versions, click on the title of the current version to expand the version history.
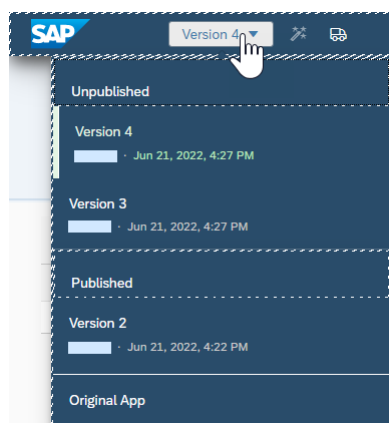
The dropdown list contains the most recent versions.

The version that is currently active is highlighted.

If you work in a test system from where UI changes are published to a productive system, you can see which versions have already been published to the productive system.

At the bottom of the list, you always find an entry for the original app.

> ### i Note
>
> If you adapted the UI of an application before the availability of the versioning feature, all changes that you made before the availability of the versioning feature are included in a version named *Version 1*.
>
> This version may contain changes that you have already published as well as changes that you have not published yet. If you have not yet published all changes, *Version 1* will appear in the *Unpublished* category, so you can publish it.

Switch to Another Version

1. Click on the title of the current version to expand the version history.
2. In the dropdown list, click on the title of the version that you want to display.

Reactivate a Version

1. Click on the title of the current version to expand the version history.
2. In the dropdown list, click on the title of the version that you want to reactivate.
3. Choose 🪄 *(Activate New Version)*.
4. Enter a title for the new version.

Restore the Original App

1. Click on the title of the current version to expand the version history.
2. In the dropdown list, select *Original App*.
3. Click 🪄 *(Activate New Version)*.
4. Enter a title for the new version.

1.4.3.1.1.8 Publishing UI Changes

You make your UI changes in a test system and publish them to transport them to the productive system. This chapter describes this publishing step for adapted apps.

Prerequisites

The following prerequisites need to be fulfilled:

- You have saved UI changes to the original app (or an existing and already published app variant) in the test system.
- The changes that you want to publish need to be included in a version. You cannot publish draft changes. You can publish either the most recent (active) version or any previous version that is yet unpublished.
- A transport request must be available for you in the test system. If you have transported UI changes or app variants in the past and the transport request that you used then is still available, you can use the same one. If your administrator has to create a new transport request for you, it's recommended to use the "customizing request" type to transport UI changes. If in doubt, ask your system administrator.

Procedure

1. In the test system, open the adapted original app that you want to publish and choose *Adapt UI* in the user actions menu.
2. In the version history, select a version that you want to publish.
3. Choose 🚚 *(Publish Version)*.
4. Select a transport from the dialog and confirm. If no transport is available, ask your administrator to create a transport request (recommendation: customizing request) for you. Afterwards, choose *Publish* again.
5. Your administrator now has to release the transport request.

After the release, all versions up to the version that you selected will be available in the productive system.

Deleting UI Changes from the Productive System

If you want to delete UI changes from the productive system, proceed as follows:

1. In the test system, open the adapted original app that you want to publish.
2. In the version history, switch to a previous version that you want to reactivate.
3. Choose 🪄 *(Activate New Version)*.
4. Enter a title for the new version.
5. Choose 🚚 *(Publish Version)*.
6. Select a transport from the dialog and confirm. If no transport is available, ask your administrator to create a transport request (recommendation: customizing request) for you. Afterwards, choose 🚚 *(Publish Version)* again.

7. Your administrator now has to release the transport request.

After the release, the reactivated version will appear in the productive system.

1.4.3.1.2   App Variants: All You Need to Know

You can create variants of apps (even if the apps are variants themselves). Your content administrator can add your app variant as a new tile to your home page of the SAP Fiori launchpad. The UI of the app variant can be adapted separately.

Purpose

App variants are apps that you create based on UI changes that you as a key user made in UI adaptation. Instead of saving your changes in your original app, you save them in a new app. These app variants are useful for special usage scenarios and their specific requirements, for example for countries, organizational units, or user groups. App variants are individual apps that are still linked to the original app. Here's an example of a sales order app with variants for Europe and China:
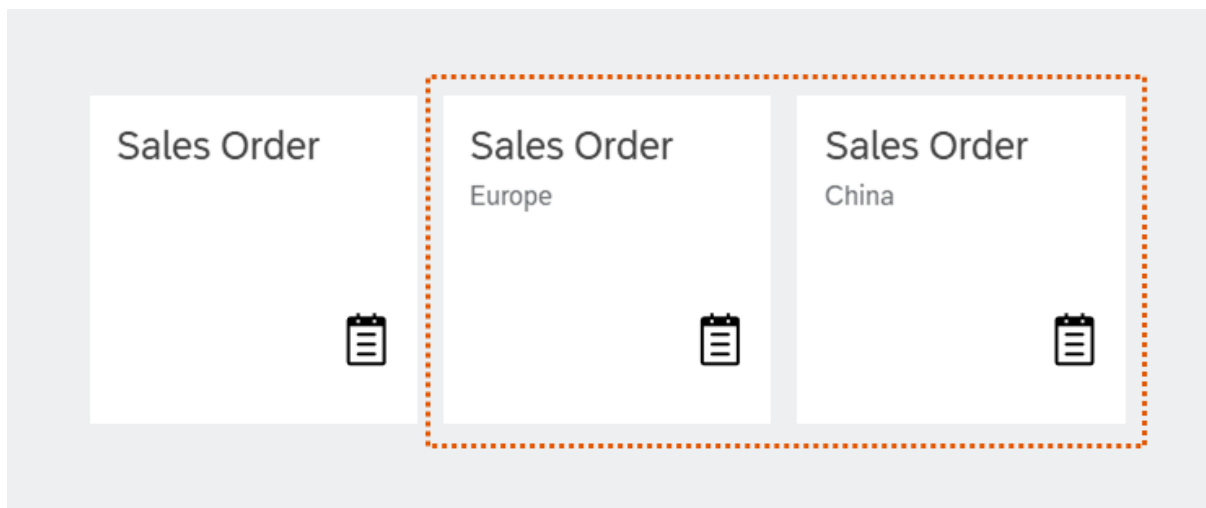
Prerequisites

Some apps don't support creating app variants. This option is then not available from the header bar in UI adaptation mode. (If you're interested in the technical details: App variants are not supported by Scaffolding-based apps or apps whose SAPUI5 version is 1.28 or lower.)

Creating App Variants - General Process

Basic Steps (for the details, see Working with App Variants [page 482]**):**

1. Adapt the UI in the test system.
2. Save your UI changes as an app variant.
3. Ask your administrator to create the target mappings and the tile for the app variant.
4. Add your app variant to your SAP Fiori launchpad home page **in your test system** via the app finder.

> → Tip
>
> Ask your administrator to add the app variant to an appropriate group, then you won't need the app finder.

5. Open the app variant in UI adaptation mode.
6. Publish the app variant. This will add the app variant to a transport request, which has to be released by your administrator.

As soon as your administrator has released the transport, the app variant is available in the productive system. You and your end users can now add the app variant to the SAP Fiori launchpad home page via the app finder (see Related Information).

> i Note
>
> For more detailed information about app variants and publishing them, see Working with App Variants [page 482] and Publishing App Variants [page 484].

Other Things to Consider

UI changes you make to the original app or a variant are not applied to other variants. But if the UI of the original app is changed outside of key user adaptation, for example, by SAP or a developer, these outside UI changes are applied to your app on as well as to all variants you've created from the app or its variants. Your key user UI changes are applied on top of such outside changes.

Here's an example:
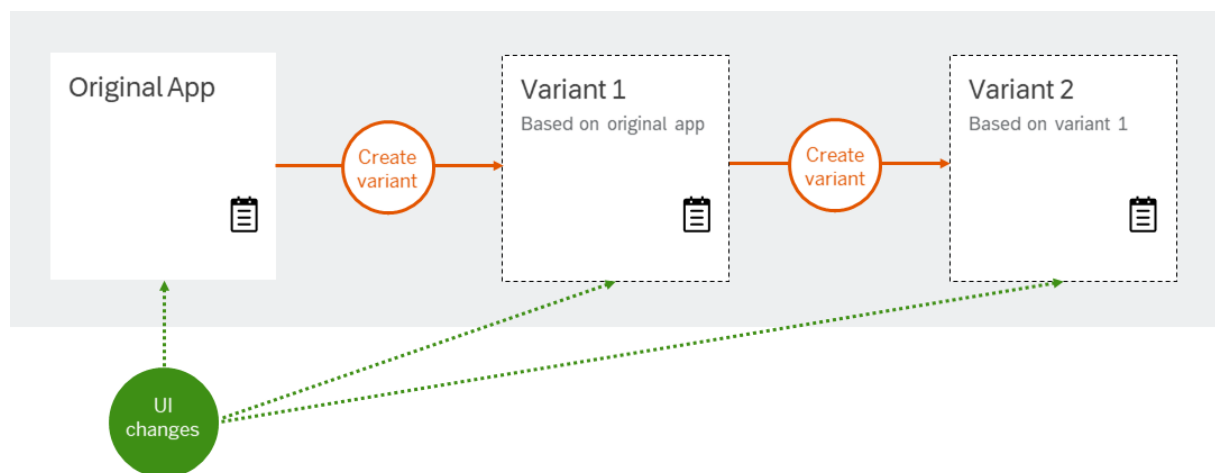
You create a new variant from an original app and a second variant from this new variant. If you as a key user then adapt the UI of the original app or one of the app variants, these changes will only apply to the app or app variant where you make the UI changes.

However, developer UI changes to the **original app** are applied to the original app **and** the variants.

Related Information

1.4.3.1.2.1 Working with App Variants

Here's how you create and manage app variants.

Prerequisites

You have started the UI adaptation mode by choosing *Adapt UI* in the user actions menu.

The changes that you want to save as an app variant need to be included in a version. You cannot save draft changes as a variant. You can save either the most recent (active) version or any previous version as an app variant.

Header Bar
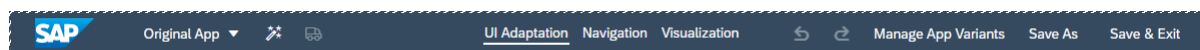
Saving an App Variant

1. In the version history, select a version that you want to save as an app variant.
2. Choose *Save As*.
3. Enter a title for your app variant. The other information is optional, but it's important to make a note of your entries, because you will need them later.

Save as New App Variant

Title: * Enter title
Subtitle:
Icon:
Description:

Save   Cancel

4. Choose *Save*.

5. In the information message, choose *Copy ID and Close* to copy the ID of the new app variant to your clipboard and close the message (you need the ID for the next step).

6. To see the new app variant on your SAP Fiori launchpad home page, ask your administrator to create the target mappings and static app launcher tile. The administrator needs the following information:

   • Name of the original app.

   • The app variant ID that you copied to your clipboard. (If necessary, you can copy the ID again via the *Manage App Variants* dialog, see below).

   • All other details about the app variant that you noted down (such as subtitle, icon, description).

   For more information, see Settings for SAPUI5 App Variants [page 220].

7. As soon as the administrator has created the target mappings and the tile, you can add the tile for the app variant to your launchpad home page in the test system via the app finder.

You can now:

• publish the app variant to a productive system by clicking the 🖫 *(Publish Version)* icon (see Publishing App Variants [page 484]).

• make further UI changes to your app variant and save them to a new version of this app variant using ⚡ *(Activate New Version)*.

• after activating a new version, save it to a new app variant using *Save As*.

Managing App Variants

If you choose *Manage App Variants*, you can perform various actions for the original app and existing app variants:

• *Adapt UI* = Opens the original app or the corresponding app variant in UI adaptation mode. Please note: This option only works after your administrator has created a target mapping (see above).

• *Copy ID* = Copies the ID of the original app or the corresponding app variant to give it to your administrator, who creates the target mapping and the tile.

• *Delete App Variant* = Deletes the corresponding app variant from the corresponding system.

> **i Note**
>
> • You can't delete the original app in this dialog.
>
> • If the *Delete App Variant* option is inactive for an app variant, ask your administrator to delete the target mappings for this app variant. As soon as this is done, the option will be active.

> • If there is no open transport request for your variant deletion, you receive a corresponding message. As soon as your administrator has created the transport request, you can repeat the deletion step.
> • If you're working with test and production systems and you want to delete an app variant altogether, delete the app variant in the test system and transport the deletion to the productive system.
>
> Also see Publishing App Variants [page 484].

Related Information

App Variants: All You Need to Know [page 480]
Publishing App Variants [page 484]
Adding Apps to the Home Page [page 504]
Static App Launcher Tiles [page 207]
Configuring Target Mappings [page 217]

1.4.3.1.2.2 Publishing App Variants

You create your app variants in a test system and transport them to the productive system. This chapter describes this publishing step for app variants.

Prerequisites

The following prerequisites need to be fulfilled:

- You have created an app variant and the app variant is available on your SAP Fiori launchpad home page.
- A transport request must be available for you in the test system. If you have transported UI changes or app variants in the past and the transport request that you used then is still available, you can use the same one. If your administrator has to create a new transport request for you, it's recommended to use the "customizing request" type to transport app variants. If in doubt, ask your administrator.

Procedure

1. In the test system, open the app variant that you want to publish and choose *Adapt UI* in the user actions menu.
2. If you have created any versions for this app variant, select the version that you want to publish from the version history.
3. Choose 🚚 *(Publish Version)*.

4. Select a transport from the dialog and confirm. If no transport is available, ask your administrator to create a transport request (recommendation: customizing request) for you. Afterwards, choose 🚚 *(Publish Version)* again.

5. Your administrator now has to release the transport request.
Your administrator also has to transport the target mapping and the tile to the same system.

After the release, the app variant is available in the productive system and can be added to the SAP Fiori launchpad home page using the app finder. See Adding Apps to the Home Page [page 504].

Deleting an App Variant from the Productive System

If you want to delete an app variant from the productive system, the steps are essentially the same: You delete the app from the test system first and then your administrator uses a transport request to transport the deletion to the productive system. See also Working with App Variants [page 482].

1.4.3.1.3 Something Isn't Working like It Should?

Click on the sections below, to find tips and solutions to the most common problems.

I can't see the *Adapt UI* option in the user actions menu

This can be due to the following reasons:

- The app is not enabled for key user adaptation.
- Your administrator has not assigned you to the authorization role SAP_UI_FLEX_KEY_USER (see Enabling Key User Adaptation [page 75]).
Contact your administrator.
- You don't use the FioriLaunchpad.html page.
Contact your administrator.

I'm getting a message that says the app is not enabled for adaptation or personalization

This means that the application developer has deactivated these options for your app. If this application was developed in your company, please contact your development team for further information.

I can't edit UI elements because I can't reach them

You have to toggle to navigation mode to edit UI elements, which are not directly on the UI but are triggered by actions or must be navigated to, such as dialogs and popups. Choose *Navigation* in the header bar and click the element you want to navigate to. To make UI changes, choose *UI Adaptation*. For more information, see Adapting the UI [page 461].
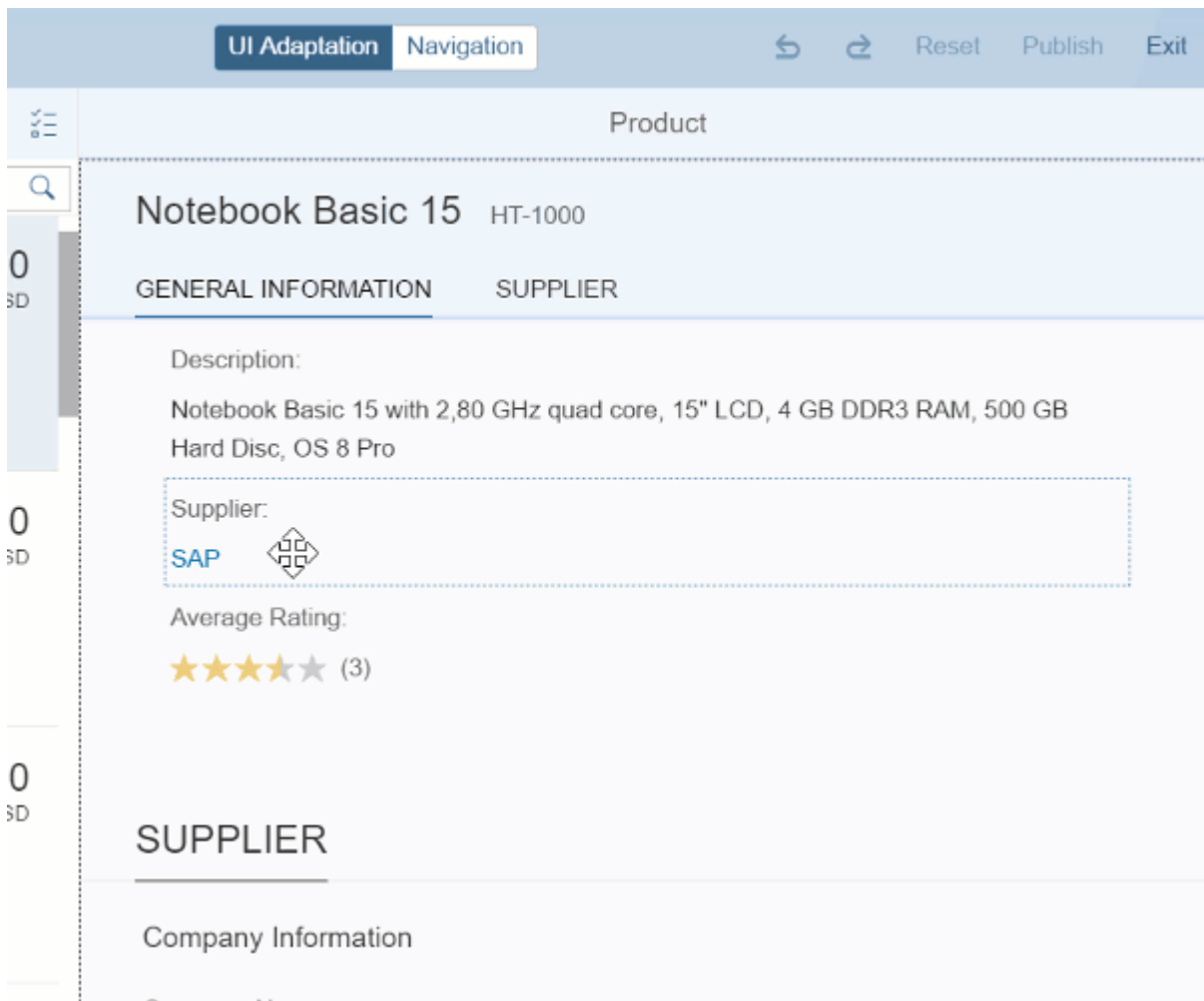
I've changed a UI element, but the change is not applied everywhere

UI elements that you change in one place might not be changed in all other places where they are used. For example, if you rename a label, it's not changed everywhere in the app, and you must rename each occurrence individually.

Also, if you make changes in edit mode of the app, they might not be visible in display mode, and vice versa. If your change is visible only in one mode, make the change also in the other (by toggling between adaptation and navigation mode as described under **I can't edit UI elements because I can't reach them**).

I cannot add UI elements

If all elements are placed on the UI, there's nothing left in the dialog of the available elements. But you could still create new custom fields and add them.

I cannot drag fields from one group to another

If you drag a field from a group and another group is not selected as a drop zone, it is not possible to drop the field in this group. The reason for this is that the fields in each group come from different data pools that are incompatible with each other.

I cannot combine fields

If you select two or three fields, the *Combine* function gets available in the context menu. If you select more fields, it's grayed out. Bear in mind that if you want to combine a combined field, it may contain only two fields.

I cannot edit a view in key user adaptation

When editing a view in key user adaptation, make sure that you work in the same system where the view has been created.

For example, if you created a view in a test system, and then transported it to the production system, you cannot change this view in the production system using key user adaptation. A view can only be adapted in the original system where it has been created.

> **i Note**
>
> However, you can always personalize views for yourself, no matter in which system they have been created.

1.4.3.2 Personalizing Apps

You can personalize SAP Fiori apps directly in the launchpad.

You can personalize apps in the following ways:

- **Save your personal settings as a view, and manage views**
  Some user interface elements, like filter bars and certain tables, allow you to do personal settings and save these as a view. If you do so, you can later re-apply the same settings just by selecting the view.
  You can also manage the views that are displayed for you in the view selector. For example, when several views are available for an app, you can select which one you want to be your default view.

- **Personalize link lists**
  Some apps provide link lists that you can personalize by adding or removing links from a predefined set of links.

Related Information

1.4.3.2.1   Save Your Personal Settings as a View

Some user interface elements, like filter bars and certain tables, allow you to do personal settings and save these as a view. If you do so, you can later re-apply the same settings just by selecting the view.
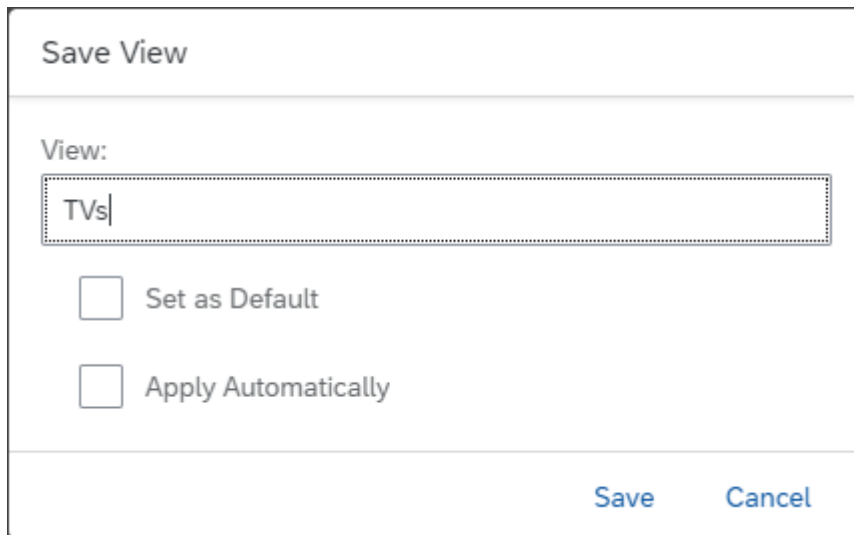
Context

If an app has a view selector where a *Save As* button is available, you can save your personal settings as a view.
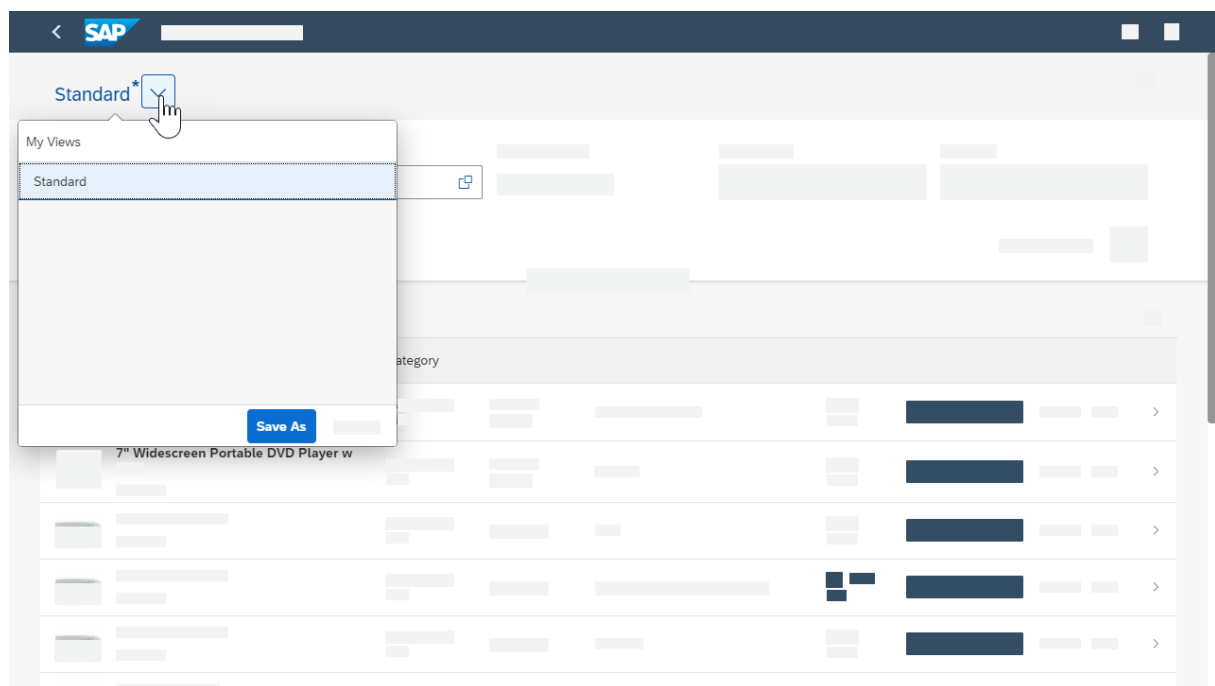
Procedure

1. Do some personal settings in your app.
2. Click on the ˅ *Dropdown* icon next to the name of the current view.
3. Choose *Save As*.
4. Enter a name for your view.

Save View

View:

TVs|

☐ Set as Default

☐ Apply Automatically

Save   Cancel

5. The following options are available:

Option	Description
Set as Default	Activate this if you want your view to be displayed by default when you enter the app.
Apply Automatically	Activate this if you want the filter settings to be applied immediately when you select the view.

6. Click *Save*.

Results

When you click on the ˅ *Dropdown* icon next to the list of views, your personalized view is available in the list of views.
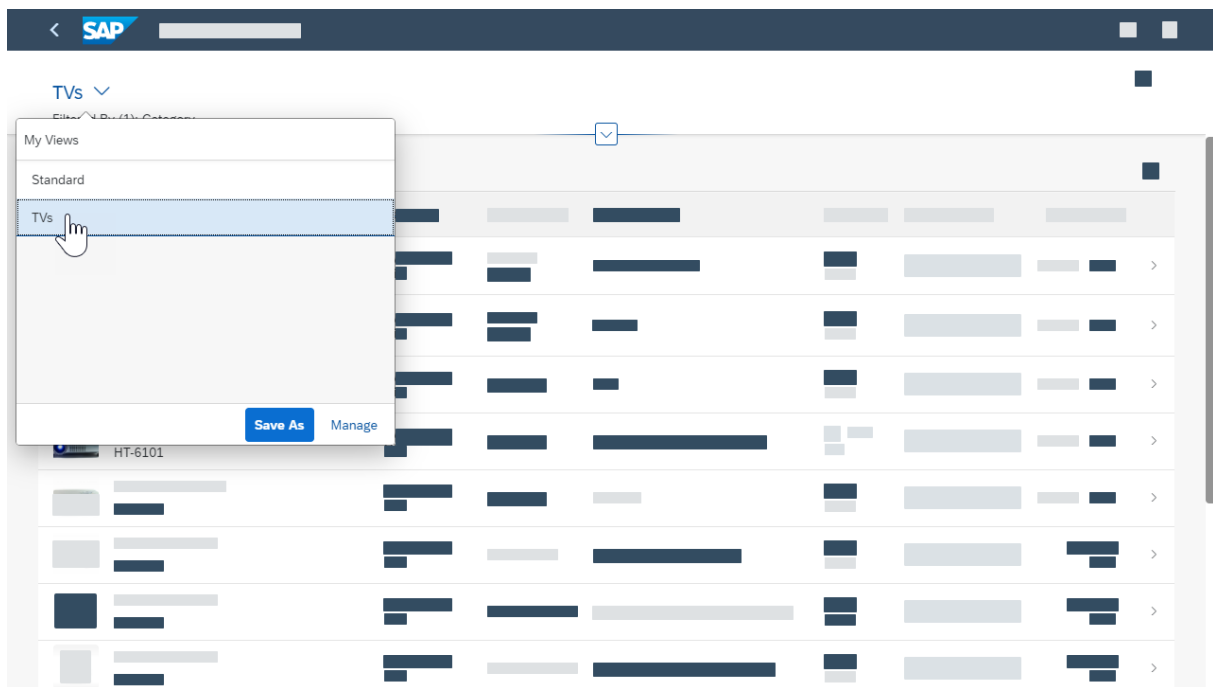
Related Information

Manage Views [page 490]

1.4.3.2.2 Manage Views

Some apps allow you to manage the views that are dislayed for you in the view selector. For example, when several views are available for an app, you can select which one you want to be your default view.

Procedure

1. Click on the ∨ *Dropdown* icon next to the name of the current view.
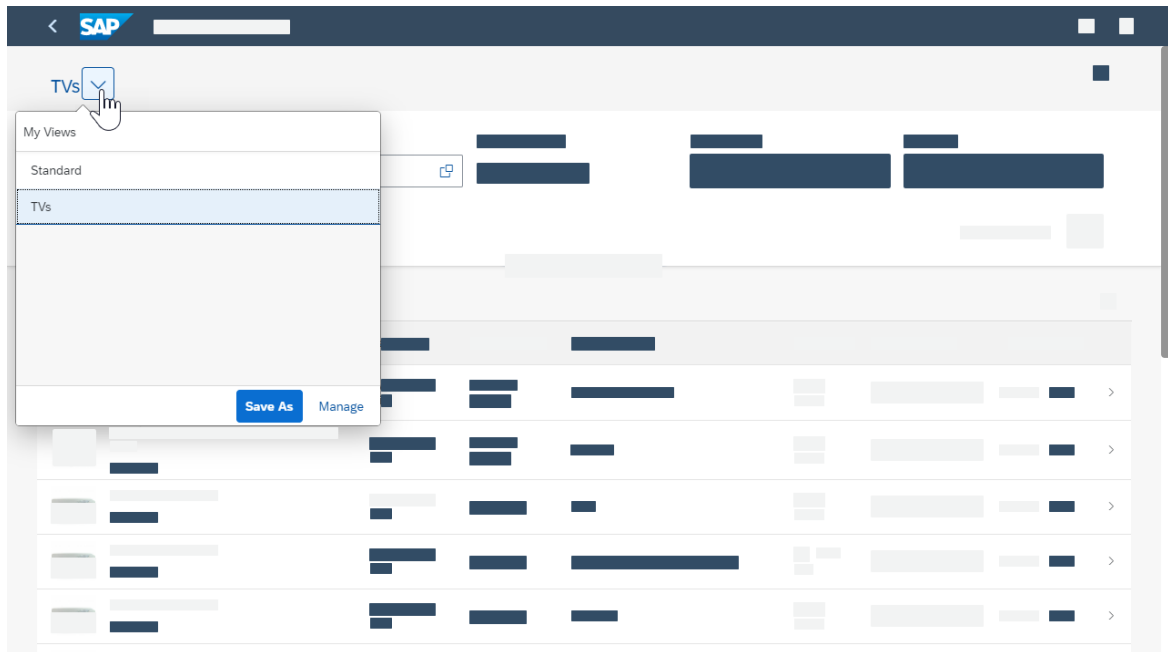
2. Choose *Manage*.

3. You can perform the following actions:

- To add or remove a view from the dropdown menu, click on the ★ *Favorite* icon.
- To rename a view, click in an input field, and overwite the old name with a new name.
- To determine which view should be displayed when you enter the app, select the *Default* radio button for the respective app.
- To determine whether the filter settings should be applied immediately when you select the view (or, for the default view, also when you enter the app), select *Apply Automatically*.
- To delete a view, click on the ⊗ *Delete* icon.

> **i Note**
>
> In case of public views: You can only delete public views that you have created yourself. Users with key user authorization can also delete public views created by other end users. See Creating and Adapting Views [page 473].

Related Information

1.4.3.2.3 Create Public Views
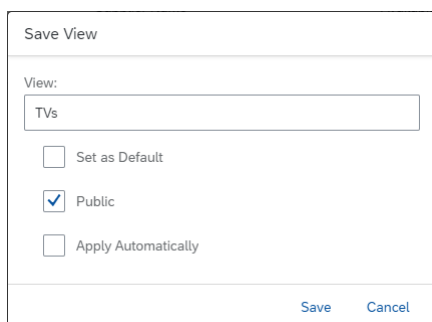
On some platforms, end users can publish a view to make it available to other end users.

Context

This feature is only available on some platforms, and it can be deactivated by your administrator.

Procedure

1. Do some settings in your app.
2. Click on the ⌄ *Dropdown* icon next to the name of the current view.
3. Choose *Save As*.
4. Enter a name for the view.
5. Activate the *Public* checkbox.

```
Save View

View:
TVs

☐  Set as Default

☑  Public

☐  Apply Automatically

                              Save    Cancel
```

   If you cannot see a *Public* checkbox in the *Save View* dialog, this feature is not available on your platform, or it has been deactivated by your administrator.
6. Click *Save*.

Results

Your view is now available to other end users. For information to modify or delete a public view, see Manage Views [page 490].

> **i Note**
>
> You can only delete public views that you have created yourself. Users with key user authorization can also delete public views created by other end users. See Creating and Adapting Views [page 473].

1.4.3.2.4 Personalize Link Lists

Some apps provide link lists that you can personalize by adding or removing links from a predefined set of links.

Context

If a link list contains a *More Links* option, you can personalize which links you want to be displayed in the link list.
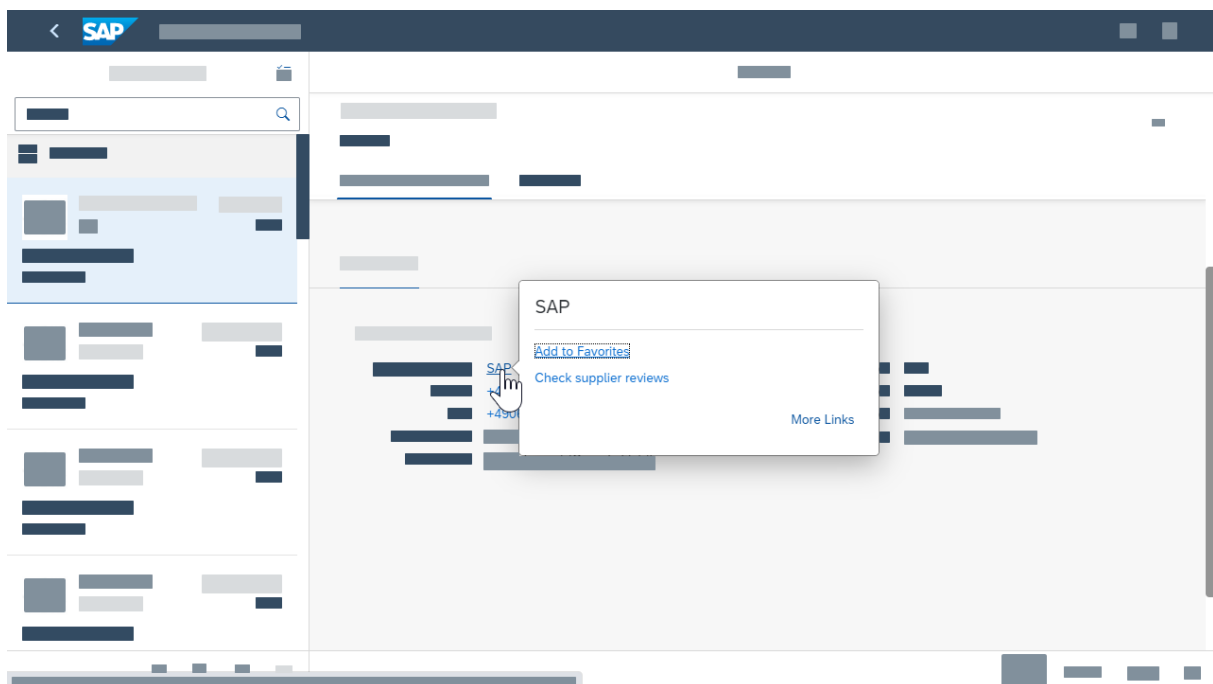
Procedure

1. Click on *More Links*.
2. Select the links that you want to be displayed in the link list. Deselect the links that you want to remove from the link list.
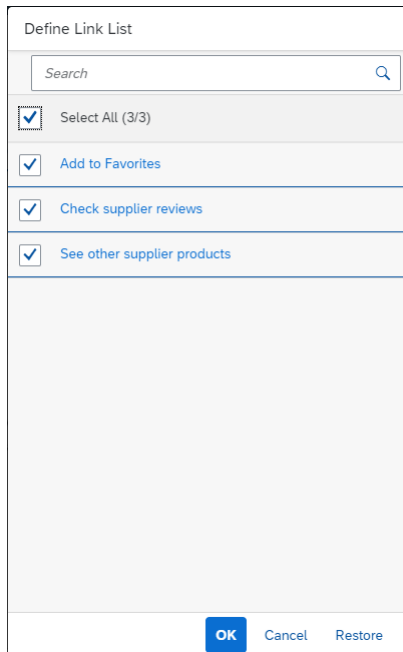
3. Click *OK*.

1.4.3.2.5 Personalizing Object Pages (Experimental)

You can personalize an object page used in your app by adding, removing or rearranging sections.

> **i** Note
>
> This feature is experimental. Experimental features are not part of the officially delivered scope that SAP guarantees for future releases - this means that experimental features may be changed by SAP at any time for any reason without notice.
>
> Experimental features are NOT FOR PRODUCTIVE USE. You may not demonstrate, test, examine, evaluate or otherwise use the experimental features in a live operating environment or with data that has not been sufficiently backed up.

Prerequisites

Your administrator enabled the personalization feature for you.

Here's how it works

Start the personalization mode by choosing *Personalize App* in the user actions menu. A footer bar at the bottom of the page indicates that you are in the personalization mode. You can edit all UI elements that get

highlighted when you hover over or select them. Bear in mind that you cannot navigate in your app or interact with it (for example, make entries) in this mode.

> **i Note**
>
> If *Personalize App* doesn't show up in the user actions menu, or no UI elements get highlighted in personalization mode, please contact your administrator.

