

SAP Copyrights, Trademarks and Disclaimers

© 2020 SAP SE or an SAP affiliate company. All rights reserved.

No part of this publication may be reproduced or transmitted in any form or for any purpose without the express permission of SAP SE or an SAP affiliate company.

SAP and other SAP products and services mentioned herein as well as their respective logos are trademarks or registered trademarks of SAP SE (or an SAP affiliate company) in Germany and other countries. Please see <http://global12.sap.com/corporate-en/legal/copyright/index.epx> for additional trademark information and notices.

Some software products marketed by SAP SE and its distributors contain proprietary software components of other software vendors.

National product specifications may vary.

This course may have been machine translated and may contain grammatical errors or inaccuracies.

These materials are provided by SAP SE or an SAP affiliate company for informational purposes only, without representation or warranty of any kind, and SAP SE or its affiliated companies shall not be liable for errors or omissions with respect to the materials. The only warranties for SAP SE or SAP affiliate company products and services are those that are set forth in the express warranty statements accompanying such products and services, if any. Nothing herein should be construed as constituting an additional warranty.

In particular, SAP SE or its affiliated companies have no obligation to pursue any course of business outlined in this document or any related presentation, or to develop or release any functionality mentioned therein. This document, or any related presentation, and SAP SE's or its affiliated companies' strategy and possible future developments, products, and/or platform directions and functionality are all subject to change and may be changed by SAP SE or its affiliated companies at any time for any reason without notice. The information in this document is not a commitment, promise, or legal obligation to deliver any material, code, or functionality. All forward-looking statements are subject to various risks and uncertainties that could cause actual results to differ materially from expectations. Readers are cautioned not to place undue reliance on these forward-looking statements, which speak only as of their dates, and they should not be relied upon in making purchasing decisions.

Typographic Conventions

American English is the standard used in this handbook.

The following typographic conventions are also used.

This information is displayed in the instructor's presentation



Demonstration



Procedure



Warning or Caution



Hint



Related or Additional Information



Facilitated Discussion



User interface control

Example text

Window title

Example text

Contents

vii	Course Overview
1	Unit 1: End-User Perspective
2	Lesson: Exploring SAP Fiori Design
9	Lesson: Exploring SAP Fiori Launchpad
18	Lesson: Personalizing SAP Fiori
23	Lesson: Exploring SAP Fiori Data Handling
32	Unit 2: Technology
33	Lesson: Explaining User Interfaces
39	Lesson: Explaining Data Services
47	Lesson: Explaining Application Types
62	Unit 3: Architecture
63	Lesson: Examining SAP Fiori for ABAP Platform
72	Lesson: Examining SAP Fiori for SAP Business Suite on HANA
82	Lesson: Examining SAP Fiori for SAP S/4HANA
96	Unit 4: Configuration
98	Lesson: Creating SAP Fiori Groups
103	Lesson: Managing SAP Fiori Catalogs
110	Lesson: Creating SAP Fiori Catalogs
116	Lesson: Creating LPD_CUST Target Mappings
120	Lesson: Creating SAPUI5 Target Mappings
122	Lesson: Creating Web Dynpro Target Mappings
125	Lesson: Creating Transaction Target Mappings
128	Lesson: Creating Back-End Catalogs
131	Lesson: Configuring SAP Fiori Launchpad
137	Lesson: Configuring Analytical Applications
150	Unit 5: Adaptation
151	Lesson: Using the UI Theme Designer
156	Lesson: Using SAP Screen Personas
165	Lesson: Using SAP Fiori Runtime Authoring
168	Lesson: Extending SAP Fiori Applications
185	Unit 6: Mobility
186	Lesson: Examining SAP Fiori Mobile
193	Lesson: Examining SAP Fiori for iOS

205 Unit 7: Integration

- | | |
|-----|--|
| 206 | Lesson: Integrating SAP Workflow in SAP Fiori |
| 212 | Lesson: Integrating SAP Fiori in SAP Enterprise Portal |
| 219 | Lesson: Exploring SAP Cloud Platform |

234 Unit 8: Further Information

- | | |
|-----|-------------------------------------|
| 235 | Lesson: Getting Further Information |
|-----|-------------------------------------|

237 Index

Course Overview

TARGET AUDIENCE

This course is intended for the following audiences:

- Application Consultant
- Developer
- Enterprise Architect
- Solution Architect
- System Administrator
- System Architect



UNIT 1 **End-User Perspective**

Lesson 1

Exploring SAP Fiori Design	2
----------------------------	---

Lesson 2

Exploring SAP Fiori Launchpad	9
-------------------------------	---

Lesson 3

Personalizing SAP Fiori	18
-------------------------	----

Lesson 4

Exploring SAP Fiori Data Handling	23
-----------------------------------	----

UNIT OBJECTIVES

- Explore the design of SAP Fiori
- Explore the SAP Fiori launchpad
- Personalize SAP Fiori
- Explore SAP Fiori data handling

Unit 1

Lesson 1

Exploring SAP Fiori Design



LESSON OBJECTIVES

After completing this lesson, you will be able to:

- Explore the design of SAP Fiori

Principles



- A collection of UX-driven apps renewing enterprise business functions
- Works seamlessly on desktops, tablets and smartphones

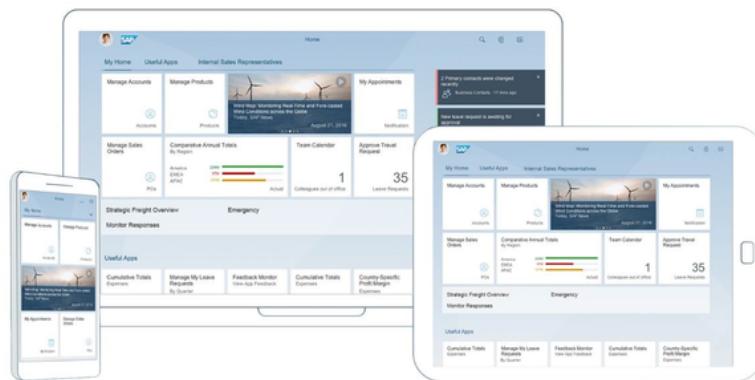
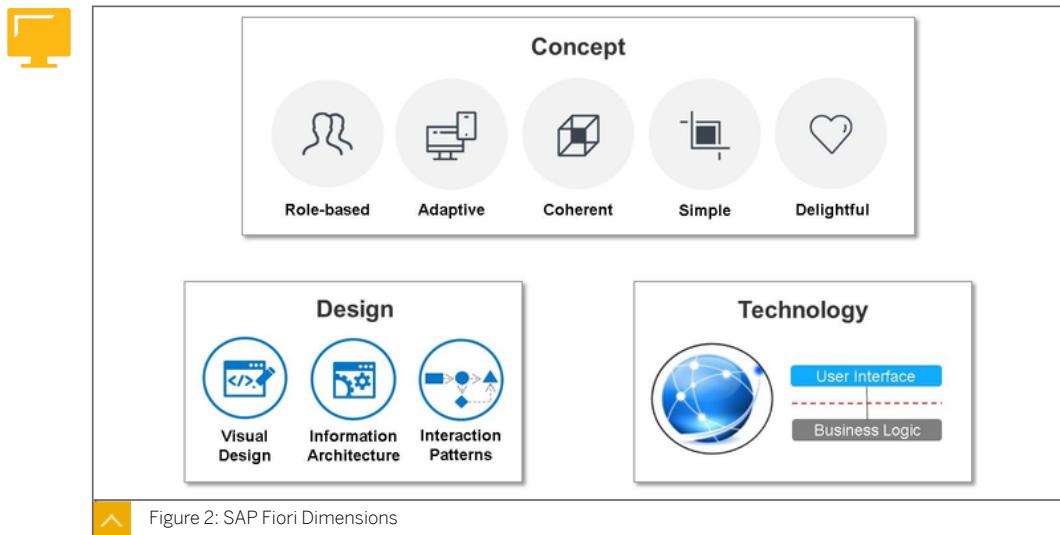


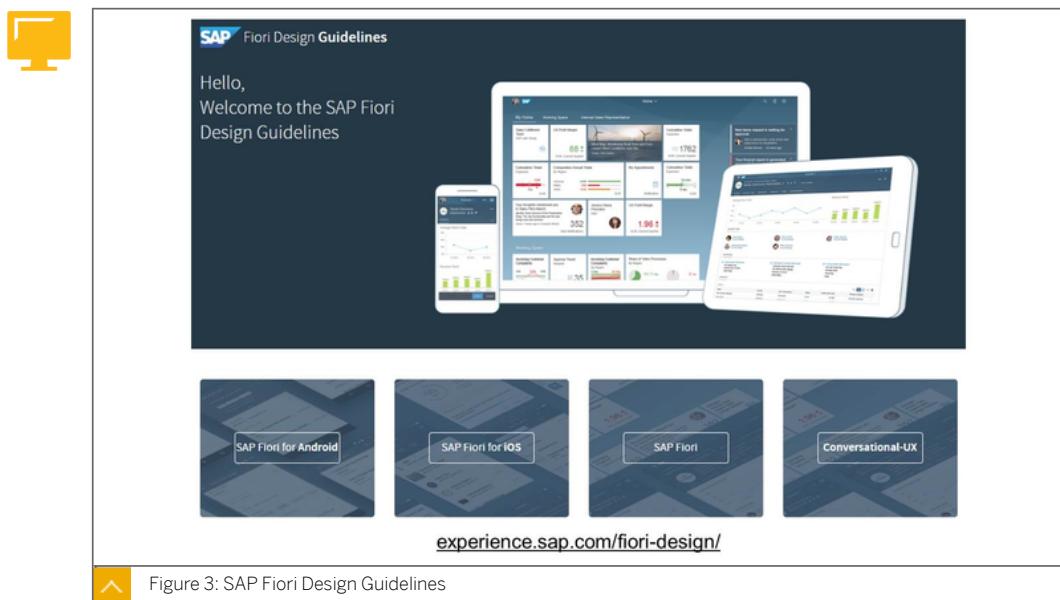
Figure 1: What Is SAP Fiori

SAP Fiori is a collection of apps that are created based on the rules of the SAP User Experience. At the point of SAPPHIRE in 2013, the first 25 apps for managers and employees with request and approval functions had been released. Since then the number of apps has increased greatly. SAP Fiori 2 was introduced with SAP S/4HANA 1610, taking the idea of Fiori to the next level.



The three dimensions in which SAP Fiori is defined are design, concept, and technology. In each dimension, rules and guidelines from optic, handling, interactions, and architectures to technologies in development and the system landscape are in place to define what SAP Fiori really is.

Design



All details of the SAP Fiori design are available as guidelines for general use. You can find all aspects of the Fiori design starting with the five core design principles up to floorplans of pages and details for UI elements. There are also several resources available for download, such as stencils and the SAP icon font, to empower customers to design their own apps.