Add section	Choose ➕ *Add Section* (or right-click the section and choose *Add Section* from the context menu), select the available sections you want to add and choose *OK*.
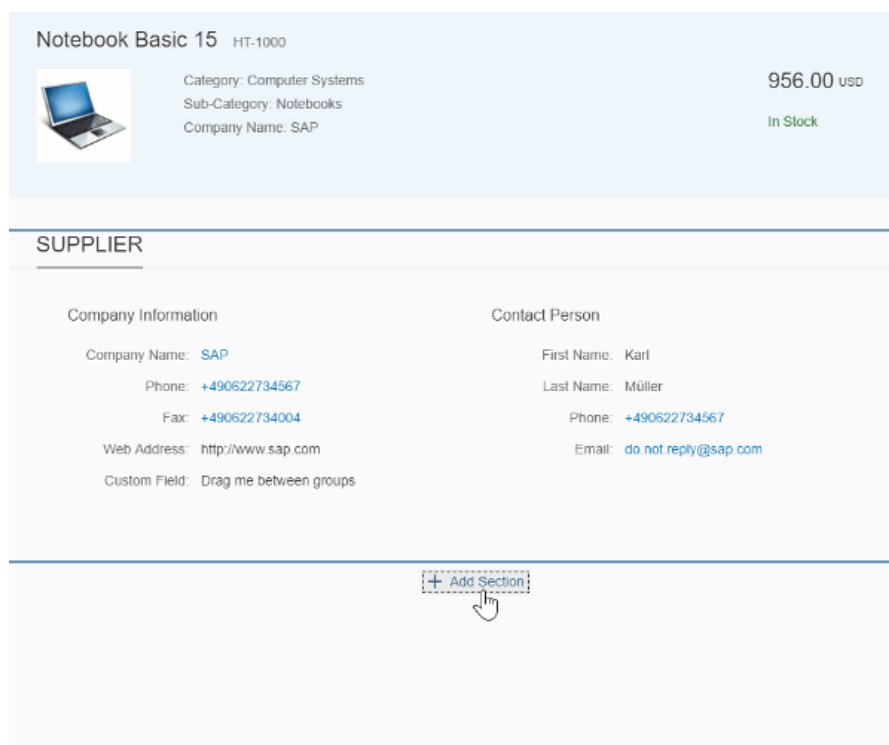
Remove section	Choose ⊗ (*Remove*) or right-click a section and choose *Remove* from the context menu. As the section is not deleted permanently but only removed from the screen, you can add it any time from the list of available sections.
Move section	Just drag a section and drop it on the section where you want to place it. Alternatively, you can cut the section (right-click it and choose *Cut* from the context menu), right-click the section where you want to place it and choose *Paste* from the context menu.

To exit the personalization mode and return to your running app, choose *Done* from the footer bar (you can also choose *Exit Personalization Mode* in the user actions menu).

1.4.4 Personalizing the Launchpad

As users you can personalize the launchpad to suit the way you work.

Follow the links to find out more about the different ways to personalize the launchpad. Note that there are different options to do this depending on the way your launchpad is configured, e.g. if you work with the home page or with spaces and pages. See Entry Page [page 416] for more information.

See sections Personalizing the Home Page [page 501] and Adding Apps to the Home Page [page 504] to learn how to adapt the home page to your needs.

If you work with spaces and pages, see sections Personalizing a Page [page 496] and Adding Apps to a Page [page 498]. Note that the personalization options might differ from the options you are used to from the classic home page.

For general information on how you can adapt the launchpad, see sections Share Apps and Create Bookmarks for Apps [page 507] and Managing Your Settings [page 509].

There is also a tutorial available that gives you a short interactive introduction to the most important personalization options in the home page.

1.4.4.1 Personalizing a Page

You can adapt and change your launchpad pages according to the way you want to work.

> **i Note**
>
> The ability to personalize the pages is enabled by your administrator. It is also important to know if you work in the spaces mode or in the classic home page. See Entry Page [page 416] for more information. How to edit the classic home page is described in section Personalizing the Home Page [page 501]. Please note that the personalization of pages does not support all features you know from the home page personalization.

When you first open the launchpad, you see one or more pages with predefined sections. In the edit mode, you can create new sections, and manage existing sections. Your personalizations are kept, even if the administrator changes the page afterwards. The tutorial Working with and Personalizing Spaces and Pages also shows you how to personalize a page.

How Can You Change the Layout?

You can drag and drop tiles to rearrange your page. You can also move tiles between sections. Note that the tile size might change when you rearrange a tile: When you drag a tile to a row with tiles of another height the tile size is adapted (if the new tile size is supported for this tile). An example: when you move a standard tile to a row that contains flat tiles, the tile is changed to a flat tile as well. When you want to drag a tile to the

link section to convert it to a link, you need to drag it next to another link. When the ⚲ symbol is shown, release the mouse button to insert the new link at that position.

Note, that drag and drop is not supported on mobile devices.

To make other changes to the page, such as moving sections or removing tiles, you need to switch to the edit mode. Select *Edit Current Page* in the user actions menu.

Now that you are in the edit mode, you can do the following:

- Tile actions

  - Rearrange or move tiles or links: Drag the tile or link to its new location—in the current section or in another section. Note that when you remove all tiles from a section, the empty section will no longer be shown in the page. It is still visible in edit mode and you can add tiles to it.
    Note that the tile size might change when you rearrange a tile: When you drag a tile to a row with tiles of another height, the tile size is adapted (if the new tile size is supported for this tile).

  - Change the tile size or convert a tile to a link: Click on ⠐⠐ *(Overflow)* and select the desired option, e g. *Convert to Flat Tile* or *Convert to Link* in the menu. Also see Tiles, Links and Tile Sizes [page 423] for a screenshot of the different tile sizes.
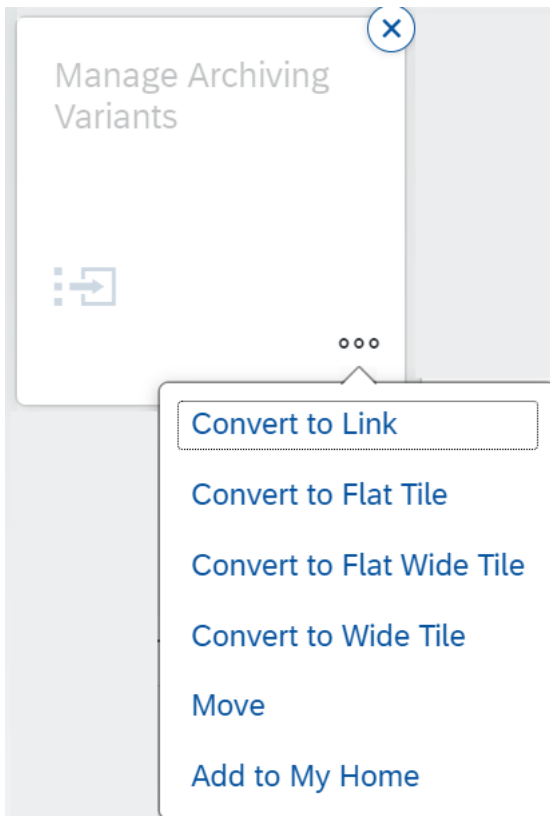
  - When My Home is enabled, you can also pin any tile from any page directly to it. Select *Add to My Home* in the tile context menu and this tile is added to the *Recently Added Apps* section on My Home. From there you can move it to any section on My Home.

- Section Actions

  - Add a section: In the space between existing sections or at the bottom, click *Add Section*. In the text box that opens, enter a name for the new section, and press `Enter`.

  - Add apps to a section: To add new apps to the page, click *Add Tile* to open the app finder where you can search for all the apps available to you. For more information, see Adding Apps to a Page [page 498].

- Rename a section: Click in the section name and edit it. To discard the change while the text box is in focus, press `Esc`.
- Move a section: Drag the section to the new location in the page.
- Hide a section: You can hide a section from the page. In the section header, click *Hide Section*. To show it again, click *Show Section*. Note that your administrator may have chosen to disable the option to hide sections.
- Delete a section: You can delete a section that you created in the personalization. In the section header, click *Delete Section*. The section is deleted and its tiles are removed from the page.
- Reset a section: You can reset a predefined section that you personalized back to its initial state. To do this, click *Reset Section*. If you have e.g. moved a tile to another section, and reset the first section, the tile will be contained in both sections.

> i Note
>
> Sometimes you see a *Recently Added Apps* section as first section. It contains the apps you have added to the page with the app finder or the bookmarking option. This is a locked section, so you can only drag apps from here to other sections but no apps to this section. When this section is empty, it will be deleted automatically.

Related Information

1.4.4.2 Adding Apps to a Page

When you work in spaces mode, you can add apps to any sections in a page, or to sections that you create.

> i Note
>
> You can only personalize your pages, when your administrator has enabled the edit mode in the launchpad configuration.

You use the app finder to search for apps and then add them to one of your pages (also see App Finder [page 427]).

Adding Apps in the App Finder

1. Select *App Finder* in the user actions menu.
2. Locate the app you want to add in the catalog. Apps are sorted into catalogs, so you can either browse all the catalogs or use the search option to find apps by their titles and keywords. You can also use the tag

filter to find apps from a specific catalog. If you do not see the search or filter options, your administrator may have disabled them. When you execute a search, the results apply only to the currently selected tab. When you change tabs, the search automatically runs in the new tab. Click ⊗ *(Clear)* to clear the filtered display.

3. Click the ✚ *(Add Tile)* or ✔ *(Edit Assignment)* icons below the app. The add tile icon indicates that the app has not yet been added to a page. The edit assignment icon indicates that the app is already assigned to at least one page.

4. A window opens with the available spaces and the pages that are assigned to the spaces. Select one or more pages.
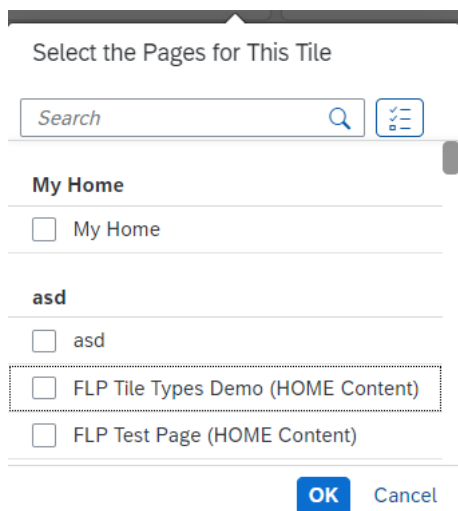
You can filter the list to show only the selected pages by clicking on the filter icon ⸬≡ . When you click *OK*, the app is added to a special section in the selected page: the *Recently Added Apps*. This section is shown as first section. It contains the apps you have added to the page with the app finder or the bookmarking option. This is a locked section, so you can only drag apps from here to other sections but no apps to this section. When the section is empty, it will be deleted automatically. When you return to the page, you can drag the app to your preferred section.

1.4.4.3 Importing Apps and Personalizations from the Classic Home Page to My Home

Learn how you can import your apps and personalized content to My Home.

> **i Note**
>
> This option is only available if you have worked with the home page before you switched to spaces and pages and if you have personalized your home page. This means, if you have added apps yourself, created groups, converted tiles to links, added bookmarks, etc. Then, you can import these adaptations to the new My Home.
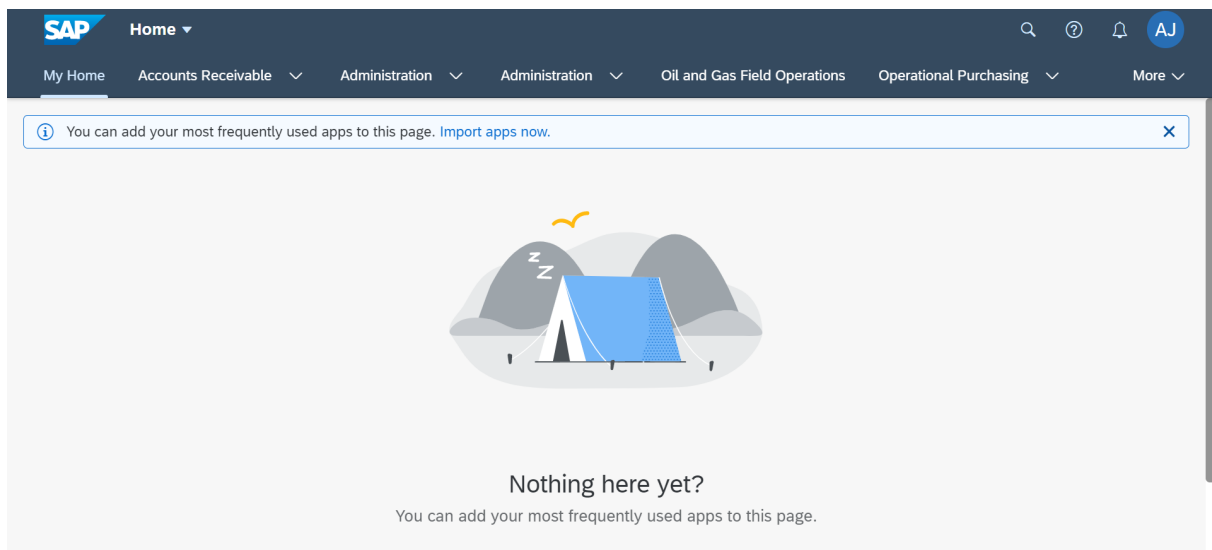
Empty My Home

When you open My Home the first time, it is empty.

How to import apps

- On the top is a message strip explaining how to import your apps. Click on *Import apps now* to open the import dialog.

- Select which groups you want to import to your My Home. All groups that were personalized in some form are preselected already.

> **→ Tip**
>
> SAP suggests that you only import your most important apps to create a rather lean page.

- Click on *Import* to start importing.
  When the import is finished, you see the content from the imported groups in sections that have the group names. You can add more apps with the app finder or bookmarks the same way as you add apps to any page.

Related Information

1.4.4.4 Personalizing the Home Page

End users can adapt and change their launchpad home page according to the way they work.

> **i Note**
>
> The ability to personalize the home page is enabled by your administrator.

When you first open the launchpad, you see predefined groups that have been configured by your administrator. You can also create new groups, and manage existing groups.

How can I change the home page?

Drag and drop tiles or links to rearrange your home page. You can also move tiles or links between groups. In addition, you can change tiles into links by dragging and dropping them in a link area. You can also convert a link into a tile with drag & drop. When a group doesn't have a link area yet, you can create a link only in the edit mode.

Note, that drag and drop is not supported on mobile devices.

There is also a tutorial available that gives you a five-minute-introduction to the most important personalization options.
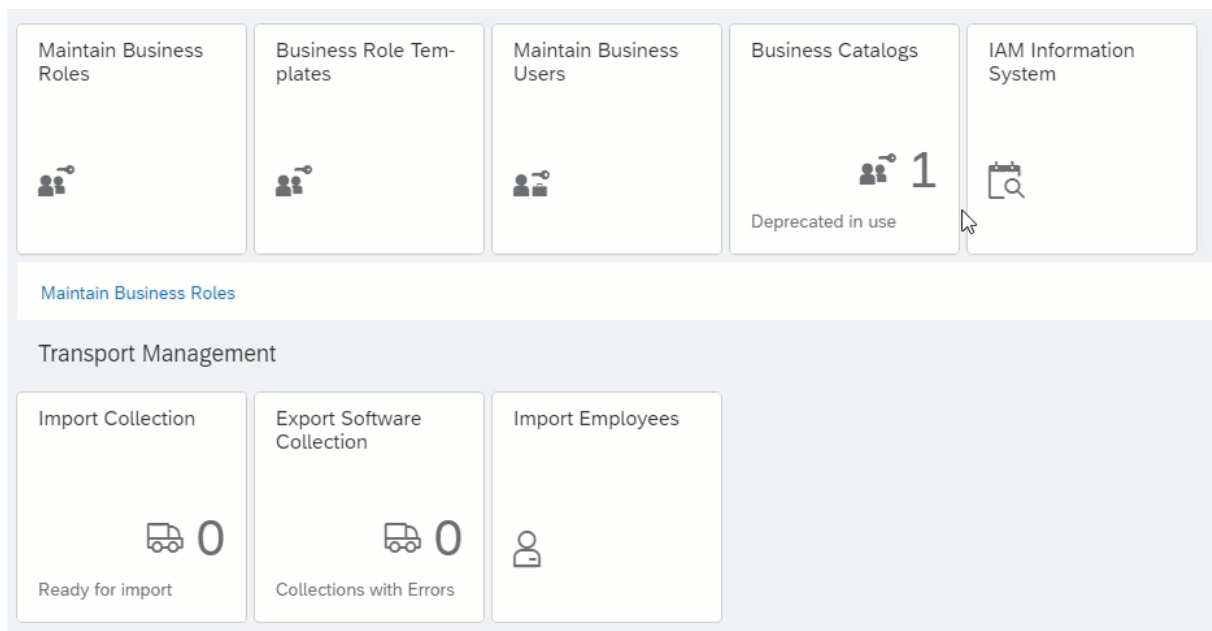
How to change the home page with drag & drop

> **→ Tip**
>
> If your home page is in tabbed mode, you can move a tile or link to a new group by dragging it to the target group in the group selection bar. The tile or link will be moved to the end of the target group.
>
> If you want to control where you want the tile to be positioned in the target group, drag the tile and hold it over the target group, and then after the new group is shown, release the tile in the precise location within the group.

To make other changes to your home page, such as moving groups, removing tiles, and renaming tiles, you need to switch to edit mode. To switch to edit mode, choose *Edit Home Page* in the user actions menu.

What can I change in the home page in edit mode?

Now that you are in edit mode you can do the following:

Action	Description
Rearrange or move tiles and links	Drag the tile or link to its new location—in the current group or in another group. You can also click or tap the tile or link and use the *Move* button to select the group to which you want to move it. This is useful if the target group isn't close to the original group in the home page.
	Note that when you remove all tiles from a group, the empty group will no longer be shown in the home page. It is still visible in edit mode and you can add tiles to it.

Action	Description
Add a group	In the space between existing groups or at the bottom of the home page, click or tap *Add Group*. In the text box that opens, enter a name for the new group, and press `Enter`. The new group, with a tile placeholder, is added to the home page.
Add apps to a group	To add new apps to the home page, click *Add Tile* to open the app finder where you can search for all the apps available to you. → Remember You cannot add tiles to locked groups. For more information, see Adding Apps to the Home Page [page 504].
Delete a group	You can delete a group that you created. In the group header, click or tap *Delete*. The group is deleted and its tiles and links are removed from the home page. Only groups created by you have a *Delete* button.
Hide a group	You can hide a group from the home page. In the group header, click or tap *Hide*. To show the group again, click or tap *Show*. i Note Your administrator may choose to disable the option to hide groups.
Reset a group	You can reset a predefined group that you personalized, back to its initial state. To do this, in the group header, click or tap *Reset*.
Rename a group	Double-click or double-tap the group name, edit it in the text box, and press `Enter`. To discard the change while the text box is in focus, press the `Esc` key.
Move a group	Drag the group title to the new location in the home page.
Rename a tile	Click or tap the tile and choose *Edit Tile Information*. In the *Tile Information* dialog box that opens, enter a new title. You can also provide a subtitle and additional information.
Rename a link	Click or tap the tile and choose *Edit Link Information*. In the *Link Information* dialog box that opens, enter a new title. You can also provide a subtitle and additional information.
Convert a tile into a link	Click or tap the tile and choose *Convert to Link*.
Convert a link into a tile	Click or tap a link and choose *Convert to Tile*.
Remove a tile or link	Click or tap the ✕ (Remove) button at the top right corner of the tile or link.

i Note

Predefined groups can be locked to prevent personalization. In edit mode, locked groups are displayed above the *My Home* group and cannot be modified.

To exit edit mode, click *Close* at the bottom of the home page or click *Exit Edit Mode* in the user actions menu.

What other changes can I make?

There are other settings for the home page that you can change, such as changing the display mode of content and setting a new theme.

For more information, see Managing Your Settings [page 509].

Related Information

User Actions Menu [page 424]

1.4.4.5 Adding Apps to the Home Page

You can add apps to any of the unlocked groups in your home page, or to new groups that you create.

Prerequisites

To personalize your home page, your administrator must first enable edit mode in the launchpad configuration.

Context

You use the app finder to search for apps and then add them to your home page. The app finder typically comprises the *Catalog* tab and the two SAP Easy Access menus, represented by the *SAP Menu* and *User Menu* tabs. For more information about these tabs, see App Finder [page 427].

> **i Note**
>
> Depending on how the administrator has configured your launchpad, you may not see all of these tabs. If your administrator has disabled both the SAP Menu and the User Menu, then the tab control is completely hidden, and you can see the *Catalog* repository only.

> **i Note**
>
> If you work in the spaces mode, this works differently. See Entry Page [page 416] and Adding Apps to a Page [page 498] for additional information.

There are a few ways to add apps using the app finder:

Action	More Information
Click *App Finder* in the user actions menu.	This option allows you to add apps to one or more groups directly in the home page without going into edit mode. You can also edit existing group assignments and create new groups on-the-fly. For more information, see Adding Apps from the Catalog in the App Finder [page 505] and Adding Apps from the SAP Easy Access Menus in the App Finder [page 506] below.
Click *Edit Home Page* in the user actions menu, and then *Add Tile* in a specific group.	This option allows you to add and remove apps one group at a time. For more information, see Adding Apps from Groups in the Home Page (Edit Mode) [page 507] below. → Tip In edit mode you can make other changes to your home page, such as moving, renaming, hiding, and deleting groups. See Personalizing the Home Page [page 501].

→ Remember

You cannot add tiles to locked groups.

Adding Apps from the Catalog in the App Finder

Procedure

1. Click *App Finder* in the User Actions Menu [page 424].
2. From the *Catalog* tab, locate the app that you want to add to your home page.

   Apps are sorted into catalogs, so you can either browse all the catalogs or use the search option to find apps by their titles and keywords. You can also use the tag filter to find apps from a specific catalog.

   i Note

   If you do not see the search or filter options, your administrator may have disabled them.

   When you execute a search, the results apply only to the currently selected tab. When you change tabs, the search automatically runs in the new tab. Click ⊗ *(Clear)* to clear the filtered display.

3. Click or tap the [📌] (Add Tile) or [📌] (Edit Assignment) icons below the app.

> → Tip
>
> The [pin icon] icon indicates that the app has not yet been added to a group. The [pin icon] icon indicates that the app is already assigned to at least one group.

4. Add the app to one or more groups in the *Add to Groups* dialog box. You can also create a new group and add the app to it, or you can remove an app from a group by clearing its checkbox in the same dialog box.

Adding Apps from the SAP Easy Access Menus in the App Finder

Procedure

1. Click *App Finder* in the User Actions Menu [page 424].
2. Click either *SAP Menu* or *User Menu*.

   The *SAP Menu* and the *User Menu* tabs are also called the SAP Easy Access menus. Each tab displays different apps in folders, as configured by your administrator. The procedure of adding apps to your home page, is the same for both menus.

   > i Note
   >
   > The SAP Easy Access menus are not supported on mobile devices.

3. If your administrator has configured more than one system, when you open the SAP Easy Access menu for the first time, select a system.

   > → Tip
   >
   > You can always select a different system by clicking *Select System* next to the name of the current system in the navigation pane.

   In the navigation pane on the left, you can now see the folders available in the selected system.

4. Navigate to the folder containing the apps that you want to add to your home page.

   You can also use the search option to find apps by their titles and keywords. If you cannot find the search option, it may be have been disabled by your administrator.

   > → Remember
   >
   > When you execute a search, the results apply only to the currently selected tab. When you change tabs, the search automatically runs in the new tab.

5. Click or tap the [pin icon] (Add Tile) or [pin icon] (Edit Assignment) icons below the app.
6. Add the app to one or more groups in the *Add to Groups* dialog box. You can also create a new group and add the app to it, or you can remove an app from a group by clearing its checkbox in the same dialog box.

Adding Apps from Groups in the Home Page (Edit Mode)

Procedure

1. Click *Edit Home Page* in the User Actions Menu [page 424].
2. In the home page, click the plus icon in a group (it's the last tile in the group) to add a tile.
3. In the *Catalog*, *User Menu*, or *SAP Menu* tab, click the [icon] (Add to Group) or [icon] (Remove from Group) icons below the app.

   The selected app is added to or removed from the group.

1.4.4.6   Share Apps and Create Bookmarks for Apps

You can share apps or search results with your colleagues or create your own tiles with access to specific app or search data.

You can choose between three options:

- Share this information on SAP Jam (if available in your system).
- Send an e-mail with a link to the search or app and your currently selected data to a colleague.
- Save the app or search as a new tile in your launchpad.

Which options are available depends on the system configuration and the specific app. If the share icon [icon] *(Share)* is available, click on it to see what you can do.

Start a search in the launchpad or open the app and select your data (e.g. by filtering or searching). Then select one of the sharing options.

Share on SAP Jam

This opens a window where you select a group in SAP Jam where you add a link to this app or search. If you have no group, you need to create one first.

Send E-Mail

An e-mail draft is created that already contains a link to this app or search and a subject. Add the receipient and some additional explaining text, if necessary. Click on *Send*.
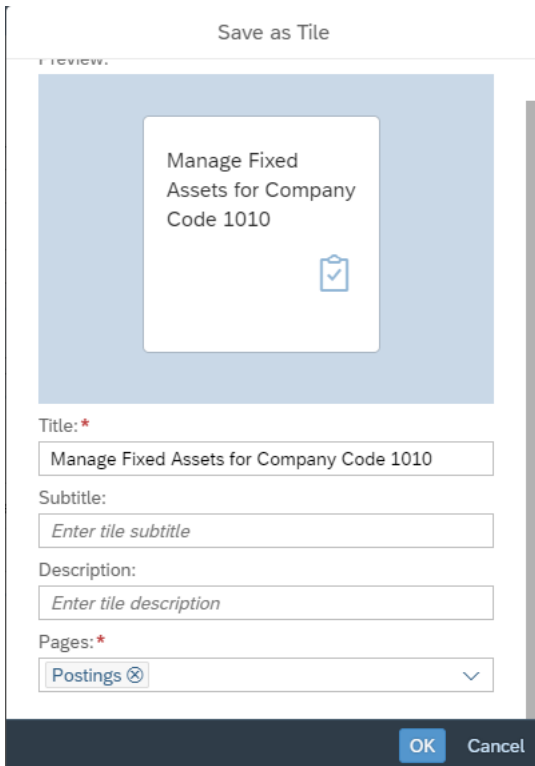
Save as Bookmark

> **i Note**
>
> This option is handled differently if you work with spaces and pages (see Entry Page [page 416] for more information). You cannot search for a personalized bookmark. We describe the differences below.

You can save the selected data as a new tile that then directly opens the selected data in the app or performs the saved search.

1. Click on the share icon ⤴ *(Share)* and select *Save as Tile*.

Creating a New Tile as Bookmark

2. Enter a title for the tile. For an app you should enter a new name to distinguish it from the existing app tile. You can enter a subtitle and a description for the tile.
3. Then you select where the tile should be added.
   - If you work with the home page, you can select a group in the field *Home Page*.
   - If you work with pages and spaces you select one or more pages. The tile is added to the *Recently Added Apps* section on the selected pages. This is a locked section, so you can only drag apps from here to other sections but no apps to this section. When the section is empty, it will be deleted automatically. When you return to the page, you can drag the app to your preferred section.

1.4.4.7 Managing Your Settings

Users maintain launchpad settings in the *Settings* dialog of their SAP Fiori launchpad.

You can find, view, and change the visual appearance of your launchpad, such as the theme, and configure various functional aspects, such as search settings, user default values, and usage analytics.

Where do I manage my settings?

In the user actions menu, choose *Settings*. A settings dialog opens for each of the relevant settings that you want to view or edit.

What settings can I view or change?

Depending on your configuration, there are several tabs that show different settings parameters. The figure shows an example of the theme settings.

> **i Note**
>
> Some settings are read-only and cannot be changed directly in the *Settings* dialog. You may also not see all the settings; they may either be disabled by your administrator or not supported on your platform.
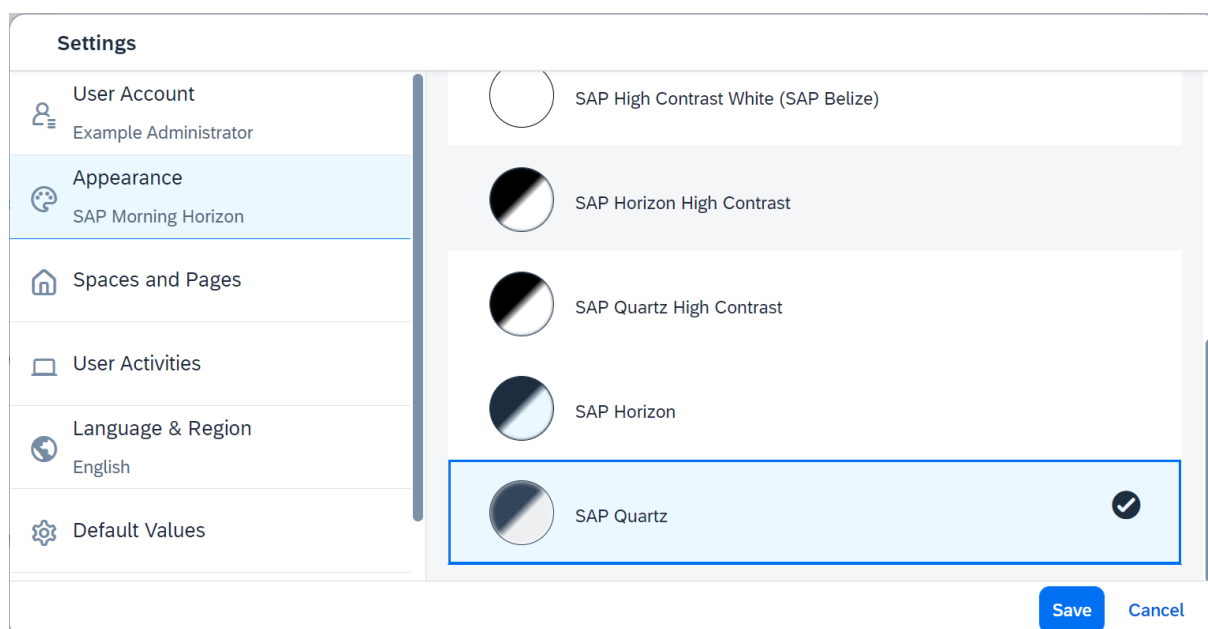
Setting	Description
User Account	View your account settings, such as name, e-mail, and server address.

Setting	Description
Appearance	Change your launchpad theme and display settings. See Themes [page 458] for more information.
	When using a hybrid device that combines touch and mouse events, you can turn on the *Optimized for Touch Input* feature which increases the size and spacing of the controls on your screen so they are easier to use with your fingertip.
	You can choose if the tiles should be displayed in a small or a large size.
	When you work with the Quartz or Horizon theme, you can use dark mode features for the launchpad. When you use a browser that supports automatic color scheme detection (e.g. Google Chrome in a current version), the current dark mode setting in your operating system settings will be reflected in your launchpad. Activate *Dark Mode* to enable or disable the automatic dark mode detection for the launchpad. If you use a browser that does not support automatic color scheme detection, you can select a dark theme instead. Note that your administrator can disable this feature.
Home Page	Choose whether to show the content of all your groups all at once on the home page or to show only one group at a time. The group selection bar at the top of the home page appears identical for both options; however, they behave differently:
	• *Show all content*: Behaves like an anchor bar; choosing a group at the top of the home page scrolls to the content assigned to the group.
	• *Show one group at a time*: Behaves like a tab bar; choosing a group displays the content assigned to the group.
	For large amounts of apps, showing one group at a time may be better.
	i Note If you work with spaces and pages, this tab is not available. See next line for more information.
Spaces and Pages	i Note Check or change the spaces setting and My Home setting for the dashboard, as described in Maintaining Settings for Spaces [page 512]. If spaces are enabled, you see your apps in sections on one or more pages. If the spaces are disabled, you see the classic home page. See Entry Page [page 416] for more information on the differences.
Language and Region	View your login language, as well as date and time formats.
	If the administrator has allowed it, you can select a different language for the launchpad and select your preferred date and time formats. See Maintaining Your Language and Regional Settings [page 512] for more information.

Setting	Description
User Activities	Specify whether to track user activity.
	Set *Track my recent activity and frequently used apps* to *NO* to stop tracking user activity. The entries *Recent Activities* and *Frequently Used* are then removed from the user actions menu.
	Click *Clear* to delete your user activities. If *Track my recent activity and frequently used apps* is set to *YES*, the *Recent Activities* and *Frequently Used* lists are cleared, but the system continues to track your activity and displays new items in the user activity lists.
	For more information, see Working with Recent Activities and Frequently Used Apps [page 431].
User Profiling	Turn on *Personalized Search* to allow the system to collect data about your search behavior. This data is used to improve future search results. You can also delete this collected data by clearing your browser history.
Default Values	View and edit default user-specific parameter values used when launching your apps.
	For more information, see Maintaining Your Default User Values [page 514].
Notifications	Personalize various settings that are related to the notifications provided to you in the launchpad.
	For more information, see Setting Notification Preferences [page 516].

→ Remember

Click *Save* to apply your changes.

When an entry you make produces an error or a warning, a message icon is displayed in the lower left corner of the user settings window. The icon shows the number of problems. Click on the icon to see detailed information about the warnings or errors. The following figure shows an example.
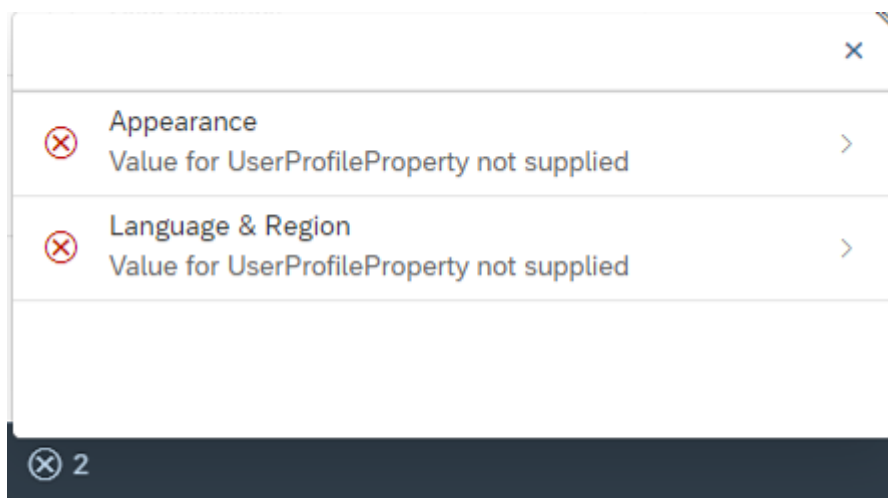
Related Information

1.4.4.7.1 Maintaining Settings for Spaces

Activate spaces and My Home in the launchpad.

On this tab you can switch to the spaces layout. If spaces are enabled, you see your apps in sections on one or more pages. If the spaces are disabled, you see the classic home page. See Entry Page [page 416] for more information on the differences.

When spaces are enabled, one or two settings can be available. Which options can be set, depends on your system setup.

- *Show My Home*: By default this is active. You can add your preferred apps to My Home and adapt it to your needs (see My Home [page 420]). When you don't want to work with My Home, you can disable it here. Then, the first page of the first space will be displayed when you start the launchpad or click on *Home*.
- *Show Import Message*: Activate this to show the message strip for import from the home page again on My Home. Then you can start importing your personalization of the home page on My Home. Note that this option is only shown when you have enabled My Home and have personalized the classic home page before (also see Importing Apps and Personalizations from the Classic Home Page to My Home [page 499]).

1.4.4.7.2 Maintaining Your Language and Regional Settings

You can see and change your language and region settings for the launchpad in the *User Settings* dialog.

Open the *User Settings* dialog from the user actions menu and go to the tab *Language & Region*. Here you see your login language, as well as the currently set date and time formats. Depending on the system settings, you can change the language and the date and time formats to be used.

User Account
John Administrator

Appearance
SAP Quartz Light

Home Page

Spaces

User Activities

Language & Region
English

Default Values

Language & Region

Language and Region:	English	⌄
Date Format:	DD.MM.YYYY (Gregorian Date)	⌄
Time Format:	24 Hour Format (Example: 12:05:10)	⌄
Time Zone:	UTC+1(+ 1 hour)	⌄
Decimal Format:	1.234.567,89	⌄

After you save your settings, the browser will refresh for the new settings to take effect.

Save   Cancel

Language and Regional Settings in the User Settings Dialog

- If the administrator has allowed it, you can select your preferred language (and region, if applicable) in the field *Language and Region*. You can also select *Browser Language* to use the language you use in your web browser.

  > **i Note**
  >
  > When you are allowed to select a language when signing in to the launchpad, this will overwrite the settings here the next time you sign in.

- When the fields *Date Format* and *Time Format* are shown (depending on the backend capabilities), you can select your preferred date and time format.
- Time Zone: For classic UI apps (e.g. Web Dynpro-based or SAP GUI apps), you should select your local user time zone. SAP recommends that you set the same time zone here as in the client to ensure a consistent behavior.
  When you don't know the type of an app, you can check it in the *About* dialog in the field *ID of the Application Framework* (see Information About App and Launchpad Version [page 443]). If you aren't sure about the settings, please contact your administrator.
  Note that the time zone setting here is only saved on the frontend server. It doesn't apply to SAPUI5-based SAP Fiori apps. These apps automatically use the browser setting for the time zone.
- Select your preferred decimal format in the field *Decimal Format*.

The launchpad is reloaded after you've saved your changes.

1.4.4.7.3 Maintaining Your Default User Values

Some applications can be launched with user-specific default values. You can view these values and edit the default values if necessary.

If you update any of the available values, the new value will be used in all applications that reference that value. You can save several value sets and switch between them.

Currently used default values

Edit currently used values

> **i Note**
>
> If you do not see this option in your launchpad, then your administrator may not have enabled it.

1. Click the *Settings* option in the user actions menu. Select *Default Values*. You see the *Saved Values*, which are the values used as default values in your system. You can change any displayed value, or just view the values.
2. Some values allow you to specify additional values. Click *Additional Values*.
3. If you have changed any of the values, a star is shown next to *Saved Values*. Click *Save* to save your changes to the applied value set.

Applying values from a stored value set

You can apply so-called views (that represent value sets) which were saved as your personal value sets or saved as public value sets.

1. Open *My Views* by clicking on the arrow next to the value set. The list shows all value sets that are marked as favorites. This usually comprises your own defined value sets and public value sets you selected as favorites.
2. If you don't see a specific value set, click on *Manage*. The *Manage Views* list shows all available value sets. Click on the star to add a value set to *My Views*.
3. To change your default values. select a different value set from the list. Check the values and then click on *Save* to import the values to the currently used values.
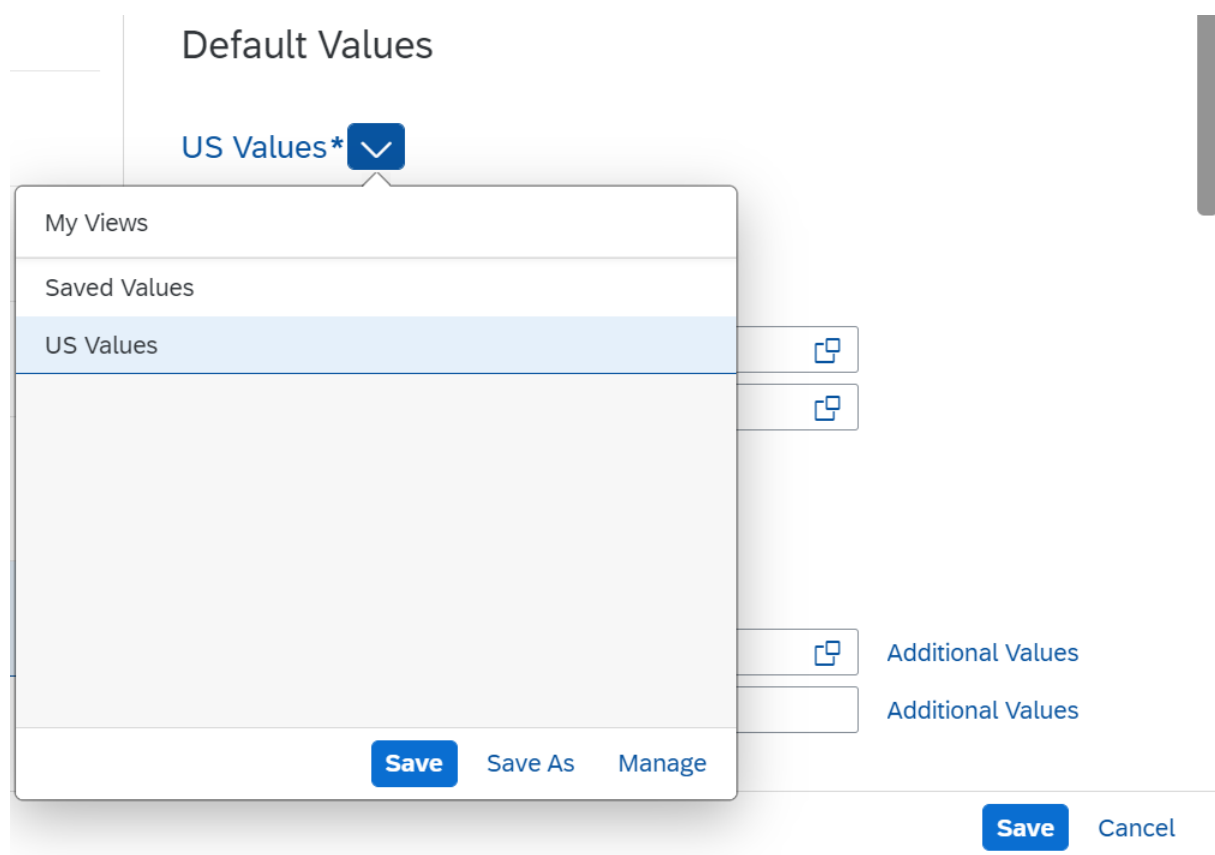
> → Tip
>
> This can e.g. be used when you normally work with a standard parameter set but need a specific parameter set for a specific business scenario. You can create two different parameter sets for the different business scenarios. You then usually work with the standard set and apply the specific parameter set when necessary. When you are done with the specific case, you apply the standard parameters again.

Saving a new value set

You can create a new parameter set, e.g for a specific business scenario.

1. Click on the arrow next to the value set name, e.g. *Saved Values* as in the image above.
2. In the opening dialog, click on *Save As*.
3. Enter a new name for the parameter set in the field *View* and select, if the value set should be public. Public means that other users in the system can use the parameter set as well.
4. If necessary, add a transport.
5. Enter the new values or change the existing values.
6. Click on the arrow next to the parameter set name and select *Save* in the opening dialog to save your changes.
7. If you want to work with the new value set directly, click on *Save* to apply it to the *Saved Values*. Those values are now activated and will be used in your system.

Managing value sets

In the Manage Views you can add value sets to your favorites. In addition, you can rename or delete your own value sets. Note that the *Saved Values* cannot be edited or deleted as they always reflect the currently active used values.

1. Click on the arrow next to the value set name, e.g. *Saved Values* as in the image above.
2. In the opening dialog, click on *Manage*.

Manage Views		
Search		
View	Sharing	Created By
★ **Saved Values**	Public	SAP
		Save Cancel
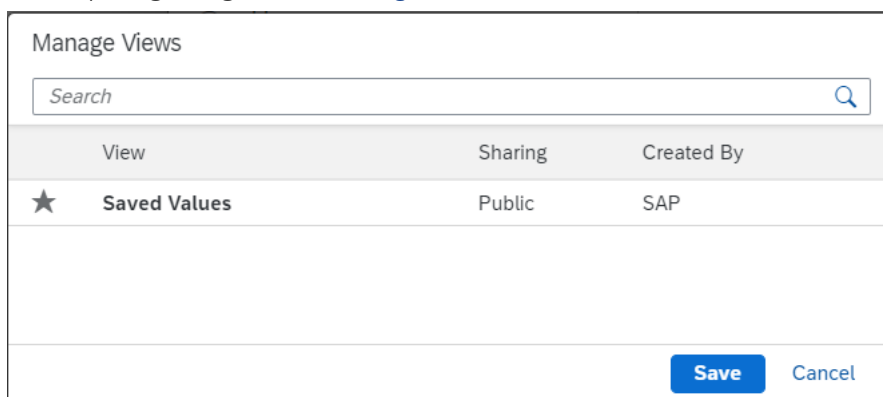
3. When you want to add a value set to your favorites, click on the star. The value set will be added to *My Views*.
4. When you want to rename the value set, enter a new name in the field *View*.
5. When you want to delete a value set, click on the x in the table row. Normally, you can only delete your own defined value sets. Only users with additional rights are allowed to edit or delete public value sets created by other users. It is good practice to edit or delete only value sets you created yourself.
6. Click on *Save*.

1.4.4.7.4 Setting Notification Preferences

The launchpad displays notifications that are related to your business role, such as leave requests or purchase orders. Usually there are different notification types available, e.g. notifications in the launchpad, mobile or e-mail notifications. You can personalize various preferences relating to which notifications you want to receive, and how and where they are displayed.

Some of the notification preferences that you set affect all types of notifications, while others are specific to a notification type.

To access your notification preferences, in the user actions menu, open ▶ *Settings* ❯ *Notifications* ❚.

General Preferences

The setting *Show High-Priority Notification Alerts* affects all notifications, regardless of their notification type. If you are working in an app or the app finder, a popup notifies you immediately when you receive a high-priority notification. When this setting is off, you will only see the high-priority notification when you are back in the home page view.

Preferences for Specific Notification Types

Below the general preferences is a table listing the different types or categories of notifications that you have received so far. This table is empty until you receive your first notification. The list expands over time as you receive different notification types relating to your business role.

You can set the following preferences separately for each notification type:

Setting	Description
High Priority	Select this option to set the priority for all notifications of the selected notification type to high. This overrides the current priority of the notification set by the notification channel. The updated priority is only set for that specific user and it is not pushed to the backend system or to other users who may have received the same notification. This setting is useful when used in combination with the *Show High-Priority Notification Alerts* setting. → Remember When you enable this setting, only new notifications are updated with the high priority flag. Notifications you received earlier will not be updated. Similarly, when you disable this setting, any earlier notifications that were flagged as high priority will not return to their original priority level.
Mobile Notifications	Select this option to also receive native notifications on your Android/iOS mobile device. This option is available for some notification types only; if this option is disabled, then the notification type cannot be broadcast to your mobile device. Your administrator also needs to configure support for mobile devices and you need to install the required client app on your device. Contact your administrator for more details.

Setting	Description
E-Mail	Select this option to receive an additional e-mail for a notification. This option is only available for some notification types; if this option is disabled, no e-mail notifications can be sent. If your e-mail address is not provided in the system, the column is not displayed and you cannot enable e-mail notifications. To check your e-mail address open ▶ *Settings* ▶ *User Account* ◀.
Enable	Turn off this option to stop receiving notifications in the launchpad and mobile/e-mail notifications for the selected notification type. → Remember When you re-enable a notification type, you will see new notifications received from that time onwards. You won't receive older notifications that were sent to you while the notification type was disabled.

Related Information

[Working with Notifications [page 438]](#)

1.5   Developer Guide

This guide provides information and code examples for developers who create applications for the SAP Fiori launchpad or who extend the functionality of the launchpad.

This image is interactive. Hover over each section for a description. Click highlighted sections for more information.
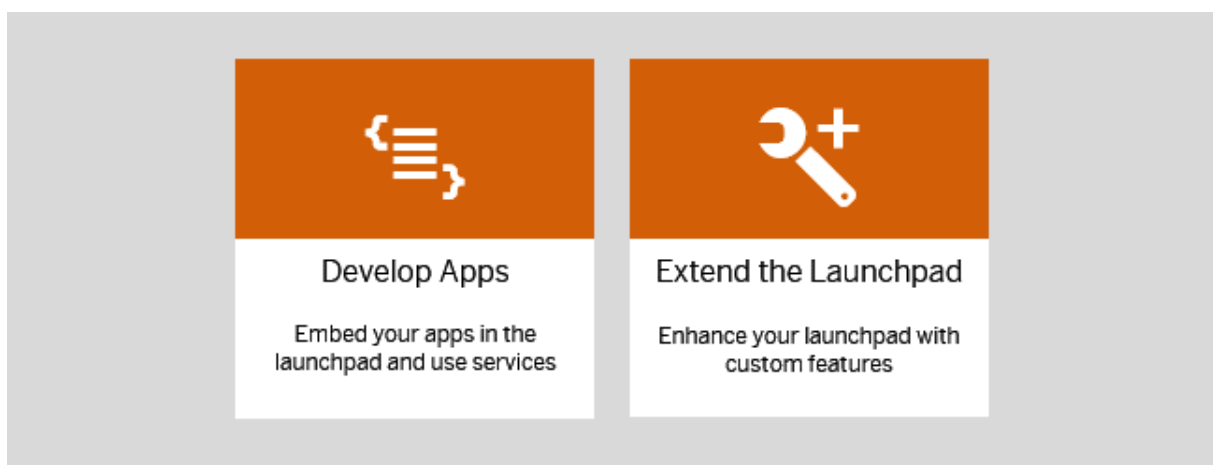
- [Developing Applications for the Launchpad [page 519]](#)

- Extending the Launchpad [page 575]

Related Information

1.5.1 Developing Applications for the Launchpad

The SAP Fiori launchpad is a runtime shell environment that hosts SAP Fiori apps, and provides the applications with services, such as navigation, personalization, embedded support, and application configuration.

SAP Fiori apps are based on SAPUI5 and are developed with the UI development toolkit for HTML5 (SAPUI5). However, there are some important aspects to consider when developing apps specifically for the SAP Fiori launchpad, and additional service APIs increase your productivity.

This guide provides an overview and more detailed information on the following topics:

- Launchpad Architecture: Overview [page 519]
- Embedding Applications in the Launchpad [page 522]
- Navigation Concepts [page 524]
- Developing Navigation [page 529]
- Local Sandbox Environment for the SAP Fiori Launchpad [page 549]
- Typical Use Cases [page 551]
- Best Practices for Developing Applications [page 574]

You can find further information at the following locations:

Description	Reference
Information about launchpad services	SAPUI5 Demo Kit: Controls and then navigate to ▶ *API REFERENCE* ▶ *sap.ushell* ▶ *services* ▶
Information about available services	http://help.sap.com/nw-uiaddon and then navigate to ▶ *JavaScript Docs for SAP NetWeaver User Interface Services* ▶

1.5.1.1 Launchpad Architecture: Overview

The major technical building blocks and how apps are embedded.

The SAP Fiori launchpad is based on the unified shell architecture. The guiding principle of the unified shell is to have a single, platform-independent, client-side runtime environment that can be hosted on different server

platforms, for example AS ABAP and SAP Cloud Platform. This means that the shell offers unified services with platform-independent interfaces (APIs) to the hosted apps and shell components. Services that need platform-specific data handling or connection management utilize platform-specific adapters.

Visualization of the shell is independent of the shell services. Apps are embedded in a so-called application container. This application container is an independent reuse component, so the embedding aspect is decoupled from the renderer. The application container can host SAPUI5 components, Web Dynpro ABAP applications, and SAP GUI for HTML transactions. The central shell container manages the shell services and renderers. It utilizes a runtime configuration that defines the implementations for services, adapters, and the shell renderer, as well as global settings like theme, language, system, and user data.

The runtime configuration is fed by the following settings:

- Static configuration settings in the hosting HTML page
- Dynamic configuration data read from the front-end server during startup
- Dynamic settings passed as query parameters in the URL

Finally, all described JavaScript components are embedded into a single HTML page. The SAP Fiori launchpad implementation of the AS ABAP front-end server contains a standard page called `FioriLaunchpad.html`.

The following figure shows the described building blocks of the SAP Fiori launchpad architecture.
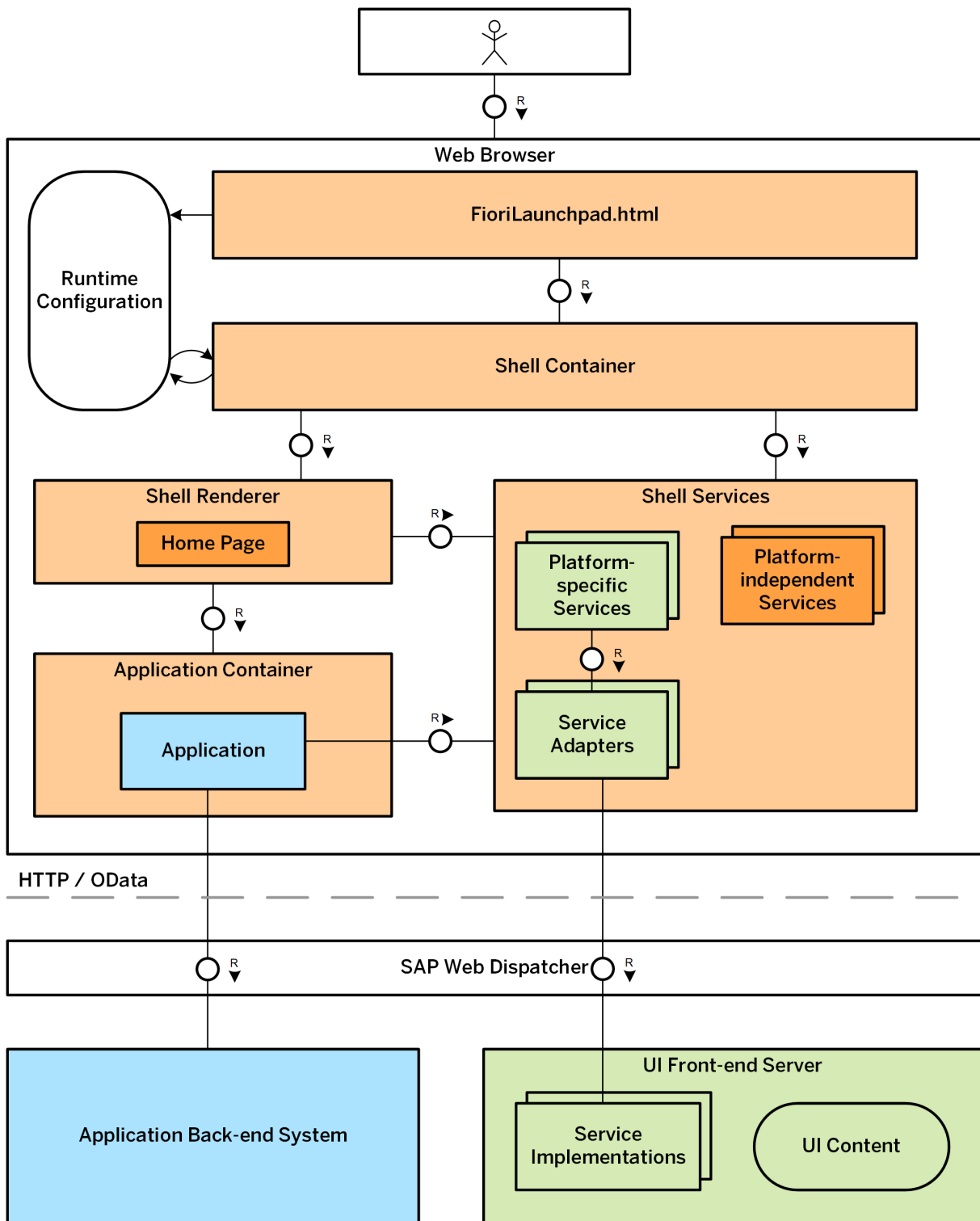
Related Information

Initial Setup of the Launchpad [page 10]

1.5.1.2 Embedding Applications in the Launchpad

Applications embedded in the SAP Fiori launchpad can be based on SAP GUI for HTML or Web Dynpro ABAP, or they can be SAPUI5 applications.

Applications based on SAP GUI for HTML or Web Dynpro ABAP can be integrated into the launchpad using configuration. For more information, see Setting Up Navigation [page 127].

For SAPUI5 applications, a tighter integration is possible. SAPUI5 applications are implemented with the same UI technology as the SAP Fiori launchpad and you can embed them directly in the launchpad by using DOM injection. When users navigate to the application, it is displayed in-place. This approach also allows smooth, animated UI transitions and the reuse of shared components, like shell services for navigation and personalization.

To embed SAPUI5 applications into the launchpad, these must be implemented as SAPUI5 components. These applications must not require to be started with their own HTML page.

Embedding SAPUI5 applications into the launchpad makes the lives of SAP Fiori application developers easier in various ways:

- SAPUI5 settings for language, date format and number format are initialized by the launchpad, using configuration settings from the front-end server. You do not need to set these values.
- Theme handling is provided by the launchpad.
- Cache busting is handled by the launchpad.
  The cache buster for the SAP Fiori launchpad and SAP Fiori apps implements the SAPUI5 cache-buster mechanism. If you develop SAP Fiori apps, you do not need to do anything for cache busting. For more information, see Cache Buster for SAP Fiori Launchpad and SAP Fiori Apps [page 367].

Related Information

Embedding SAPUI5 Applications [page 522]

1.5.1.2.1 Embedding SAPUI5 Applications

Applications are embedded into the SAP Fiori launchpad as self-contained SAPUI5 components using DOM injection.

The application container identifies the application using the name of the SAPUI5 component. Based on the name of the SAPUI5 component, the server-side application index infrastructure provides the application URL as well as any dependencies that need to be loaded.

The application container registers the component namespace as the module path for the application URL. For more information, see in the SAPUI5 documentation. As this is a global setting, it is essential that you use fully qualified names for the applications modules, for example, component and views.

File Structure

The SAPUI5 component is defined in the `Component.js` file, which should be located in the root folder of your application. For more information, see in the SAPUI5 documentation.

Descriptor for Applications, Components, and Libraries

The descriptor provides a central location for metadata associated with an application, an application component, or a library. For more information, see .

Attributes Used by the SAP Fiori Launchpad

The SAP Fiori launchpad evaluates the following attributes of the descriptor for applications, components and libraries:

- `title`
- `icons`
- `i18n`

Make sure that these attributes are maintained in the descriptor.

Path Resolution

For all properties that reference files, there are several ways to define them:

- Absolute reference, typically starting with "/", meaning that it is an absolute path on the same host
- Relative reference pointing to a location outside the application, starting with "../"
- Relative reference pointing to a location within the application, starting with any file or folder name below the root folder of the application

The SAP Fiori launchpad ensures that relative references are relative to the location of the `Component.js` file, and **not** relative to the location of the HTML file displayed by the web browser. This is consistent with the behavior for other references in component metadata, for example the `includes` property.

Passing Parameters to Components

The shell passes startup parameters to the component instance as part of the `componentData`, see getComponentData. They are set in the `startupParameters` property and always passed as arrays. The parameters can either be set as static query parameters in the component URL (as part of the launchpad configuration) or passed dynamically during cross-application navigation as part of the shell's URL hash.

Related Information

1.5.1.3   Navigation Concepts

SAP Fiori launchpad supports intent-based navigation and navigation based on URL fragments as well as cross-app and inner-app navigation.

The SAP Fiori launchpad is displayed in a web browser and offers the web-like navigation experience users are accustomed to. The user can for example, create bookmarks and use the web browser's back and forward buttons. Links for navigation within an app or to another app are HTML links, meaning that browser functions like *Open in New Tab* can be used out-of-the-box.

The URLs in the browser address bar are human-readable. They express a navigation intent that users can relate to. Users can send the URL of launchpad apps to other users by e-mail, and if you use the capabilities of the launchpad to their full potential, users with different roles who navigate to the same URL may see different content.

When starting the SAP Fiori launchpad, the user can access apps by clicking on the respective tile on the homepage or the user can search for an app. This navigation is handled within the same web app because all apps are hosted in one document: The `FioriLaunchpad.html` file bootstraps the unified shell, which provides the services, the JavaScript APIs, that are common for all apps, independent of their server platform. The `FioriLaunchpad.html` file is the only HTML document that the browser loads and, thus, it is also the only SAPUI5 root application. All following interactions are implemented by dynamic modifications of the hosting HTML document using JavaScript.

From a technical point of view, SAP Fiori apps are SAPUI5 components which are loaded into an application container. They are dynamically injected into the DOM of the `FioriLaunchpad.html` file.

Related Information

1.5.1.3.1   Navigation Based on URL Fragments

URL fragments can be used for inner-app- and cross-app navigation in the SAP Fiori launchpad.

The following URL shows an example of a navigation based on fragments of a standard URL. The URL part beginning with the hash (#) character is the fragment that is used for navigation:

`http://<server>:<port>/<path>/FioriLaunchpad.html#<fragment>`

> **i Note**
>
> In web browsers you may have to encode some parts of the URL.

As you load only one HTML page from the server, all information about the apps to be loaded has to be encoded in the URL fragment. To do the actual navigation, the following two options exist:

- Setting the browser URL fragment explicitly with a JavaScript call
- Implicitly by rendering an href-based link containing the relative URL with the hash fragment. In this case, the hash change is performed by the browser itself when the user clicks on the link.

The unified shell services provide a listener which is registered for the hash change event and loads the app according to the information encoded in the URL hash.

1.5.1.3.2 Intent-based Navigation

Intent-based navigation allows you to decouple the navigation triggers from the actual navigation targets, which can be flexibly configured for different roles and device types.

Rather than directly encoding the name of the target app into the URL fragment, app developers provide a navigation intent. An intent expresses what you want to do next (rather than how to do it). The following example shows an intent consisting of a combination of a semantic object and an action:

Example:

`http://<server>:<port>/<path>/FioriLaunchpad.html#Employee-display`

This approach allows you to decouple the intent, which is coded into the source app, from the actual navigation target, see the figure below. The target app can be defined using configuration in a so-called target mapping that maps an intent to a concrete app. This approach has several benefits:

- You can develop and roll out new apps quickly, and adapt navigation targets later using configuration only. Example: You develop a new app to display employee data. For displaying employee data, we assume that you already have a Web Dynpro application available, but you are planning to replace it by an SAPUI5 application in the future.
  With the decoupling you can develop and roll out your new app now independently from the employee-display app. In your source app, you just code the navigation intent (rather than a concrete navigation target). In the configuration, you map this intent to your existing Web Dynpro application. When your new SAPUI5 app for displaying employee data is ready, you can simply map the same intent to the new SAPUI5 app in the configuration, without changing any line of code.
- You can start different apps depending on the device type.
  Example: If a user clicks on a link in your app in a desktop environment, you can start a desktop application to display employee data. If the same user taps on the same link in your app on a mobile device, you can start a lightweight mobile app that displays the same data in a different way. You can do this by simple configuration, without changing any line of code in your source app.

- You can start different apps depending on the role of the user.
  Example: A manager in your organization gets a different view on employee data than regular employees. You have created two views for this in your employee data app. Again, no modification is required in the code of the source app. For one intent, you can configure different navigation targets for different roles: With the same URL, the user's role determines which of the two different apps will be started.
- You can extend and customize SAP Fiori scenarios without modifying any SAP Fiori app code, just by configuration.

The following graphic shows an example for an intent coded into a source app. The intent expresses what you want to do next on an abstract level.
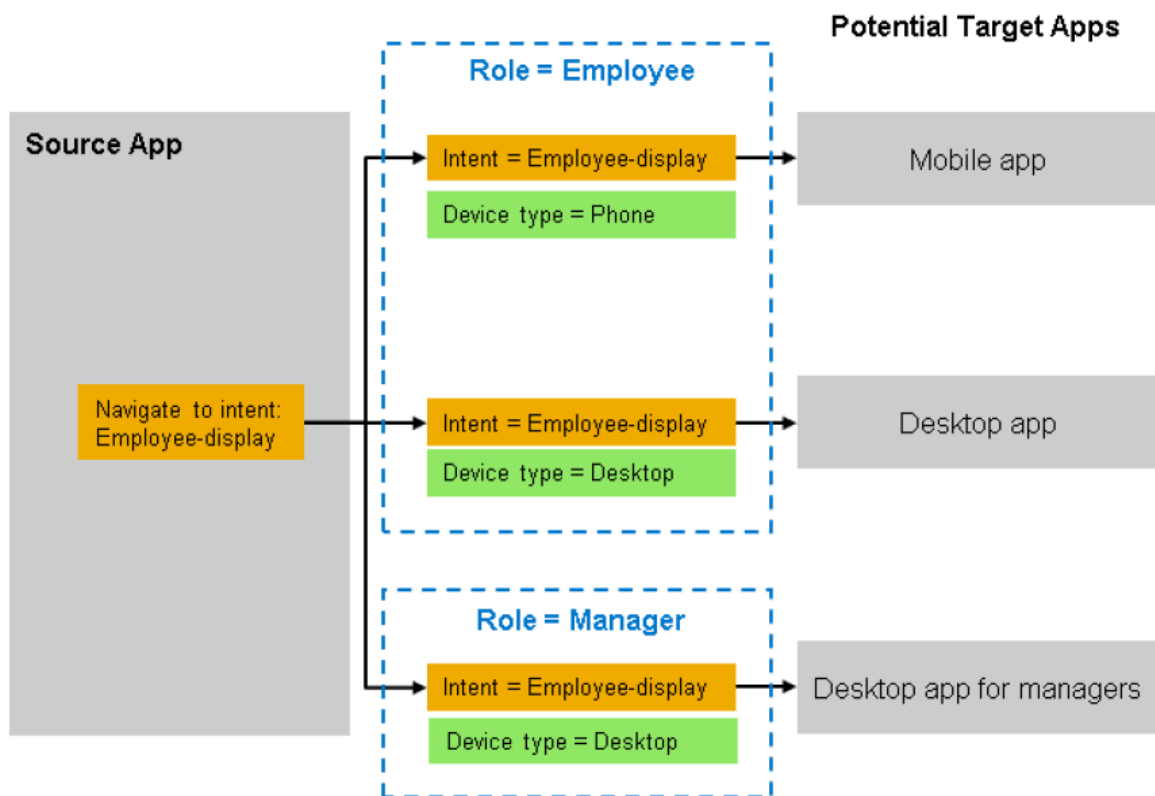
Depending on the device type where the navigation is triggered as well as the user's role, the same intent is resolved to different target apps. The target apps express how the intent can be achieved.

Related Information

Setting Up Navigation [page 127]

1.5.1.3.3 Navigation Operations: Inner-App and Cross-App Navigation

From a technical perspective, two types of navigation operations exist: cross-app navigation and inner-app navigation.

Cross-app navigation is navigation from one app to another. This is handled by the unified shell services. Inner-app navigation is navigation within a single app. This includes navigation between different views in an app, like a list view and a details view, but it also includes state changes like setting a filter in a drill-down app. This type of navigation has to be performed by the app itself.

For both cases, you use the intent-based approach: The hash consists of a general part that is controlled by the shell services and, optionally, an app-specific part that denotes the inner-app UI state. This has to be taken into account when running apps that are embedded in the SAP Fiori launchpad. These apps must not directly modify the part of the URL hash that is interpreted by the unified shell services. To support you in developing navigation, APIs are available both for cross-app navigation and inner-app navigation.
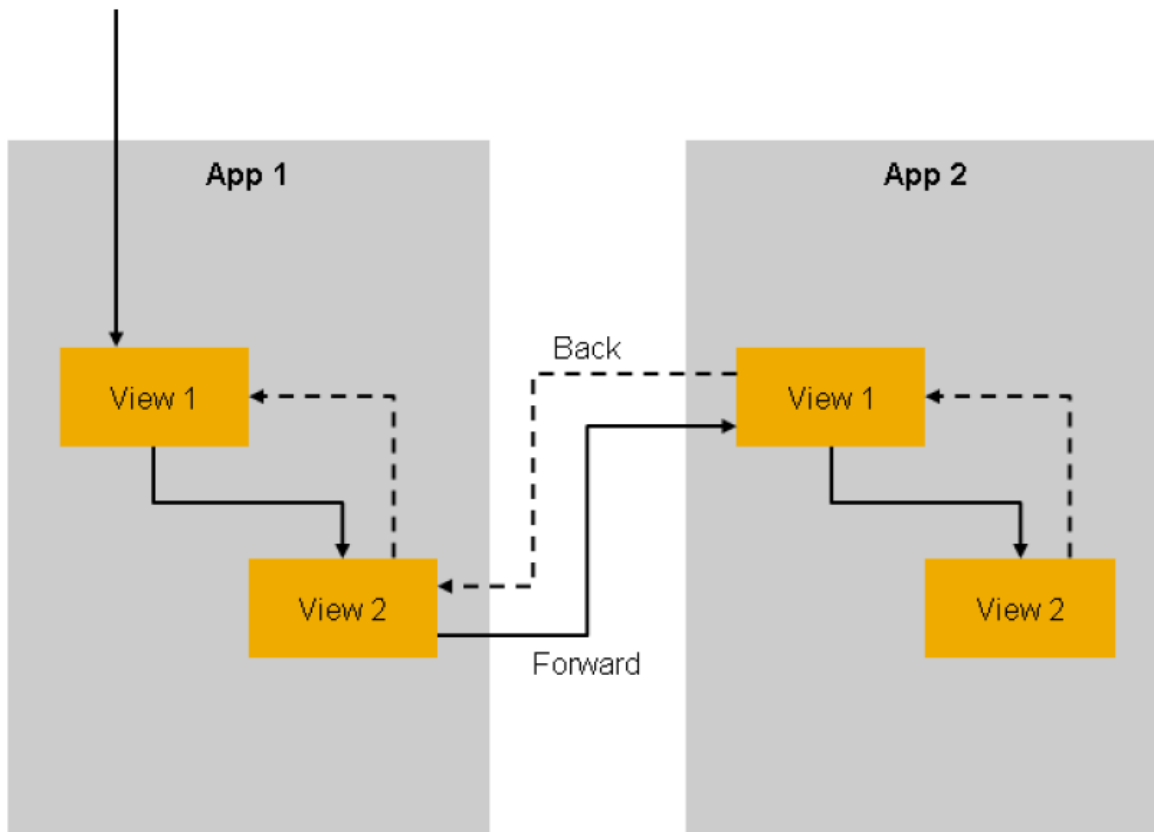
You can combine the two navigation operations and navigate to an app in a specific state. This happens, for example, during the back navigation from App2 to App1 in the example above. To enable back navigation to View 2 of App 1, the app needs to be able to restore the previous UI state from the app-specific information in the URL fragment, which usually contains parameters for loading the required data. In the above figure, all four states can be bookmarked. This is the main advantage of this approach. But it also implies the following best practices for app developers:

- Do **not** clutter the browser history by adding meaningless states to the URL fragment.
- Be prepared to reestablish the state of an app based on the app URL fragment at any time.

1.5.1.3.4 Overwriting the System Alias for the Target Application

To overwrite the system alias that you defined in the launchpad designer or in transaction `LPD_CUST`, you use the `sap-system` parameter for Web Dynpro applications, SAP GUI for HTML transactions, and SAPUI5 Fiori apps.

This topic is relevant only for application developers.

The `sap-system` parameter has to be passed as part of the intent, for example, `#SalesOrder-display?sap-system=<system alias>`.

The value of the `sap-system` parameter is the valid system alias defined in *RFC Destinations (Display/Maintain)* (transaction SM59).

> **i Note**
>
> You have to provide the value for the system alias first in transaction SM59. For more information, see the documentation for SAP Business Client under *Remote Systems*.

After overwriting the system alias, the navigation is directed to the target system with the given system alias. For example, if end users search for a purchase order in the SAP Fiori launchpad, you are directed to the target system such as an ERP or SRM system at runtime based on the search result.

The application changes the system for the target application dynamically at runtime based on the value of the `sap-system` parameter.

The target application can be a Web Dynpro application, an SAP GUI for HTML transaction, or an SAPUI5 application.

If you provide a system alias from the `sap-system` parameter that cannot be resolved, there is a fallback to the front-end server system.

Related Information

Configuring Remote Systems [page 308]

1.5.1.4   Developing Navigation

The URL expresses an intent regarding the navigation target. This is something users can easily relate to, and with the help of the browser's history cache, they can easily return to their frequently-used apps, just by typing a few characters in the browser's address bar.

The complete syntax for the URL fragment looks as follows:

```
#<SemanticObject>-<Action>?<parameter-1>=<value-1>&…&<parameter-n>=<valuen>[&/<
innerappspecificfragment>]
```

The URL fragment has a shell part and an app-specific part:

The shell part of the URL fragment is used for cross-app navigation. It is independent from the concrete implementation of an app.

The app-specific part of the URL is dependent on the concrete implementation of an app.

> **i Note**
>
> You might come across URLs with the following pattern:
>
> ```
> #<SemanticObject>-<Action>[~<InternalTargetID>]?<parameter-1>=<value-1>&…
> &<parameter-n>=< value-n>[&/<innerappspecificfragment>]
> ```
>
> The `<InternalTargetID>` (the part of the URL after the tilde (~) character) is a hash value that is set by the navigation target resolution service. While the combination of a semantic object and an action specify a navigation intent, the unique target ID identifies a specific target app. This part of the URL is usually not shown to end users and should not be touched by developers.

Cross-App Navigation

The following example shows a link for cross-app navigation:

```
<a href="#DaysSalesOutstanding-drilldown?kpiId=1234&amp;variantId=abcd"
target="_self">Details</a>
```

To make sure that the URL format is always correct, use one of the following services to construct such links:

- When developing SAPUI5 applications, use the `CrossApplicationNavigation` service.
  For more information, see Cross-Application Navigation for SAPUI5-based Apps [page 531].
- When developing Web Dynpro or SAP GUI applications, use one of the methods available in ABAP.
  For more information, see Cross-Application Navigation for ABAP-based Apps [page 537].

Inner-App Navigation

The SAPUI5 core library provides an API for navigating with URL fragment changes. When an app is running within the SAP Fiori launchpad, the implementation considers the shell-specific navigation parts and allows both inner-app and app-to-app navigation by similar means. The navigation API provides the following functionality:

- Directly trigger navigation to a parameterized navigation target
- Register navigation routes and listeners for URL fragment changes to restore a specific state

Related Information

1.5.1.4.1 Cross-Application Navigation for SAPUI5-based Apps

The following concepts support the cross-app navigation between SAPUI5-based apps.

- The `CrossApplicationNavigation` service enables you to construct shell fragments and links from local fragments.

> ### i Note
>
> - Do **not** construct the shell fragments yourself.
> - Do **not** use deep links to the app-specific fragment for passing information to the next app.

  For more information, see Developing Navigation to Another SAP Fiori App [page 531].

- For single-value parameters, you can use startup parameters to pass information to the next app.
  For more information, see Handling Startup Parameters [page 534].
- For navigation with complex application contexts, selection and presentation variants enable you to share well-defined data sets across users, apps, and technologies, without transferring business data.
  For more information, see Handling Complex Application Contexts [page 543].

1.5.1.4.1.1 Developing Navigation to Another SAP Fiori App

You can develop navigation from an SAP Fiori app to another SAP Fiori app using the `CrossApplicationNavigation` service.

In your application code, you specify a navigation intent and call the `CrossApplicationNavigation` service. Here are its most important methods:

Method	Description
`hrefForExternal`	With this method you can construct links to other apps.
`isNavigationSupported`	With this method you can check for a given intent whether navigation to a target app is actually possible for the current user on the current device.
`toExternal`	With this method, you can directly trigger navigation to another app within the SAP Fiori launchpad.

Navigation is performed using URL fragments. A navigation intent consists of a semantic object and an action. In addition, you can pass parameter values to the target application.

When using either of the methods above, you can either provide a semantic object, an action, and optional parameters, or you can provide a shell hash. Both patterns are shown in the examples below.

> **i Note**
>
> The service also offers the method `hrefForAppSpecificHash`, which creates a complete shell hash from an inner-app navigation pattern. For more information on inner-app navigation, see in the SAPUI5 documentation.

> **⚠ Caution**
>
> It is crucial to use this service for navigation. Do **not** use `window.location.hash`.

Obtaining the Interface

> **⇶ Sample Code**
>
> ```
> var oCrossAppNav =
> sap.ushell.Container.getService("CrossApplicationNavigation");
> ```

Generating a Link to Another App from a Semantic Object and Action

Generate a URL to put into a link tag:

> **⇶ Sample Code**
>
> ```
> var hrefForProductDisplay = oCrossAppNav.hrefForExternal({
>   target : { semanticObject : "Product", action : "display" },
>   params : { ProductID : [ "102343333" ] }
> });
> ```

Generating a Link to Another App from a URL

If you retrieve a shell hash from a URL, you can use `{target : {shellHash : "..."}}` to set the hash:

> **⇶ Sample Code**
>
> ```
> var hrefForProductAction = oCrossAppNav.hrefForExternal({
>   target : { shellHash : "Product-action?ProductID=102343333" }
> });
> ```

The following example generates a URL to navigate to the start page. In the SAP Fiori launchpad, this is the home page. In other scenarios this may be a different start page, for examle an initial application.