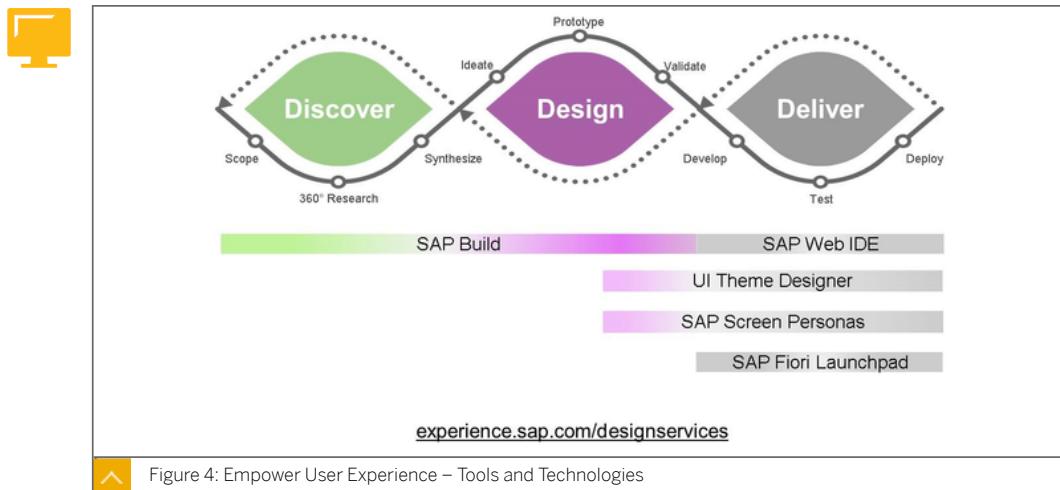
Since May 2016, SAP Fiori is also available for Apple iOS. Apple is a strong partner, especially in terms of design. There is a close cooperation between Apple and SAP to provide not only a

Unit 1: End-User Perspective

merged design but also guidance and tools for developers to develop native apps for iOS. In addition a growing number of apps is developed by this cooperation leveraging the features of Apple mobile devices.

Since June 2018, SAP Fiori is also available for Google Android. It provides a merged design and guidance for developers to develop native apps for Android. But in contrast to SAP Fiori for iOS, no ready-to-use apps are provided to end users.

Since February 2019, design principles for conversational user experience (CUX) in SAP Fiori are available. It is already possible in SAP Fiori 2.0 to use the SAP Conversational AI, and will be a central part of the SAP Fiori 3 design.



The source of the SAP UX Strategy and, therefore, SAP Fiori, is design thinking. This mindset embodies a human centered process where diverse thinkers build on each other's ideas to understand and solve problems. The following are the main phases in this process:

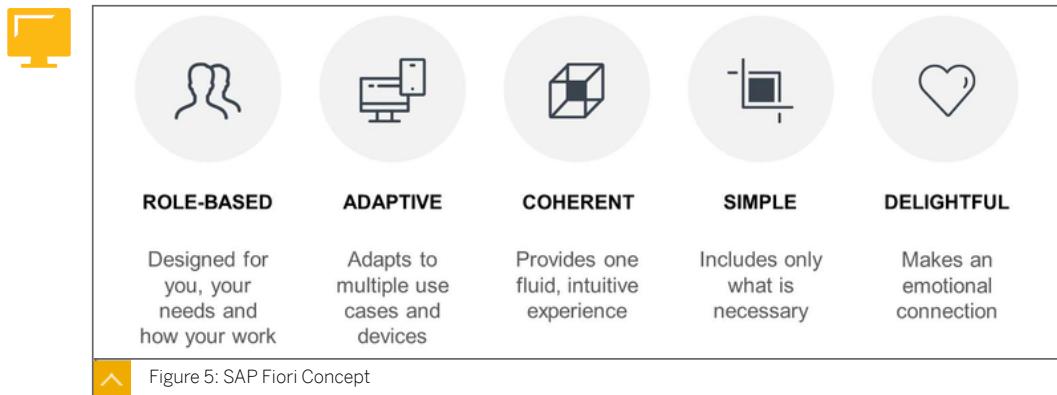
1. Discover: which combines scoping, 360° research, and synthesis
2. Design: which combines ideation, prototyping, and validation
3. Deliver: which combines developing, testing, and deploying

In line with these phases, SAP offers tools and technologies to empower customers to realize user experience on their own:

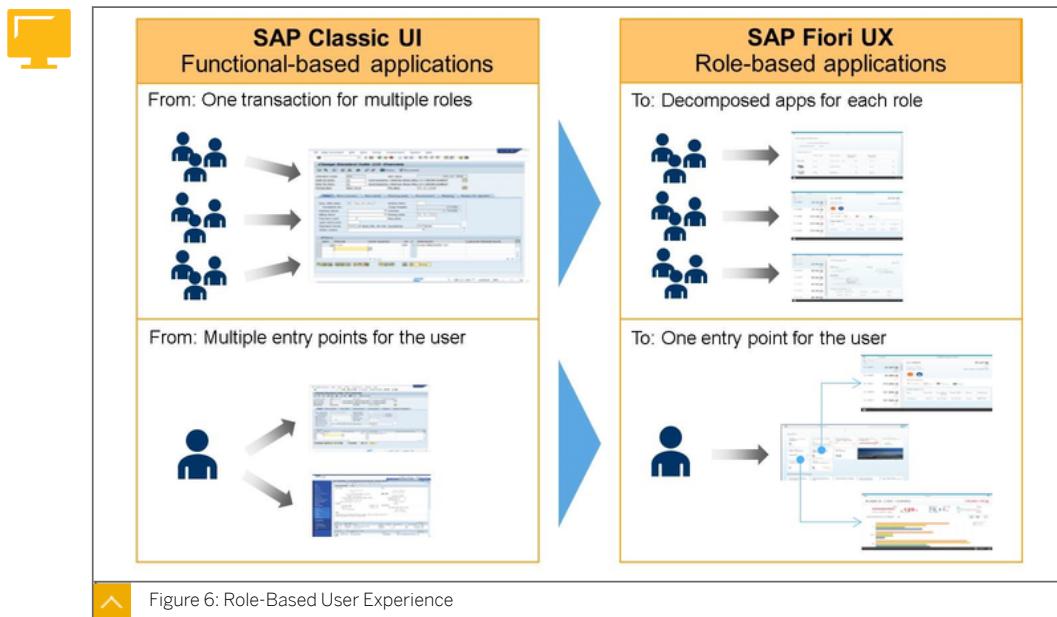
- **SAP Build** is a tool for prototyping web applications using Fiori UI elements or imported images of user interfaces. The resulting SAPUI5 project can be imported in **SAP Web IDE**, providing a stub for developers.
- **SAP Web IDE** is the Integrated Development Environment (IDE) for the web technology SAPUI5, the basis of all Fiori web apps.
- **UI Theme Designer** is a browser-based, graphical WYSIWYG (what you see is what you get) editor for designing the look of web apps.
- **SAP Screen Personas** is a web-based layer for personalization of SAP GUI Dynpro and Web Dynpro ABAP.
- **SAP Fiori launchpad** is the customizable role-based start page and frame for Fiori web apps.

In addition, SAP can provide support in all phases of the design thinking process to the extent that the customer wants. SAP User Experience Design Services offer services ranging from individual consulting for certain tools, tasks or phases to conducting the whole implementation process.

Concept



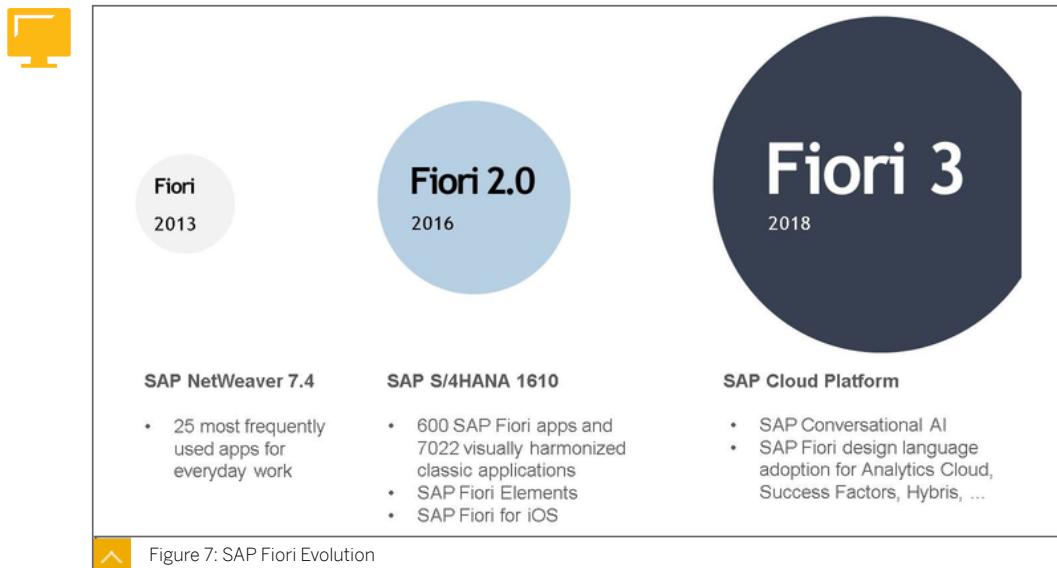
In the second dimension, SAP Fiori offers a unified user experience for a variety of clients. Users should have a consistent, coherent, simple, intuitive, and delightful user experience on all devices to be able to work better and more efficiently. The five design principles of SAP Fiori are at the core of every Fiori app to fulfill these goals. The role-based approach is, therefore, the biggest change in comparison with classic user interfaces.



A role-based user experience means that end users get all information and functions they need for their daily work - but no more. The classic SAP User Interface (UI) often offers a single complex transaction for many user roles. The tasks of a single user then make it necessary to use multiple transactions or additional applications.

Unit 1: End-User Perspective

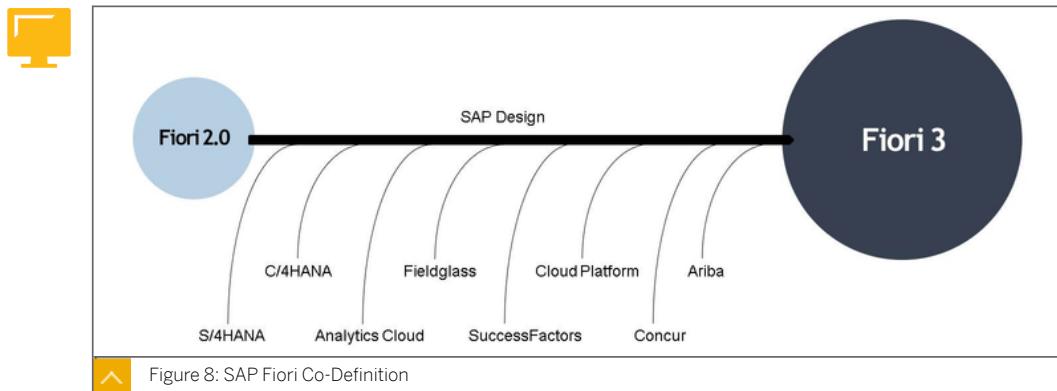
SAP Fiori decomposes these big transactions in several discrete apps suited to the user roles. All apps are connected to each other so all the tasks of the transaction are still possible. However, they are only performed if the user really wants. The *SAP Fiori launchpad* then serves as the central entry point for all of the apps of a user.

Technology

Beginning in 2013, SAP delivered the first 25 apps shortly followed by several more. These apps were delivered in “waves” extending the number of functions available to end users step by step. It was already possible to use analytical apps if an SAP HANA was available in the systems.

In October 2016, SAP shipped the SAP S/4HANA 1610 including the completely reworked SAP Fiori 2.0. All former apps could use the new design and features out-of-the-box. Although it was already possible via customizing, now SAP included thousands of classic applications automatically in SAP Fiori. With SAP Fiori elements and SAP Fiori for iOS, new ways to develop SAP Fiori applications arose and got fine-tuned in the following years.

In October 2018, the new design SAP Fiori 3 was introduced. But instead of the previous versions, SAP did not deliver everything at once. SAP chose to deliver the SAP Fiori 3 innovations in a stepwise approach.



SAP defined SAP Fiori 3 jointly with all the various product units, taking SAP Fiori 2.0 as a starting point and looking to see which aspects would need to be added to best support them all.