Checking whether the User Can Actually Navigate to a Target App on the Current Device

Navigation to a target app is only possible if a target app is mapped to the respective semantic object, action and parameters, for the current user and for the current device type. The `isNavigationSupported` method allows you to check whether this prerequisite is met. This is useful if you want to hide links created by the `hrefForExternal` method if there is no target application assigned for the current user on the current device type.

Note that you can test an array of links in one request.

> **≡ Sample Code**
>
> ```
> var oCrossAppNav.isNavigationSupported([
>   { target : { shellHash : "SalesOrder-approve?SOId=1234"} },
>   { target : { shellHash : "SalesOrder-delete?SOId=5678"} }
> ])
> .done(function(aResponse) {
>   aResponse.map( function(elem, index) {
>     if (elem.supported===true){
>       // enable link
>     }
>     else {
>       // disable link
>     }
>   })
> })
> .fail(function() {
>   // disable all links
> });
> ```

> **→ Tip**
>
> You can use the `getLinks` method of the `CrossApplicationNavigation` service to find out all applications that are assigned to the current user for a given semantic object and optionally a given parameter. For example, this can be useful if you want to generate a list of "See also" links in your application.

Triggering Navigation to Another App

The `toExternal` method directly triggers the navigation to another app.

Perform a direct navigation:

```
// trigger navigation
var oCrossAppNav.toExternal({
  target : { semanticObject : "Product", action : "display" },
  params : { ProductID : [ "102343333" ] }
});
```

URL to navigate to the start page:

```
var oCrossAppNav.toExternal({
  target : { shellHash : "#" }
});
```

i Note

Do **not** use `#Shell-home` or `Shell-home` to navigate to a specific home page.

The following example navigates to a fact sheet:

```
var oCrossAppNav.toExternal({
  target: { semanticObject : "SalesOrder", action: "displayFactSheet" },
  params : { SalesOrderID : "27" }
});
```

Related Information

sap.ushell.services.CrossApplicationNavigation

1.5.1.4.1.2 Handling Startup Parameters

Find out how startup parameters are handled that are encoded in the URL. This refers to target apps based on SAPUI5.

Tips

Consider the following tips when using startup parameters in the URL:

- Use only single-value parameters in the launchpad intent; multi-valued parameters are **not** recommended.

- The startup parameters are part of the URL and stored in the browser history. This may be a security or data protection issue. If you pass security-critical content, use an anonymized format and keep the sensitive data within your back end application.
- The length of a browser URL is limited and truncation may occur. Keep the length of the URL fragment below 512 characters.
- Information is transferred to the front-end server as part of navigation target resolution. This information may be persisted on the front-end server.

Passing Startup Parameters Dynamically with JavaScript

An app can dynamically pass parameters to another app using the `CrossApplicationNavigation` service. The following code snippet shows an example:

```
var href_For_Product_display = ( sap.ushell && sap.ushell.Container &&
sap.ushell.Container.getService("CrossApplicationNavigation").hrefForExternal({
   target : { semanticObject : "Product", action : "display" },
   params : { "ProductID" : "102343333", SupplierId : "90210" }
})) || "";
```

Receiving Startup Parameters Dynamically with JavaScript

Startup parameters are received by an embedded component by calling the `getComponentData` method, which returns a `startupParameters` member. The following code snippet shows an example:

```
sap.ui.core.UIComponent.extend("AppNavSample.Component", {
  ...
  createContent : function() {
    /*
     * getComponentData() returns an object like { startupParameters : { AAA :
["BBB"], DEF: ["HIJ","KLM"] } }
     * NOTE: parameters values are passed via arrays
     */
    var oComponentData = this.getComponentData();
    jQuery.sap.log("app was started with parameters " +
JSON.stringify(oComponentData.startupParameters || {} ));
  ...
  }
});
```

> i Note
>
> The parameter values are always passed into an array inside the object returned by `getComponentData()`.

Consuming Startup Parameters from Inside an Embedded View

For views created by the SAPUI5 routing framework or directly from within the component, you can locate and identify the component using `getOwnerIdFor`. The following code snippet shows an example:

```
// view controller.
   getMyComponent: function() {
     "use strict";
     var sComponentId = sap.ui.core.Component.getOwnerIdFor(this.getView());
     return sap.ui.component(sComponentId);
   }
   onCreate : function() {
     …
     var oStartupParameters =
 this.getMyComponent().getComponentData().startupParameters;
   }
...
```

For more information, see sap.ui.core.Component.

Other Sources of Startup Parameters

To set startup parameters for an app, the following options exist:

- Passing the startup parameters dynamically from the source app as described above.
- Setting the startup parameters in the configuration of a navigation target in transaction LPD_CUST, see Changing LPD_CUST Entries for Navigation Targets [page 228].
- Defining the startup parameters in configuration (target mapping).
- User enters them manually in the URL.

> **i Note**
>
> Startup parameters are transparent to the end user in the URL and may be tampered with or serialized with minimal efforts.

Passing Query Parameters in a Stand-alone Index Page

If you use a stand-alone index page for testing your app in a local environment, you can pass the query parameters to the component as shown in the following example:

```
<head>
…
<script>
  var oStartupParameters = jQuery.sap.getUriParameters().mParams;
  var oComponent = sap.ui.getCore().createComponent({
    name: "sap.samples.SampleComponent",
    settings: {
      componentData: { startupParameters: oStartupParameters }
    }
  });
  new sap.ui.core.ComponentContainer({
```

```
    component: oComponent
  }).placeAt("content");
</script>
</head>
<body>
  <div id="content"/>
</body>
```

> **i Note**
>
> Note that this code (`getUriParameters().mParams`) should not be used in productive environments.

1.5.1.4.2 Cross-Application Navigation for ABAP-based Apps

Learn how to enable cross-app navigation for ABAP-based apps, e.g. Web Dynpro or SAP GUI apps.

When developing ABAP-based applications (e.g. for Web Dynpro or SAP GUI), use one of the methods available in class CL_LSAPI_MANAGER to either generate a SAP Fiori launchpad URL or to trigger an intent-based navigation.

Prerequisites

You have established a connection from the back-end system to the front-end server. For more information, see Establish a Connection from Back-End System to Front-End Server [page 310].

Generate Launchpad URL for Delayed Execution

The class CL_LSAPI_MANAGER has the method CREATE_FLP_URL. Use this method to create a direct SAP Fiori launchpad startup URL. This is usefull to generate a URL to use in an e-mail or to embed in applications (reuse component with HTML viewer).

Sample Code for CREATE_FLP_URL:

> **⇆ Sample Code**
>
> ```
> DATA: lt_parameters TYPE tihttpnvp,
>       lv_url        TYPE cl_lsapi_manager=>url.
> FIELD-SYMBOLS: <param> LIKE LINE OF lt_parameters.
> APPEND INITIAL LINE TO lt_parameters ASSIGNING <param>.
> <param>-name = `par1`.
> <param>-value = `val %20`.
> " Home URL
> lv_url = cl_lsapi_manager=>create_flp_url( ).
> " URL => https://abc.sap.corp:44355/sap/bc/ui2/flp?sap-client=000&sap-
> language=EN#Shell-home
> " Intent URL
> ```

```
    lv_url = cl_lsapi_manager=>create_flp_url( object = `Action` action = `doIt`
    parameters = lt_parameters ).
    " URL => https://abc.sap.corp:44355/sap/bc/ui2/flp?sap-client=000&sap-
    language=EN#Action-doIt?par1=val%2520%252520
    " Standalone Application URL
    lv_url = cl_lsapi_manager=>create_flp_url(
        component = `sap.ushell.demo.AppNavSample`
        component_url = `/sap/bc/ui5_demokit/test-resources/sap/ushell/demoapps/
    AppNavSample`
        config = cl_lsapi_manager=>gc_flp-shell_config-standalone
        parameters = lt_parameters ).
    " URL => https://abc.sap.corp:44355/sap/bc/ui2/flp?sap-client=000&sap-ushell-
    config=standalone&sap-language=EN
    "        #Shell-runStandaloneApp?sap-ushell-
    SAPUI5.Component=sap.ushell.demo.AppNavSample
    "        &sap-ushell-url=%252fsap%252fbc%252fui5_demokit%252ftest-
    resources%252fsap%252fushell%252fdemoapps%252fAppNavSample
    "        &par1=val%2520%252520
```

Trigger Direct Navigation to Intent

Applications based on Web Dynpro ABAP or SAP GUI technologies have their own methods for navigation and resource addressing that do not map to intent-based navigation. Due to technology difference, implementing intent-based navigation is not always straightforward and needs different approaches, depending on the UI technology on which the application is based and the shell in which it is running in. The API tries to hide the mentioned complexity from the application, providing a stable interface for triggering intent-based navigation from ABAP that can be consumed in the same way by any UI technology and shell combination. Use method CL_LSAPI_MANAGER=>NAVIGATE_TO_INTENT directly or IF_LSAPI=>NAVIGATE with preconstructed intent-based navigation location URI.

Sample Code for NAVIGATE_TO_INTENT:

⇖ Sample Code

```
DATA: lt_parameters   TYPE /ui2/if_lsapi=>t_parameters,
      ls_parameter    LIKE LINE OF lt_parameters,
      lv_ibn_location  TYPE string.
ls_parameter-name  = 'id'.
ls_parameter-value = '7010'.
INSERT ls_parameter INTO TABLE lt_parameters.
" *********************************************************************
" create an IBN URL for delayed usage
lv_ibn_location = cl_lsapi_manager=>create_ibn_url(
  object = 'SalesOrder'
  action = 'display'
  parameters = lt_parameters ).
" manually trigger the navigation, providing pre-constructed IBN URL
cl_lsapi_manager=>get_instance( )->navigate( location = lv_ibn_location ).
" *********************************************************************
" create an IBN URL and trigger navigation in one step
cl_lsapi_manager=>navigate_to_intent(
  object = 'SalesOrder'
  action = 'display'
  parameters = lt_parameters
  navigation_mode = if_wd_portal_integration=>co_show_external ).
```

Trigger Navigation to a SAPGUI Transaction

With method NAVIGATE_TO_TRANSACTION you can launch SAPGUI transactions from various environments or technologies such as Web Dynpro ABAP or ITS. The current environment/shell in which the source application is running (Business Client, NWBC for HTML, Fiori, stand-alone) will be kept. Thus, if navigation is for example launched out of an application that is embedded in Fiori, the new transaction will be started in the context of Fiori as well.

Object Based Navigation (OBN)

You can launch an OBN in the SAP Business Client, NWBC for HTML or the Enterprise Portal with method NAVIGATE_TO_OBJECT. More information can be found here.

Related Information

Blog: Legacy Navigation to IBN

1.5.1.4.3   Tagging Navigation Targets with Parameters

You can tag navigation targets using the `sap-tag` parameter. The `CrossApplicationNavigation` service allows you to get navigation targets based on these tags.

Generic Use Case

You can define a group of intents by assigning them the same tag.

To tag an intent, simply add the `sap-tag` parameter to the intent. The default value for the parameter represents the name of your tag.

Parameters:		Name	Mandatory	Value	Is Regular Expression	Default Value	Target N...
	☐	sap-tag				foo	

> **i Note**
>
> For each navigation target only one tag can be assigned.

At runtime you can query the content for these intents using the `getLinks` method of the `CrossApplicationNavigation` service, and passing the tag as a parameter.

> **⇶ Sample Code**
>
> Querying content for tagged parameters
>
> ```
> oCrossAppNavigator.getLinks({semanticObject: "MySemanticObject", tags:
> ["foo"]}).done( function(oResult) {
>   // Do something with the result
> })
> ```

Primary Actions

For each semantic object you can define a navigation target to serve as the default navigation target. For example, you can define a specific default app for displaying an instance of the respective business object.

To define a primary action for a semantic object, add the `sap-tag` parameter with the default value `primaryAction` to the intent.

Parameters:	☑	Name	Mandatory	Value	Is Regular Expression	Default Value	Target N...
	☑	sap-tag				primaryAction	

Make sure to define no more than one primary action per semantic object.

The following example shows how to use the API:

> **⇶ Sample Code**
>
> ```
> CrossApplicationNavigation.getPrimaryIntent("MySemanticObject",
> {}).done( function (oResult) {
>   // Do something with the result
> });
> ```

Related Information

sap.ushell.services.CrossApplicationNavigation

1.5.1.4.4 Navigation: Passing Large Amounts of Data to Another App

When developing navigation, there are several ways to pass startup parameters to the target application.

The easiest way to pass startup parameters to the target application is to add them to the URL hash. The maximum length of URLs is limited by web browsers. If you have a large number of parameters and / or parameters with long values, this condition may lead to a truncation of the URL. Therefore, it is a best practice to store a set of parameters with a key, and pass only the key in the URL.

> **i Note**
>
> Do not put any security-critical information in the URL.

The following table gives an overview of the different possibilities:

Method	Recommendation	Roundtrip Required?	URL Works	Longevity	Cleanup	Security	Comment
URL	For non-security critical data	No	Always and for everybody	Eternal	Yes	Persisted on the client (browser history)	
Shortened URL	For non-security critical data	Yes	For everybody	As long as key data is retained	Via aging, cleanup via batch report	Persisted on the server. Anybody guessing or obtaining the key can access the data.	Only for SAP Fiori apps running in the same launchpad
Browser local storage, pass hashtag	OK, if URL stability is not required	No	For me - in the same browser session	Session only	No cleanup, the space may be exhausted by other apps	Persisted on the client	Data in storage session is limited; some browsers do **not** support this in private mode
Read/write to dedicated app-specific back-end service; pass hashtag in URL	Optimal for large amounts of data	Yes	Yes - for a given time and for everybody; can be configured, to be defined by the app	For longer time, controlled by the app	Via aging, cleanup via batch, and so on		Extra roundtrips; all parameters can be configured by app (lifetime)

Method	Recommendation	Roundtrip Required?	URL Works	Longevity	Cleanup	Security	Comment
							per user / per role; data amount unlimited

The following variants are **not** recommended:

- Storing data in a window local JavaScript variable: This may no longer work if, in future, every application in the SAP Fiori launchpad is launched in a new window.

Automatic URL Shortening for SAP Fiori Apps

For navigation between SAP Fiori apps running in the same launchpad, automatic transparent URL shortening allows applications to overcome the limitation regarding URL length.

> i Note
>
> Currently, this mechanism only works for SAPUI5-based Fiori apps that run within the same launchpad on the same front-end server.

Automatic URL shortening allows apps to pass large-size parameter bags to other apps as startup parameters. When the URL hash exceeds size limits, the parameters are automatically replaced by a key and persisted on the front-end server. This mechanism is transparent to the application. This allows application developers to pass any number of parameters, with any name/value length, and not have to be concerned about exceptional cases when the URL becomes too long. The SAP Fiori launchpad decides which parameters are compressed and which ones remain readable/changeable in the URL. This depends on the length of the key of the compressed data in the URL as well as on internal considerations.

When URL shortening occurs, parts of the URL are stored on the server, in a storage that is accessible across users. If you would like to avoid this, make sure to work with hashes well below 500 characters.

A report is available for cleaning up data that was written to the server for URL shortening. For more information, see.

> ⚠ Caution
>
> Do not put any security-critical information into the URL. If your input parameters contain any security-critical information, pass a neutral key instead, and store the data in an application-specific service in the back-end system as described below.

Considerations for an Application-Specific Back-End Service

We recommend that the length of the URL hash on the client side has is less than 500 characters. If the amount of data exceeds this limit and cannot be reduced, the information has to be stored in a different location, and a corresponding key should be used for the URL.

To generate the key, you can develop an application service. You can then decide how you protect the data, especially if it can be accessed across users.

The following aspects have to be considered:

- If the amount of data does **not** fit the URL, is it possible to reduce the size of the information, or can it be safely stored locally?
- If the same user opens the link of the called application again in the same session, for example using the browser's back button or in a different browser window or session, what will happen? Do you want the selection still to be available? Where and how long do you want the information to be persisted? How unique is the storage key?
  For this, the following options exist:
  - You can generate a unique hashtag for a new set every time.
  - You can use a fixed key or a limited set of fixed keys.
- How is persisted data cleaned up?

1.5.1.4.5   Handling Complex Application Contexts

The SAP Fiori launchpad enables navigation between two SAP Fiori apps that are assigned to the same end user via their role: By means of so-called selection variants and presentation variants, a standardized handling is introduced that enables an efficient context-sensitive navigation between any two SAP Fiori apps.

Selection and presentation variants enable the sharing of well-defined data sets across users, apps, and technologies, without transferring business data.

- A selection variant represents all settings which determine the data record selection and the total of all values in an upcoming result request. From a technical point of view, a selection variant is a set of parameter values and filters that influences the request path and the `$filter` query option in OData `GET` requests on `EntitySets`. It can also represent an ABAP `select` option. Some objects in SAP Fiori already provide selection variants, such as the `SmartFilterBar` or the Smart Business KPI evaluation.
  For more information, see Selection Variants [page 544].
- A presentation variant represents settings which determine the representation of a retrieved result set on the user interface, including, for example, layout, sequence of fields, sort order, grouping, totals, and visualizations. The presentation variant influences query options such as `$select` and `$orderby` in OData `GET` requests on `EntitySets`. Presentation variants are specified in semantic terms, so that they are independent of the UI technology, the channels, or protocols.
  For more information, see Presentation Variants [page 546].

Selection and presentation variants are provided as an extension in the data model and can be used in, for example, KPIs and reports by any consumer. They can be preconfigured as part of the application configuration for, for example, embedded charts and tables.

1.5.1.4.5.1 Selection Variants

Selection variants define the contract for passing complex selections between a navigation source and a navigation target in SAP Fiori applications.

For analytical applications in particular, the supported selection context is often not exposed as a navigation target parameter for one of the following reasons:

- Analytical applications often accept static or dynamic filters on almost any attribute of the semantic object. In OData terms, the `$filter` query option is available for most properties of the underlying entity type.
- At design time of the navigation target, the restrictions are not known statically. The concrete list of fields and their values may depend on the runtime application context in the UI of the navigation source.
- Unlike "regular parameters", these restrictions are not bound to single values or patterns, but may incorporate expressions up to the complexity of any OData `$filter` expression for use cases where the navigation source uses OData to access its data. The complex expressions may contain up to 1000 single values, intervals, exclude/not-equal statements, and logical expressions. An important subset of a complex expression in the context of the SAPUI5 ALV and Design Studio as well as ABAP-based apps in general is the functional scope of the ABAP select option, which are typically a limiting factor on the UI as well as on provider side.

For these reasons, the navigation target exposes the capability to handle a complex selection context in terms of an OData filter expression or an ABAP select option instead of exposing each property of the underlying data model as a parameter. This selection context is called selection variant. Some components already contain such selections and may have SAPUI5 variants, such as the `SmartFilterBar` or the Smart Business KPI evaluation.

> i Note
>
> A selection variant needs to contain parameters and filters. It does not necessarily boil down to a filter variant because the underlying data model, such as CDS views, CalcViews, and BEx queries, may expose parameters which are not part of the result set and cannot be restricted at runtime via a filter query option.

A selection variant is a specific subset of an application context.

Navigation Parameters Representing Field Names

The parameters and filters of a selection variant refer to the underlying data model. When you use selection variants for navigation, it is no longer necessary to explicitly pass navigation parameters representing field names. The parameters that already have been introduced by the existing application do not need to be removed. The following rules apply:

- If you use selection variants, all restrictions must be part of the selection variant. This also applies to the existing explicit parameters.
- In addition, single values may be passed via explicit parameters. The value of the parameter must then be identical to the single value restriction in the selection variant.

Field Name Mapping Between Source and Target

To allow a peer-to-peer contract between the navigation source and the navigation target, the launchpad allows explicit parameters to map the source and target field names. The result is that most field names are not exposed as parameters per default, that nothing can be mapped in the navigation configuration, and that a global set of semantic field names is available that is known to the source and the target. This is stipulated by the concept of semantic objects in the SAP Fiori launchpad as well as the SAP Business Suite's concept of global field names, which both indicate the app-independent semantics in their names. So, the navigation source exposes semantic field names and the navigation target maps these names to local names of the target app, if required.

Client-side JSON Model

The following code snippet shows an example of the client-side JSON model.

> **i Note**
>
> The type definition follows the corresponding term definition for selection variant in the OData 4.0 Vocabularies - SAP UI, see SelectionVariantType.

The app or navigation source `AccountBalances-list` uses the OData service `/(…)/AccountBalance` with a query that supports an ad-hoc currency translation. This translation requires the `DisplayCurrency` and `ExchangeRateType` parameters which are represented by the entity types `AccountBalanceQueryParameters` and `AccountBalanceQueryResult`.

The app is configured for the use of the `ExchangeRateType` 'M' and the end user has selected 'EUR' as `DisplayCurrency`. In the filter bar, the user has selected the following filters for the displayed result:

- Company code = '0001', '0002' (multiple single values);
- Fiscal year = '2014'
- GL account = ['10000', '20000'], '3000' (interval plus single value)

This results in the following JSON representation of the selection variant:

> **⇒ Sample Code**
>
> ```
> SelectionVariant =
> {
>   "SelectionVariantID" : "20141023134315",
>   "ParameterContextUrl" : "/(..)/AccountBalance/
> $metadata#AccountBalanceQueryParameters",
>   "FilterContextUrl" : "/(..)/AccountBalance/
> $metadata#AccountBalanceQueryResult",
>   "Text" : "Temporary Selection Variant, Account Balance, 24.10.2014
> 13:43:15",
>   "Parameters" : [
>                 {
>                         "PropertyName" : "DisplayCurrency",
>                         "PropertyValue" : "EUR"
>                 },
>                 {
>                         "PropertyName" : "ExchangeRateType",
>                                 "PropertyValue" : "M"
> ```

```
                }
        ],
        "ODataFilterExpression" : "",
        "SelectOptions" : [
                {
                        "PropertyName" : "CompanyCode",
                        "Ranges" : [
                                {
                                        "Sign" : "I",
                                        "Option" : "EQ",
                                        "Low" : "0001",
                                        "High" : null
                                },
                                {
                                        "Sign" : "I",
                                        "Option" : "EQ",
                                        "Low" : "0002",
                                        "High" : null
                                }
                        ]
                },
                {
                        "PropertyName" : "FiscalYear",
                        "Ranges" : [
                                {
                                        "Sign" : "I",
                                        "Option" : "EQ",
                                        "Low" : "2014",
                                        "High" : null
                                }
                        ]
                },
                {
                        "PropertyName" : "GLAccount",
                        "Ranges" : [
                                {
                                        "Sign" : "I",
                                        "Option" : "BT",
                                        "Low" : "10000",
                                        "High" : "20000"
                                },
                                {
                                        "Sign" : "I",
                                        "Option" : "EQ",
                                        "Low" : "30000",
                                        "High" : null
                                }
                        ]
                }
        ]
}
```

1.5.1.4.5.2 Presentation Variants

Define the contract for passing layout- and display-relevant information between a navigation source and a navigation target in SAP Fiori apps.

For most use cases, the supported presentation context is not exposed as navigation target parameters for one of the following reasons:

- The variety of settings which a caller may want to influence is too large and too diverse to be statically defined at design time of the app.

- Many presentation related settings require a complex handover format, which does not fit (single value) parameters.
- Template-based applications are driven by annotations for which no URL syntax exists. For handing over an annotation, the annotation document URI would have to be used as parameter.
- The relevant presentations settings may depend on the runtime application context in the navigation source UI.

For this reason, instead of exposing each presentation-relevant setting as a (URL) parameter, the navigation target rather exposes the capability to handle a complex presentation context in terms of UI annotations. The so-called presentation variant, as specified on the level of CDS for domain-specific annotations and OData for UI annotations, has been introduced to represent the interoperable UI state with regard to layout settings and display settings.

The concept of a presentation variant is a specific subset of a general concept of an application context. In order to cover all future applications of the presentation variant, the structure of the presentation variant must be described in a public format. This has been done as part of the SAP UI Vocabulary (complying to OData V4). From the vocabulary term definition, a representation in JSON is derived, which covers all settings that are relevant in the context of UI navigation.

Client-side JSON Model

The most relevant parts of the presentation variant UI annotation are redefined in JSON, meaning that the presentation variant for UI navigation is defined as a true subset of the UI annotation. The following code snippet shows a concrete example for the specified representation.

> i Note
>
> The type definition follows the corresponding term definition for presentation variant in the OData 4.0 Vocabularies - SAP UI, see PresentationVariantType .

The app or navigation source `AccountBalance-list` uses the OData service `/(..)/AccountBalance`. The OData service has an aggregating entity set with, among others, the following dimensions: Ledger, CompanyCode, GLAccount, FiscalYear, and FiscalPeriod. The ledger and the fiscal year are fixed to a single value and not displayed in the result. The GL account is restricted to a meaningful range.

This results in two presentations of the data on the screen: A chart with the fiscal period as category and an analytical table with the fiscal period as group and the GL account as further detail.

This example is represented in the following JSON representation of the presentation variant:

> ≒ Sample Code
>
> ```
> PresentationVariant =
>     {
>         "PresentationVariantID" : "20160803134315",
>         "ContextUrl" : "/(..)/AccountBalance/
> $metadata#AccountBalanceQueryResult",
>         "Text":"GL Account Balances by Fiscal Period ",
>         "MaxItems":10,
>         "SortOrder":
>         [
>             {
>                 "Property":"FiscalPeriod",  // PropertyPath
> ```

```
                "Descending":false    // i.e. ascending
            }
        ],
        ....
        "GroupBy":["FiscalPeriod","…"], // Collection of Property Paths (from
context)
        "TotalBy":["CompanyCode"], // Collection of Property Paths (from
context)
        "Total":["Amount","…"], // Collection of Property Paths (from context)
        "IncludeGrandTotal":true,  // json boolean value
        "InitialExpansionLevel":1,  // json integer number
        "RequestAtLeast":["Ledger"] // Collection of Property Paths (from
context),
        "Visualizations":[{   // at least one entry (current visualization in
nav source)
        "Type": "LineItem",  // optional: concat #<qualifier>
        "Content":
            [{
                "Value": "GLAccount", // Property Path
                "Label": "G/L Account",  // literal (source annotation paths
have to be resolved)
                "IconUrl": null, // string, literal
                "Criticality": null, // property path; contains members of
UI.CriticalityType
                "CriticalityRepresentation": "WithoutIcon"  // member of
UI.CriticalityRepresentationType
            },
        {
..
        }
        ]
    },
        {
            Type: "Chart", // optional: concat #<qualifier>
            Content:
                {
                    "Title":"",  // chart title , literal
                    "Description": "", // literal
                    "ChartType": "Line", // member of EnumType @UI.ChartType
                    "Measures": ["Amount"], // Collection of Property Paths
(from context)
                    "MeasureAttributes":[{     // optional
                    "Measure": "Amount", // Propery Path
                    "Role": "Axis1",  // member of UI.ChartMeasureRoleType
                },
                {
                    …
                }
        ],
            "Dimensions": ["FiscalPeriod","GLAccount"], // Collection of
Property Paths
            "DimensionAttributes":[{ // optional
            "Dimension": "FiscalPeriod", // Propery Path
            "Role": "Category",  // member of UI.ChartDimensionRoleType
            },
        {
            …
        }]
        }
    },
        {
            …
        }
    ]
}
```

Example: Assume the navigation target is a document line item list report without a chart display, which provides a further detail to an account balance. This means that all properties which are used in the contextual presentation variant are actually available 1:1 in the navigation target. In this case, the the line item list is expected to take over the groups and totals from the context and include all table columns from the context, plus some specific columns for the line item detail.

1.5.1.5   Local Sandbox Environment for the SAP Fiori Launchpad

The sandbox environment is a simplified SAP Fiori launchpad environment that allows you to test your applications and make sure that they can be embedded properly into the SAP Fiori launchpad.

With the local sandbox environment you can:

- Ensure that your SAPUI5 Fiori app can run embedded in the SAP Fiori launchpad; that is, it must not define any script, stylesheet, or other dependencies in its HTML file, but only in its main SAPUI5 component. This main SAPUI5 component is defined in the sandbox configuration.
- Use SAP Fiori launchpad services either implicitly (the environment does the job for you without any coding) or through API calls without front-end server configuration.
- Test the unified shell services to ensure they work properly for your SAP Fiori apps as far as possible in a purely local environment (apart from application-specific OData services called).

The sandbox uses a local configuration file instead of services on the ABAP front-end server or any other platform running the SAP Fiori launchpad. This configuration file contains the configuration of SAP Fiori applications and their navigation targets. For information on the syntax of the local configuration file, see Local Configuration File for the Launchpad Sandbox [page 550].

The local sandbox environment for the SAP Fiori launchpad is provided as part of the SAPUI5 demo kit. You can use the sandbox e.g. in the following development environments:

- SAP Web IDE
  For more information, in the SAP Web IDE documentation, see Running Applications in the SAP Fiori Launchpad .
- SAP Business Application Studio
  For more information, see App Development Using SAP Business Application Studio.

Related Information

Intent-Based Navigation [page 108]

1.5.1.5.1 Local Configuration File for the Launchpad Sandbox

The local configuration is stored in JSON format. You can configure a list of applications. The applications are launched with an intent that is composed of the syntax `<SemanticObject>-<Action>`. For each application, specify the following properties:

Property	Description
`additionalInformation`	The `additionalInformation` property in this object refers to the name of the root component of your SAPUI5 Fiori application (without ".Component").
`applicationType`	The application type for SAPUI5 applications is always **URL**.
`url`	The **URL** property is set to the root path of your application (the folder where you would have your `Component.js` files).
`title`	The title of your application.
`description`	A description for the application.

> ⚛ Example
>
> ```
> {
>     "applications" : {
>         "MyFioriObject-display" : {
>             "additionalInformation" :
> "SAPUI5.Component=com.mycompany.MyFioriApplication",
>             "applicationType" : "URL",
>             "url" : "/MyFioriApplication",
>             "title" : "My Fiori Application",
>             "description" : "My first Fiori application"
>         },
>         "MySecondFioriObject-display" : {
>             "additionalInformation" :
> "SAPUI5.Component=com.mycompany.MySecondFioriApplication",
>             "applicationType" : "URL",
>             "url" : "/MySecondFioriApplication",
>             "title" : "My Second Fiori Application",
>             "description" : "My second Fiori application"
>         }
>     }
> }
> ```

> ⚠ Caution
>
> The intents "MyFioriObject-display" and "MySecondFioriObject-display" must be constructed from two identifiers with a dash, not "MyFioriObject display" or "MyFioriObjectdisplay". For more information, see Intent-Based Navigation [page 108].

1.5.1.6 Typical Use Cases

Collection of typical use cases for developing apps for the SAP Fiori launchpad.

When you develop apps for the launchpad, make sure that you define dependencies and extensions using the SAPUI5 descriptor for applications, components, and libraries. Performance optimizations in the launchpad rely on dependencies being maintained correctly in this location. For more information, see .

Related Information

1.5.1.6.1 Saving an Application State to a Tile

The `Bookmark` service allows you to add bookmark functionality to an app. Users then can create, update, and delete tiles on the home page or a page.

The use case is as follows: The user starts an app and does some filtering, grouping, or a search. The user then wants to create a tile on the home page or page that opens the app in the same application state. In the following sections you learn how you can define that this bookmark option shall be available in the share option of an app.

You have the following options to define a bookmark option for an app:

`AddBookmarkButton`	A simple way to add a *Save as Tile* button (or link) to your application using the current location as tile URL.	Adding Bookmarking Functionality to an App with the Bookmark Button [page 552]
`Bookmark` service	Calling the `Bookmark` service gives you more flexibility when creating tiles and also allows you to update and delete tiles.	Adding Bookmark Service for Creating, Updating, and Deleting Bookmark Tiles [page 553]

1.5.1.6.1.1 Adding Bookmarking Functionality to an App with the Bookmark Button

In your application, you can provide a button or a link that allows users to add a tile referencing the current application state to a page or the home page of the SAP Fiori launchpad.

Prerequisites

Your application updates the URL hash according to its current state. This ensures that the tile allows the user to jump directly to the current state or view of the application.

The application runs in a unified shell environment.

Context

The `AddBookmarkButton` is part of the `sap.ushell` library.

The procedure uses the following services and controls:

- Inner app navigation, see
- `sap.ushell.ui.footerbar.AddBookmarkButton`

Procedure

1. Include the `AddBookmarkButton` button in a view of your application.

   Use the following Code snippet in the `view.xml` file:

   ```
   <core:View
   xmlns:footerbar="sap.ushell.ui.footerbar"
   ...
   >
   ...
   <footerbar:AddBookmarkButton id="addToHome" />
   ...
   </core:View>
   ```

2. During view initialization, predefine the required parameters by setting `appData` of the `AddBookmarkButton`.

   The code snippet shows predefined tile parameters:

   ```
   onInit: function () {
       var oView = this.getView();
       var oAddToHome = oView.byId("addToHome");
       oAddToHome.setTitle("My Bookmark"); // default: ""
       oAddToHome.setSubtitle("(via button)"); // default: ""
       oAddToHome.setInfo("any info"); // default: ""
   ```

```
    oAddToHome.setIcon("sap-icon://home"); // default: undefined
    oAddToHome.setNumberUnit("EUR"); // default: undefined
    oAddToHome.setServiceUrl("/any/service/$count"); // default: undefined,
string or a JS function
    oAddToHome.setServiceRefreshInterval(100); // default: undefined
    oAddToHome.setCustomUrl("http://www.sap.com"); // default: undefined,
string or a JS function
    },
```

3. If a `serviceUrl` is set either as a string or as a dynamic JavaScript function, an `App Launcher - Dynamic` tile is created on the user's home page. Otherwise, an `App Launcher - Static` tile is created. If the application wants to change values before the *Save as Tile* dialog is opened, the application can register a `BeforePressHandler`.

   Update of `serviceURL` before opening the *Save as Tile* dialog:

```
oView.byId("addToHome").setBeforePressHandler(function () {
    oAddToHome.setServiceUrl("/my/updated/service/$count");
});
```

4. If a `customUrl` is set either as a string or a dynamic JS function, it will be used as the navigation target for the tile that is about to be created.

Next Steps

If the configuration options for the button are not sufficient, for example if you create a tile referencing a different app, use the `Bookmark` service API directly, as described in the next section.

Related Information

1.5.1.6.1.2 Adding Bookmark Service for Creating, Updating, and Deleting Bookmark Tiles

The `Bookmark` service allows users to create, update, and delete tiles on the launchpad page or home page.

Creating a Tile

```
var oBookmarkService = sap.ushell.Container.getService("Bookmark");
oBookmarkService.addBookmark({
  title: "My Bookmark",
  url: "#UI2Fiori2SampleApps-appnavsample",
  icon: "sap-icon://home",
  info: "Any Info",
```

```
   subtitle: "Any Subtitle",
   serviceUrl: "/sap/opu/odata/myservice/Object?$count",
   serviceRefreshInterval: 600,
   numberUnit: "Objects"
}).done(function () {
   sap.m.MessageToast.show("Bookmark added");
}).fail(function (sMessage) {
   sap.m.MessageToast.show("Failed to add bookmark: " + sMessage);
});
```

If a `serviceUrl` is set, an *App Launcher - Dynamic* tile is created on the user's home page. Otherwise, an *App Launcher - Static* tile is created. For more information on tile types, see Configuring Tiles [page 207].

Note that the texts in the example above (title, subtitle, and number unit) are not translated.

The app is added to the default group on the home page or to the default page when you work with pages and spaces. When you want to control this behavior, check out the parameter `vContainer` (see sap.ushell.services.Bookmark in the SAPUI5 Demo Kit for detailed information).

The URL fragment in the code above is just an example. In your code, you construct the URL from the launchpad. When constructing the URL, make sure to use the external format, for example, a URL generated by the cross-application navigation service. For more information, see Developing Navigation to Another SAP Fiori App [page 531].

Checking Whether a Tile Already Exists

The following example shows how to count the number of existing tiles for a given URL on the user's launchpad. This is useful for finding out whether any tile for a given URL already exists on the current user's home.

```
var oBookmarkService = sap.ushell.Container.getService("Bookmark");
oBookmarkService.countBookmarks("#UI2Fiori2SampleApps-
appnavsample").done(function (iCount) {
  sap.m.MessageToast.show("Number of bookmarks: " + iCount);
}).fail(function (sMessage) {
  sap.m.MessageToast.show("Failed to count bookmarks: " + sMessage);
});
```

Deleting Tiles

The following example shows how to delete all tiles with a given target URL on the user's launchpad:

```
var oBookmarkService = sap.ushell.Container.getService("Bookmark");
oBookmarkService.deleteBookmarks("#UI2Fiori2SampleApps-
appnavsample").done(function (iCount) {
  sap.m.MessageToast.show(iCount + " bookmarks deleted");
}).fail(function (sMessage) {
  sap.m.MessageToast.show("Failed to delete bookmarks: " + sMessage);
});
```

Updating Tiles

The following example shows how to update all existing tiles with a given URL on the user's launchpad:

```
var oBookmarkService = sap.ushell.Container.getService("Bookmark");

oBookmarkService.updateBookmarks("#UI2Fiori2SampleApps-appnavsample", {
  title: "My Bookmark",
  url: "#UI2Fiori2SampleApps-appnavsample",
  icon: "sap-icon://home",
  info: "Any Info",
  subtitle: "Any Subtitle",
  serviceUrl: "/sap/opu/odata/myservice/Object?$count",
  serviceRefreshInterval: 600,
  numberUnit: "Objects"
}).done(function (iCount) {
  sap.m.MessageToast.show(iCount + " bookmarks updated");
}).fail(function (sMessage) {
  sap.m.MessageToast.show("Failed to add bookmark: " + sMessage);
});
```

Related Information

sap.ushell.services.Bookmark

1.5.1.6.2 Referencing Resources Inside an SAPUI5 Component

Your app is embedded in the unified shell as an SAPUI5 component and you want to reference resources that are inside the same SAPUI5 component, such as JavaScript modules or stylesheets.

Referencing JavaScript Modules

For referencing JavaScript modules of your own component, always use the fully qualified module name. The unified shell registers a module path for the root of the component. Example on an AS ABAP front-end server: Component name is `mycompany.samples.mysample` and it is deployed as BSP application `mycompany/my_sample_bsp`. In this case, the module path `mycompany.samples.mysample` is mapped to `/sap/bc/ui5_ui5/mycompany/my_sample_bsp`.

The unified shell then locates your elements as follows:

- It retrieves the `mycompany.samples.mysample.Component` component from `/sap/bc/ui5_ui5/mycompany/my_sample_bsp/Component.js`.
- It retrieves the `sap.samples.mysample.view.S2` view from `/sap/bc/ui5_ui5/my_sample_bsp/view/S2.view.xml`.

You can then reference all modules contained in the component with the fully qualified module name using the `require` mechanism. For more information, see in the documentation of SAPUI5. For example, you can load a JavaScript file located in `/sap/bc/ui5_ui5/my_sample_bsp/morejs/MyJSFile.js` by using `jQuery.sap.require("mycompany.samples.mysample.morejs.MyJsFile");`.

Including Custom Stylesheets

To include custom stylesheets, use the `includes` property of the component metadata as shown in the example in in the documentation of SAPUI5.

Referencing Other Files

For referencing other files, you may need to build a URI for a resource, for example, when creating a resource model. You can do this by calculating the absolute path based on the relative module path of your own component. The following code snippet shows an example:

```
var effectiveUrl = jQuery.sap.getModulePath("mycompany.samples.mysample") + "/" +
"i18n/i18n.properties"
var resourceBundle = jQuery.sap.resources({
   url : effectiveUrl
});
```

1.5.1.6.3   Persisting Personalization Data

The personalization service allows you to store user-specific settings for your applications, like the column sequence of a table, or query parameters for data selection.

The functionality is implemented as a shell service and has several platform-specific adapters:

- In the local development scenario, you can use the sandbox shell, which stores personalization data in the browser's local storage.
- In a productive environment, like the ABAP shell or the HANA Cloud Platform, personalization data is stored on a front-end server.

> **i Note**
>
> "Personalization" in this context does **not** mean handling tiles on the user's home page. For this purpose the bookmark service is available. For more information, see Adding Bookmark Service for Creating, Updating, and Deleting Bookmark Tiles [page 553].

> **i Note**
>
> Do not use the personalization service to persist business data. As business data is typically related to a specific application, we recommend you persist it on the respective back-end system where the application data resides, using a specific OData service.

Modes

The personalization service can be used in either of the following modes:

- Direct mode
- Container mode

The following table lists criteria for when to use which mode:

Criteria	Direct Mode	Container Mode
Personalization for ...	One entity, for example settings for one table	Several entities, for example settings for a table and query parameters for the content of this table
Complexity	Easy, one step	Separate load and save operations
Variants Available?	No	Yes

Direct Mode

You can use direct mode if you want to store one entity only. Personalization data should be one simple or structured value.
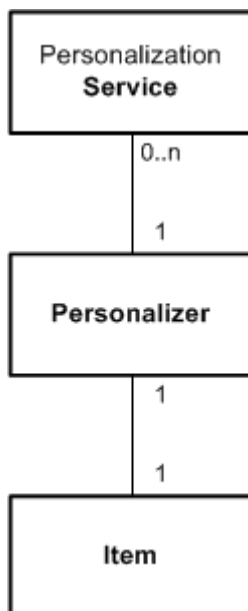
In this mode a **personalizer** object is the only object used. The personalization service has a factory method for it. Each personalizer is dealing with exactly one item. The value of an item is stored as a JSON object.

Container Mode

Use this mode if you want to store personalization data for more than one entity, if you want to store more than one value for an entity (several variants), or if you want to control the saving and loading of the data.

A container can contain several items. We recommend you assign items directly to the container if you do not need any variants.
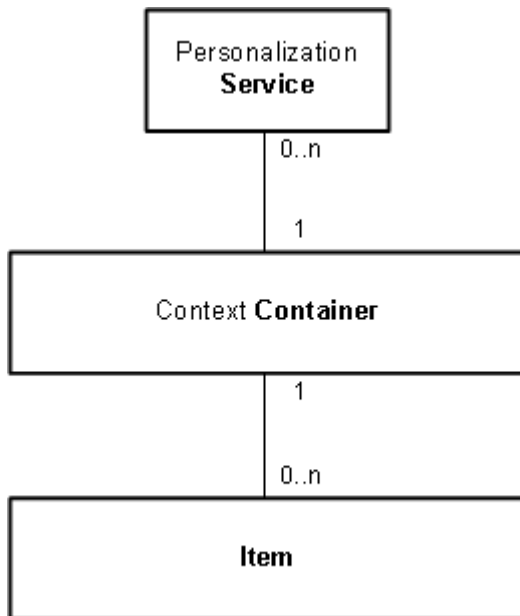
A container can be mapped to a variant set adapter, which contains one or more variant sets. A variant set contains variants, which contain the actual personalization data as item values. In small applications you typically need only one variant set.
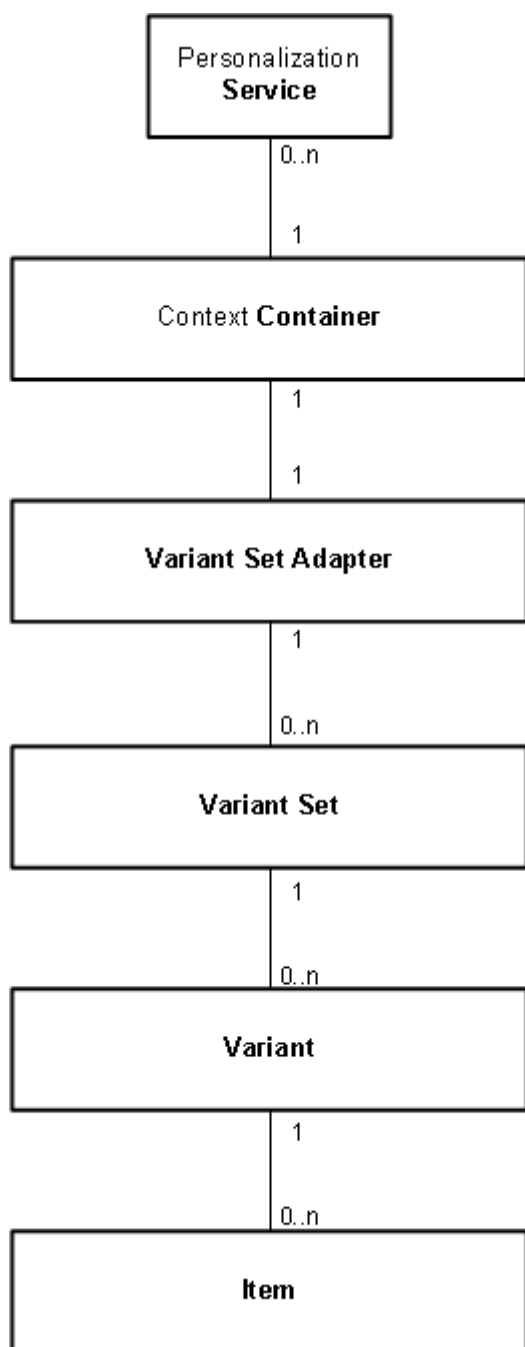
The container object is created by a factory method of the personalization service.

The container is the entity that is loaded and saved. You would usually have one container for one application - which is a hint for naming the container. The length of the container key must not exceed 40 characters. If you would like to use personalization data in several applications you can use the same container in those applications. The amount of data stored in one container influences the startup performance of an application, as personalization data is typically loaded during startup.

Table Personalization

For personalization of tables, specific personalization functionality is available to make it easier for application developers..

Table personalization always needs a personalizer (see the section *Direct Mode* above). In container mode a transient personalizer object is used, which stores the table personalization data in its memory. To persist personalization data, you have to explicitly copy it to an item inside the container.

Demo App

In addition to the code samples in this section, you can find a demo app in the SAPUI5 demo kit under the following path:

`/test-resources/sap/ushell/demoapps/AppPersSample/App.view.xml`

`/test-resources/sap/ushell/demoapps/AppPersSample/App.controller.js`

For example, on an ABAP server, the path may be as follows:

`http://<server>:<port>/sap/public/bc/ui5_ui5/demokit/test-resources/sap/ushell/demoapps/AppPersSample/`

JSDocs

You can find the JSDocs for the personalization service under the following path in the SAPUI5 demo kit:

`#docs/api/symbols/sap.ushell.services.Personalization.html`

1.5.1.6.3.1 Selected Method Parameters

.The following parameters are used in the factory methods of the personalization service, for example `getContainer` and `getPersonalizer`.

oScope

The scope object contains parameters for controlling the behavior of objects of the personalization service. Note the following parameters:

- **validity**

This parameter is used in container mode to control how long a container is valid. After the validity has expired, the data may still be stored on the front-end server, but it is no longer used. The `validity` parameter can have the following values:

Value	Description
0	The data is persisted in the launchpad window, and is lost when the window is closed or when the launchpad is reloaded.
`<nn>`	Number of minutes the data is to be persisted on the front-end server, for example **30**.
infinity	The data is persisted on the front-end server without any time restrictions.

> i Note
>
> For validity > 0 (including `infinity`) a copy of the container is stored in the volatile memory of the launchpad window in the browser. Therefore, no roundtrip is necessary for reloading a container at a later point in time. Only if the launchpad itself is reloaded, the copy is lost and a roundtrip is necessary.

- **Parameters controlling the browser caching of personalization data**
  The following parameters control whether personalization data stored in a container is cached.

Property	Value	Default Value	Description	Access to Enum Constants
keyCategory	FIXED_KEY | GENERATED_KEY	GENERATED_KEY	FIXED_KEY is a static hard-coded key which does not change. This value is required for caching to take place. GENERATED_KEY is for all other cases.	constants.keyCategory.FIXED_KEY = "FIXED_KEY" constants.keyCategory.GENERATED_KEY = "GENERATED_KEY"

Property	Value	Default Value	Description	Access to Enum Constants
`writeFrequency`	`HIGH	LOW`	`HIGH`	Use `LOW` if the key is typically rarely updated, for example in explicit table personalization by end users.	constants.writeFrequency.HIGH= "HIGH" constants.writeFrequency .LOW = "LOW"
`clientStorageAllowed`	`true	false`	`false`	Security-relevant data must not be stored on client side, for instance in local storage or in the HTTP browser cache. Applications should set this parameter to `true` only if the data is not critical from a security point of view.	n/a

Caching of personalization data reduces read calls to the front-end server. The cache buster uses one key for all cached containers. As a consequence, if one cached personalization container is changed, the cache for all personalization containers is invalidated.

oComponent

The component object of the current app is stored together with the personalizaton data. It can be retrieved in a controller object as follows:

⚛ Sample Code

```
var oComponent = sap.ui.core.Component.getOwnerComponentFor(this.getView());
```

You can use the component ID as a query parameter for selective cleanup of personalization data.

Related Information

1.5.1.6.3.2 Direct Mode: Simple Use Case

In this example, the personalization data is a set of flags set by the user and then explicitly saved.

Implementation Steps

We need the following steps in our application:

1. Application initialization:
   1. Get the personalizer.
   2. Read the personalization data.
2. React on the *Save* button.
   1. Write the personalization data.

Get the Personalizer

There is one personalizer per item. Technically, each item is part of a container. So a personalizer is identified by the container and the item.

> ⇘ Sample Code
>
> ```
> oPersonalizationService = sap.ushell.Container.getService("Personalization");
> oComponent = sap.ui.core.Component.getOwnerComponentFor(this.getView());
> oScope = {
>   keyCategory : oPersonalizationService.constants.keyCategory.FIXED_KEY,
>   writeFrequency: oPersonalizationService.constants.writeFrequency.LOW,
>   clientStorageAllowed : true
> };
> ...
> oPersId = {
>   container : "sap.ushell.samples.persSample1",
>   item : "fruits"
> };
> ...
> oPersonalizer = oPersonalizationService.getPersonalizer(oPersId, oScope,
> oComponent);
> ```

Read the Personalization Data

The `getPersData` method is asynchronous and uses the deferred object functionality (`jQuery.Deferred`).

> **⟐ Sample Code**
>
> ```
> oReadPromise = oPersonalizer.getPersData()
>   .done(function(oPersData){
>     // Pseudo code: set the checkboxes on the UI according to the read
> personalization data
>   })
>   .fail(function() {
>     jQuery.sap.log.error("Reading personalization data failed.")
>   });
> ```

Write the Personalization Data

The save method is asynchronous as well.

> **⟐ Sample Code**
>
> ```
> oSavePromise = this.oPersonalizer.setPersData(oPersData)
>   .done(function(){
>     // Tell the user that the data was saved
>   })
>   .fail(function() {
>     jQuery.sap.log.error("Writing personalization data failed.")
>   });
> ```

The personalization service ensures that the save and load operations to the front-end server are sequentialized, so that a subsequent operation is only started when the previous one is finished.

1.5.1.6.3.3 Direct Mode: Table Personalization

Table personalization is available for the table control of UI5 Mobile. This example uses framework functionality that allows you to develop table personalization easily.

A personalizable table has a button for starting personalization. Personalization of tables means switching on and off the visibility of columns and changing the order of the columns.

In the view.xml, there is a button for starting personalization. No click handler needs to be assigned to this button. When the view is initialized, the personalization data is read (in the ABAP shell from the front-end server; in the sandbox shell from the browser local storage) and automatically applied to the table control.

When a user clicks the button, a dialog box is displayed where the user can switch the visibility of columns on and off, and change the order of the columns using arrow keys. When the user clicks OK, the personalization data is stored on the front-end server (or in the browser local storage if the sandbox is used).

Implementation Steps

We need the following steps in our application:

- In the view containing the table:
  - Add a button for starting personalization.
- Application initialization:
  - Create the table personalizer.

Add a Button for Starting Personalization

We recommend you add a button for starting personalization to the header Toolbar of the table. In views, static IDs have to be used for tables and columns.

⇥ Sample Code

```
<Table id="SampleTableMobile"headerText="Table header"
  showSeparators="Inner"items="{ path : '/rows' }">
  <headerToolbar>
    <Toolbar>
      <Label text="Test Table for Personalization"></Label>
      <ToolbarSpacer></ToolbarSpacer>
      <Button id="personalize"icon="sap-icon://table-view"/>
    </Toolbar>
  </headerToolbar>
  <columns>
    <Column width="100px">
    ...
```

Create the Table Personalizer

Note that the table personalizer is a different object than the (personalization) personalizer.

⇥ Sample Code

```
oComponent = sap.ui.core.Component.getOwnerComponentFor(this.getView());
oPersonalizationService = sap.ushell.Container.getService("Personalization");
oPersId = {
  container : "AppPersSample2",
  item : "mobiletable"
};
// Get the table control and the button control
oMobileTable = this.getView().byId("SampleTableMobile");
oStartPersButton = this.getView().byId("personalize");
// Get a Personalizer
oScope = {
  keyCategory : oPersonalizationService.constants.keyCategory.FIXED_KEY,
  writeFrequency : oPersonalizationService.constants.writeFrequency.LOW,
  clientStorageAllowed : true
};
oPersonalizer = oPersonalizationService.getPersonalizer(oPersId, oScope,
oComponent);
```

```
  // Create a table personalization controller
oTablePersoController = newsap.m.TablePersoController({
  table : oMobileTable,
  persoService : oPersonalizer
});
// Use the personalization data to configure the table accordingly
oTablePersoController.activate();
// Attach the personalization button in the table to the personalization popup
oStartPersButton.attachPress(function() {
  oTablePersoController.openDialog();
};
...
```

1.5.1.6.3.4 Container Mode: Simple Use Case

This example shows how to use the container mode.

In this example we create the same user interface as in .

Implementation Steps

We need the following steps in our application:

- Application initialization:
  - Get the container and read the personalization data.
- React to the Save button:
  - Write the personalization data to the container and save the container.

Get the Container and Read the Personalization Data

The factory method `getContainer` is asynchronous as it loads the personalization data from the storage. In the respective `done` method the data from the respective item is read.

> ⇆ Sample Code
>
> ```
> var that = this;
> ...
> oComponent = sap.ui.core.Component.getOwnerComponentFor(this.getView());
> oPersonalizationService = sap.ushell.Container.getService("Personalization");
> ...
> oScope = {
>   keyCategory : oPersonalizationService.constants.keyCategory.FIXED_KEY,
>   writeFrequency: oPersonalizationService.constants.writeFrequency.HIGH,
>   clientStorageAllowed : false,
>   validity : 30
> };
> oPersonalizationService.getContainer("sap.ushell.samples.persSample1",
> oScope, oComponent)
>   .fail(function() {
> ```

```
    jQuery.sap.log.error("Loading personalization data failed.")
  })
  .done(function(oContainer) {
    that.oContainer = oContainer;
      that.oFruits = oContainer.getItemValue("fruits");
      that.oVegetables = oContainer.getItemValue("vegetables");
      // Pseudocode: apply the personalization data to the UI checkboxes
  };
```

Write the Personalization Data to the Container and Save the Container

> ⌨ Sample Code
>
> ```
> var that = this;
> ...
> this.oContainer.setItemValue("fruits", this.oFruits);
> this.oContainer.setItemValue("vegetables", this.oVegetables);
> this.oContainer.save()
>   .fail(function() {
>     jQuery.sap.log.error("Saving personalization data failed.")
>   })
>   .done(function() {
>     // Tell the user that the data is saved
>   };
> ```

1.5.1.6.3.5 Container Mode: Table and Query Personalization

In this example we use the same table personalization as in the direct mode example above, and we add a query value for controlling the entries displayed in the table.

In addition to a button for starting the table personalization, the UI for this example needs a UI element for specifying the query value, for example an input field and a *Save* button to save the personalization data for the table and for the query value. The purpose of this example is to show a simple example before moving on to the fullyfledged scenario that also includes variants.

Table personalization requires a personalizer to handle the table personalization data. Since the personalizer built for the direct mode saves implicitly, it is not suitable for the container mode. For this reason there is a transient personalizer that does not save to the front-end server.

Implementation Steps

We need the following steps in our application

- In the view containing the table:
  - Add a button to the table for starting personalization.
  - Add a *Save* button.

- Application initialization:
  - Create the transient personalizer and attach it to the table personalization controller.
  - Get the container and load its data.
  - Execute the application data query and set the query result in the UI element.
  - Read the table personalization.
- React on the *Save* button:
  - Get the personalization values and save them.

Add a Button for Starting Personalization

We recommend to add a button for starting the personalization to the header Toolbar of the table. In views, static IDs have to be used for tables and columns.

⌨ Sample Code

```
<Table id="SampleTableMobile"headerText="Table header"
  showSeparators="Inner"items="{ path : '/rows' }">
  <headerToolbar>
    <Toolbar>
      <Label text="Test Table for Personalization"></Label>
      <ToolbarSpacer></ToolbarSpacer>
      <Button id="personalize" icon="sap-icon://table-view"/>
      <Button id="save" icon="sap-icon://save"/>
    </Toolbar>
  </headerToolbar>
  <columns>
    <Column width="100px">
    ...
```

Create the Transient Personalizer and Attach it to the Table Personalization Controller

⌨ Sample Code

```
oPersonalizationService = sap.ushell.Container.getService("Personalization");
oComponent = sap.ui.core.Component.getOwnerComponentFor(this.getView());
...
// Get the table control and the button control
oMobileTable = this.getView().byId("SampleTableMobile");
oStartPersButton = this.getView().byId("personalize");
oSaveButton = this.getView().byId("save");
// Get the transient personalizer
this.oPersonalizer = oPersonalizationService.getTransientPersonalizer(); //
no params here
// Create a table personalization controller
this.oTablePersoController = newsap.m.TablePersoController({
  table : oMobileTable,
  persoService : this.oPersonalizer
});
// Attach the personalization button in the table to the personalization popup
oStartPersButton.attachPress(function() {
```

```
    this.oTablePersoController.openDialog();
};
// Attach the save button to the save method
oSaveButton.attachPress(function(){
  this.oSavePersonalization();
};
...
```

Get the Container, Read and Use the Personalization Data

The factory method `getContainer` is asynchronous as it loads the personalization data from the storage. In the respective `done` method all subsequent initialization steps have to be done.

≡, Sample Code

```
var that = this;
...
oScope = {
  keyCategory : oPersonalizationService.constants.keyCategory.FIXED_KEY,
  writeFrequency: oPersonalizationService.constants.writeFrequency.LOW,
  clientStorageAllowed : true,
  validity : Infinity
};
oPersonalizationService.getContainer("sap.ushell.samples.persSample2",
oScope, oComponent)
  .fail(function() {
    jQuery.sap.log.error("Loading personalization data failed.")
  })
  .done(function(oContainer) {
    that.oContainer = oContainer;
    that.oQueryParam = oContainer.getItemValue("queryParam");
    // Pseudocode: set the query param in the UI element
    // Pseudocode: read the table content considering oQueryParam
    that.oPersonalizer.setValue(oContainer.getItemValue("table"));
    // Use the personalization data to configure the table accordingly
    that.oTablePersoController.activate();
  };
```

Get the Personalization Values and Save Them

≡, Sample Code

```
this.oSavePersonalization = function(){
  // Pseudocode: get the query parameter value from the UI into
this.oQueryParam
  this.oContainer.setItemValue("queryParam", this.oQueryParam);
  this.oContainer.setItemValue("table", this.oPersonalizer.getValue());
  this.oContainer.save()
    .fail(function() {
      jQuery.sap.log.error("Saving personalization data failed.")
    })
    .done(function() {
      // Before the next save is triggered the last one has to be finished.
      // Could be done by disabling the save button during the  save.
```

```
        };
    };
```

1.5.1.6.3.6 Container Mode: Table and Query Personalization with Variants

A variant is a set of personalization settings that users can save with a name. They can later restore these settings by selecting the variant name, for example from a dropdown list.

This example is based on the example Container Mode: Table and Query Personalization [page 567]. In addition, users can select personalization variants that they have previously saved.

In this example we deal with only one variant set. The following figure shows the interactions regarding personalization.

The personalized values for the query and the table are only persisted when the user explicitly saves them as variants. The user can select a variant to be applied. The application can start with a default personalization, or it can start with the personalization variant the user last selected. The latter alternative is shown in the activity diagram.

Implementation Steps

We need the following steps in our application:

- In the view containing the table:
  - Add a button for starting personalization (see the previous example for details)
  - Add a button, a popup, and a dropdown list for the variant handling (not described here)
- Application initialization:
  - Create the transient personalizer and attach it to the table personalization controller (see the previous example for details).

- Get the container and read the personalization data for the current variant.
- Get the names and keys for all variants in order to populate a selection control (dropdown list).
- Execute the query and set the query value in the UI element.
- Apply the table personalization.
- React on the user's variant selection action:
  - Read the variant data and apply it.
  - Set the selected variant as current variant.
- React on user's variant save action:
  - Get the current personalization values and save them to the variant of the name supplied.

Get the Container, Read and Apply the Personalization Data for the Current Variant

The factory method **getContainer** is asynchronous as it loads the personalization data from the storage. In the respective **done** method all subsequent initialization steps have to be done.

⇛ Sample Code

```
var that = this;
...
oPersonalizationService = sap.ushell.Container.getService("Personalization");
oComponent = sap.ui.core.Component.getOwnerComponentFor(this.getView());
...
oScope = {
  keyCategory : oPersonalizationService.constants.keyCategory.FIXED_KEY,
  writeFrequency: oPersonalizationService.constants.writeFrequency.LOW,
  clientStorageAllowed : true,
  validity : Infinity
};
oPersonalizationService.getContainer("sap.ushell.samples.persSample2",
oScope, oComponent)
  .fail(function() {
    jQuery.sap.log.error("Loading personalization data failed.")
  })
  .done(function(oContainer) {
    that.oContainer = oContainer;
    that.oVariantSetAdapter = new
sap.ushell.services.Personalization.VariantSetAdapter(that.oContainer);
    that.oVariantSet = that.oVariantSetAdapter.getVariantSet("orderTable");
    // select control
    that.oVariantList = that.oVariantSet.getVariantNamesAndKeys();
    // Pseudocode: Assign the entries of oVariantList to the selected control
    // get variant data
    that.sVariantKey = that.oVariantSet.getCurrentVariantKey();
    applyVariant(that.sVariantKey);
  };
...
function applyVariant(sVariantKey) {
  var oVariant =    {};
  oVariant = that.oVariantSet.getVariant(sVariantKey);
  // query params
  this.oQueryParam = this.oVariant.getItemValue("queryParam");
  // Pseudocode: set the query param in the UI element
  // Pseudocode: read the table content considering oQueryParam
  // table
  this.oPersonalizer.setValue(oVariant.getItemValue("table"));
```

```
    // use the personalization data to configure the table accordingly
    this.oTablePersoController.activate();
}
...
```

React on the User's Variant Selection Action

≒ Sample Code

```
...
  applyVariant(sSelectedVariantKey);
  this.oVariantSet.setCurrentVariantKey(sSelectedVariantKey);
  this.oContainer.save() // save the whole container!
    .fail(function() {
      jQuery.sap.log.error("Saving personalization data failed.")
    })
    .done(function() {
      // Tell the user that the personalization data was saved
    };
...
```

React on the User's Variant Save Action

≒ Sample Code

```
...
bDone = false;
oVariant = undefined;
do{
  //  Pseudocode: Popup for variant name
  switch(sVariantNamePopupInput) {
    case"cancel":
      bDone = true;
      break;
    case"save":
      break;
  }
  if(!bDone) {
    sVariantKey = this.oVariantSet.getVariantKeyByName(sVariantNameInput);
    if(sVariantKey) { // variant with this name exists
      // Pseudocode: Popup to ask the user if he/she wants to overwrite the
existing variant
      switch(sVariantOverwritePopupInput) {
        case"cancel":
          // user has to enter a new name
          break;
        case"overwrite":
          this.oVariant = this.oVariantSet.getVariant(sVariantKey);
          break;
      }
    } else{
      this.oVariant = this.oVariantSet.addVariant(sEnteredVariantName);
    }
    if(this.oVariant) {
```

```
      // Do save
      this.oVariant.setItemValue("queryParam", this.oQueryParam);
      this.oVariant.setItemValue("table", this.oPersonalizer.getValue());
      this.oVariantset.setCurrentVariant(this.oVariant.getVariantKey());
      this.oContainer.save() // save the whole container!
        .fail(function() {
          jQuery.sap.log.error("Saving personalization data failed.")
        })
        .done(function() {
          // Tell the user that the personalization data was saved
        };
      bDone = true;
    }
  }
} while(!bDone);
...
```

Related Information

Container Mode: Table and Query Personalization [page 567]

1.5.1.6.3.7 Troubleshooting Personalization

Personalization data is stored on the front-end server in the following tables:

Table	Description
/UI2/CONTAINER	Data of context containers
	Validity `Infinity` is stored as `0` on the front-end server.
	Do not confuse with the validity `0` of the JavaScript API.
	Containers stored with the validity `0` of the JavaScript API are not stored on the front-end server at all.
/UI2/ITEM	Data of the items in context containers

You can use transaction `SE16` and filter by your user to check if the personalization data is stored correctly.

For cleaning up personalization data, you can use program `/UI2/PERS_EXPIRED_DELETE` or transaction `/UI2/PERS_DEL`.

Related Information

Cleanup of Expired Personalization [page 359]

1.5.1.7 Best Practices for Developing Applications

Best Practices for developing SAP Fiori apps that run in the SAP Fiori launchpad

Best practices intend to ensure that your apps run without side effects that may be difficult to analyze in the SAP Fiori launchpad:

- Build your apps as self-contained SAPUI5 components.
  "Apps" in the SAP Fiori launchpad are SAPUI5 components. The launchpad instantiates your app by loading the `Component.js` file. Use an app-specific `index.html` file for standalone apps only. The unified shell does **not** load the `application.js` and `index.html` files.
- Declare the configuration information, for example, the location of icons and library dependencies, in the application descriptor. For more information, see .
- Do **not** use global variables. If you cannot avoid a global variable, use qualified names to ensure uniqueness.
- If you need an event bus, use the event bus of the component (`sap.ui.core.Component.getEventBus`). By this, you avoid conflicting event names and make sure that your listeners are automatically removed when the component is unloaded. Do **not** use the global event bus (`sap.ui.getCore().getEventBus()`).
- Register models on the root component or on single views of your apps.
  Example: `this.getView().setModel("MyModel",aModel);` Do **not** use `sap.ui.getCore()` to register models.

> → Tip
>
> To get the component that owns an embedded view, you can use the following line of code in the view controller:
>
> ```
> sap.ui.core.Component.getOwnerComponentFor(this.getView());
> ```

  For more information, see in the SAPUI5 Developer Guide.
- Let SAPUI5 generate IDs for global elements.
  Do not set explicit IDs for global elements in your code. This may prevent you from running an app several times in the SAP Fiori launchpad. Do **not** rely on `sap.ui.getCore().byId()` for global location of elements. Always use local names `getView().byId()` and let SAPUI5 generate the IDs of views and components.
- Use only the SAPUI5 APIs to manipulate the location hash.
  Do **not** read or write directly to `window.location.hash` or `window.location`. The SAP Fiori launchpad uses URL hashes for its own navigation. Direct manipulation of the location hash would interfere with the lauchpad navigation. For cross-app navigation, use the `CrossApplicationNavigation` service. For more information on this service, see sap.ushell.services.CrossApplicationNavigation .
- Ensure that all controls created by your component are destroyed when the component is destroyed.
  All controls in the component's control tree that are defined in an XML view in a static way or explicitly added to a parent control in a JavaScript view are destroyed automatically. Controls that are not part of a parent, however, are **not** automatically destroyed, for example dialog instances. You ensure proper destruction by adding the respective controls to the corresponding view with the method `addDependent` of `sap.ui.core.Element`.
- Avoid using `sap.ui.localResources` inside your `Component.js` file.

sap.ui.localResources registers a path relative to the FioriLaunchpad.html page. Components must **not** make implicit assumptions on this location.

- Avoid using jQuery.sap.registerModulePath inside your Component.js file. This creates a dependency to a specific platform.

- Do **not** program against a specific intent or action. The intent is distinct from the physical application. It may be reconfigured.

- Use the data source mechanism of the app descriptor to create the OData model. This is an important prerequisite for the framework to propagate the segment parameter.

1.5.2 Extending the Launchpad

This section provides information about tasks that you can perform to extend SAP Fiori launchpad.

In this interactive graphic, explore the components of the launchpad that you can extend. Hover over each callout for a description. Click the callouts for more information.

- Developing Plug-Ins [page 601]
- Adding a Title to the Shell Bar [page 614]
- Displaying and Hiding the Shell Bar [page 613]
- Adding Custom Icons to the Shell Bar [page 615]

1.5.2.1 Custom Tiles and Tile Types

Custom tile types can be created for specific tasks when configuring a static or dynamic tile does not fulfill the requirement, e.g. when the tile should show different information.

When you want to create your own tiles for the launchpad there are several options: You can configure one of the tiles that are delivered by SAP (e. g. static or dynamic tiles), you can use Smart Business to create KPI tiles (when you use SAP HANA, see e.g. Configuring the SAP Smart Business KPI Tile) or you can create your own custom tile type if you need specific features, e.g. different content shown on a tile.

How to create those tiles in the launchpad app manager (recommended) is described in Maintaining Tiles [page 156]; how to configure and create static and dynamic app launcher tiles with the launchpad designer is described in Configuring Tiles [page 207]. In the following sections you'll learn what you need to do to create a custom tile type and then use it to create a custom tile.

Related Information

Tiles [page 105]

1.5.2.1.1 What Is a Custom Tile Type?

Generally, when you see it from the launchpad perspective, a custom tile type is not just a control. It is rather a self-contained "mini-UI5 app" with a controller that includes its own (business) logic and a view with the actual user interface. It runs in a launchpad page or home page. The controller can trigger network requests to fetch data, for example. In addition, it has access to APIs to communicate with the launchpad.

Schematic Display of a Tile (Type)

On the ABAP platform, the mentioned APIs are part of the UI2 services framework. The UI2 services framework injects the CHIP API (getComponentData().chip) into the component, which provides different functions depending on what contracts have been consumed in the CHIP Definition XML. Look for example at the namespace contract to see what contracts comprise. In Accessing the CHIP API and the Contracts [page 583] we describe which contracts are available for a custom tile type.

The following figure shows the relation between the custom tile, the APIs and the framework.

Custom Tile and Framework

1.5.2.1.2   Implementing a Custom Tile Type

Learn how to implement a custom tile type.

Context

The following steps provide an overview of the procedure. Follow the links for more information on the single steps.

Procedure

1. Implement the tile logic and UI based on a SAPUI5 component. For more information, see .

   For the tile type's view, we recommend that you use the `sap.m.GenericTile` to maintain compatibility with updates in the future. See class sap.mGenericTile🔗 for additional information.

2. Create a descriptor file (`chip.xml`) for the new tile type. For more information, see Creating a CSR CHIP [page 583]. Consume framework services (contracts) as required; for example, to access the configuration of a tile instance.

   Accessing the CHIP API and the Contracts [page 583] and the following sections show which contracts are supported for a custom tile and give an example.

   > ℹ️ Note
   >
   > For more information, see CHIP Definition XML [page 580] and Accessing the CHIP API and the Contracts [page 583].

3. To expose the new tile type in the launchpad designer, the administrator needs to register the type in the ABAP Frontend server. For more information, see Assigning a CHIP to the Universal Fiori Launchpad Catalog [page 596].

4. When you want to work with the custom tile in other launchpad design-time tools, e.g. in the launchpad app manager or in the *Manage Launchpad Pages* app, you need to implement BAdI /UI2/ BADI_FDM_CUST_TILE. SAP recommends this step. See Exposing Custom Tile Type Information to Launchpad Design-Time Tools [page 599] for more information. If you still work with the launchpad designer, you can create a custom tile with this tile type there. Open the launchpad designer and select the catalog for which you want to create a new tile. Click on the + icon to create a new tile and select the new custom tile type. Also see Creating or Deleting Tiles or Target Mappings [page 203].

1.5.2.1.2.1 CHIP Development

A CHIP (Collaborative Human Interface Part) is an encapsulated, stateful piece of software used to provide functions in collaboration with other CHIPs in a Page Builder page or side panel. All available CHIPs are registered in a library (CHIP catalog). The CHIP model describes capabilities of a CHIP and is not based on a specific UI technology. Client-side rendering CHIPS (CSR CHIPs) are CHIPs that use client-side rendering capabilities, such as SAPUI5.

When you want to create a custom tile, a CHIP is needed to register the tile in the universal catalog.

1.5.2.1.2.1.1 CHIP Specification Schema

The CHIP specification schema is described in an XSD file. The current schema is part of your installation and can be found here: `http://<server>:<port>/sap/public/bc/ui2/services/sap/ui2/srvc/ chipdefinition.xsd`

The CHIP description consists of the following elements:

- `implementation`
  This element specifies the implementation details and is mandatory.
- `appearance`
  This element specifies the title and a description for the CHIP.
- `contracts`
  This element specifies the contracts that can be implemented..
- `parameters`
  This element specifies additional parameters that can be defined.
- `sources`
- `property paths`

1.5.2.1.2.1.2 CHIP Definition XML

In the CHIP definition file you define the elements of the CHIP specification schema used in your CHIP. Whatever you define here can be accessed from your coding using the CHIP API.

Element	Description
`implementation`	As implementation only `sapui5` is supported.
	You can reference an SAPUI5 Component. Also see Accessing the CHIP API and the Contracts [page 583].
`appearance`	Set the title and the description of your CHIP.
`contracts`	Define the contracts used in your CHIP. Also see Accessing the CHIP API and the Contracts [page 583].
`parameters`	These parameters can, for example be used to determine in which catalogs your CHIP is displayed.
`sources`	
`property paths`	

CHIP Definition XML Referencing an SAPUI5 Component

For `componentName` specify the path to the folder where your `Component.js` file is located.

⌨ Sample Code

```
<?xml version="1.0" encoding="UTF-8"?>
<!-- Copyright (c) 2013 SAP AG, All Rights Reserved -->
<chip xmlns="http://schemas.sap.com/sapui2/services/Chip/1">
  <implementation>
    <sapui5>
      <basePath>../../../../../../</basePath>
      <componentName>sap.hba.apps.kpi.WebContent.KPITile.resources</
componentName>
```

```
      </sapui5>
    </implementation>
</chip>
```

Specifying the Base Path

If your component is located under a different root folder than your CHIP definition XML, you should add a `basepath` element to indicate the common root folder of both files. You can use a virtual namespace for this. When the code is deployed e.g. in the BSP Repository of an ABAP system, it may have a path like /sap/bc/ui5_ui5/Z_MY_KPI_TILE that does not represent the logical namespace which might be something like sap.hba.apps.kpi.KPITile.

The following example shows a CHIP implementation without a virtual namespace. This is used e.g. in older CHIP implementations. You can use a virtual namespace in addition, this is shown in the next section.

≡ Sample Code

/hana/sap/hba/apps/kpi/WebContent/KPITile/resources/Kpi.xml

```xml
<?xml version="1.0" encoding="UTF-8"?>
<chip xmlns="http://schemas.sap.com/sapui2/services/Chip/1">
  <implementation>
    <sapui5>
      <basePath>../../../../../../</basePath>
      <viewName>sap.hba.apps.kpi.WebContent.KPITile.resources/
view.Kpi.view.js</viewName>
    </sapui5>
  </implementation>
</chip>
```

This will result in the following call:

≡ Sample Code

```
jQuery.sap.registerModulePath("sap.hba.apps.kpi.WebContent.KPITile.resources",
  "/hana/sap/hba/apps/kpi/WebContent/KPITile/resources")
```

The module name prefix is derived from the `<viewName>` element. The slash (`/`) indicates the end of the prefix. This makes it possible for the contoller to reside in a sibling folder of the `view` folder, named `contoller`.