Figure 9: SAP Fiori 3 Outlook

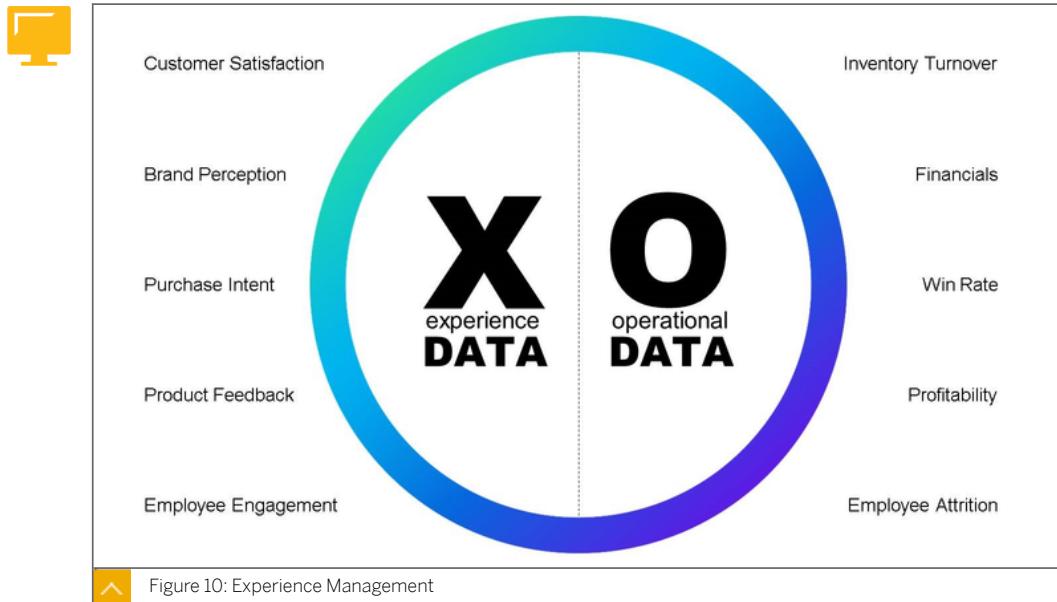
One of the most requested new aspects was being able to provide more information on home pages, which the design is supporting via cards. In addition, SAP re-imagined how aspects of intelligence such as natural language interaction and machine intelligence can become a key part of the experience by deeply integrating them into the foundation of the design to provide:

- A new theme for a fresh, modern look and feel.
- Intelligent home pages showing everything the user needs to focus on.
- Digital assistant with fully integrated conversational user experience.
- Active business situation handling, with explanations, insights and proposed actions.
- Dynamic content to support intelligent apps with embedded artificial intelligence (AI).

Unit 1: End-User Perspective

- Improved search previews, results pages, and result visualizations.
- A central inbox, also integrated with the digital assistant.

The goal of SAP Fiori 3 is to provide the perfect user interface for the Intelligent Suite fully supporting experience management.



The experience economy has changed the way businesses compete. Today, organizations must continuously listen to the beliefs, emotions, and intentions of customers, employees, suppliers, partners, and all other stakeholders. This type of data is called experience data, or X-data.

Businesses must have the ability to interpret and analyze large amounts of X-data, and understand its relationship to the operational data of the business, or O-data.

To achieve breakthrough results, businesses need to move beyond systems of record to new systems of action, specifically designed to intelligently use both X, and O-data to manage and improve the four core experiences of business: Customer, employee, product, and brand.



LESSON SUMMARY

You should now be able to:

- Explore the design of SAP Fiori

Unit 1

Lesson 2

Exploring SAP Fiori Launchpad

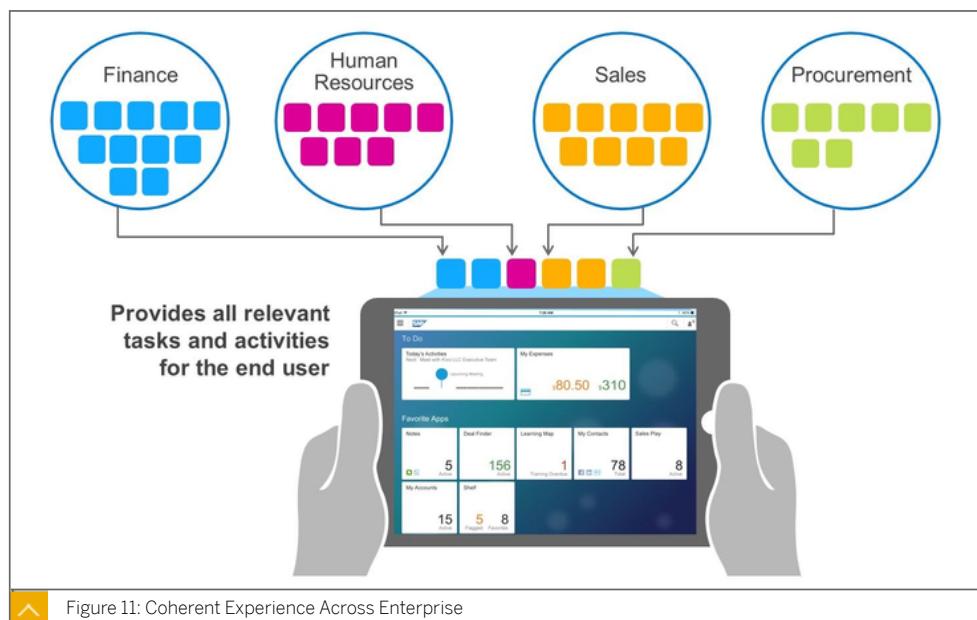


LESSON OBJECTIVES

After completing this lesson, you will be able to:

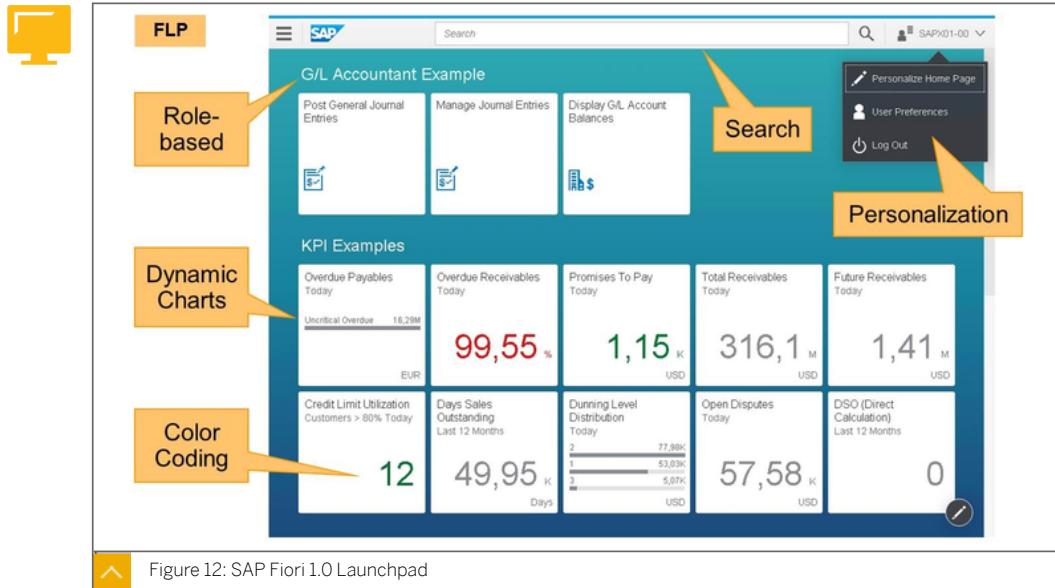
- Explore the SAP Fiori launchpad

Features

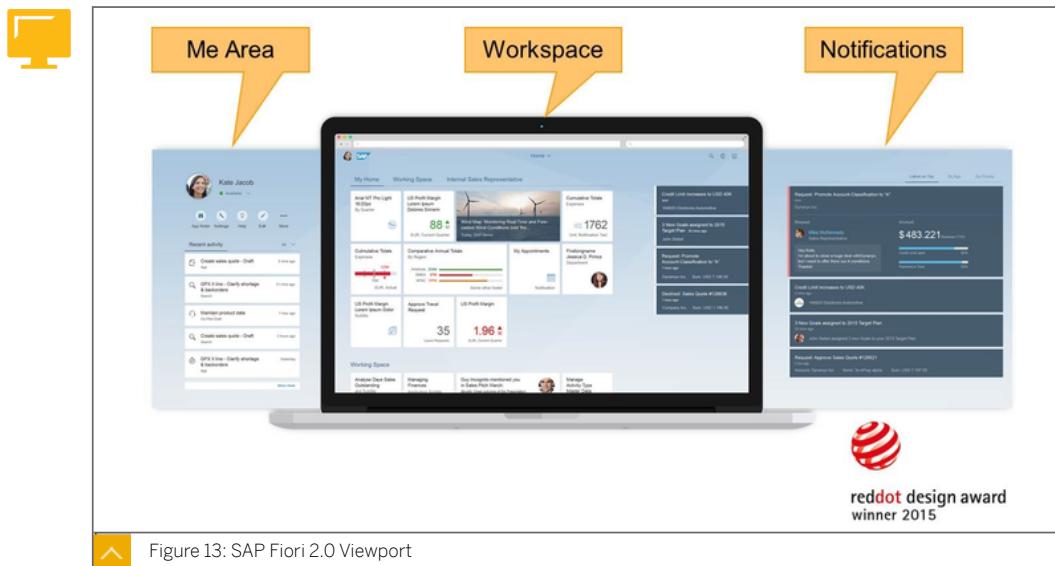


The SAP Fiori launchpad (FLP) offers a coherent user experience across different enterprise solutions by utilizing the capabilities of the user role to combine the decomposed apps in one surface. Each app represents one individual step for one specific role. Therefore, several apps combined represent a complete process covering different enterprise solutions and systems. This is a shift from monolithic solutions to activity-based apps and a role-based simplification of business processes.

Unit 1: End-User Perspective



The FLP offers several features to make it easier for end users to do their work. Depending on the role of the user, certain groups consisting of apps are displayed. Each app is represented by a tile, which can be clicked or touched to start the app. Tiles can offer summarized information about the data available for the user in the app. This information can be visualized using a simple number, color coding based on Key Performance Indicators (KPIs), and even charts showing comparisons, trends, or contributions.

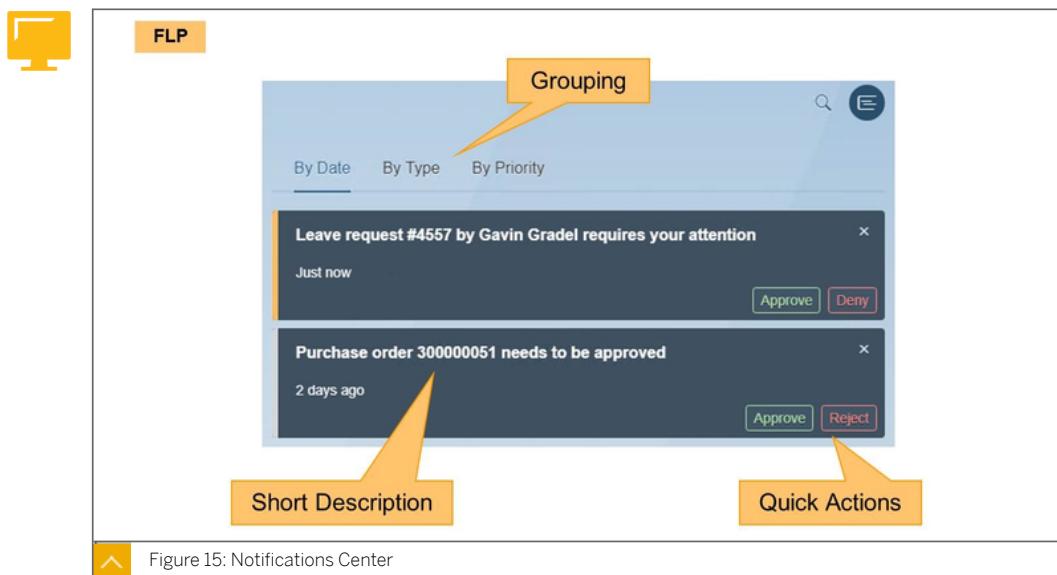


With SAP S/4HANA 1610, SAP Fiori 2.0 is shipped to customers. The SAP Fiori *viewport* now offers a viewport consisting of the *Me Area* (to the left), the *Workspace* (in the center), and the *Notifications* (to the right). The *Me Area* is an enhanced user menu providing access to frequently used apps and recent activities. The *Notifications* provide fast access to information and tasks including simple actions. The *Workspace* now offers additional

possibilities to navigate between apps like the navigation menu at the top. The whole design of SAP Fiori 2.0 was rewarded with the “Red Dot Design Award” in 2015.