The URL prefix is computed from the following parts:

- The path to the CHIP definition XML (`/hana/sap/hba/apps/kpi/WebContent/KPITile/resources`)
- The `<basePath>` element (`../../../../../../`)
- The module name prefix (`sap.hba.apps.kpi.WebContent.KPITile.resources`)

Specifying a Virtual Namespace

Please refer to the following CHIP implementation as an example using a virtual namespace. Please note that the "WebContent" and "resources" segments can be removed from the namespace.

```
/hana/sap/hba/apps/kpi/WebContent/KPITile/resources/Kpi.xml
<?xml version="1.0" encoding="UTF-8"?>
<chip xmlns="http://schemas.sap.com/sapui2/services/Chip/1">
  <implementation>
    <sapui5>
      <basePath>./</basePath>
      <componentName virtualNamespace="true">sap.hba.tiles.kpi</componentName>
    </sapui5>
  </implementation>
</chip>
```

This SAPUI5 implementation definition will result in the following code to be executed when this CHIP's SAPUI5 implementation is created:

Result of virtual namespace

```
jQuery.sap.registerModulePath(
    "sap.hba.tiles.kpi",
    "/hana/sap/hba/apps/kpi/WebContent/KPITile/resources"
);
return new sap.ui.core.ComponentContainer({
    component: sap.ui.getCore().createComponent({
        componentData: {
          chip: oApi // CHIP API
        },
        name: "sap.hba.tiles.kpi"
    })
});
```

CHIP Definition XML Referencing an SAPUI5 View (deprecated)

i Note

This option is deprecated and will be removed in a future version. We recommend using CHIP Definition XML referencing an SAPUI5 component instead, as described above.

For `viewName` specify the name of the `sapui5 view` containing the content of your CHIP. The `viewName` comprises package path, the name of the view and the file extension, for example `tests.sample.Myview.view.xml`.

The following `sapui5` view types are supported:

- Javascript with file extension `js`
- XML with file extension `xml`
- JSON with file extension `json`

The following example shows a CHIP definition with an `sapui5` view of type `xml`, with a title and two contracts: `fullscreen` and a configuration that specifies a catalog:

```
<?xml version="1.0" encoding="UTF-8"?>
```

```
<chip xmlns="http://schemas.sap.com/sapui2/services/Chip/1">
  <implementation>
    <sapui5>
      <viewName>sap.ui2.tests.fullsize.Full.view.xml</viewName>
    </sapui5>
  </implementation>
  <appearance>
    <title>Chip taking full size</title>
  </appearance>
  <contracts>
    <consume id="fullscreen"/>
    <consume id="configuration">
      <parameters>
        <parameter name="parameterName">defaultValue</parameter>
      </parameters>
    </consume>
  </contracts>
</chip>
```

1.5.2.1.2.1.3 Creating a CSR CHIP

Context

For developing client-side rendering CHIPs (CSR CHIPs) you can use any code editor, e.g. SAP WebIDE. An SAPUI5 application is extended by a CHIP definition, which is used to enable the application to run in a page builder.

Procedure

1. Create an SAPUI5 application with a component.
2. Create a CHIP definition xml file, e. g. called mychip.chip.xml and define the entries. For more information, see CHIP Definition XML [page 580].
3. To access the CHIP API, proceed as described under Accessing the CHIP API and the Contracts [page 583].
4. Create the content of your CHIP with the SAPUI5 means.

1.5.2.1.2.1.4 Accessing the CHIP API and the Contracts

Context

With the CHIP API you can access the contracts and all elements defined in the CHIP definition file.

Procedure

1. To access the CHIP API from the component of your project, retrieve it via the component data:

```
sap.ui.define([
    "sap/ui/core/UIComponent"
], function (UIComponent) {
    "use strict";
    return
UIComponent.extend("sap.ushell.demotiles.abap.customDemoTile.Component", {
        metadata: {
            manifest: "json"
        },
        createContent: function () {
            var oChipData = this.getComponentData() &&
this.getComponentData().chip;
            // ...
        }
    });
});
```

2. You now can access the contracts as properties as described in the following topics.

   As a prerequisite for using a contract, you must consume this contract in the CHIP definition XML. The respective property is then available in the CHIP API object.

Related Information

CHIP Definition XML [page 580]

1.5.2.1.2.1.4.1 Accessing the "bag" Contract

You can use property bags to persist any key-value pairs in order to store settings for a CHIP instance.

Using property bags, you can persist page settings in the following scopes:

- **Personalization**: User-specific settings (These settings supersede Customizing and Configuration settings.)
- **Customizing**: Client-specific settings (These settings supersede *Configuration* settings, but can be superseded by *Personalization* settings.)
- **Configuration**: System-wide settings (These settings can be superseded by *Customizing* and *Personalization* settings.)

As a prerequisite, you have to consume the bag contract in the CHIP definition XML.

The bag property is then available in the CHIP API object.

> **i Note**
>
> CHIPs are not the only objects that can store properties in bags. Property bags may also be created by a client-side page builder.

Example: Code Example

```
var oChipApi = this.getComponentData().chip;
var aBagIds = oChipApi.bag.getBagIds();
var oBag = oChipApi.bag.getBag("sample_bag");
```

Related Information

CHIP Definition XML [page 580]

http://help.sap.com/javadocs/ui-services/services/symbols/chip.bag.html

1.5.2.1.2.1.4.2 Accessing the "search" Contract

This contract interface allows the page builder to get the CHIP-specific search keywords. In addition, it can have the CHIP highlight words in its UI based on the search terms used.

As a prerequisite, you have to consume the `search` contract in the CHIP definition XML.

The `search` property is then available in the CHIP API. It provides the following methods:

- `chip.search.setKeywords()`
  You can use this method to set CHIP-specific keywords for finding the CHIP in an overall CHIP search. If the CHIP does not set any keywords, the page builder can only search based on CHIP metadata like the CHIP title.
- `chip.search.attachHighlight()`
  You can use this method to attach an event handler to the "highlight" event, which is fired whenever the user executes a search over all CHIPs using specific search terms. The words to be highlighted, which are provided as parameters to the event handler, are derived from the search terms. The CHIP can highlight these in its UI.

Example: Code Example

```
init: function() {
  var oChipApi = this.getComponentData().chip;
  var oResourceBundle = this.getView().getModel("i18n");
  oChipApi.search.setKeywords([oResourceBundle.getProperty("info"],
oResourceBundle.getProperty("subtitle"]);
  oChipApi.search.attachHighlight(function(aHighlightWords) {
    //enable CHIP UI to highlight words
  });
}
```

In this example, the CHIP instance sets its language-dependent, CHIP-specific search keywords in the component's init method. In addition, it attaches an event handler for highlighting words from the search in its UI.

Related Information

1.5.2.1.2.1.4.3 Accessing the "writeConfiguration" Contract

As a prerequisite, you have to consume the `writeConfiguration` contract as well as the `configuration` contract in the CHIP definition XML.

The `writeConfiguration` property is then available in the CHIP API object. It provides the method `chip.writeConfiguration.setParameterValues()`, which allows mass updates of parameters that have been declared in the CHIP definition XML.

> **i Note**
>
> Only string values are supported.
>
> The default error handler is usually provided by the page builder.

Example: Code Example

```
var oChipApi = this.getView().getViewData().chip;
oChipApi.writeConfiguration.setParameterValues({
  columns: "2",
  id: "07UI2_EPM0BUI2_EPMDEMO00",
  showChildren: "false"
}, function () {
  // continue in case of success...
}, function (sErrorMessage) {
  alert("write configuration failed: " + sErrorMessage);
});
```

Related Information

1.5.2.1.2.1.4.4 Accessing the "fullscreen" Contract

As a prerequisite, you have to consume the `fullscreen` contract in the CHIP definition XML.

The `fullscreen` property is then available in the CHIP API object. It provides the following methods:

- `chip.fullscreen.attachFullscreen()`
  You can use this event handler to find out when your CHIP should display a different view, for example when the user switches the fullscreen mode.
- `chip.fullscreen.getFullscreen()`
  You can use this method to find out whether fullscreen mode is on.
- `chip.fullscreen.setFullscreen()`
  You can use this method to switch fullscreen mode from within your CHIP.

Example: Code Example

```
var oChipApi = this.getComponentData().chip;
var bIsFullscreen = oChipApi.fullscreen.getFullscreen();
```

Related Information

http://help.sap.com/javadocs/ui-services/services/symbols/chip.fullscreen.html

1.5.2.1.2.1.4.5 Accessing the "refresh" Contract

As a prerequisite, you have to consume the `refresh` contract in the CHIP definition XML.

The `refresh` property is then available in the CHIP API object. It provides the method `chip.refresh.attachRefresh()` to learn when a refresh is requested.

Example: Code Example

```
var oChipApi = this.getComponentData().chip;
oChipApi.refresh.attachRefresh(function (){
  // do refresh now
});
```

Related Information

1.5.2.1.2.1.4.6 Accessing the "searchProvider" Contract

As a prerequisite, you have to consume the `searchProvider` contract in the CHIP definition XML.

The `searchProvider` property is then available in the CHIP API object. It provides the method `chip.searchProvider.addSearchProvider()` for adding a search provider.

Example: Code Example

```
var oChipApi = this.getComponentData().chip;
oChipApi.searchProvider.addSearchProvider("http://foo.sap.corp/
searchProvider.xml");
```

Related Information

1.5.2.1.2.1.4.7 Accessing the "url" Contract

As a prerequisite, you have to consume the `url` contract in the CHIP definition XML.

The `url` property is then available in the CHIP API object. It provides the method `chip.url.toAbsoluteUrl()` to convert a relative URL into an absolute URL.

Example: Code Example

```
var oChipApi = this.getComponentData().chip;
var sImageUrl = oChipApi.url.toAbsoluteUrl("sap/ui2/chips/sample/links/
loading.gif");
```

Related Information

http://help.sap.com/javadocs/ui-services/services/symbols/chip.url.html

1.5.2.1.2.1.4.8 Accessing the "configuration" Contract

As a prerequisite, you have to consume the `configuration` contract and declare names for all configuration parameters in the CHIP definition XML. Optionally, you can also define default values for the configuration parameters.

The `configuration` property is then available in the CHIP API object. It provides the method `chip.configuration.getParameterValueAsString()`, which allows you to read parameters that have been declared in the CHIP definition XML.

Example: Code Example

```
var oChipApi = this.getComponentData().chip;
var sConfigurationValue =
oChipApi.configuration.getParameterValueAsString("parameterName");
```

Example: Example for Configuration

```
<?xml version="1.0" encoding="UTF-8"?>
<chip xmlns="http://schemas.sap.com/sapui2/services/Chip/1">
  <implementation>
    <sapui5>
      <!-- ... -->
    </sapui5>
  </implementation>
  <appearance>
    <title>Link List Sample Chip</title>
  </appearance>
  <contracts>
    <consume id="configuration">
      <parameters>
        <parameter name="columns">3</parameter>
        <parameter name="id"></parameter>
        <parameter name="showChildren">true</parameter>
      </parameters>
    </consume>
    <consume id="fullscreen" />
    <consume id="navigation" />
    <consume id="refresh" />
    <consume id="url" />
  </contracts>
</chip>
```

Related Information

http://help.sap.com/javadocs/ui-services/services/symbols/chip.configuration.html

1.5.2.1.2.1.4.9 Accessing the "visible" Contract

This contract interface allows the page builder to notify the CHIP when the CHIP's visibility changes.

As a prerequisite, you have to consume the `visible` contract in the CHIP definition XML.

The `visible` property is then available in the CHIP API object. It provides the following methods:

- `chip.visible.attachVisible()`
  You can use this method to attach an event handler to the `visible` event which is fired whenever the CHIP's visibility changes.
- `chip.visible.isVisible()`
  You can use this method to find out whether the CHIP is currently visible.

Example: Code Example

```
init: function () {
  var oChipApi = this.getComponentData().chip;
  oChipApi.visible.attachVisible(function (bVisible) {
    if(bVisible) {
      // Do something expensive which you would like to avoid if the CHIP isn't
shown yet
    }
  });
}
```

Related Information

https://help.sap.com/doc/javadocs_ui-services/1.0/en-US/services/symbols/contract.visible.html

1.5.2.1.2.1.4.10 Accessing the "configurationUi" Contract

This contract allows you to provide a CHIP-specific configuration UI where an administrator can set parameters for the CHIP.

As a prerequisite, you have to consume the `configurationUi` contract in the CHIP definition XML.

The `configurationUi` property is then available in the CHIP API object. It provides the following methods:

- `chip.configurationUi.attachCancel()`
  You can use this method to register an event handler to be notified when the user cancels the configuration.

- `chip.configurationUi.attachSave()`
  You can use use this method to register an event handler to be notified when the user triggers a save action for the current configuration. The purpose of the event handler is to persist the configuration asynchronously (the configuration is not persisted by the embedding application, this is the responsibility of the CHIP). The event handler must return a `jQuery.Deferred` object's promise to inform the embedding application whether the save operation has been successful. If it has not been successful, an error message is provided.

- `chip.configurationUi.display()`
  You can use this method to inform the embedding application that the user has triggered configuration of this CHIP.

- `chip.configurationUi.isEnabled()`
  You can use this method to find out whether configuration of CHIPs is enabled.

- `chip.configurationUi.setDirtyProvider()`
  You can use this method to to define the callback function which provides the configuration UI's "dirty" state for this CHIP. The callback function has to return a boolean value telling whether this CHIP's configuration UI is currently in a "dirty" state, that is, whether it contains unsaved changes.

- `chip.configurationUi.setUiProvider()`
  You can use this method to define the callback function which provides the configuration UI for this CHIP. Since version 1.21.0, the function might get a map of parameters that can be used to set some default configuration values, for example.

Example: Code Example: Accessing the "configurationUi" Contract

```
init: function() {
  varoChipApi = this.getComponentData().chip;
  if(oChipApi.configurationUi.isEnabled()) {
    oChipApi.configurationUi.setUiProvider(function(mParameters) {
      // get configuration UI as sap.ui.core.Control
      var oConfigurationUi = ...;
      // make sure cancel/save handlers are attached now
      oChipApi.configurationUi.attachCancel(function() {
      });
      oChipApi.configurationUi.attachSave(function() {
        var oDeferred = newjQuery.Deferred();
        // either reject or resolve, maybe later
        return oDeferred.promise();
      });
      oChipApi.configurationUi.setDirtyProvider(function() {
        return false; // tell about unsaved changes here!
      });
      return oConfigurationUi;
    });
    // enable some trigger to start configuration:
    // oTrigger.attachPress(function () {
    //   oChipApi.configurationUi.display();
    // });     // Note: do NOT use to display live data etc.!
  }
}
```

In this example, the CHIP instance checks whether configuration is enabled in the component's method. If configuration is enabled, it sets a callback function which will later provide the configuration UI. That callback is the last point in time to make sure that cancel/save handlers are attached and a "dirty" provider is set - of course this can also be done earlier. In addition, the CHIP enables a trigger (for example, click on the CHIP) for the user to request the configuration UI to be displayed. This trigger should call `oChipApi.configurationUi.display()` when invoked.

Example: Code Example: Configuration UI

```
<?xmlversion="1.0"encoding="UTF-8"?>
<chip xmlns="http://schemas.sap.com/sapui2/services/Chip/1">
  <implementation>
    <sapui5>
      <!-- ... -->
    </sapui5>
  </implementation>
  <appearance>
    <title>Configuration UI</title>
  </appearance>
  <contracts>
    <consumeid="configurationUi"/>
  </contracts>
</chip>
```

Related Information

http://help.sap.com/javadocs/ui-services/services/symbols/chip.configurationUi.html

1.5.2.1.2.1.4.11 Accessing the "preview" Contract

This contract allows you to display a preview of a CHIP. This can be used to display a list of available CHIPs in a catalog, for example.

The preview does not display any live data, but it contains a link to the same target application as the corresponding live CHIP.

As a prerequisite, you have to consume the `preview` contract in the CHIP definition XML.

The `preview` property is then available in the CHIP API object. It provides the following methods:

- `chip.preview.getTitle()`
  You can use this method to get the CHIP title. The CHIP may display this in the UI that it shows in preview mode.
- `chip.preview.getDescription()`
  You can use this method to get the CHIP description. The CHIP may display this in the UI that it shows in preview mode.
- `chip.preview.isEnabled()`

You can use this method to find out whether the preview mode of CHIPs is enabled.

- `chip.preview.setPreviewIcon()`
  This method defines the preview icon to be used for this CHIP.

- `chip.preview.setPreviewTitle()`
  This method defines the preview title to be used for this CHIP. It is necessary to explicitly set a title, otherwise no title will be displayed in the preview.
  We recommend to use the same title for the preview as for the live CHIP. To achieve this, set the preview title to the return value of the `oChipApi.preview.getTitle()` method.
  If you want to define a separate title for the preview, you should additionally use the bag contract to store the title in a translatable bag property.

- `chip.preview.setTargetUrl()`
  You can use this method to define the URL of the underlying application for this CHIP.

Example: Code Example

```
init: function() {
  var oChipApi = this.getComponentData().chip;
  if(oChipApi.preview.isEnabled()) {
    // Determine target URL e.g. from CHIP configuration
    var sUrl = ... // typically #<SemanticObject>-<Action>?
Parameter1=Value1&Parameter2=Value2, e.g. #DaysSalesOutstanding-Display?
kpiId=KPI&variantId=VARIANT
    oChipApi.preview.setTargetUrl(sUrl);
    // Note: do NOT use to display live data etc.!
    //   You may use oChipApi.preview.getTitle() and
oChipApi.preview.getDescription() to display on the UI

oChipApi.preview.setPreviewIcon(oChipApi.configuration.getParameterValueAsString(
"icon")); //sample: set preview icon from CHIP configuration
    oChipApi.preview.setPreviewTitle(oChipApi.preview.getTitle());
  }
}
```

In this example, the CHIP instance checks whether preview mode is enabled in the component's init method. If preview mode is enabled, it computes and sets the target URL in the `preview` contract. In addition, it ensures that rendering is done in preview mode. For example, it makes sure not to use any live data and uses title and description accessed using the preview contract for its UI. If the embedding application does not render the CHIP in preview mode, it provides an icon and a title to be displayed by the embedding application for preview purposes.

Related Information

http://help.sap.com/javadocs/ui-services/services/symbols/chip.preview.html

1.5.2.1.2.1.5 API Reference

You can find the JSDocs for CHIP development at the following locations:

Namespace chip.bag

Namespace chip.configuration

Namespace chip.configurationUi

Namespace chip.fullscreen

Namespace chip.navigation

Namespace chip.preview

Namespace chip.refresh

Namespace chip.search

Namespace chip.searchProvider

Namespace chip.url

Namespace chip.visible

Namespace chip.writeConfiguration

1.5.2.1.2.1.6 Registering a CSR CHIP in a Catalog

You define and store a CSR CHIP in the back-end system. Then you register the CHIP in the Web Dynpro configuration so that it can be used in the page building service.

Context

Procedure

1. Define the CHIP definition XML in an editor.

   Optional: Specify the parameters for a CHIP, for example `/UI2/Tag` and `/UI2/BusinessObject`. The tag **Parameters** is optional.
2. Store the CHIP in the back-end system. The CHIP definition XML is stored in the MIME repository of the Object Navigator (transaction `SE80`).
3. To register the CHIP in the CHIP catalog, go to transaction `/UI2/CHIP`.
4. Enter the name for the CHIP (for example, `MY_CHIP`) and choose *Create*.
5. Enter the required data for CHIP attributes:

a. `Field URL`: Specify the path of the CHIP definition XML in the MIME repository, for example /sap/public/bc/ui2/test/test_chip_1.xml.

b. `Field Display Name`: Specify the name under which the CHIP is displayed in the CHIP catalog.

c. `Field Description`: Specify a short text for the CHIP.

d. `Field Configuration String`: This is a string container to be interpreted by UI Services. The field is optional.

6. Save your changes.

7. The path of the CHIP definition XML is checked. If the local HTTP request reading the file fails, an error message is displayed. You will be asked to specify a transport request for the CHIP registration data. During the process, a BAdI is called that stores the CHIP in the catalog cache. Check that the *Exists in Catalog Cache* checkbox is selected after saving.

Next Steps

For more information, see CHIP Definition XML [page 580] and in the SAPUI5 Developer Guide.

1.5.2.1.2.1.7 Changing a CSR CHIP

Use

After creating and registering a client-side rendering CHIP (CSR CHIP), you might need to do some changes.

Changing the CHIP XML

If you edit the CHIP XML (for example, to change the catalog filter parameters), you must make sure that the catalog cache is updated. In order to update the catalog cache, you have the following possibilities:

- Open the CHIP in transaction `/UI2/CHIP` and save it.
- Run the program `/UI2/CHIP_SYNCHRONIZE_CACHE` in transaction `SE38`.

Changing the CHIP Configuration

You can change the CHIP configuration in transaction `/UI2/CHIP`.

For more information, see Registering a CSR CHIP in a Catalog [page 594].

1.5.2.1.2.2 Assigning a CHIP to the Universal Fiori Launchpad Catalog

You can enhance the standard tile template list of the launchpad designer by adding a new tile template to the Universal Fiori Launchpad Catalog in the back end.

Prerequisites

- The desired custom tile type has been implemented.
  See Implementing a Custom Tile Type [page 578].
- You have registered the corresponding CHIP in the universal catalog with transaction /UI2/CHIP.
  See Registering a CSR CHIP in a Catalog [page 594].

Context

Before you can configure custom tile types, the new tile types have to be exposed in the SAP Fiori launchpad designer.

For this, the catalog /UI2/FLPD_CATALOG must be enhanced. The catalog is stored as a Web Dynpro configuration of the Web Dynpro component /UI2/CHIP_CATALOG_CONFIG called /UI2/FLPD_CATALOG.

Proceed as follows to enhance this catalog:

Procedure

1. Start *Repository Info System* (transaction SE84) in the back-end ABAP system.
2. Open the node ▶ *Repository Information System* ❯ *Web Dynpro* ❯ *Web Dynpro Applicat.* ❯.
3. Enter CUSTOMIZE_COMPONENT in the *Web Dynpro Application* field and choose *Execute*. The CUSTOMIZE_COMPONENT application application is listed, double click it to open. Choose *Test/Execute*.
4. In the browser window, enter /UI2/CHIP_CATALOG_CONFIG as *Component Name* and /UI2/FLPD_CATALOG as the *Configuration ID*.

   CUSTOMIZE_COMPONENT changes the catalog on the Customizing (client-specific) layer. Do not change the configuration layer as this is the UI add-on standard delivered version.
5. Choose *Create* to enhance an existing configuration for the first time. Afterwards, continue in change mode.
6. In the *Create Customizing* dialog, enter a description and choose *OK*. Select a transport request of type *Customizing Request* or create a new transport request with this type.
7. In the *Configuration Context* with the node tree of the configuration, select *parameterMultiVal (CHIP_NAME)* and select ▶ *New* ❯ *values* ❯. Enter the value of your CHIP.

Make sure that you start the name of the CHIP with the prefix `X-SAP-UI2-CHIP`. For example, if you registered a CHIP `MYCHIP` in transaction `/UI2/CHIP`, the value of the node tree must be `X-SAP-UI2-CHIP:MYCHIP`.

8. Save your entries.

Related Information

1.5.2.1.2.3 Enabling Tile Size Change for Custom Tiles

To ensure that custom tiles are displayed correctly in the available sizes, you need to enable the tile size changes.

There are two types of tile size changes that we need to distinguish:

- Tile formats like standard tile, link, flat wide and flat. This is mostly relevant when you work with spaces and pages. Administrators can select the different sizes when creating or editing a page. Users that are allowed to personalize can select the different sizes when editing a page. Also see Tiles, Links and Tile Sizes [page 423]
- Tile size changes between small and responsive. If they are allowed to, users can set small size for all tiles. Also see Managing Your Settings [page 509].

> i Note
>
> We recommend that you use the `sap.m.GenericTile` to maintain compatilibity with updates in the future.
>
> See class sap.mGenericTile for additional information.

Support Different Tile Sizes

In order to enable that users can change the custom tile size, the following code has to be implemented. This is part of step 1 in Implementing a Custom Tile Type [page 578]. To enable the display format change, you can add the following code to your custom tile component.

> ⚞ Sample Code
>
> ```
> sap.ui.define([
>     "sap/ui/core/mvc/Controller"
> ], function (Controller) {
>     "use strict";
>     return Controller.extend("custom.controller.Tile", {
>         onInit: function () {
> ```

```
        var oChipApi = this.getView().getViewData().chip;
        oChipApi.types.attachSetType(this._setDisplayFormat);
        },
        _ setDisplayFormat: function (sDisplayFormat) {
            // Set the property frameType on the sap.m.GenericTile control
used in the custom tile implementation according to the display format
        }
    });
});
```

You also have to declare the available types in the tile's CHIP.xml file in the types contract as follows. This is part of step 2 in Implementing a Custom Tile Type [page 578].

⇆ Sample Code

```
        <consume id="types">
            <parameters>
                <parameter name="supportedTypes">tile,link,flat,flatWide</
parameter>
                <parameter name="defaultType">tile</parameter>
            </parameters>
        </consume>
```

This example supports all tile sizes available with spaces and pages and indicates that "tile" is the preferred tile size for this tile.

Guidelines for the tile sizes:

- `tile` must always be supported.
- `link` controls if the custom tile can be converted to a link on the classic home page. On a page, a tile can always be displayed as a link if it has a title and a target that is known to the SAP Fiori launchpad.
- `flat` and `flatWide` are optional and are only supported on pages.
- The default type controls which display format is displayed as 'preferred' in the Manage Launchpad Pages app.

Support Small Tile Size

To enable the size change between small and responsive, you can add the following code to your custom tile component.

⇆ Sample Code

```
sap.ui.define([
    "sap/ui/core/mvc/Controller"
], function (Controller) {
    "use strict";
    return Controller.extend("custom.controller.Tile", {
        onInit: function () {
            sap.ushell.Container.getServiceAsync("Configuration")
                .then(function (configurationService) {
                    // Keep the eventRegistry to be able to detach the event
later
                    this.oEventRegistry =
configurationService.attachSizeBehaviorUpdate(this._onSizeBehaviorUpdate.bind(
this));
                }).bind(this);
```

```
        },
        _onSizeBehaviorUpdate: function (updatedSizeBehavior) {
            // Set the property on the sap.m.GenericTile control used in the
custom tile implementation
        },
        onExit: function () {
            // Always detach the event to avoid memory leaks
            this.oEventRegistry.detach();
        }
    });
});
```

Afterwards, you can decide if users are allowed to change the tile size themselves. This is done by enabling the launchpad configuration parameter HOMEPAGE_TILES_SIZE_ENABLE_USER. See Launchpad Configuration Parameters [page 29] for further information.

Related Information

class sap.ushell.services.Configuration🔗
class sap.m.GenericTile🔗

1.5.2.1.3 Exposing Custom Tile Type Information to Launchpad Design-Time Tools

To pass information on your custom tile type to launchpad design-time tools, you need to implement Business Add-In (BAdI) /UI2/BADI_FDM_CUST_TILE.

Prerequisites

- You have implemented your own custom tile type. See Implementing a Custom Tile Type [page 578].
- You have created a tile using your custom tile type. See Creating or Deleting Tiles or Target Mappings [page 203].

Context

The BAdI implementation is required for the following:

- Display custom tile information (such as tile title, semantic object, etc.) in different launchpad design-time tools (e.g. launchpad content manager, launchpad app manager).
- Add references to custom tiles in the launchpad content manager. See Adding Reference Tiles and Target Mappings to a Catalog [page 250].

Procedure

1. Create an enhancement implementation for the BAdI `/UI2/BADI_FDM_PUB_CUST_TILE`.

   For more information, see How to Implement a BAdI.

2. Create an implementing class.

   > **⁘ Example**
   >
   > You can use the example implementation `/UI2/CL_FDM_PUB_BADI_STD_TILE` as a template.

3. Implement method `GET_CUSTOM_TILE_DETAILS` of BAdI interface `/UI2/IF_BADI_FDM_PUB_CUST_TILE`.

   This method gets general custom tile details (such as tile title, semantic object, etc.) for display in various design-time tools (e.g. launchpad content manager, launchpad app manager).

   See Parameters of Interface /UI2/IF_BADI_FDM_PUB_CUST_TILE [page 601].

4. In the BAdI implementation, specify the ID of your custom tile type (i.e. the CHIP ID) as value of the filter `TILE_TYPE` (for example, **MYCHIP**).

   The filter allows switching between your own BAdI implementation and the BAdI implementations created by SAP for SAP-delivered custom tile types (e.g. SAP Smart Business tiles).

   For more information, see How to Use Filters .

Related Information

Setting Up Business Catalogs with the Launchpad Content Manager [page 238]

1.5.2.1.3.1 Parameters of Interface /UI2/ IF_BADI_FDM_PUB_CUST_TILE

Method GET_CUSTOM_TILE_DETAILS

Method `GET_CUSTOM_TILE_DETAILS` of interface `/UI2/IF_BADI_FDM_PUB_CUST_TILE` has the following parameters:

Importing Parameters

Parameter	Description
IS_TILE_KEY	Key that is generated when you use your custom tile type to create a custom tile (e.g. in the launchpad designer)
IV_TILE_TYPE	ID of your custom tile type (i.e. the CHIP ID)
IV_TILE_CONFIGURATION	Tile configuration as defined in the configuration contract of the custom tile type implementation. See Accessing the CHIP API and the Contracts [page 583].
IT_TILE_BAG_PROPERTY	Translatable texts of the custom tile as defined in the bag contract of the custom tile type implementation. See Accessing the CHIP API and the Contracts [page 583].

Changing Parameters

Parameter	Description
CS_TILE_DISPLAY	Display information of the tile (e.g. icon or title text).
CS_TILE_NAVIGATION	Determine possible navigation targets of the tile
CT_MESSAGES	Report if there were issues processing the custom tile information

1.5.2.2 Developing Plug-Ins

Plug-ins allow you to extend the functionality of the launchpad; for example, to add new elements to the user interface.

Plug-ins have the following characteristics:

- Plug-ins are automatically loaded and initialized when the launchpad is started.
- Plug-ins are implemented as SAPUI5 components and provide all standard deployment and life cycle features of SAPUI5.

Developing a plug-in involves the following steps:

1. Implementing the plug-in. This is described in Implementing a Plug-In [page 604]. The section also contains an example showing how to create a plug-in that adds buttons to the shell header bar.

2. Consider bundling your implementation for better performance. Also see UI5 Tooling ↗ for detailed information.

3. Deploy your implementation to the front-end server. Also see The SAPUI5 ABAP Repository and the ABAP Back-End Infrastructure - Documentation - Demo Kit SAPUI5 SDK (ondemand.com) for detailed information on the different setups.

4. Activating and configuring the plug-in. Plug-ins need to be activated. When activating a plug-in, you can specify values for configuration parameters to be passed to the plug-in. How to do this, is described in Activating Plug-Ins on the ABAP Platform [page 606].

> → Tip
>
> Plug-in Development Guidelines [page 602] shows do's and don'ts for development.

1.5.2.2.1 Plug-in Development Guidelines

Guidelines and best practices for developing SAP Fiori launchpad plug-ins.

Provide an application descriptor and set the FLP type "plugin"

SAP Fiori applications must specify metadata in the application descriptor (`manifest.json`). See Implementing a Plug-In [page 604] for detailed information. For more information about the application descriptor, see .

Implement platform-independent plug-ins

Plug-ins shall always be implemented in a platform-independent way. Platform-specific configuration is allowed.

Make sure that your plug-ins are loaded with a minimal performance footprint

Plug-ins are loaded during the start-up of the SAP Fiori launchpad. This is a critical point in time for performance, as the SAPUI5 framework and the apps also need loading time and browser processing time. It is therefore important to load plug-ins with minimal performance footprint.

Common use cases for plug-ins are extending the SAP Fiori launchpad with additional actions in the launchpad header. UI extension plug-ins must only execute the code to add this button, but not more. Any other processing should only be done at the point in time when the user clicks the button.

Provide a Component preload file

To ensure that a Component-based SAPUI5 project (app or plug-in) will be loaded with the smallest number of roundtrips possible, package it into a single `Component-preload` file. This ensures that all files are loaded with the initial request and no additional loading will be triggered at runtime. See Component Preload Generation ↗ for details.

Don't make any assumptions on the order or the timing of loading

There is no guarantee from the launchpad at which point in time a plug-in is initialized. The implementation must be fail-safe and handle situations when required resources might not be ready yet, or certain processing has already been executed. The loading time might be subject to change.

Make sure that your plug-in is robust if more than one instance is loaded

Plug-ins which have the same implementation (SAPUI5 Component) might be configured several times with different parameters. In this case, the launchpad creates several instances of the SAPUI5 component that implements the plug-in. Plug-in implementations must take care of this. Don't assume that a only a single instance of the plug-in component is created. Even if your plug-in implementation is a singleton by nature, multiple instances might be created due to configuration, and your plug-in must be able to handle this in a robust way.

Keep the number of plug-ins in one launchpad to a minimum

If you implement several smaller scenarios, like adding a button to the launchpad header and adding an entry to the user actions menu, aggregate them into one plug-in rather than creating multiple plug-ins. This reduces the initialization during the start-up of the SAP Fiori launchpad. To control different scenarios of this plug-in, plug-in configurations can be used.

Declare your plug-in's dependencies

To ensure future compatibility, for example with regard to Asynchronous Module Definition (AMD) compliance and future UI5 versions, every plug-in must declare all its dependencies towards UI controls, helpers, etc. using the `sap.ui.define` syntax.

For more information, see .

1.5.2.2.2   Implementing a Plug-In

One of the most common use cases for plug-ins is to add user interface elements to the SAP Fiori launchpad. This example shows how to implement such a plug-in.

Plug-ins have to be implemented as standard SAPUI5 components, similar to SAPUI5 applications hosted inside the SAP Fiori launchpad. All guidelines for SAPUI5 components are also applicable to plug-in implementations. Plug-ins shall also provide a descriptor (`manifest.json`) file. For more information, see .

The SAP Fiori launchpad loads and instantiates the plug-ins automatically during start-up. Plug-ins are typically plain `Component` instances and do not inherit from `UIComponent`. Plug-ins can contribute UI controls to the SAP Fiori launchpad using the APIs of the shell renderer.

Plug-ins are loaded and instantiated asynchronously. When multiple plug-ins are loaded, do not assume any specific instantiation order of the components. Do not rely on the point in time when the plug-in is loaded. The current behavior might change in the future.

In your plug-in, you would typically use the following APIs:

- SAP Fiori launchpad UI services, accessible via sap.ushell.Container.getServiceAsync(…)
- Renderer APIs

You can find an example how to use this API in the SAPUI5 demo kit at https://sapui5.hana.ondemand.com/test-resources/sap/ushell/demoplugins/SimpleUIExtensionPluginSample/Component.js.

In addition to the standard settings for SAPUI5 apps, you need to do the following settings for SAP Fiori launchpad plug-ins:

```
"sap.app": {
  "type": "component"
},
...
"sap.flp": {
  "type" : "plugin"
}
```

For more information about the application descriptor, see .

The UI extension API is specific to the "fiori2" shell renderer implementation, which is the standard renderer for SAP Fiori launchpad.

The number of icons that you can display in the shell bar is restricted. For more information, see Adding Custom Icons to the Shell Bar [page 615].

Example: Implement a Plug-In

The following example shows how to create a plug-in that adds buttons to the shell header bar. This is realized with a single, re-usable SAPUI5 component that can be dynamically configured. The plug-in configuration can be obtained from the component data "config" object (using `this.getComponentData().config`).

> ⇶ Sample Code

```
sap.ui.define([
    "sap/ui/core/Component",
```

```
    "sap/base/Log",
    "sap/ui/core/IconPool",
    "sap/m/MessageToast"
], function (Component, Log, IconPool, MessageToast) {
    "use strict";
    var sComponentName =
"sap.ushell.demoplugins.SimpleUIExtensionPluginSample";
    return Component.extend(sComponentName + ".Component", {
        metadata: {
            manifest: "json"
        },
        init: function () {
            var oRenderer = sap.ushell.Container.getRenderer(),
                oPluginParameters = this.getComponentData().config,
                oShellHeaderItemProperties = {
                    tooltip: oPluginParameters.tooltip || "",
                    ariaLabel: oPluginParameters.tooltip || "",
                    icon: IconPool.getIconURI(oPluginParameters.icon ||
"question-mark"),
                    press: function () {
                        MessageToast.show(oPluginParameters.message ||
"Default Toast Message");
                    }
                };
            if (oPluginParameters.position === "end") {
                oRenderer.addHeaderEndItem(
                    oShellHeaderItemProperties,
                    true,
                    false // visible in all states
                );
            } else if (oPluginParameters.position === "begin") {
                oRenderer.addHeaderItem(
                    oShellHeaderItemProperties,
                    true,
                    false // visible in all states
                );
            } else {
                Log.warning("Invalid 'position' parameter, must be one of
<begin, end>. Defaulting to 'end'.", undefined, sComponentName);
                oRenderer.addHeaderEndItem(
                    oShellHeaderItemProperties,
                    true,
                    false // visible in all states
                );
            }
        }
    });
});
```

Note that the component that acts as shell plug-in is a plain component, not a UIComponent. It shall only have minimal dependencies and load very fast. The main functionality of the plug-in shall rather be loaded lazily, i.e. when the button is pressed. Depending on the use case, the press handler could then instantiate a separate UIComponent which implements the plug-in functionality. The activation and configuration of the plug-in depends on the corresponding target platform. See for example Activating Plug-Ins on the ABAP Platform [page 606].

The next steps are bundling, deploying and activating the plug-in (see Developing Plug-Ins [page 601] for an overview).

Related Information

1.5.2.2.3 Activating Plug-Ins on the ABAP Platform

On the ABAP platform, you can activate plug-ins in Customizing or by creating a target mapping.

Context

If a plug-in should be active for all users of a client, it can be configured and activated in Customizing. If you want to activate specific plug-ins only for specific user groups, create a target mapping.

Activating Plug-Ins in Customizing

Procedure

1. Open transaction `/UI2/FLP_CUS_CONF`.
2. In the section *Define Launchpad Plugins*, create a new entry for your plug-in.
3. In *SAP Reference IMG* (transaction SPRO), choose ▶ *SAP NetWeaver* ❭ *UI Technologies* ❭ *SAP Fiori* ❭ *SAP Fiori Launchpad Settings* ❭ *Display Cross-Client Settings* ❭.
4. In the section *Launchpad Plugins*, create a new entry for your plug-in, and set the activity state to *Active*.
5. Optional: If you want to define your plug-in configuration and activation cross-client, you can also use transaction `/UI2/FLP_CONF_DEF` and follow steps 2 to 4.

Related Information

Configuring and Activating Plug-Ins in a Target Mapping

Procedure

1. In the launchpad app manager (see e.g. Maintaining Launchpad App Descriptor Items [page 148]), create a target mapping with the intent `Shell-plugin` and the following configuration:

Field	Value
Semantic Object	**Shell**
Action	**plugin**
Application Type	**SAP UI5 Fiori App**
Title	Enter a meaningful title for your plug-in.
URL	Enter the the root path to the location where the `Component.js` file is located, without the file name itself.
ID	Enter the app descriptor or the component ID of the SAPUI5 component which is used for loading and instantiation. For more information on app descriptors, see Maintaining Launchpad App Descriptor Items [page 148]. The component ID is the name of the component's module without *.Component*. For more details on this, see .
Information	Optional: Enter a description of your plug-in.
Device Types	Select the device or devices that are supported by your plug-in, e.g. desktop or phone. The plug-in will only be shown on these selected devices.
Parameters	Define parameters that you want to pass to the plug-in component by entering a name and a default value for each parameter.

> i Note
>
> As target mappings with the intent `Shell-plugin` have a special meaning, they do not appear in standard services such as the `TargetMappings` feed, `resolveLink`, or `getLinksForSemanticObjects`.

2. In the launchpad content manager, assign the catalog to a role. See Adding and Removing Catalogs, Groups and Spaces to/from Roles [page 256] for detailed information.

3. If you have not already done this, assign the role to the respective users (see Assigning Users to Roles [page 339]).

1.5.2.3 Extending the Launchpad Shell

Read the following sections to find out, which components of the launchpad you can extend.

1.5.2.3.1 Adding and Removing Custom Items to or from the User Actions Menu

Use the methods of the `Renderer` API to add items to or remove added items from the user actions menu.

Syntax

`addUserAction(oParameters)`

`showActionButton(aIds, bCurrentState, aStates, bIsFirst)`

`hideActionButton(aIds, bCurrentState, aStates)`

These methods are part of SAPUI5 Demo Kit (▶ *API REFERENCE* ❭ *sap.ushell.renderers.fiori2* ❭ *Renderer* ❭).

> **i** Note
>
> The `addUserAction` method replaces the `addActionButton` method. `addUserAction` returns an asynchronous response, while `addActionButton` returns a synchronous response.

> **i** Note
>
> You can only remove entries that were added to user actions menu by the `addActionButton` method.

Parameters

addUserAction

Parameter	Type	Description
oParameters	Object	Contains the required fields for creating and showing the new control object:

- String `controlType`: The class name of the control type to create. Currently, the only supported type is `sap.m.Button`.
- Object `oControlProperties`: The properties that will be passed to the created control. For `sap.m.Button`, the properties should include at least "text" (i.e., title) and "press" (i.e., action press handler).
- Boolean `bIsVisible`: When `true`, displays the control after the item is created. If `false`, you can later use the `showActionButton` method to display the control.
- Boolean `bCurrentState`: When `true`, adds the control to the current rendered shell state only. When the user navigates to a different state, or to a different application, then the control is removed.
- String[] `aStates`: Specifies an array of launchpad states (`sap.ushell.renderers.fiori2.Renderer.LaunchpadState`) in which to add the control. Valid only if `bCurrentState` is set to `false`. If no launchpad state is passed, the button is added in all states.
- Boolean `bIsFirst`: When set to `true`, displays the item as the first item in the user actions menu after the app finder, which is always displayed first.

Optionally you can add an `icon` for the menu entry to be displayed. We recommend this for a consistent user experience.

showActionButton / hideActionButton

Parameter	Type	Description
aIds	String	Array of IDs of the button controls that should be shown or hidden.
bCurrentState	Boolean	If `true`, the created control is added to the current rendered shell state. When the user navigates to a different state, or to a different application, then the control is removed.
		If `false`, the control is added to the LaunchpadState.

Parameter	Type	Description
`aStates`	String	A list of the launchpad states (`sap.ushell.renderers.fiori2.Renderer.LaunchpadState`) in which to add the control. Valid only if `bCurrentState` in `addActionButton(controlType, oControlProperties, bIsVisible, bCurrentState, aStates)` is set to `false`. If no launchpad state is passed, the item is added in all states.
`bIsFirst`	Boolean	i Note Due to the restructuring of the menu, this parameter is obsolete and will be removed with the next version.

Example

```
<SCRIPT language ="JavaScript">
var oRenderer = sap.ushell.Container.getRenderer("fiori2"),
    oAddActionButtonProperties = {
        controlType : "sap.m.Button",
        oControlProperties : {
            text : "NewAction",
          icon: "sap-icon://refresh",
            press : function () {
                alert("ActionPerformed!")
            }
        },
        bIsVisible : true,
        bCurrentState : true,  // => added to the current state and removed when
navigation away from it
        bIsFirst : true
    };
oRenderer.addUserAction(oAddActionButtonProperties);
</SCRIPT>
```

Item Order

Menu entries are sorted in the following order if all of them are displayed:

- Recent Activities
- Frequently Used
- App Finder
- Settings
- Edit Mode
- Contact Support

- Your own actions
- Sign Out

1.5.2.3.2 Adding Items to the User Settings Dialog Box

Use the methods of the `Renderer` API to add items to the users' *Settings* dialog box.

Syntax

`addUserPreferencesEntry(entryObject);`

This method is part of the SAPUI5 `sap.ushell.renderers.fiori2` namespace. For more information, go to SAPUI5 Demo Kit and then navigate to ▶ *API REFERENCE* ❯ *sap.ushell.renderers.fiori2* ❯ *Renderer* ▶.

Properties of the `entryObject` Object

Property	Type	Description
`entryHelpID`	String	(Optional) The ID of the object.
`title`	String	The title of the entry to be presented in the list in the *Settings* dialog box. We recommend using the string from the translation bundle.
`value`	{String}/{Function}	A string to be presented as the value of the entry. This parameter can be either a string or a function that supplies a string via the returned `jQuery.Deferred.promise` object.

Property	Type	Description
onSave	Function	A function that implements entry-specific save functionality, triggered when the user clicks *Save* in the *Settings* dialog box. The function returns a `jQuery.Deferred.promise` object.
		If an error occurs, pass the error message via the `jQuery.Deferred.promise` object. Errors are displayed in the log.
onCancel	Function	A function to call to close the *Settings* dialog box without saving any changes.
content	Function	A function to call to retrieve a `jQuery.Deferred.promise` object that consists of a `sap.ui.core` control to be displayed in a follow-on dialog box. An SAPUI5 view instance can also be returned.
		The function is called each time the user opens the *Settings* dialog box.

Example

⟨≡⟩ Sample Code

```
<SCRIPT language ="JavaScript">
var oRenderer = sap.ushell.Container.getRenderer("fiori2");
var oEntry = {
  title: "title",
  value: function() {
      return jQuery.Deferred().resolve("entryTitleToBeDisplayed");
  },
  content: function() {
      var deferred = jQuery.Deferred();
      deferred.resolve(new sap.m.Button("userPrefEntryButton", {text:
"Button"}));
      return jQuery.Deferred().resolve(new
sap.m.Button("userPrefEntryButton", {text: "Button"}));
  },
  onSave: function() {
      return jQuery.Deferred().resolve();
  };
oRenderer.addUserPreferencesEntry(oEntry);

</SCRIPT>
```

Additional Information

- The `content` function, which is provided as part of the entry configuration object, is called each time the user opens the *Settings* dialog box.
- The SAPUI5 view or SAPUI5 control that returns as the result of the `content` function is destroyed when the user closes the *Settings* dialog box (by clicking on *Cancel* or *Save*). This is triggered by the generic *Settings* dialog box and is not the responsibility of the plug-in developer.

1.5.2.3.3 Displaying and Hiding the Shell Bar

Use the methods of the `Renderer` API to specify when the shell bar is displayed or hidden.

Syntax

`setHeaderVisibility(bVisible, bCurrentState, aStates)`

These methods are part of the SAPUI5 `sap.ushell.renderers.fiori2` namespace. For more information, go to SAPUI5 Demo Kit and then navigate to ▶ *API REFERENCE* ❯ *sap.ushell.renderers.fiori2* ❯ *Renderer* ❯.

Parameters

Parameter	Type	Description
bVisible	Boolean	The visibility of the shell bar.
bCurrentState	Boolean	If `true`, the visibility is set according to the current rendered shell state. When the user navigates away from the current app (or home page), the visibility is reset.
		If `false`, visibility is determined according to the values of the `aState` parameter.
aStates	String	Valid only if `bCurrentState` is set to `false`
		A list of the launchpad states in which the shell bar visibility flag is set.

To determine when the shell bar is displayed, the launchpad state parameters can have the following values of the `sap.ushell.renderers.fiori2.Renderer.LaunchpadState` enum:

- `Launchpadstate.App`: The launchpad state when running an SAP Fiori app.
- `Launchpadstate.Home`: The launchpad state when the home page is open.

If no launchpad state is provided, the flag is set for all states.

Example

```
<pre>
var oRenderer = sap.ushell.Container.getRenderer("fiori2");
oRenderer.setHeaderVisibility(false, false, ["home", "app"]);
</pre>
```

1.5.2.3.4　Adding a Title to the Shell Bar

Use the method of the `Renderer` API to add a title in the shell bar.

Syntax

`setHeaderTitle(sTitle)`

This method is part of the SAPUI5 `sap.ushell.renderers.fiori2` namespace. For more information, go to SAPUI5 Demo Kit and then navigate to ▶ *API REFERENCE* ❯ *sap.ushell.renderers.fiori2* ❯ *Renderer* ❯.

> ℹ Note
>
> The search control and the title occupy the same space in the shell bar. Opening the search hides the title.

Parameters

Parameter	Type	Description
Title	String	The text of the title to display in the shell bar.

Example

> 🖥 Sample Code
>
> ```
> <SCRIPT language ="JavaScript">
> var oRenderer = jQuery.sap.getObject("sap.ushell.renderers.fiori2.Renderer");
> oRenderer.setHeaderTitle("Some Title");
>
> </SCRIPT>
> ```

1.5.2.3.5 Adding Custom Icons to the Shell Bar

Use the methods of the `Renderer` API to add add icons to the shell bar.

Syntax

`addHeaderItem(oControlProperties, bIsVisible, bCurrentState, aStates)`

This method is part of the SAPUI5 `sap.ushell.renderers.fiori2` namespace. For more information, go to SAPUI5 Demo Kit, and then navigate to ▶ *API REFERENCE* ❯ *sap.ushell.renderers.fiori2* ❯ *Renderer* ❭.

The method creates and displays a UI control (icon) in the shell bar of the SAP Fiori launchpad. The new header control is of type `sap.ushell.ui.shell.ShellHeadItem`, and is created with the properties passed by the caller.

The UI control is added to the right of any existing items on the **left-hand side** of the shell bar, in accordance with the given display parameters and shell bar states (see: `sap.ushell.renderers.fiori2.Renderer.LaunchpadState`).

The function returns the newly created item of type `sap.ushell.ui.shell.ShellHeadItem`.

Adding icons to the **right-hand side** of the shell bar is done in a similar way using the renderer API function `addHeaderEndItem`.

> ℹ Note
>
> The shell bar can contain a maximum of three items on the left-hand side of the shell bar. The two standard icons on the left-hand side, ❮ (Back) and ⌂ (Home or company logo) are configured by default and take preference over any custom-added icons. Take into account that when an application is displayed and depending on the configured state of the shell bar, all three icons may be displayed which would prevent any custom icons from appearing.
>
> On the right-hand side, the shell bar can contain an additional three items apart from the default standard icons: 🔍 (Search) ☰ (Notifications) and 👤 (User Actions Menu). Custom icons cannot replace the default icons in the shell bar. In addition, you can also move actions from the user actions menu to the right-hand side of the shell bar. These actions in the shell bar take preference over custom icons added using the `addHeaderEndItem` API function. For more information, see Customizing the Shell Bar [page 73].

Parameters

Parameter	Type	Description
oControlProperties	Object	The properties of the new `sap.ushell.ui.shell.ShellHeadItem` control. Mandatory properties: • `icon`: Path to the icon source. Example: `"sap-icon://documents"` • `press`: Handler function that is called on the press event.
bIsVisible	Boolean	Specifies whether the item is displayed after being created. If `true`, the control is displayed according to the `bCurrentState` and `aStates` parameters. If `false`, the control is created, but not displayed. For more information, see the `showHeaderItem` and `showHeaderItem` methods.
bCurrentState	Boolean	If `true`, the created control is added to the current rendered shell state. When the user navigates to a different state, or a different application where the state remains the same, then the control is removed. If `false`, the control is added to the shell states given in the `aStates` parameter.

Parameter	Type	Description
aStates	String	Valid only if **bCurrentState** is **false**.
		An array of shell states (see **LaunchpadState** in the **sap.ushell.renderers.fiori 2.Renderer** documentation) in which the control is added.
		Example: **[oRenderer.LaunchpadState. App, oRenderer.LaunchpadState.H ome]**
		If no launchpad state is passed, the UI control is added in all states.

Example

```
<SCRIPT language ="JavaScript">
var oRenderer = sap.ushell.Container.getRenderer("fiori2");
var oItem = oRenderer.addHeaderItem({
    icon: sap.ui.core.IconPool.getIconURI("menu2"),
    press: function (oEvent) {
        // behavior/functionality of click event
  }
}, true, false, [oRenderer.LaunchpadState.App]);

</SCRIPT>
```

1.5.2.3.6   Adding a Sub-Header

Use the methods of the `Renderer` API to add a sub-header area under the shell bar, which can be used as a message area or for adding additional buttons to the shell bar.

Syntax

addShellSubHeader(oParameters)

This method is part of the SAPUI5 sap.ushell.renderers.fiori2.renderer namespace. For more information, go to SAPUI5 Demo Kit and then navigate to ▶ *API REFERENCE* ❯ *sap.ushell.renderers.fiori2* ❯ *Renderer* ❯.

> **i Note**
>
> The `addShellSubHeader` method replaces the `addSubHeader` method. `addShellSubHeader` returns an asynchronous response, while `addSubHeader` returns a synchronous response.

Parameters

Parameter	Type	Description
`oParameters`	Object	Contains the required fields for creating and showing the new control object: • String `controlType`: The class name of the control type to create. • Object `oControlProperties`: The properties that will be passed to the created control. • Boolean `bIsVisible`: When `true`, displays the sub header control after being created. • Boolean `bCurrentState`: When `true`, adds the current control to the current rendered shell state only. When the user navigates to a different state, or to a different application, then the control is removed. • String[] `aStates`: Specifies the list of launchpad states (`sap.ushell.renderers.fiori2.Renderer.LaunchpadState`) in which to add the control. Valid only if `bCurrentState` is set to `false`. If no launchpad state is passed, the button is added in all states.

Example

> **⇶ Sample Code**
>
> ```javascript
> <SCRIPT language ="JavaScript">
> var oRenderer = sap.ushell.Container.getRenderer("fiori2"),
>     oAddSubHeaderProperties = {
>         controlType : "sap.m.Bar",
>         oControlProperties : {
>             id: "testBar",
>             contentLeft: [new sap.m.Button({
>                 text: "Test SubHeader Button",
>                 press: function () {
>                     sap.m.MessageToast.show("Pressed");
>                 }
>             })]
>         },
>         bIsVisible : true,
>         bCurrentState : true};
>     oPromise = oRenderer.addShellSubHeader(oAddSubHeaderProperties);
> oPromise.done(function (oShellSubHeader) {
>     // performed after subHeader creation
> });
> </SCRIPT>
> ```

1.5.2.3.7 Adding a Tool Area to the Launchpad Shell

You can use the methods of the Renderer API to create a tool area for the launchpad shell, which can e.g. be used to display a customized menu or additional icons.

Syntax

For a tool area there are four methods available.

Syntax	Description
removeToolAreaItem(vIds, bCurrentState, aStates)	Removes the given tool area item from the launchpad in the given launchpad states. This API is meant to be used for implementing custom elements in the launchpad. We do not recommend using it on a standard launchpad element, as this may interfere with the standard launchpad functionality.
addToolAreaItem(oControlProperties, bIsVisible, bCurrentState, aStates)	Creates a tool area item in the launchpad and adds it to the tool area in the given launchpad states. Once the item is added, the tool area is rendered on the left side on the launchpad shell. ⇥ Sample Code ```sap.ushell.Container.getRenderer("fiori2").addToolAreaItem({
id: "testButton",
icon: "sap-icon://documents",
expandable: true,
press: function (evt) {
window.alert("Press");
},
expand: function (evt) {
// This function will be called on the press event of
the "expand" button.
// The result of "expand" event in the UI must be
determined by the developer
window.alert("Expand");
}
}, true, false, ["home"]);```
showToolArea(sLaunchpadState, bVisible)	Sets the tool area visibility.
showToolAreaItem(vIds,	Displays the tool area items on the left side of the launchpad shell in the given launchpad states.

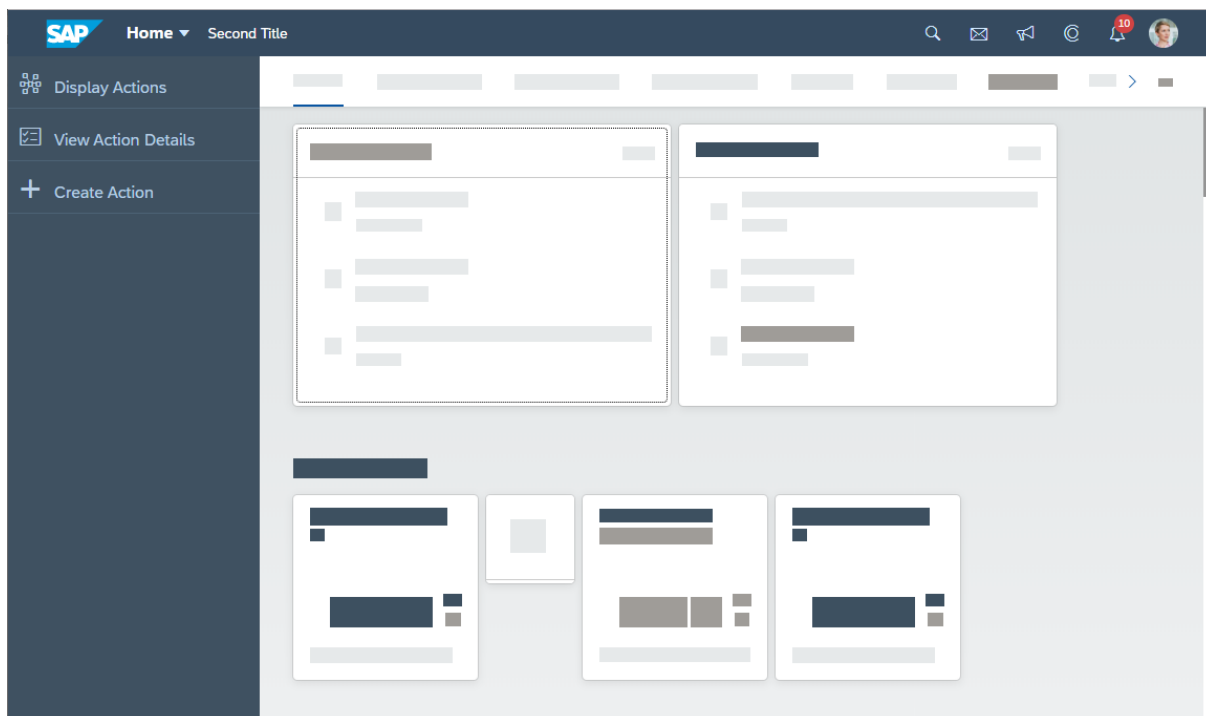| Syntax	Description
bCurrentState, aStates)	⮂ Sample Code ``` var renderer = sap.ushell.Container.getRenderer("fiori2"), button1 = new sap.ushell.ui.shell.ToolAreaItem({ icon: "sap-icon://wrench" }); renderer.showToolAreaItem(button1.getId(), false, ["home", "app"]); ```

These methods are part of the SAPUI5 sap.ushell.renderers.fiori2.renderer namespace. For more information, open the SAPUI5 Demo Kit (or go to the demo kit and then navigate to ▶ *API REFERENCE* ▶ *sap.ushell.renderers.fiori2* ▶ *Renderer* ▶).

Parameters

Parameter	Type	Description
vIds	String	A single ID or an array of IDs.
bCurrentState	Boolean	If true, add the items to the currently rendered shell. If false, add the items to the launchpad state itself, causing the items to be rendered everytime the given states are active.
aStates	String	Only valid if bCurrentState is set to false. An array of shell states (i.e. sap.ushell.renderers.fiori2.Renderer.LaunchpadState) from which the controls are removed. If no launchpad state is provided, the items are removed from all states.
oControlProperties	Object	The properties object that will be passed to the constructor of sap.ushell.ui.shell.ToolAreaItem.
bIsVisible	Boolean	Specify whether to display the tool area item.
sLaunchpadState	String	Specifies the launchpad state in which to show/hide the tool area.
bVisible	Boolean	Specifies whether to display the tool area or not.

The screenshot shows an example for a menu in a tool area on the left side.

1.5.2.3.8 Adding a Footer Bar to the Launchpad Shell

Use the methods of the Renderer API to create and display an SAPUI5 control as the footer bar of the SAP Fiori launchpad shell, which can, for example, be used as a message area or for adding additional buttons to the shell footer.

Syntax

`setShellFooter(controlType, oControlProperties)`

The function creates and adds a footer control. The footer is displayed in all states of the shell bar, including the `headerless` state.

This method is part of the SAPUI5 sap.ushell.renderers.fiori2.renderer namespace. For more information, go to SAPUI5 Demo Kit and then navigate to ▶ *API REFERENCE* ❯ *sap.ushell.renderers.fiori2* ❯ *Renderer* ❯.

> **i Note**
>
> The `setShellFooter` method replaces the `setFooterControl` and `setFooter` methods. `setShellFooter` returns an asynchronous response, while `setFooterControl` and `setFooter` return a synchronous response.

Parameters

Parameter	Type	Description

oParameters	Object	Contains the required fields for creating and showing the new control object:
		
		• String `controlType`: The class name of the control type to create. Example: `sap.m.Bar`
		
		• Object **oControlProperties**: The properties that will be passed to the created control. Example:
		
		``
		{
		  id: "testBar",
		  contentLeft: [new sap.m.Button({ <button data>
		})]
		}
		``

Example

🗎 Sample Code

```JavaScript
<SCRIPT language ="JavaScript">
var oRenderer = sap.ushell.Container.getRenderer("fiori2"),
    oFooterControlProperties = {
        controlType : "sap.m.Bar",
        oControlProperties : {
            id: "testBar",
            contentLeft: [new sap.m.Button({
                text: "Test Footer Button",
                press: function () {
                    sap.m.MessageToast.show("Pressed");
                }
            })]
        }
    };
oPromise = oRenderer.setShellFooter(oFooterControlProperties);
oPromise.done(function (oShellFooter) {
    // performed after footer creation
});
</SCRIPT>
```

1.5.2.4 Adding Tile Actions to All Tiles

Use the method of the `LaunchPage` API to register a callback function for external tile action providers, and add actions to all tiles in the launchpad when the user clicks a tile in action mode.

Syntax

`registerTileActionsProvider(function)`

The function returns an array of tileActions objects.

This method is part of the SAPUI5 `sap.ushell.services.launchpage` namespace. For more information, see LaunchPage (or go to SAPUI5 Demo Kit and then navigate to ▶ *API REFERENCE* ❭ *sap.ushell* ❭ *services* ❭ *launchpage* ❭).

Note that this method is only relevant when your users work with the home page and groups. It is not supported in pages. See Entry Page [page 416] for detailed information.

Properties of the tileActions Object

Properties	Type	Description
`text`	String	The text to display in the actions menu when the user clicks the tile.
`icon`	sap.ui.core.URI	The icon to display next to the text.
`press`	Function	The action to perform when the user clicks the tile action. You can define a targetURL instead.

Properties	Type	Description
targetURL	String	You can enter any URL, for example **http://help.sap.com**. You have to provide a complete URL including the protocol (typically **http://** or **https://**). > i Note > > If you choose to use semantic object navigation, the target URL will be automatically computed and inserted into this field. Note that the target URL is only used when you haven't defined a `press` function. When a `press` function is defined, it is prioritzed.

Example

> ≡, Sample Code

```
this.oPageBuilderService.registerTileActionsProvider(function (oTile){
        return [
                {
                        text: "Some Action",
                        icon: "sap-icon://action",
                        targetURL: "#SemanticObject-Action"
                },
                {
                        text: "Settings",
                        icon: "sap-icon://action-settings",
                        press: function (){
                            //Open settings UI
                        }
                }
        ];
    }
);
Returns the tile actions immediately without any additional server access.
This eliminates
                delays in rendering the action list in the browser.
```

1.5.2.5   Logging User Activities

Use the method of the `logRecentActivity` API to display URL applications and custom tiles in the *Recent Activities* and *Frequently Used* lists in the Quick Access dialog.

Syntax

`logRecentActivity(oRecentEntry)`

This method is part of the SAPUI5 `sap.ushell.renderers.fiori2` namespace. For more information, go to SAPUI5 Demo Kit and then navigate to ▶ *API REFERENCE* ❯ *sap.ushell* ❯ *renderes* ❯ *fiori2* ◀(Renderers for Fiori2).

Properties of the `oRecentEntry` Parameter

Property	Type	Description
`title`	String	The text to display in the User Activities lists
`url`	String	The navigation target URL

Example

🖎 Sample Code

```
<SCRIPT language ="JavaScript">
var oRecentEntry = {},
    oRenderer = sap.ushell.Container.getRenderer("fiori2");
oRecentEntry.title = oConfig.display_title_text;
oRecentEntry.url = oConfig.navigation_target_url;
oRenderer.logRecentActivity(oRecentEntry);
</SCRIPT>
```

Related Information

Working with Recent Activities and Frequently Used Apps [page 431]

1.5.2.6 Implementing Data Loss Protection

Use the methods of the `Container` API to set the "dirty" (unsaved) state of a page and trigger the data loss protection mechanism when needed. Set the `dirty` flag to `true` when there are unsaved changes in the application that can lead to potential data loss.

Syntax

`setDirtyFlag(bIsDirty)`

`getDirtyFlag()`

These methods are part of the SAPUI5 `sap.ushell.services.container` namespace. For more information, go to SAPUI5 Demo Kit and then navigate to ▶ *API REFERENCE* ❯ *sap.ushell.services* ❯ *Container* ❯.

Parameters

Parameter	Type	Description
bIsDirty	Boolean	Indicates whether the dirty flag should be true or false.

Example

> ⌕ Sample Code
>
> ```
> <SCRIPT language ="JavaScript">
> ...
> var isDataChanged = sap.ushell.Container.getDirtyFlag()
> ...
> if(<change condition>) {
> sap.ushell.services.Container.setDirtyFlag(false);
> }
> ...
> </SCRIPT>
> ```

1.6   Security Aspects

This section discusses several security aspects you need to consider when using SAP Fiori launchpad.

Related Information

1.6.1 Security Aspects for Catalogs and Groups

- Assigning catalogs and groups to user roles makes the tiles available to the user.
    - Adding catalogs and groups to PFCG makes them visible in the launchpad.

    > **i Note**
    >
    > Entries for catalogs and groups are also displayed in SAP Easy Access Menu in SAP GUI.
    >
    > To suppress this behavior, you can make a setting in PFCG. In PFCG, open the role and select the *Hide Menu from SAP Easy Access* checkbox in the menu options.

    - The tile content may require additional PFCG authorization profiles which have to be assigned separately to the user.
- The authorization object `/UI2/CHIP` can be used to control access to tiles.

For more information, see SAP Help Portal at http://help.sap.com/nw-uiaddon. Under *Application Help*, open *SAP Library* and choose *SAP Fiori Launchpad* *Setting Up the Launchpad* *Configuring Authorization Roles* *Assigning Tile Catalogs and Groups to Roles* .

Related Information

1.6.2 Ensuring Complete Logout from Integrated Systems

When using other systems that are integrated with SAP Fiori launchpad, after logging out, an open browser window may still contain session cookies. A user who has access to the open browser window can access these

systems without having to authenticate. The solution described in this topic ensures that session cookies of all systems are removed when logging out from SAP Fiori launchpad.

SAP Fiori launchpad allows access to systems other than the SAP Fiori launchpad front-end server, which serves the URL to start:

- SAP Fiori launchpad
- UI-related resources
- REST and OData services for running the launchpad

> i Note
>
> To keep the complexity of the system landscape to a minimum, we recommend having only one SAP Gateway server for all OData services used in SAP Fiori scenarios.

For the following scenarios, logging out is performed completely and you do not need to perform the extra logout configuration described in this topic.

- SAP Fiori launchpad front-end server.
- Enterprise Search system (AS ABAP).
- Systems used to load remote tile catalogs, such as SAP HANA KPI tiles for SAP Smart Business.
- SAP Lumira™ running in SAP Fiori launchpad.

  > i Note
  >
  > For releases of SAP Lumira prior to 1.18, it is necessary to perform the configuration as described in SAP Note 2010502 .

- Systems that are accessed to start Web Dynpro ABAP or SAP GUI for HTML applications in the SAP Fiori launchpad based on report launchpad customizing with specific application types (not plain URL) - either through SAP Web Dispatcher or directly on the system.

For all other scenarios, it is necessary to do the following:

- On SAP Web Dispatcher: Maintain a logout rule symmetric to the rule causing the system login so that the logout URLs are routed to the correct system.
- On SAP Fiori launchpad front-end server: Define a custom logout page for the ICF node `/sap/public/bc/icf/logoff`, which ensures that all logout URLs are requested. Thus the missing logouts are guaranteed to take place when logging out from the front-end server.

Related Information

1.6.3 Security Aspects for Client-Side Caching of Target Mappings

When using SAP Fiori launchpad, target mappings are stored in the client cache.

For improved performance, target mappings are stored in the client cache. The data stored in the client cache conforms to the SAP security guidelines. It contains configuration data only.

The following information may be stored in the client cache:

- The list of intents and applications or application variants assigned to a specific user
- URLs of associated target systems, transaction codes, Web Dynpro ABAP applications and/or configurations
- If present, links to foreign URLs

Related Information

Cache Buster for Target Mappings and UI2 Services [page 366]

1.6.4 Security Aspects for Launchpad Configuration Files (Deprecated)

In a launchpad configuration file, you can set parameters for the SAP Fiori launchpad. It is important that you make sure that only authorized persons can edit these files.

For security reasons, the system considers a launchpad configuration file only if it originates from a path that is included in the `configurationFileFolderWhitelist` parameter of the `FioriLaunchpad.html` file.

> **i Note**
>
> Note that the the `configurationFileFolderWhitelist` parameter is depreccated and will be removed in a future version. The configuration with this parameter will then no longer be possible.

In the configuration delivered by SAP, a launchpad configuration file may only be located in one of the following folders:

- In the same folder as the `FioriLaunchpad.html` file
- `cfg/` (relative path to the `FioriLaunchpad.html` file)
- `cfg/sap/` (relative path to the `FioriLaunchpad.html` file)
- `/sap/ushell_config` (absolute path on the front-end server)
- `/sap/bc/ui5_demokit/test-resources/sap/ushell/demoapps/LaunchpadConfigFileExamples/` (absolute path on the front-end server)

Make sure that only authorized persons have write access in these folders.

If you would like to restrict the list of allowed paths further, or allow additional paths, you can edit the `FioriLaunchpad.html` file. This file is located in the BSP repository of your front-end server, in the folder ▶ */UI2/USHELL* 〉 *Page Fragments* 〉 *shells* 〉 *abap* ◣ .

> ⚠ Caution
>
> The `FioriLaunchpad.html` file is delivered by SAP. If you modify this file, your changes might be overwritten when you upgrade your system to a new release or support package. For this reason, remember to create a backup copy of the `FioriLaunchpad.html` file before upgrading.

1.6.5 Clickjacking Framing Protection

Clickjacking is an attempt to trick users into clicking hidden or masked user interface elements without the user realizing it. The user thinks he or she is clicking on the underlying element in the presented context, but is actually clicking on an action chosen by the attacker.

To prevent malicious applications from using SAP Fiori launchpad for clickjacking attacks, clickjacking framing protection is enabled by default.

Clickjacking framing protection ensures that your application only runs in trusted environments when other applications frame it. If clickjacking framing protection determines it is not already in a safe environment, clickjacking framing protection detects the origin of the framing window and compares it against a fixed value or list. The function prevents SAP Fiori launchpad applications from being embedded into other web applications, unless you trust the embedding application. You define trusted domains in an allowlist for clickjacking framing protection.

> ℹ Note
>
> Consider adding domains (such as `*.example.com`) to the allowlist for ease of maintenance, but weigh this risk against your current security measures for your network infrastructure.

To enable the standard clickjacking framing protection for SAP Fiori launchpad and SAP Fiori launchpad designer, maintain the allowlist for clickjacking framing protection. For more information, see Using a List for Clickjacking Framing Protection (for the current SAP NetWeaver Application Server version) or look at ▶ *https://help.sap.com/viewer/p/SAP_NETWEAVER* 〉 `<Your release>` 〉 *SAP NetWeaver Security Guide* 〉 *Security Guides for SAP NetWeaver Functional Units* 〉 *Security Guides for the Application Server* 〉 *Security Guides for AS ABAP* 〉 *SAP NetWeaver Application Server for ABAP Security Guide* 〉 *Special Topics* 〉 *Using a list for Clickjacking Framing Protection* ◣ .

If you do not activate the list, the launchpad defaults to a more restrictive clickjacking mechanism. It only allows a web site to frame the launchpad if it has the same origin. In this case it needs to be hosted on exactly the same domain.

Related Information

SAP Note 2057847🔗

1.7 Troubleshooting

This section provides a central starting point for solving problems in the launchpad or launchpad designer.

Something Is Not Working as Expected?

Administrators and developers use one of the support tools to analyze the error and find out possible root causes. In many cases, problems occur because the content is not configured correctly. In this troubleshooting guide, you get guidance on how to solve common issues such as missing content.

If there is an issue with a specific app, end users can use the *App Support* tool for help. If the plug-in is installed, it can be accessed via the user actions menu. For more information, see App Support [page 393]. With *App Support*, end users can immediately see if there are any configuration or authorization errors. The logs can also be forwarded to the admin for further analysis.

If an error occurs in the launchpad, end users can contact their support team using the *Contact Support* option in the launchpad, provided this option was enabled by an administrator. See Contacting Support [page 442] and Configuring the Option to Contact Support [page 95].

Contacting SAP Support

If this troubleshooting guide does not help to solve the issue, administrators or developers can create an incident to contact SAP support.

- How do I find the appropriate support component for an issue with SAP Fiori launchpad or SAP Fiori launchpad designer? See SAP Note 2348123.
- What do I need to consider for efficient incident analysis? See SAP Note 2116090.

Related Information

Launchpad Support Tools [page 375]
Launchpad Troubleshooting [page 631]
Launchpad Designer Troubleshooting [page 655]
Launchpad Content Exposure Troubleshooting [page 651]

1.7.1 Launchpad Troubleshooting

This section provides a central starting point for solving problems in the launchpad.

Here is an overview of the major types of problems that may occur:

- Tile, group or catalog is not displayed on the home page
- Navigation to an application does not work (failed intent resolution)
- Issues with the system aliases configured for the launchpad
- Problems with translated texts in tiles, catalogs, or groups

For an overview of tools that you can use for launchpad troubleshooting, see Launchpad Support Tools [page 375].

Related Information

1.7.1.1　Issues with Tiles, Groups or Catalogs

This section helps you to analyze and solve issues that may occur for tiles, groups and catalogs.

Issue	Possible Reasons	Error Analysis and Possible Solution
A tile is displayed with the error "Cannot load tile".	An end user personalized the group, e.g. by moving, deleting or adding tiles. An administrator deletes the tile from the catalog which is referenced by the group the end user personalized.	The end user needs to remove the orphaned tile from the group. See Personalizing the Home Page [page 501]. To prevent this issue, an admin can switch off the personalization of groups. See Editing Groups [page 306].
	An administrator deleted the tile from the catalog which is referenced by a group. After deleting the tile, the administrator did not clean up customized or configured groups which use the tile.	Use Fiori Launchpad Checks (transaction /UI2/FLC) to identify the orphaned tiles in groups. Delete these tiles in the SAP Fiori launchpad designer. Alternatively, you can identify orphaned tiles in groups using *Show Usage in Groups* in the launchpad content manager. See Where-Used Checks for Tiles [page 246].

Issue	Possible Reasons	Error Analysis and Possible Solution
	An administrator made changes in customizing and you upgraded the front-end server (or applied a support package).	The Fiori launchpad content changes from configuration (CONF scope) cannot be applied because customizing already exists.
		To resolve this issue, in the SAP Fiori launchpad designer locate the outdated catalog in customizing and click on *Outdated* in the error message. Confirm the message and refresh the page.
		Without customizing, the configured content shines through and the tile can be loaded.
		For more information, see Launchpad Designer Troubleshooting - Outdated Catalog [page 657].
A tile is not displayed.	There is no matching target mapping (semantic object/action) or device type.	Use Fiori Launchpad Intent Analysis (transaction `/UI2/FLIA`) for error analysis.
		In the SAP Fiori launchpad designer, make sure the target mapping configuration is correct.
		See Configuring Target Mappings [page 217].
	The tile is a reference tile and the reference is broken because the original tile was deleted.	Use Fiori Launchpad Checks (transaction `/UI2/FLC`) to identify broken references.
		Broken references cannot be restored. The tile needs to be created from scratch.
		Refer to
		> i Note > 2495630
		to find out how to further analyze the error.