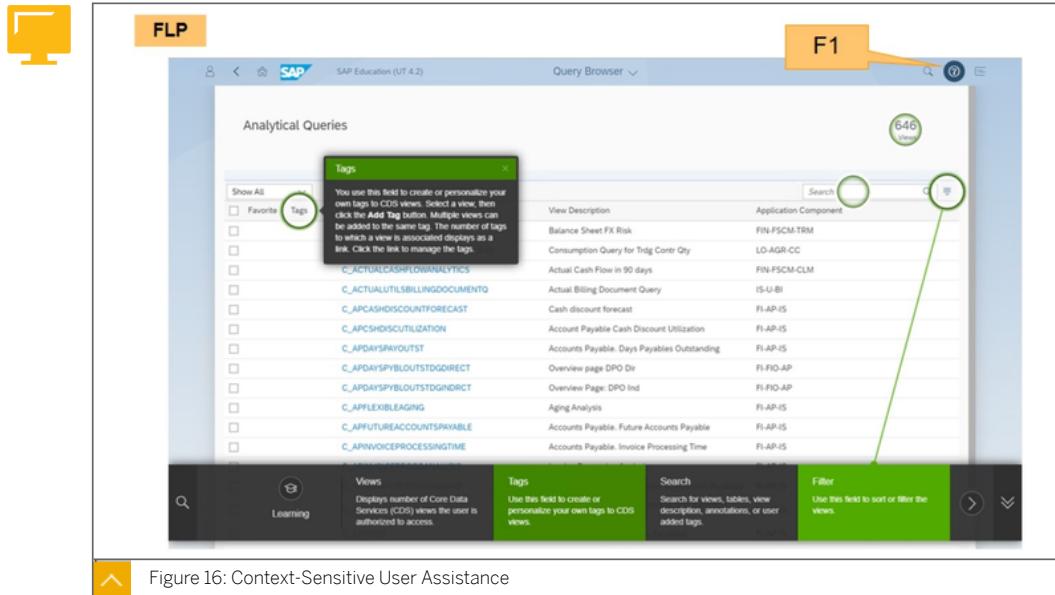
The features in the *Me Area* can be configured per user role. The user area can show a photo of the user and include collaboration features via SAP Jam. The options in personalization can be reduced to disallowing everything or extended to adapting apps at runtime. The recent activities and frequently used apps can also be deactivated.



The notifications center can be enabled per user role in the *SAP Fiori launchpad*. Notifications are created by notification providers and can be based on SAP Workflow or the ABAP

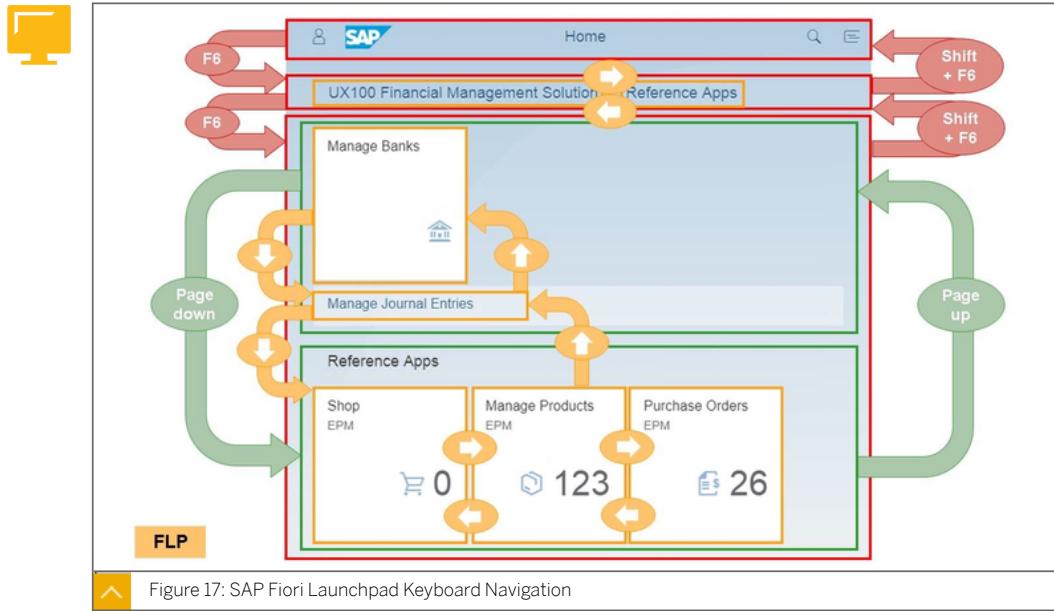
notification framework. They can offer quick actions and are able to start a suitable app showing details for the notification topic.

Accessibility



The SAP *Fiori launchpad* offers help functionality that includes a guided tour for an app. This context-sensitive user assistance can be started via the question mark in the upper right corner or by pressing F1. A help panel consisting of tiles overlays the app at the bottom as well as bubbles pointing out the key elements of the app. By selecting a tile or its bubble, an information dialog box appears offering more information about the functionality.

The help content for this user assistance is provided by the SAP Content Server hosted by SAP, an SAP Enable Now Manager hosted by the customer, or a mixture of both. This also includes access to learning material available in the Learning Center.



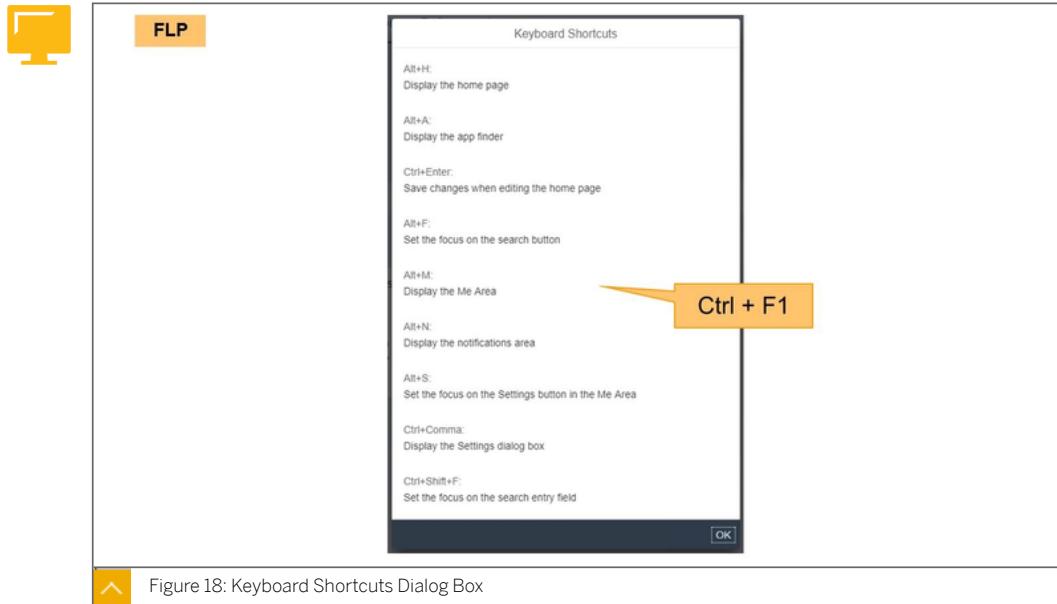
The SAP Fiori *launchpad* can be fully operated via a keyboard. Its workspace consists of blocks, groups, and tiles/links. There are four blocks:

- Header
- Anchor bar
- Groups
- Footer (optional)

Switching between blocks is done by pressing F6 for downward and Shift + F6 for upward.

Switching between groups is done by pressing Page Up and Page Down, and switching through tiles or links in the anchor bar or groups is done via the arrow keys. It is also possible to use Tab to cycle through all elements.

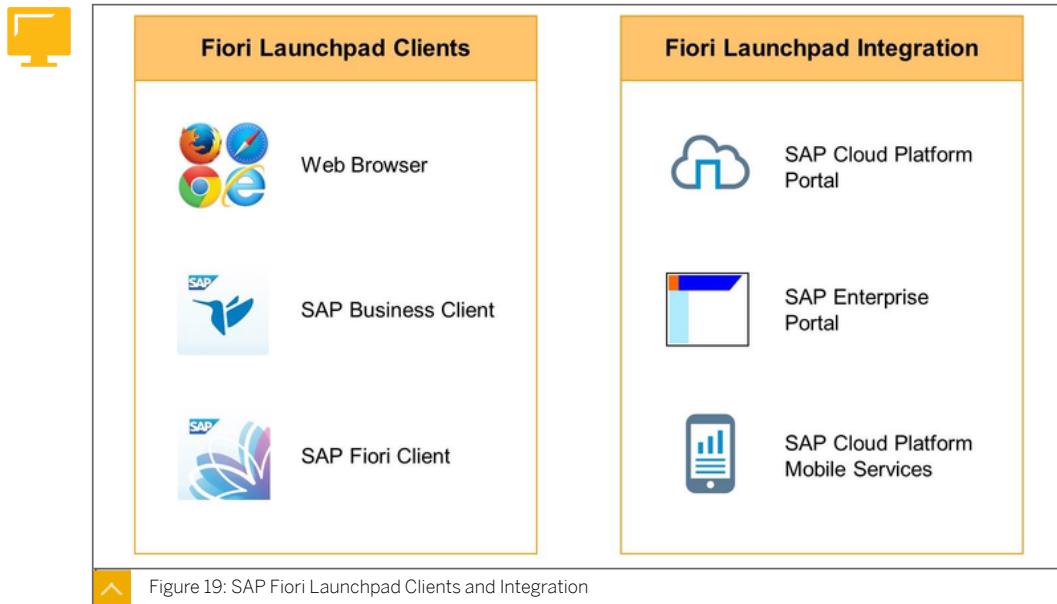
Unit 1: End-User Perspective



A list of shortcuts can be shown by pressing Ctrl + F1. Some of these shortcuts may interfere with shortcuts of the browser, depending on its brand and settings.

Additional usability features include a high contrast theme recommended for users who are visually impaired. The standard High-Contrast Black (HCB) theme for SAPUI5 is a predefined theme that is distinguished for this purpose. A screen reader can also be used to read out the contents of the user interface. Labels, headings, and descriptions help describe the contents and visual elements of an application.

Client Integration



There are three clients available to access the FLP:

Web Browser

Any HTML5-ready browser can be used.

SAP Business Client

As of SAP Business Client 6.0, it is possible to create SAP Fiori launchpad system connections.

SAP Fiori Client

On mobile devices from Apple, Google, and Microsoft, the SAP Fiori Client is available.

It is also possible to integrate the FLP in other system areas to add additional values like logon features or increased distribution range. Details and benefits of these integration options will be discussed later or considered in more depth in other trainings:

- SAP Cloud Platform Portal
- SAP Enterprise Portal
- SAP Cloud Platform Mobile Services

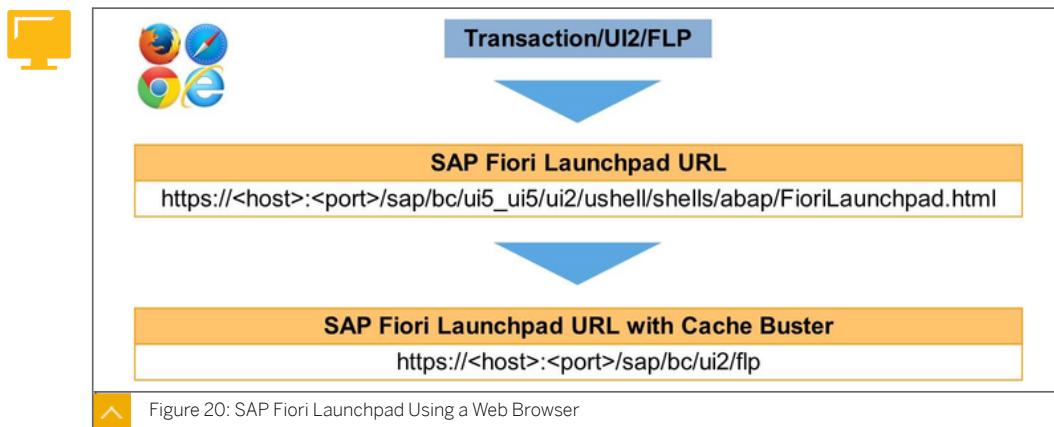


Figure 20: SAP Fiori Launchpad Using a Web Browser

The easiest way to start an FLP is to enter the URL <https://<host>:<port>/sap/bc/ui5/ui2/ushell/shells/abap/FioriLaunchpad.html> in a HTML5-ready browser such as Apple Safari, Google Chrome, Microsoft Internet Explorer, or Mozilla Firefox. This URL has been available since the beginning of SAP Fiori. Over time, SAP added additional options to start the launchpad.

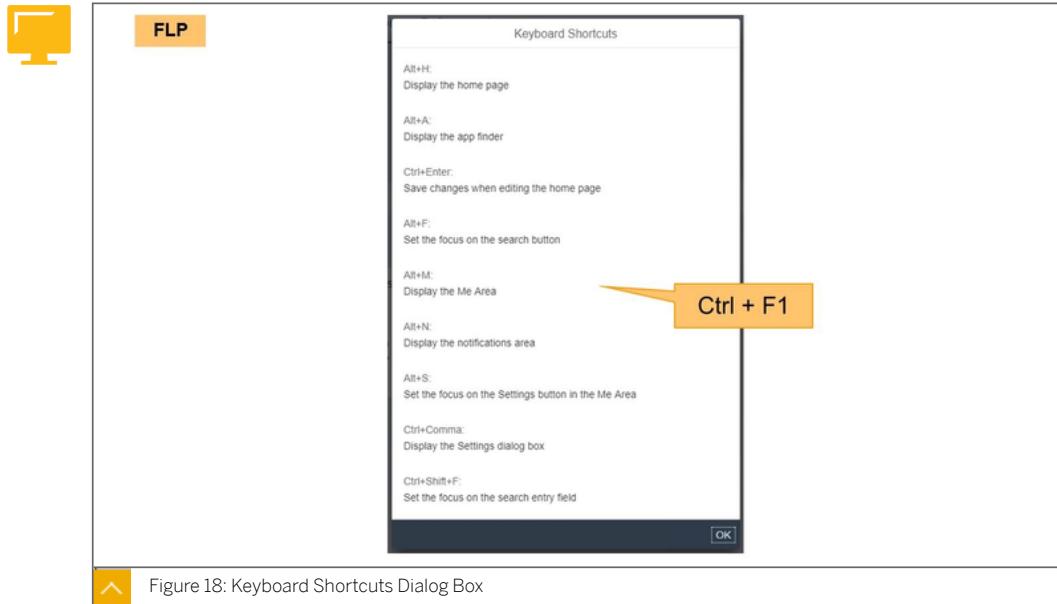
Many customers are used to accessing SAP software via a logon to an Application Server ABAP (AS ABAP). Therefore, the transaction /UI2/FLP was added. This is able to log on the user automatically with the credentials they used to log on to the AS ABAP. This approach is meant to be used by users working in both worlds, that is, ABAP transactions and SAP Fiori web apps, if no business client is available.



Note:

The transaction /UI2/FLP starts the FLP via HTTP by default. An entry in the database table HTTPURLLOC can be used to call a reverse proxy like the SAP Web Dispatcher via HTTPS.

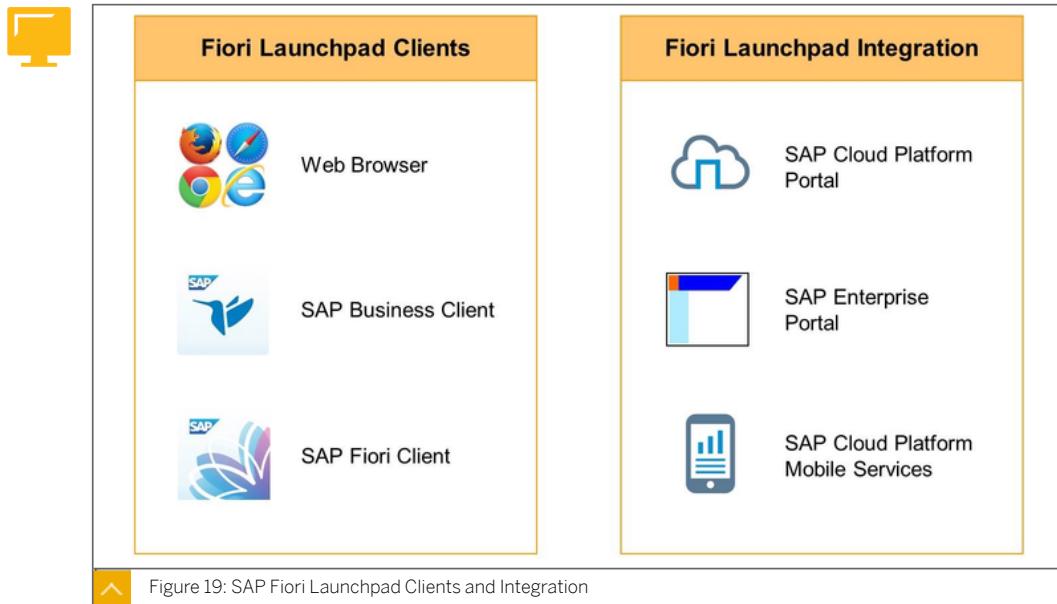
Unit 1: End-User Perspective



A list of shortcuts can be shown by pressing Ctrl + F1. Some of these shortcuts may interfere with shortcuts of the browser, depending on its brand and settings.

Additional usability features include a high contrast theme recommended for users who are visually impaired. The standard High-Contrast Black (HCB) theme for SAPUI5 is a predefined theme that is distinguished for this purpose. A screen reader can also be used to read out the contents of the user interface. Labels, headings, and descriptions help describe the contents and visual elements of an application.

Client Integration



There are three clients available to access the FLP:

Web Browser

Any HTML5-ready browser can be used.

SAP Business Client

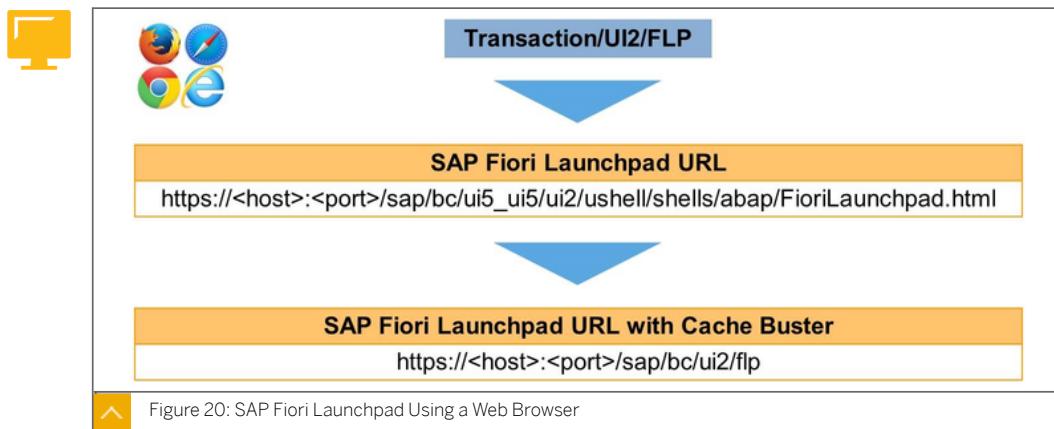
As of SAP Business Client 6.0, it is possible to create SAP Fiori launchpad system connections.

SAP Fiori Client

On mobile devices from Apple, Google, and Microsoft, the SAP Fiori Client is available.

It is also possible to integrate the FLP in other system areas to add additional values like logon features or increased distribution range. Details and benefits of these integration options will be discussed later or considered in more depth in other trainings:

- SAP Cloud Platform Portal
- SAP Enterprise Portal
- SAP Cloud Platform Mobile Services



The easiest way to start an FLP is to enter the URL <https://<host>:<port>/sap/bc/ui5/ui2/ushell/shells/abap/FioriLaunchpad.html> in a HTML5-ready browser such as Apple Safari, Google Chrome, Microsoft Internet Explorer, or Mozilla Firefox. This URL has been available since the beginning of SAP Fiori. Over time, SAP added additional options to start the launchpad.

Many customers are used to accessing SAP software via a logon to an Application Server ABAP (AS ABAP). Therefore, the transaction /UI2/FLP was added. This is able to log on the user automatically with the credentials they used to log on to the AS ABAP. This approach is meant to be used by users working in both worlds, that is, ABAP transactions and SAP Fiori web apps, if no business client is available.



Note:

The transaction /UI2/FLP starts the FLP via HTTP by default. An entry in the database table HTTPURLLOC can be used to call a reverse proxy like the SAP Web Dispatcher via HTTPS.

Another option to start the FLP is by using the URL <https://<host>:<port>/sap/bc/ui2/flp>. This URL is much shorter than the original one so it is easier to memorize. Even more important is the cache buster feature. This technique causes web browsers to load content from the server rather than from the browser cache if activated. The cache buster for SAP Fiori uses versioned URLs containing tokens to signal the browser that new resources are available on the server. Instead of forbidding caching or setting a lifetime for the resources, the system invalidates the cache only when resources are actually updated on the server. When activating the cache buster in a system via the official task list, an external alias for the original URL that points to the new URL is created. So, the end user does not need to change the older URL and still has the benefit of the cache buster feature.