Issue	Possible Reasons	Error Analysis and Possible Solution
	An admin has added a tile to a group. End users cannot see the new tile because they personalized the group.	Affected end users can delete the personalization in the SAP Fiori launchpad by resetting the group. See Personalizing the Home Page [page 501]
		Alternatively, an administrator can delete the personalization for the affected users. See Delete Personalization Data [page 355].
		To prevent this issue, an admin can switch off the personalization of groups. See Editing Groups [page 306].
A group or catalog is not displayed.	The catalog or group is not assigned to an authorization role.	Use Fiori Launchpad Checks (transaction /UI2/FLC) for error analysis.
		Assign the group or catalog to an authorization role. See Assign Business Catalogs to Roles [page 330].
	The group or catalog is assigned to a composite role. Settings for composite roles override the settings of included roles.	Use Fiori Launchpad Checks (transaction /UI2/FLC) to find out to which role the group or catalog is assigned. You can then search for the role in transaction PFCG to check whether it is a composite role.
		Assign the group or catalog to a specific role rather than a composite role.
	The group or catalog is assigned to a reference user.	To ensure that the role assignments of the reference user are also taken into account by the SAP Fiori launchpad, in table SSM_CUST, set parameter REFUS_NWBC to YES. See SAP Note 2540430 .
An administrator has changed groups but the change is not reflected for an end user.	The end user has personalized the group before the administrator made the changes.	The end user can reset the group. See Personalizing the Home Page [page 501].
		Alternatively, an administrator can delete the personalization for the affected users. See Delete Personalization Data [page 355].

General Recommendations

The following guidelines help to avoid some of the issues mentioned above:

- Do not delete published tiles but adapt existing tiles to their new purpose or change the target mappings which the intent of a tile is referencing.
- You as an admin do not fully control the lifecycle of personalized groups. Since this can cause issues (e.g. tiles cannot be loaded or changes to groups are not reflected), you should consider switching off the personalization of groups to prevent end users from removing or editing them.
  See Editing Groups [page 306].

Related Information

Checking Launchpad Content [page 382]
Determining Roles for an Intent [page 388]
Blog: Where is my tile?

1.7.1.2 Issues with Navigation

This section helps you to analyze and solve issues occurring when launchpad users try to navigate to a target application.

Symptoms

An end user wants to navigate from a tile on the launchpad homepage to an application. One of the following problems may occur:

- The target application cannot be loaded and an error is displayed (e.g., an HTTP 404 error, or the message "App could not be opened because the SAPUI5 component of the application could not be loaded.").
- The target application is not started in the right system (e.g. it is loaded on the frontend server instead of in the backend system).

Possible Reasons

- The intent referenced in the URL is not valid.
- The catalog referencing the application is not assigned to the same authorization role as the user who should be able to access it.
- The target system is not evaluated correctly because the system alias or RFC destination is not configured correctly.
- The application index is not up to date or there is an issue with the configuration.
- The target application was referenced from a tile in a remote catalog but the corresponding application components are not installed on the front-end server. See SAP Note 2390564 and Accessing Remote Catalogs [page 193].

How to Identify the Root Cause

Support Tool	Description	More Information
Message logging: • Network trace (browser developer tools) • UI2 application log (transaction SLG1 with object `/UI2/BE`)	If the navigation from the launchpad to an application fails, a log is written in the application log. To display a message containing an application log ID, use the console log of your browser. You can use the log ID to search for the relevant entry in the application log. To get more details on the root cause of the failed intent resolution, you can navigate from the message text of the log to transaction `/UI2/FLIA`.	Display Logs in the Browser [page 401] Display Logs for User Interface Services [page 402]
Intent analysis (transaction `/UI2/FLIA`)	This tool is useful if you already know the affected intent which references the application that is not displayed: • Check to which authorization role the intent is assigned and if a given user is assigned to this role. • Find out which target mapping belongs to the affected intent.	Analyzing Intents for SAP Fiori Launchpad [page 387]
System alias checks (transaction `/UI2/FSAC`)	Use this tool to check whether the system alias or RFC destination is defined correctly.	System Alias Checks [page 399]
Application log of SAPUI5 application index (Execute report `/UI5/APP_INDEX_CALCULATE` and choose *Display All Logs*.)	Use this tool to check whether the calculation of the application index was performed and whether issues occurred. There should be no warnings and errors logged. Especially keep an eye on errors related to the SAPUI5 repository or component that corresponds to a failing app. It's recommended that the application index is calculated regularly. There should be a general log and it shouldn't be too old (and at least reflect the latest updates of apps).	

How to Solve the Issue

Issue	Message Log Type	Possible Solution
The intent does not exist	Warning in UI2 application log	Make sure the intent that is referred to in the URL exists. Correct the intent reference in the URL or create an intent. See Configuring Target Mappings [page 217]
The intent exists and is valid but it is not assigned to the current user's authorization role	Warning in UI2 application log	If the current user should be able to navigate to the link, make sure that the user is assigned to an authorization role that is assigned to the catalog containing the app launcher tile and the target mapping. See Assign Business Catalogs to Roles [page 330].
The intent exists and is assigned to the user's authorization role, but the configuration of the navigation target is not consistent.	Error in UI2 application log	The target mapping references a target (launchpad role, launchpad instance, application alias/application ID) that is not configured in transaction LPD_CUST. Start transaction LPD_CUST and make sure a valid target is customized. See Changing LPD_CUST Entries for Navigation Targets [page 228]. Make sure the target mapping configuration is correct. See Configuring Target Mappings [page 217].
There's no general log or it's outdated (for example, it doesn't reflect the latest updates of apps).	Error in SAPUI5 application index	Report /UI5/APP_INDEX_CALCULATE is not scheduled regularly or not triggered automatically. The application index might be outdated. Make sure that the application index is up-to-date. For more information, see .

Issue	Message Log Type	Possible Solution
The component ID of your app or library in the ABAP system is not unique.	Error in SAPUI5 application index	If other repositories are reported with the same ID, contact the owners so that you or they can change the component IDs. See SAP Note 2177717 . We recommend that the owners or the administrator delete BSP applications deployed in $TMP only for testing purposes using the delete option of repository upload and download report /UI5/UI5_REPOSITORY_LOAD.
An issue with the virus scan prevents the recalculation of the application index. Report /UI5/APP_INDEX_CALCULATE indicates the following error messages: • No virus scan profile selected as default (profile) • Failed to replicate descriptor of application <name>	Error in SAPUI5 application index	Use transaction VSCANPROFILE to check whether the virus scan profile /UI5/UI5_INFRA_APP/REP_DT_PUT is inactive and configured correctly. See SAP Note 2041207 .
A dependency is missing.	Error in SAPUI5 application index	This could either be an issue with the app itself or a required software component is not installed.
An app indicates errors and might not run correctly.	Error in browser console	First of all, please check the log of the application index using report /UI5/APP_INDEX_CALCULATE. If there are no errors in the log, the app probably does not exist at all and is referred to by incorrect navigation customizing.
The component URL in the target mapping does not match the URL retrieved from the application index.	Error in browser console	This message is an indicator for incorrect configuration. In most cases this is not the root cause for an app to fail. Therefore, please check other console entries first.
Other errors	Error in browser console	There are most probably issues with the app itself. In case the app is running with outdated sources after applying a fix, see Outdated or Unavailable Resources (Caching Issues) [page 640].

Issue	Message Log Type	Possible Solution
A request failed for an URL containing `/sap/opu/odata/`.	Error in network trace	The used OData service is not available or does not work properly. Please check the service configuration.
A resource is outdated (HTTP 404 error).	Error in network trace	See Outdated or Unavailable Resources (Caching Issues) [page 640]
A file with suffix `-preload` is not found (HTTP 404 error).	Error in network trace	This is usually not a functional error and should not prevent the app from loading. However, the performance can suffer as single source files are loaded instead of one package. Usually, this indicates that a build step before deploying the app is missing.
A file is not found (HTTP 404 error).	Error in network trace	Depending on the URL, you have the following options: • URL contains path segment `/resources/`. The error is possibly caused by a dependency that could not be resolved. Please check the messages in the browser console for errors in transitive closure (such as no descriptor or multiple descriptors found) and also check the application index logs. If there are no errors, the app might not have declared a required dependency in its `manifest.json` file. • URL starts with `/sap/bc/ui5_ui5/<Namespace>/<Name>`. The corresponding ICF node might be inactive. Please activate it using transaction SICF, if applicable. Otherwise it should be a problem with the application itself.

Related Information

Logging and Tracing for Launchpad and Launchpad Designer [page 401]
Determining Roles for an Intent [page 388]

1.7.1.3 Outdated or Unavailable Resources (Caching Issues)

This section helps you as an expert to analyze and solve caching issues.

Symptoms

One of the following problems may occur due to caching issues:

- The browser log contains a failed request with an HTTP 404 error (REQUESTED RESOURCE OUTDATED).

  > **i Note**
  >
  > This is possibly a problem with internal cache management. Please do not clear the UI2 caches but open a ticket for component CA-UI2-INT-BE. See SAP Note 2362875 and Performance [page 362].

- At runtime, users do not get the resources implemented in the backend but outdated, cached versions.

Possible Reasons

- The request to fetch resources might not use cachebuster or caching might be active and simple caching is active instead. As a consequence, changes don't reach a client immediately but only after a certain expiration time (for example, one day later).
- The cachebuster tokens for the resource might be outdated on the frontend server or in the browser of the users.
- Single source files were changed in the backend but this change is not reflected in the `-preload.js` bundles loaded by the browser.

How to Solve the Issue

Step	What to Do?	Description	Background Information
Step 1	Always create a network trace and try to find the outdated resource.	If you know the resource name, filter the trace for it. If you implemented an SAP Note, you can look up the resource name in the correction instructions.	Changed texts are usually contained in `.properties` files, in `$metadata` files, or in annotation files. The changed logic is usually contained in `.js` files, in most cases there are `-preload.js` bundles.
Step 2	Determine the type of resource from the URL path.	`/sap/opu/odata` indicates OData service data. Ensure that there's no caching active. `/sap/bc/ui5_ui5` indicates app sources. If the path contains the path segment `/resources/`, the resource is probably part of the SAPUI5 distribution layer.	
Step 3	Determine whether cachebuster is used.	A cachebuster URL always contains a cachebuster token. If not, cachebuster is inactive and probably not implemented correctly. For sources of SAPUI5 apps cachebuster tokens are path segments such as `/~<Token>~/`. You have the options below.	Using cachbuster is recommended for all SAP solutions, except for older applications not running in SAP Fiori launchpad. Customer implementations not running in the launchpad have to actively enable cachebuster.

Step	What to Do?	Description	Background Information
	Option A: Cachebuster is active	Check whether the following solves the issue: • Reload SAP Fiori launchpad and check whether this solves the issue. (At startup, the launchpad receives all relevant tokens, but they might be outdated after a change in the system.) • Do a full calculation of the application index using report `/UI5/ APP_INDEX_CALCULAT E`. For more information, see . • If you own the resource that causes issues and you can change it, change the resource (for example, by deploying it again). Then reload the launchpad and check whether the new request fetches the resource and the token was changed.	
	Option B: Cachebuster is not working	For more information, see: • • • Cache Buster for SAP Fiori Launchpad and SAP Fiori Apps [page 367]	

Step	What to Do?	Description	Background Information
	Option C: Cachebuster is not implemented and other caching techniques are active	1. Check whether the request was served by the server or by the browser cache. If the response was not taken from the browser cache, but still does not correspond to the state of the backend, there could be other caches in between, for example ICM or web dispatcher caches. 2. Determine the cache headers sent by the server. Resources which are not under the control of cachebuster are expected to have either no caching at all or only a short-term caching (for example, two hours or one day). You can specify the caching behavior using the Business Add-In `/UI5/BADI_CONFIG_HTTP_HANDLER`. See SAP Note 2289829.	

1.7.1.4 Issues with OData Metadata Caching

This section helps you analyze and solve issues with OData metadata caching of SAP Fiori apps.

Prerequisites for OData Metadata Caching

OData Metadata Caching Update

Administrators need to schedule the `/UI5/UPD_ODATA_METADATA_CACHE` report for updating the cache buster tokens. For more information, see Scheduling Update of OData Metadata Caching [page 26]. In S/4HANA cloud systems, the cache update is processed automatically.

SAPUI5 Best Practices

The app itself needs to fulfill certain prerequisites. App developers need to make sure of the following criteria:

- The app must not be based on the deprecated `sap.ca.scfld.md` library.
- The app must have a `manifest.json` file.
- The app must declare a `sap.ui5/dependencies/minUi5Version` higher than 1.28.
- The app must declare the OData services under `sap.app/datasources`.
- The app must declare the SAPUI5 models under `sap.ui5.models` so that they are created.
    - The model must refer to the data source. It must not specify the URL directly.
    - The model must use a standard SAPUI5 OData model class. It must not specify a customized subclass.
- In case an app uses a reuse component: The reuse component that declares the OData service has to be declared either under `sap.ui5/dependencies/components` or preferably `sap.ui5/dependencies/componentUsages`. It is not sufficient to only declare the embedding library because it does'nt declare the OData service.

For more information, see Descriptor for Applications, Components, and Libraries🔗 in the SAPUI5 documentation.

How to verify if the app follows these criteria:

1. Before you launch the app, open the developer tools of your browser. The way to do this varies between browsers. In most browsers, you can right-click anywhere within your browser window and choose *Inspect*.
2. Launch the app.
3. In the browser developer tools, go to the *Network* tab and filter for `manifest.appdescr`.
4. Look for the manifest that belongs to the app (there might be other manifests for shell components). If no `manifest.appdescr` is found, the app violates the development criteria above.
5. If a manifest is found, double-click it to open it and verify the remaining criteria mentioned above.
6. In case of reuse components, also filter for `manifest.json` and verify the found manifests.

> **ⅈ Note**
>
> If you find any violations or if you have questions, contact the app developer.

Symptoms

OData metadata documents or annotation files aren't cached on the client, the caching performance is bad, or other symptoms connected to OData metadata caching arise.

Possible Reasons

- The report for updating the cache buster tokens hasn't been scheduled.
- The app doesn't follow SAPUI5 best practices [page 643].
- There's an issue with the particular OData service. For more information, see *How to Identify the Root Cause*.

- Shared memory is insufficient.

How to Identify the Root Cause

1. Identify the OData service that corresponds to the app that you want to analyze.
2. For detailed logs for each service, execute the `/UI5/APP_INDEX_CALCULATE` report and choose *Display all Logs*.
3. For each client, open the logs for the subobject *Backend Context Token Collection (Client Dependent)*.
4. Check the status for the requested token and the error messages of the respective service.

> **i Note**
>
> SAP Gateway returns the status for the requested token according to specific rules.
>
> There are the following statuses:
>
> - `ERROR`: There's an error in the OData service or there's a communication problem.
> - `UNKNOWN`: The service is unknown or isn't configured. The following reasons for this status can apply:
>   - The corresponding app doesn't work because the service doesn't exist or isn't configured.
>   - The corresponding app works because the app doesn't need the service and declares it by mistake.
>   - The corresponding app works and SAP Gateway isn't able to return the requested token.
> - `AMBIGUOUS`: The service isn't supported, for example, because user-dependent routing is configured.

How to Solve the Issue

Most importantly: Schedule the `/UI5/UPD_ODATA_METADATA_CACHE` report for updating the cache buster tokens. For more information, see Scheduling Update of OData Metadata Caching [page 26].

Solving OData Metadata Caching Issues

Issue	Possible Solution
Error in service	If the service is an SAP service, contact SAP support.
Communication problem	Solve the system communication problem.
The corresponding app doesn't work because the OData service doesn't exist or isn't configured.	Check if the app is really used.
The corresponding app works because the app doesn't need the service and declares it by mistake.	You can ignore this warning.

Issue	Possible Solution
The corresponding app works and SAP Gateway isn't able to return the requested token.	Contact SAP support.
The service isn't supported, for example, because user-dependent routing is configured.	Do not use user-dependent routing.
• Bad performance when executing `/UI5/UPD_ODATA_METADATA_CACHE` • Calculated cache tokens aren't stable and change after every run of the report.	Increase the shared memory so that the general system load stays below 90%. To do this, go to transaction RZ11 and change the `abap/shared_objects_size_MB` parameter accordingly. For more information about shared memory, see 1322182.

Other Issues

Why do I no longer find any `$metadata` requests when I start an app?

The SAPUI5 OData model introduced another cache for metadata, which is stored in a browser database. It uses the same cache tokens. You can clear this database in the browser developer tools on the `Application` tab.

What restrictions exist in older SAPUI5 versions?

If your app is based on a SAPUI5 version below 1.66, it's not possible to propagate cache tokens to reuse components. In this case, `$metadata` calls for services of reuse components can't be cached.

Why are the cached `$metadata` and annotations outdated?

Four layers of caches are involved:

- SAP Gateway metadata cache
- ICM cache
- Browser cache
- SAPUI5 local database cache

If metadata changes in the back end, all layers need to be invalidated. The invalidation depends on the `last-modified` timestamp of the service. If metadata changes, `last-modified` must reflect this and all caches are invalidated.

SAP Gateway and the service implementation need to ensure a proper update of last-modified (duty of service).

If this fails and if you have verified that the have been met, create a customer incident for component CA-UI5-ABA-AIDX.

Related Information

Cache Buster for OData Metadata of SAP Fiori Apps [page 370]

1.7.1.5 Issues with Translated Texts

This section helps you to analyse and solve issues with translated texts in the launchpad.

Symptoms

- An end user of the launchpad has reported an issue with a translated text in a tile, catalog, or group title.
- A text is not displayed in the correct language

Related Information

Translating Launchpad Texts [page 408]
Text Displayed in the Wrong Language [page 647]

1.7.1.5.1 Text Displayed in the Wrong Language

This section provides information on what to do if texts (e.g. tile titles) are not displayed in the expected language.

Prerequisites

- You have made sure that the requested language is installed in your system landscape.
- You have checked that the default logon language was not overwritten. See *Specifying Language Settings (Optional)* in the SAP Fiori: Setup and Configuration guide.

Context

Depending on the adaptation layer, there are different reasons why a text is not displayed in the requested language:

- Case 1: The translation was not imported on customer side or the translation was not delivered by SAP. In this case, the text is displayed in the source language in the configuration layer rather than in the selected translation language.
- Case 2: A text delivered by SAP was changed in the customization layer but the changed text was not translated. In this case, the text is displayed in the source language in the customizing layer (not the corresponding translated text in the configuration layer).
  Note that any change in the customization layer affecting a catalog or group results in a changed title of the catalog or group.
- Case 3: A text was changed in the personalization layer. Personalized texts are not translatable.

The steps below will help you to narrow down the root cause of the missing translation.

Procedure

1. Start *Fiori Launchpad Checks* (transaction `/UI2/FLC`) in the system in which you run the SAP Fiori launchpad and in the source language of your project (usually English).
2. In the *Adaptation Layer* section, choose *Customization*.
3. Choose *Execute*.

   We recommend you use the layout variant 2SAP_TRANS to display all relevant fields to analyze issues with translated texts.

   You can use the ALV export function to save the table in the result screen as a local file.
4. In the result screen, search for the text that is displayed in the wrong language.
5. Note down the corresponding catalog ID, tile ID, and group ID.
6. Log on in the required translation language.
7. Repeat steps 2 and 3.
8. In the result screen for the translation language, search for the IDs determined in step 5 and check the entries of the affected text fields:

   - The field is empty.
     This means that the translation was not imported on customer side or not delivered by SAP. Check with your system administator if the required language was imported.
   - The field displays the text in the source language.
     The text was not translated by the customer. Determine the configuration ID and the text ID and contact your translation department (see Translate Texts from Launchpad Designer (Customizing Scope) [page 408]).
9. If you can exclude missing language import and missing translation (case 1 and 2), the reason why the text is not displayed in the expected language could be that it was changed in the personalization layer (case 3).

   To fix this, administrators can delete all personalizations for a specific configuration ID.
   1. To determine the configuration ID, see step 1 under Translate Texts from Launchpad Designer (Customizing Scope) [page 408].
   2. Launch the Web Dynpro ABAP application `WD_ANALYZE_CONFIG_USER`.
   3. Enter the configuration ID and select "Personalization" as user scope.
   4. Select all entries and choose *Delete*.

   For more information, see section "Analyzing Customizing and Personalization with WD_ANALYZE_CONFIG_USER" under Analyzing Web Dynpro ABAP Adaptations .

Related Information

1.7.1.6 Issues with SAP Easy Access Menu

This section helps you to analyze and solve issues that may occur for the user menu or the SAP menu in the app finder.

Issue	Possible Reasons	Error Analysis and Possible Solution
Entries from the user menu in the ABAP system are not displayed in the *User Menu* tab of the app finder.	In the Menu Options dialog box in transaction **PFCG**, the *Hide Menu from NWBC, Suite Page Builder, and Fiori Launchpad* checkbox is selected for the relevant role.	Deselect the checkbox. See Assign Business Catalogs to Roles [page 330].
	The user menu entry in the transaction **PFCG** has an assigned RFC destination.	To integrate apps of a remote system, we recommend creating a technical back-end catalog in the respective remote system and replicating it to the SAP Fiori front-end server. See Advanced Scenario - Adapt SAP Template Content and Add Your Own Content [page 123].
It takes very long to load entries in the *SAP Menu* tab of the app finder or there is a timeout and the entries are not displayed at all.	• The caching mechanism provided by report /UI2/ EAM_BUILD_CACHE is not available in the system. • The caching mechanism is not activated. • The caching mechanism is activated but the cache is not filled yet.	Install SAP_UI 751 SP05 (or higher), SAP_UI 752 SP02 (or higher) or SAP Note 2545066 . Schedule the report /UI2/ EAM_BUILD_CACHE to run periodically. See Scheduling Caching of SAP Menu Entries [page 27].
Tiles are displayed duplicated in the *SAP Menu* tab of the app finder, that is, they call the same app, but one tile is displayed with info text and icon, and one tile without.	To display the apps in the *SAP Menu* tab of the app finder, two data sources are used: • *SAP Menu* or *User Menu* • Launchpad app descriptor items	The system behaves correctly. The system enables the end user to transfer the tile with icon and info text to his home page, which increases the recognition.

Related Information

1.7.1.7 Theming Issues

This section helps you to analyze the root cause if the SAP Fiori launchpad is not rendered with the expected theme.

Symptom	How to Solve the Issue
The launchpad is not rendered in the default theme SAP Belize for users who have already assigned the SAP Blue Crystal theme in the personalisation.	See SAP Note 2408337
After upgrading software component SAP_UI from version 750 to 751 or higher, the launchpad displays a black screen at runtime.	See SAP Note 2417862

Related Information

1.7.1.8 Performance Issues

This section helps you to analyze the root cause of performance issues in the SAP Fiori launchpad.

A slow start-up performance could be caused by a very large number of tiles on the home page. If users experience long waiting times, we recommend administrators to check which groups and tiles are not required on the home page and remove the relevant assignments of groups to roles or tiles to groups. Users can still access the apps using the app finder and add the corresponding tiles back to the home page if required.

For more information on root causes for performance issues, see SAP Note 2447857.

Related Information

1.7.2 Launchpad Content Exposure Troubleshooting

This section provides a central starting point for solving issues related to the exposure of launchpad content to the launchpads on SAP BTP.

Overview

Here is an overview of the major types of issues that may occur:

- Some content was not exposed, e.g. a tile is missing or the navigation does not work.
- The content was exposed but is structured in a different way, e.g. a tile is duplicated.
- The content was exposed but is not up-to-date.

The most common root causes for issues with launchpad content exposure include the following:

- The end-to-end process was not set up correctly, e.g. system configuration and user authorizations.
- Content issues within the classic launchpad content (launchpad content on ABAP platform) can also cause issues when the content is exposed. These issues are typically also visible in the design-time tools (e.g. launchpad content manager).
- Especially when it comes to the tile/target mapping combinations in catalogs, the requirements for the content structure are stricter in the CDM conversion than for the classic launchpad content model. For example, when a tile is intended to launch the app referenced by the target mapping. tile and target mapping have to reside in the same catalog in order for the navigation to work.
  The ideal content structure is simple tile/target mapping combinations with unique intents in a catalog.
- No content exposure was triggered after the content was updated or there was an issue with the import to the launchpads on SAP BTP.

Checklist to Prevent Issues

Certain boundary conditions have to be met in order for the content exposure to be successful. Before you start with your error analysis, check the following:

- Make sure the prerequisites listed under Manage Launchpad Content for Exposure [page 343] are met.
- Make sure the role selection was saved at least once.
- Some features are not yet supported for content exposure. Please check the list of restrictions. See Restrictions - Federation. For example, custom tile types are not supported for content exposure and are therefore not exposed.
- Make sure the required boundary conditions are met for the entities you want to expose. See Checklist for Successful Content Exposure [page 342].
- To ensure that the exposed content is up-to-date, content change notification should be in place. See Manage Content Change Notifications [page 348].

Error Analysis

For a more detailed error analysis of exposure issues, you can use the error log in *Manage Content for Launchpads on SAP BTP* (transaction `/UI2/CDM3_EXP_SCOPE`):

- Choose *View Log* to display issues after the exposure was triggered
- Choose ▶ *Preview* ▶ *Display Log* ▌ to display issues before you trigger the exposure.

Only content without errors can be exposed.

The log bundles messages for entities relevant for exposure. It starts with a header entry identifying the date, time, and user details of the exposure, followed by a breakdown into general information and information on the entities roles, business apps, catalogs, spaces, pages, groups, and URL templates.

The entities contain a list of messages and other information relevant for error analysis, such as the application title or the app resource.

For error analysis, check the log entries with the following status:

- 🔴 (Error): The entity is not exposed and an action is necessary to fix this.
- ⚠️ (Warning): The entity is exposed but there could potentially be an issue for the end user.

Depending on the error, you can choose the error log message and click on *Open in Design-Time Tool*. This will navigate you to the design-time tool in which the tile/target mapping was maintained.

You can also use the launchpad content manager to follow up on errors displayed in the log. See Displaying Issues with Launchpad Content [page 261].

The *Tile/Target Mapping ID* column shows the ID that corresponds to the business app ID.

For more information on how the entities selected for exposure are converted to CDM format, see Conversion of Entities to the Common Data Model (CDM) [page 652].

Related Information

Exposing Launchpad Content to SAP Business Technology Platform [page 340]
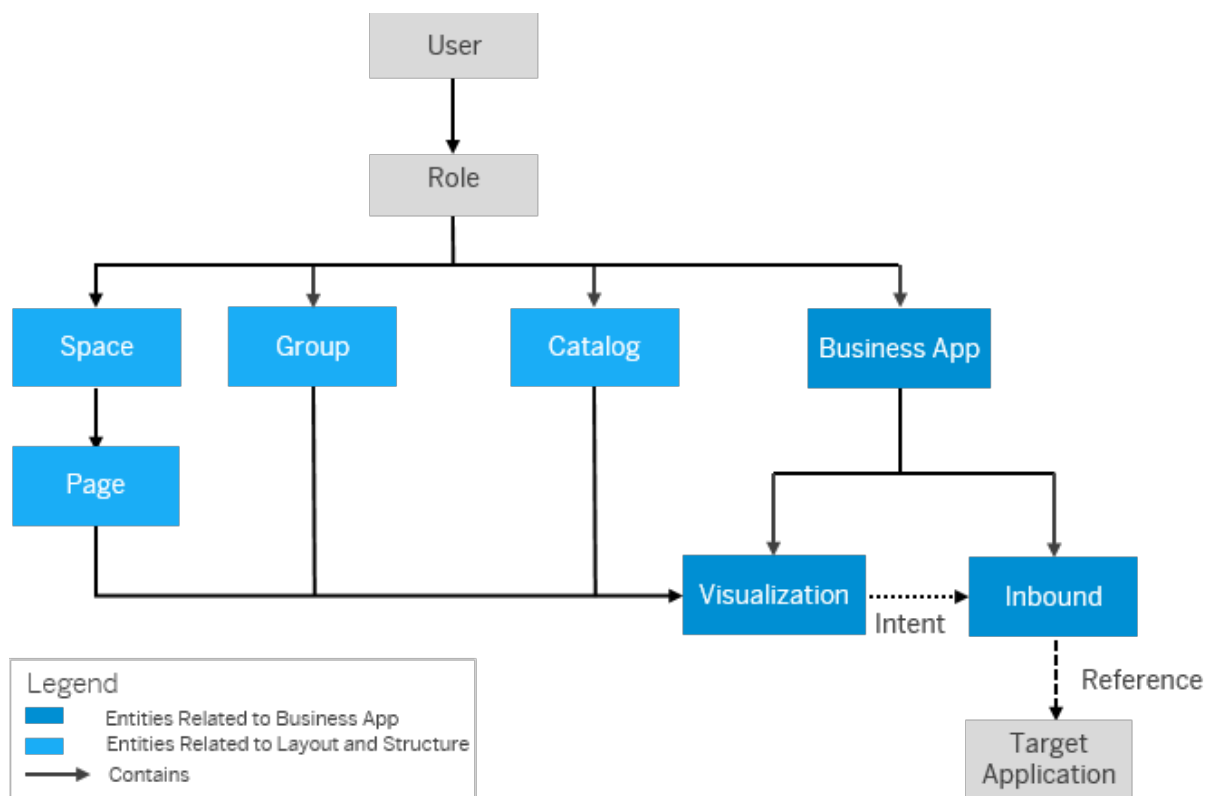
1.7.2.1 Conversion of Entities to the Common Data Model (CDM)

Get an overview of the CDM entities and how they relate to the entities in the data model of the launchpad on ABAP Platform (classic data model).

During the launchpad content exposure, the entities selected for exposure are converted to CDM format.

CDM is a stack-agnostic format to describe the launchpad content, which allows to describe all the objects used and their relationship to each other.

The following graphic provides an overview of how the CDM entities relate to each other.

Legend
- ■ Entities Related to Business App
- ■ Entities Related to Layout and Structure
- → Contains

For a detailed description of the classic data model, see About Launchpad Content [page 103].

Role

Apps are assigned to users via roles.

In the classic data model, tile/target mapping combinations can only be assigned to roles via catalogs.

In CDM, business apps can directly be assigned to roles.

Business App

Tile/target mapping combinations within a catalog are converted to business apps in CDM.

The main entities of a business app are:

- Visualization - Corresponds to the tile, i.e. defines how the app is rendered.
- Inbound - Corresponds to the target mapping, i.e. defines the navigation target that is used for intent-based navigation.

A business app defines an application from the business perspective. This means that a business app has a certain business purpose or task to fulfill.

Business apps are directly assigned to roles, while tile/target mappings are assigned to catalogs, which in turn are assigned to roles.

There are different constellations of tile/target mapping combinations. The ideal constellation for the CDM conversion are tile/target mapping combinations with unique intents in a catalog. This allows a unique matching.

The following constellations can lead to issues during the conversion to CDM:

- In the classic data model, tile and target mapping can reside in different catalogs. As long as both are assigned during runtime, you can launch the app by clicking the tile. In the exposed content, this case will no longer work.
- If a tile can potentially navigate to multiple target mappings, it is not possible to statically determine the preferred navigation target. This leads to a duplication of tiles after the conversion to CDM.

See Checklist for Successful Content Exposure [page 342].

Catalog

In the classic data model, the catalog is the entity primarily holding authorization information and grouping tiles and target mappings for a specific business use case.

In CDM, catalogs only define how the visualizations of apps are displayed in the app finder, according to the apps that are assigned to the catalog.

Space and Page

Using spaces, the user can have more than one (home) page to access their launchpad content. Each space can have one or more pages. Within a page, the content can be divided into sections, which are similar to groups.

Pages are assigned to users via spaces that are assigned to roles.

In the classic data model, tiles and target mappings are assigned to pages.

In CDM, visualizations are assigned to pages.

Group

A group defines how the visualizations of apps are grouped on the launchpad home page.

In the classic data model, this is based on the assigment of tiles to groups.

In CDM, this is based on the assignment of visualizations to groups.

Related Information

Exposing Launchpad Content to SAP Business Technology Platform [page 340]

1.7.3 Launchpad Designer Troubleshooting

This section provides a central starting point for solving problems in the launchpad designer.

Prerequisites

You have maintained the following settings:

- If you are in configuration scope, you have maintained the workbench request and package name
- If you are in customization scope, you have maintained the customizing request.

You may encounter certain errors even after following the above steps.

If an error occurs in the launchpad designer when entities (e.g. catalogs or tiles) are created, updated, or deleted, a log is written in the UI services application log (object /UI2/BE). You can use the application log to analyse the root cause of an error. For more information, see Logging and Tracing for Launchpad and Launchpad Designer [page 401].

Here is a list of some of the common errors encountered with possible causes and solutions:

Issue	Possible Reason	Possible Solution
Unable to create or delete group	- Back end is not available - Workbench request not maintained - Customizing request not maintained	- If you are in customization scope, maintain the customizing request under the *Settings* button on the launchpad designer page. - If you are in configuration scope, maintain the workbench request under the *Settings* button on the launchpad designer page. - Contact the system administrator.
Unable to create or delete catalog	- Back end is not available - Workbench request not maintained - Customizing request not maintained	- If you are in customization scope, maintain the customizing request under the *Settings* button on the launchpad designer page. - If you are in configuration scope, maintain the workbench request under the *Settings* button on the launchpad designer page. - Contact the system administrator.

Issue	Possible Reason	Possible Solution
Unable to add or delete a tile in catalog or group	• Correct path to the tile not provided • Workbench or customization request not maintained • Back end service (OData) failure	• Maintain the transport request appropriately. • Contact the system administrator. • The tile templates are in the universal catalog (`FLPD_CATALOG`). If you want to add a new tile template, it has to be registered to the universal catalog using back end transaction (`UI2/CHIP`). For more information, see Registering a CSR CHIP in a Catalog [page 594].
SAP note is applied, but the changes are not reflected.	Browser cache contains old resources	Clear the browser cache to fetch the latest resources
Tile configuration changes are not saved	• Customization or workbench request or package not maintained • Back end service failure	Maintain the transport request appropriately
Changes made for a catalog or a group in configuration scope are not reflected in customizing scope.	Catalog or group is outdated.	Delete the responsible catalog or group in customizing scope and refresh. (All the customization changes done are deleted).
You want to make changes to a catalog created in the configuration scope using the customizing scope but the catalog is read-only in the customizing scope.	See SAP Note 2971366 for more information.	
I delete a catalog or group in customizing scope, but it appears again when I refresh the launchpad designer tool.	Catalog or group is created in configuration scope.	If you really want to delete the catalog or group, then delete in configuration scope as this might impact other clients.
I see an error tile in some of the groups. Why?	• You have no authorization for this tile. • The tile might have been deleted from the parent catalog.	• Check for authorizations • Delete the tile from group even if it was deleted from the catalog.

Given below is a list of common error messages or system messages:

Message	Possible Reason	Possible Solution
`Cannot login with PERS scope; login with CUST or CONF`	PERS scope is not supported in the launchpad designer	Either remove the scope parameter or add `&scope=CONF` at the end of the URL.

Message	Possible Reason	Possible Solution
`Remote catalog '<catalog_id>' cannot be retrieved`	Some of the SAP HANA catalogs are not supported due to the system landscape setup.	• Check whether the system landscape was set up correctly and check the authentication to the remote catalog, for example, SAP HANA based Smart Business catalog works correctly. • Contact the system administrator.
`Operation Failed. Transport information not specified.`	• If the application is launched in configuration scope, default workbench request and default package are not maintained. • If the application is launched in customizing scope, default customizing request is not maintained.	• Check the user parameter settings in back end ABAP system or settings on the launchpad designer page. • You can work with catalogs or groups in 'Z' and 'Y' namespace in case no transport to other systems is required. • Select the *Local Object* checkbox under the *Settings* button on the launchpad designer page.
`ID: <catalog_id> outdated`	Changes made for a catalog or a group in configuration scope are not reflected in customizing scope.	Click on *Outdated* next to the responsible catalog or group in customizing scope and confirm the message. i Note This action deletes all changes in customizing scope.

Related Information

[Updating Content Created with the Launchpad Designer \[page 182\]](#)

Important Disclaimers and Legal Information

Hyperlinks

Some links are classified by an icon and/or a mouseover text. These links provide additional information.
About the icons:

- Links with the icon : You are entering a Web site that is not hosted by SAP. By using such links, you agree (unless expressly stated otherwise in your agreements with SAP) to this:

  - The content of the linked-to site is not SAP documentation. You may not infer any product claims against SAP based on this information.

  - SAP does not agree or disagree with the content on the linked-to site, nor does SAP warrant the availability and correctness. SAP shall not be liable for any damages caused by the use of such content unless damages have been caused by SAP's gross negligence or willful misconduct.

- Links with the icon : You are leaving the documentation for that particular SAP product or service and are entering an SAP-hosted Web site. By using such links, you agree that (unless expressly stated otherwise in your agreements with SAP) you may not infer any product claims against SAP based on this information.

Videos Hosted on External Platforms

Some videos may point to third-party video hosting platforms. SAP cannot guarantee the future availability of videos stored on these platforms. Furthermore, any advertisements or other content hosted on these platforms (for example, suggested videos or by navigating to other videos hosted on the same site), are not within the control or responsibility of SAP.

Beta and Other Experimental Features

Experimental features are not part of the officially delivered scope that SAP guarantees for future releases. This means that experimental features may be changed by SAP at any time for any reason without notice. Experimental features are not for productive use. You may not demonstrate, test, examine, evaluate or otherwise use the experimental features in a live operating environment or with data that has not been sufficiently backed up.
The purpose of experimental features is to get feedback early on, allowing customers and partners to influence the future product accordingly. By providing your feedback (e.g. in the SAP Community), you accept that intellectual property rights of the contributions or derivative works shall remain the exclusive property of SAP.

Example Code

Any software coding and/or code snippets are examples. They are not for productive use. The example code is only intended to better explain and visualize the syntax and phrasing rules. SAP does not warrant the correctness and completeness of the example code. SAP shall not be liable for errors or damages caused by the use of example code unless damages have been caused by SAP's gross negligence or willful misconduct.

Bias-Free Language

SAP supports a culture of diversity and inclusion. Whenever possible, we use unbiased language in our documentation to refer to people of all cultures, ethnicities, genders, and abilities.

THE BEST RUN **SAP**