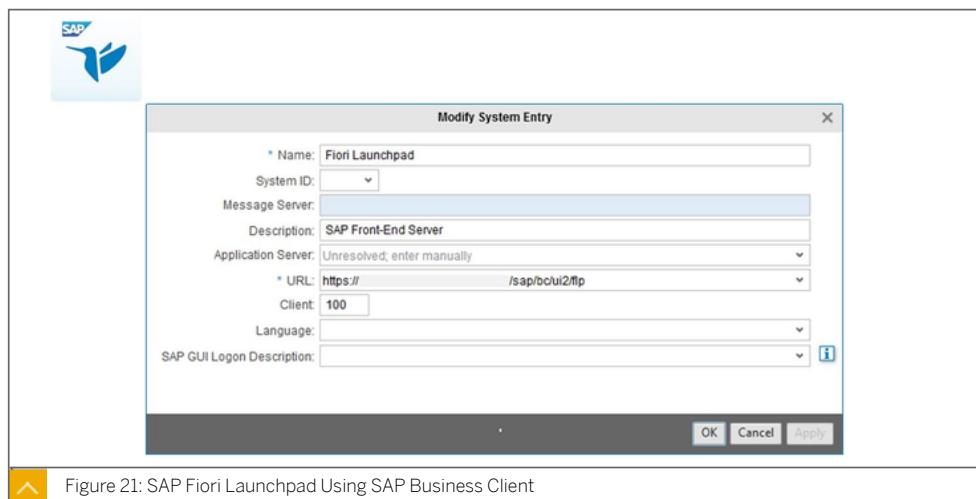


Figure 21: SAP Fiori Launchpad Using SAP Business Client

SAP Business Client makes it the possible to access SAP GUI and web applications in one client software. Therefore, in *SAP Business Client 6.0*, the possibility to add system connections for FLP was introduced. The benefit is that the end user only needs one tool to access all the functions of an AS ABAP. It is even possible to start ABAP transactions in the FLP, which opens a new SAP GUI tab in the *SAP Business Client*. SAP Logon is completely integrated in *SAP Business Client*.



Hint:

For *SAP Business Client 6.0*, the term NetWeaver was dropped from the name. Previous releases are still called *SAP NetWeaver Business Client (NWBC)*.

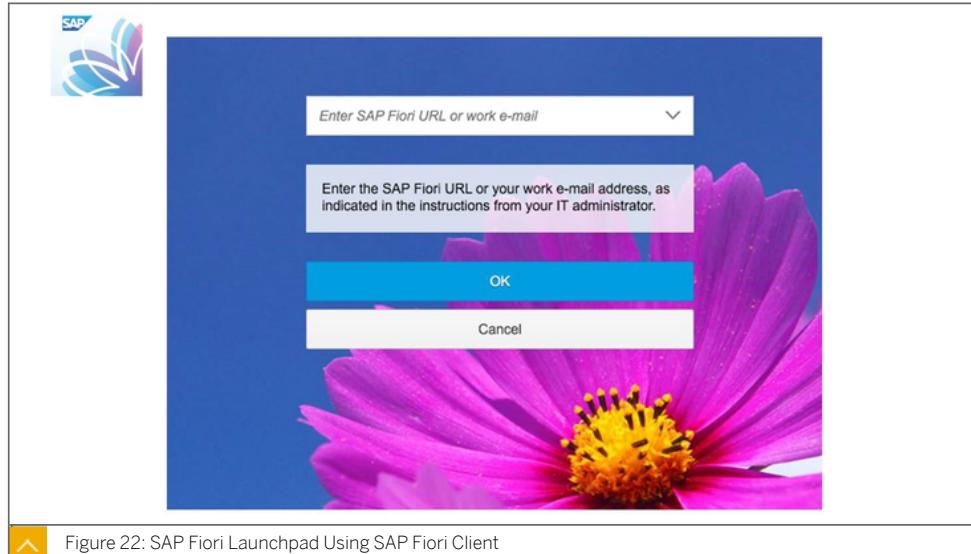


Figure 22: SAP Fiori Launchpad Using SAP Fiori Client

For mobile devices, it is possible to access the FLP using the native web browser of the device. But the web browser still handles the FLP as a web page showing a frame with a URL header and navigation buttons. To get a real mobile experience when using SAP Fiori, accessing the FLP via the *SAP Fiori Client* is recommended. All browser frames are removed, showing the FLP in full-screen like any other mobile app. In addition, a delta caching is in place, speeding up the handling of SAP Fiori on the device, and an attachment viewer is available for the most common files used in Fiori apps. *SAP Fiori Client* is available free of charge in the native stores of Apple, Google, and Microsoft for tablets and smartphones.

When combining *SAP Fiori Client* with SAP Cloud Platform Mobile Services, additional benefits such as single-sign on and offline data are available, depending on the individual Fiori app.



LESSON SUMMARY

You should now be able to:

- Explore the SAP Fiori launchpad

Unit 1

Lesson 3

Personalizing SAP Fiori



LESSON OBJECTIVES

After completing this lesson, you will be able to:

- Personalize SAP Fiori

Personalization

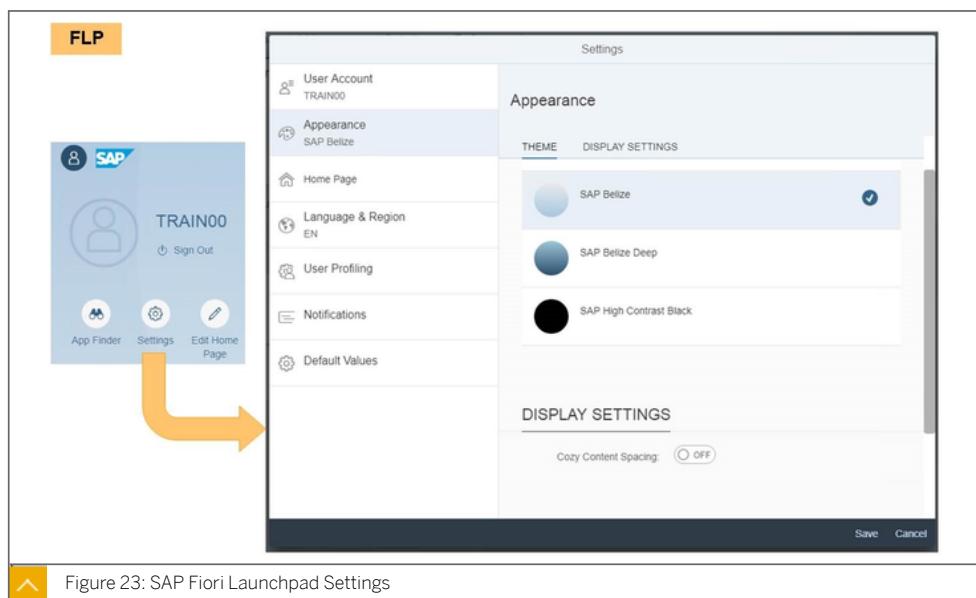
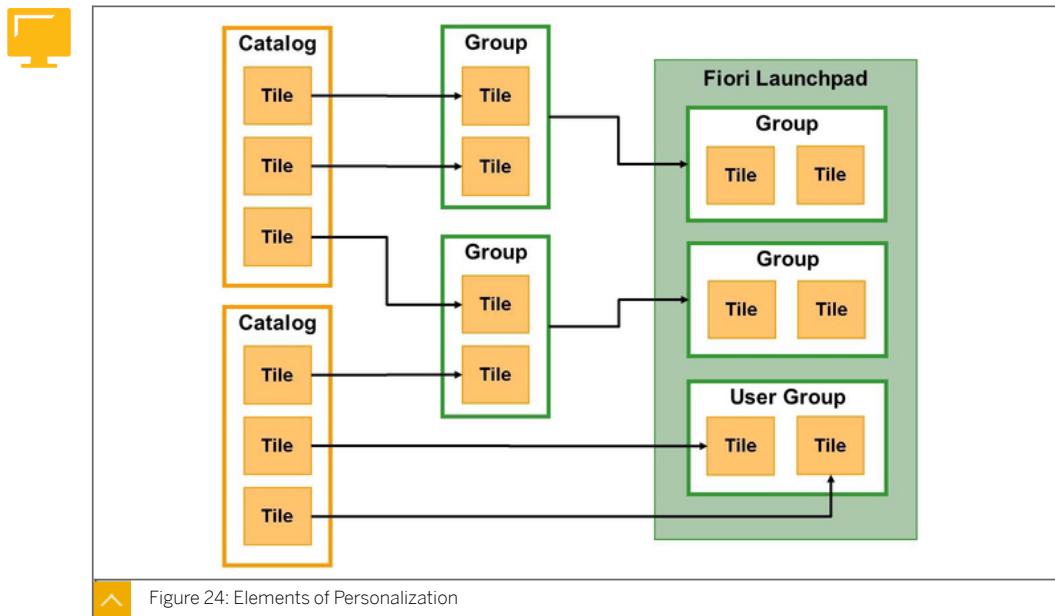


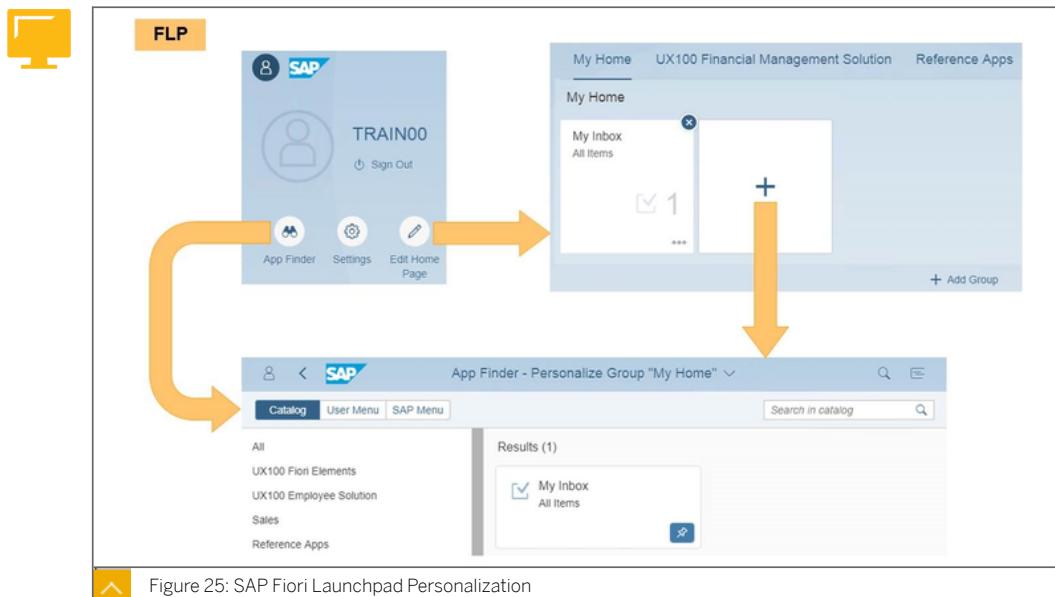
Figure 23: SAP Fiori Launchpad Settings

End users can personalize their own variant of the SAP Fiori launchpad (FLP). The settings can be accessed via the *Me Area*. It is possible to get information about the user account, home page, language, and region. Depending on the configuration of the FLP for the user, the following settings can be changed:

- Selection of design theme
- Activation of user profiling
- Appearance and behavior of notifications
- Maintenance of default values



As well as changing general settings like language, users can rearrange groups and tiles. create new groups, and add or delete tiles to or from an existing group. Tiles are organized in catalogs in the system. These catalogs hold all technical information to start an application. To show a tile in the FLP, it must be embedded in a group. Groups can be created centrally in the system and then added to the launchpad, or the user can create an own group and add tiles from catalogs.



To enter the action mode for personalization, a user needs to choose *Edit Home Page* in the *Me Area* of the FLP. In this mode, tiles can be removed from existing groups and new groups can be created and rearranged. When adding a new tile to a group in the action mode or

Unit 1: End-User Perspective

choosing *App Finder* in the *Me Area*, the app finder is shown. Here, the user can choose tiles from all catalogs assigned to their user role.

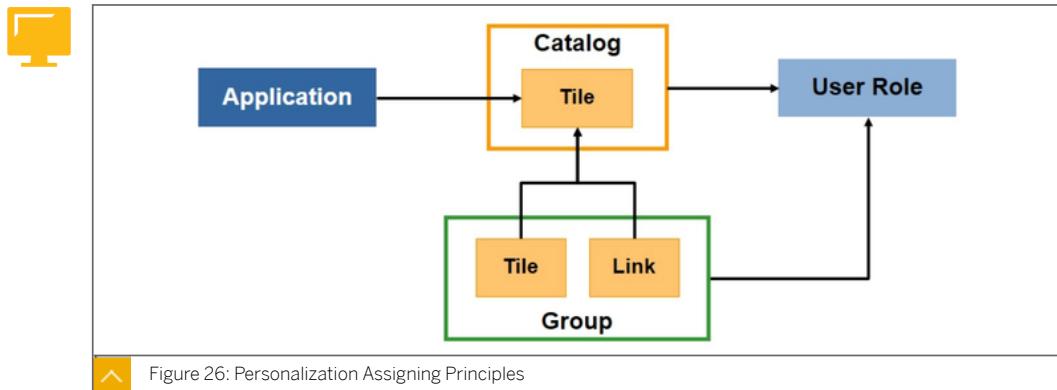


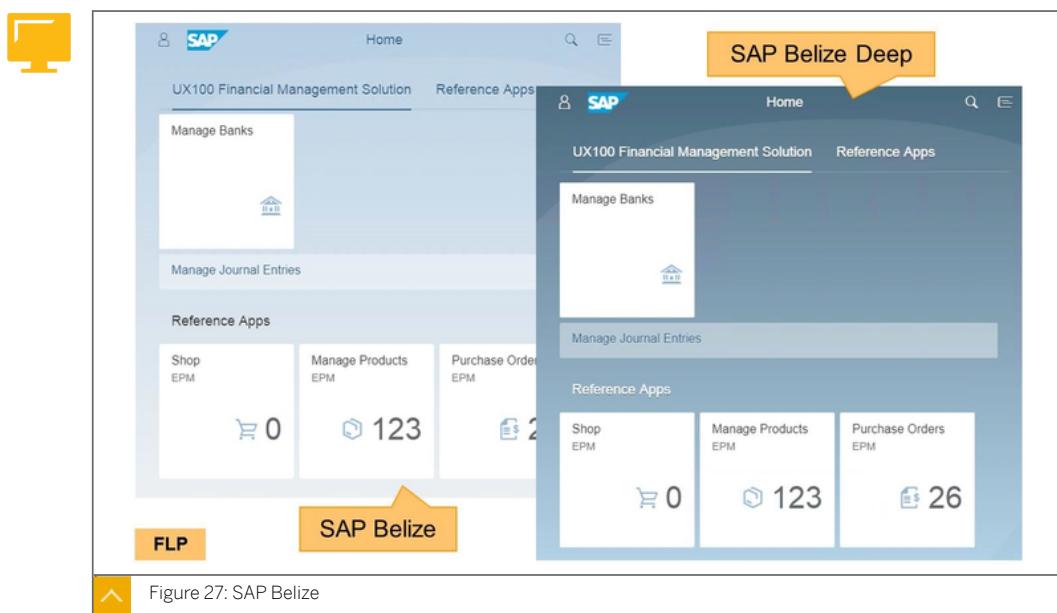
Figure 26: Personalization Assigning Principles

A tile connects the Fiori launchpad with the app implementation in the system. This includes starting the correct app with the correct parameters and the information shown on the tile. Tiles are created in catalogs that collect all apps of one solution area. Apps, tiles, and catalogs are delivered by SAP, but they can also be created by customers.

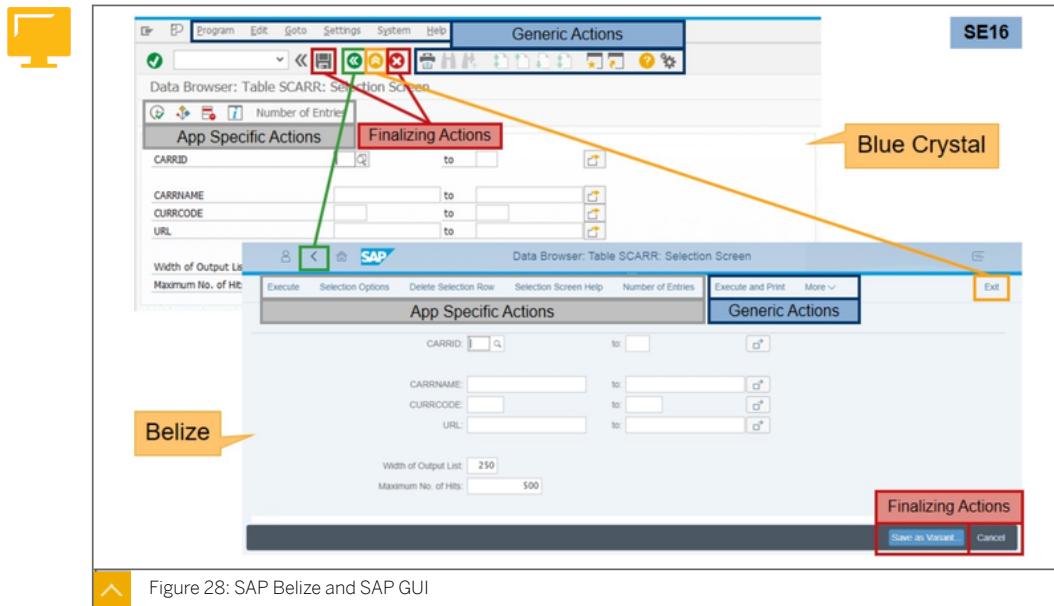
Tiles of catalogs can be assigned to groups and then visualized as tiles or links. Groups do not add any additional settings to the tiles and have no direct connection to the apps. Both catalogs and groups can then be assigned to user roles. This works like adding transactions to user roles and grants the user role access to all tiles of the catalog or group.

SAP also delivers template groups and roles. However, these should be adapted to the needs of the customer, or, to be more precise, the needs of the users of the customer.

Classic UI Integration



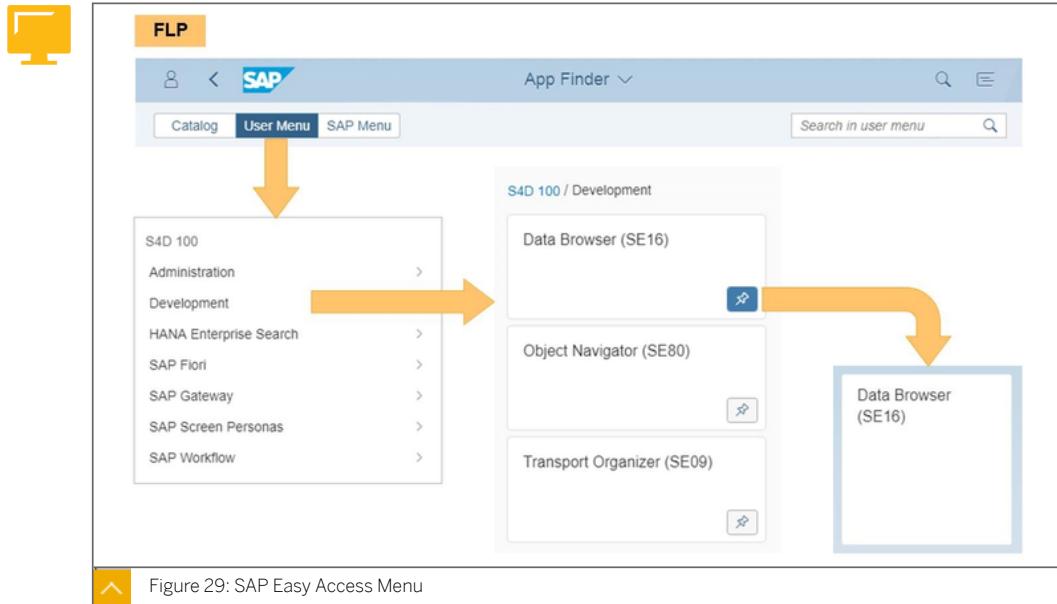
The design for FLP used in SAP Fiori 1.0 is SAP Blue Crystal. In SAP Fiori 2.0, this design can no longer be used. SAP Belize (Deep) is the new design offering additional features such as more visual layers, animations, and more details in micro interactions and typography.



Most importantly, SAP Belize is not only a design for SAPUI5, but also for *Web Dynpro ABAP* (WDA) and SAP GUI in particular. For WDA and SAP GUI, this means that well-known functions like **SAVE** or **BACK** change their position according to the rules of SAP Fiori. Finalizing actions are defined in the SAP Fiori design guidelines to be visible in the lower-right corner or the **BACK** button has to be in the upper left corner next to **Home**. This all works out-of-the-box by using the SAP Belize theme with SAP GUI.

SAP Belize is available as of SAP GUI for HTML with SAP Kernel 7.49 and SAP GUI for Windows 7.50, but only if the user connects to an SAP S/4HANA 1610 or a newer SAP S/4HANA release. With SAP GUI for Windows 7.60, SAP Belize is the default design for all SAP products. A full documentation of all designs for SAP GUI and their prerequisites is available in SAP note [710719](#).

Unit 1: End-User Perspective



Adding transactions to the FLP has been possible since the release of SAP Fiori 1.0. However, in SAP Fiori 2.0, with the automatic adaptation of the design and behavior of SAP GUI to SAP Fiori, it is even more attractive. The easiest way to add transactions is by accessing the user or SAP menu in a mapped ABAP system. The mapping has to be done by an administrator, but once this is complete, creating a tile for a transaction is as easy as adding any other app to the FLP.



LESSON SUMMARY

You should now be able to:

- Personalize SAP Fiori

Unit 1

Lesson 4

Exploring SAP Fiori Data Handling

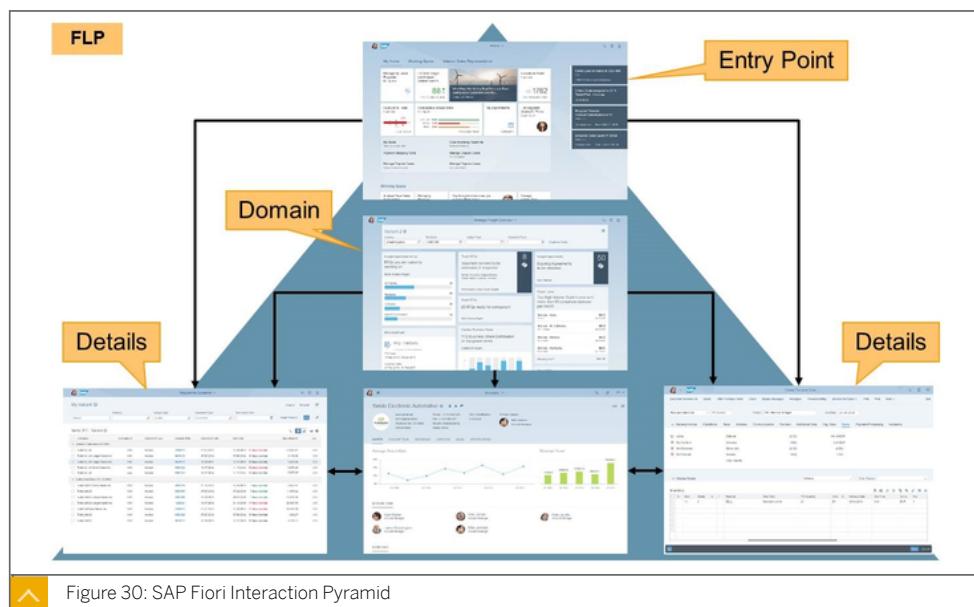


LESSON OBJECTIVES

After completing this lesson, you will be able to:

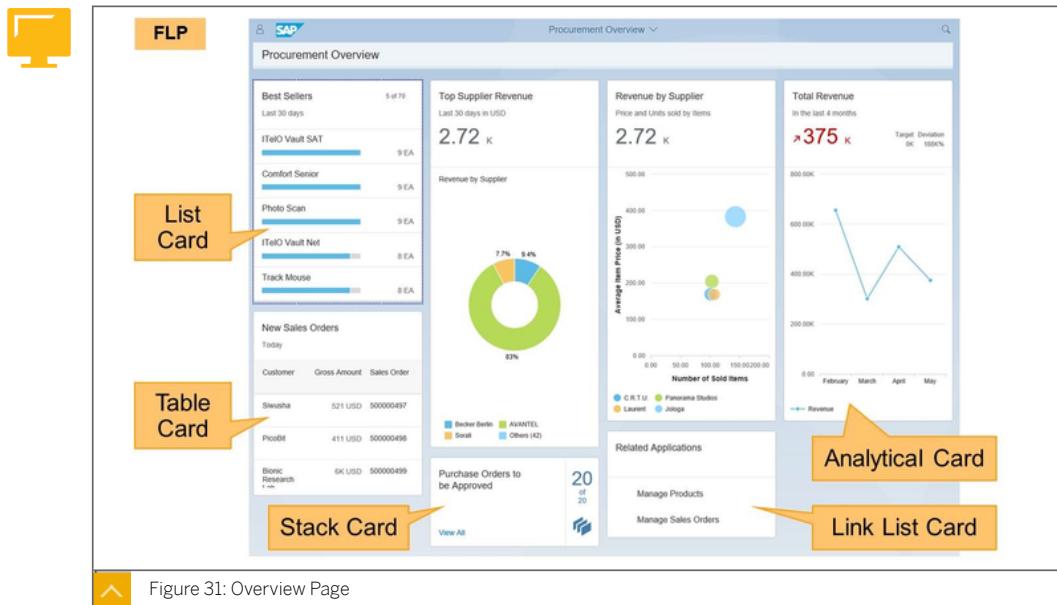
- Explore SAP Fiori data handling

Interaction Floorplan

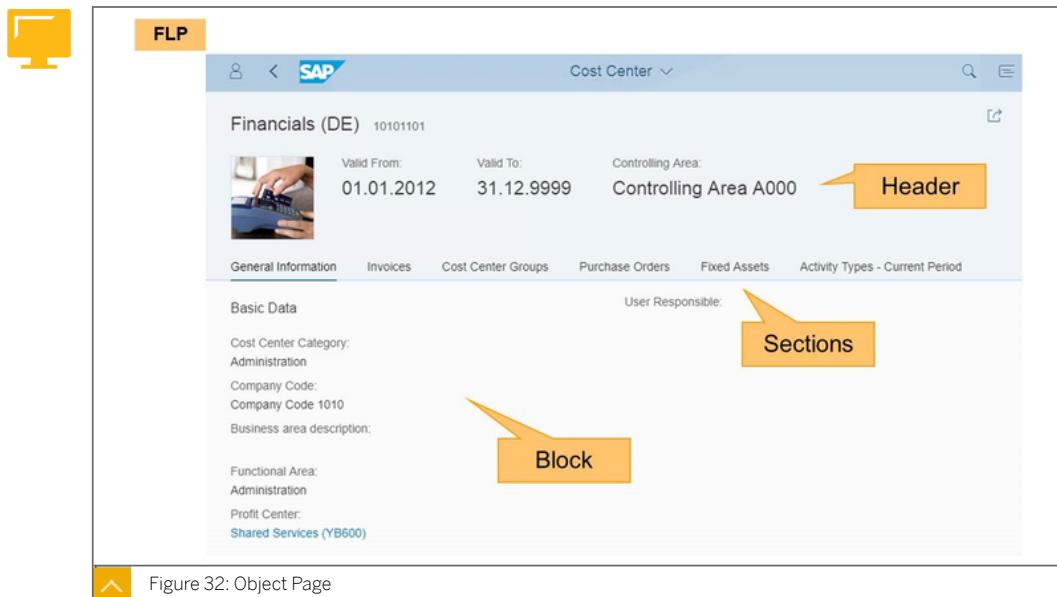


The SAP Fiori interaction pyramid visualizes the connection between different layers of apps. The overall content scope becomes more focused with each interaction step. The FLP used as an entry point contains all of the favorite apps of a user. An overview page focuses on the domain-specific key tasks and contains only the most frequently used apps for a role. A pool of apps show the details and offer actions for certain business objects with object pages standing in the center of cross-app navigation.

Unit 1: End-User Perspective



The overview page (OVP) is a data-driven SAP Fiori app type and floorplan that provides all the information a user needs in a single page, based on the specific domain or role of the user. It allows the user to focus on the most important tasks and view, filter, and react to information quickly. Each task or topic is represented by a card (or content container). Different types of cards allow the visualization of information in an attractive and efficient way.



The object page allows the user to display, create, or edit a business object. It comes with a flexible header; a choice of anchor or tab navigation; and a flexible, responsive layout. These

features make it adaptable for a wide range of use cases. The object page, like the overview page, is based on *SAP Fiori elements* technology and uses an annotated view of app data.

SAP Fiori Search

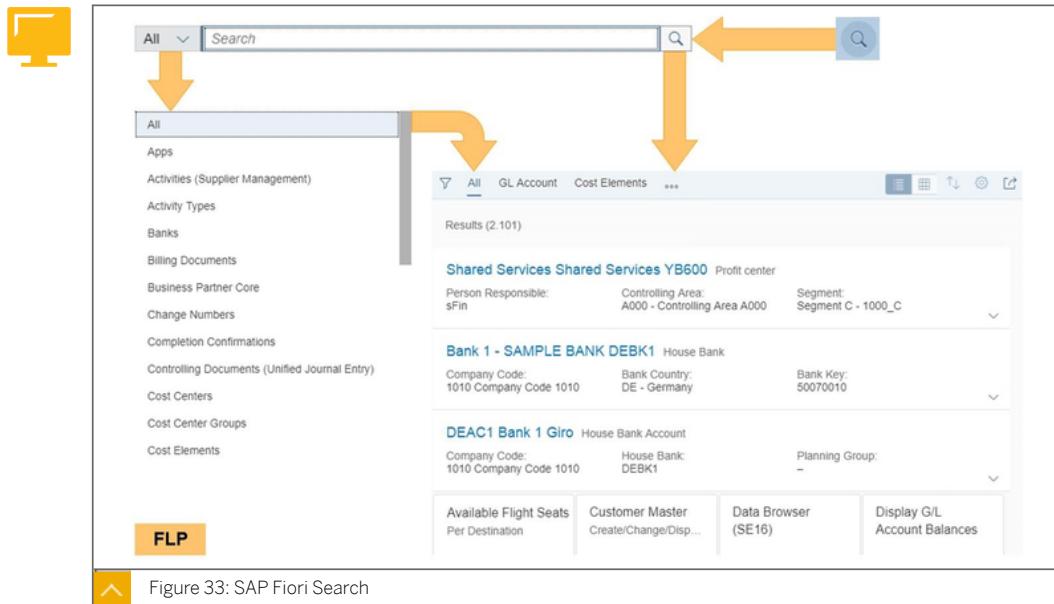


Figure 33: SAP Fiori Search

SAP Fiori offers a powerful search to find contents in the *SAP Fiori launchpad (FLP)*. Using any database, it is possible to search for text fragments of Fiori tiles. However, when using SAP HANA, all business data of the back-end server (BES) can be searched quickly.

To access the SAP Fiori search, open the *Search* field at the top of the FLP, select a data area, and enter any value of a dataset in the BES. You may use the wildcard * as a placeholder.

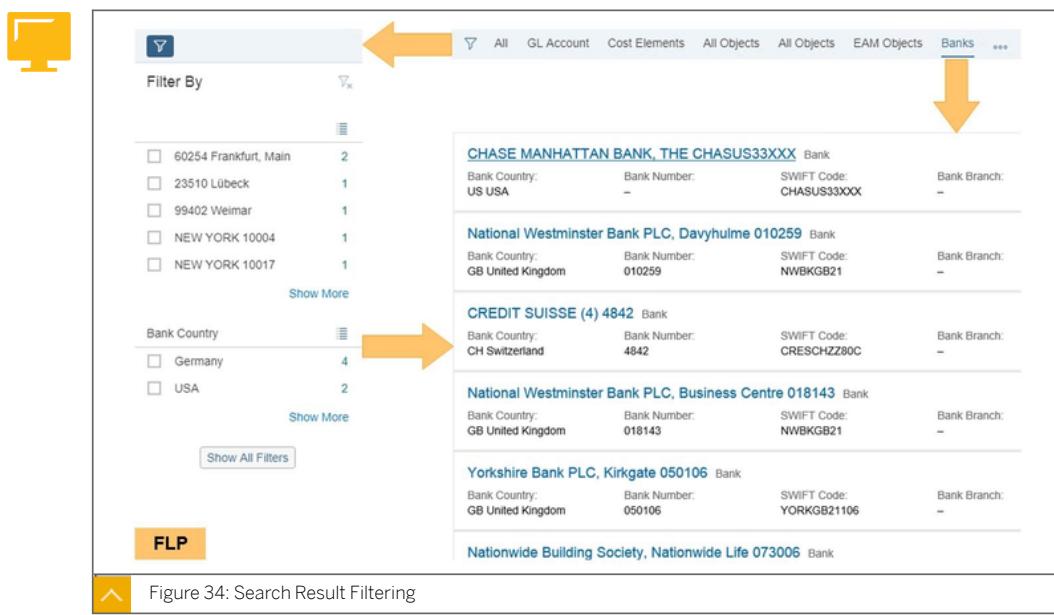
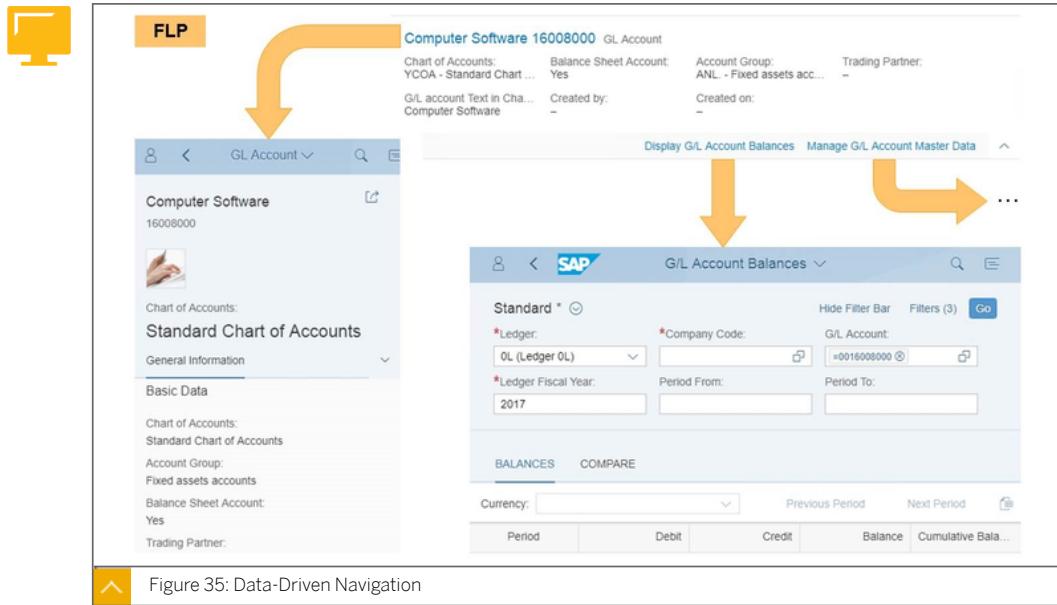


Figure 34: Search Result Filtering

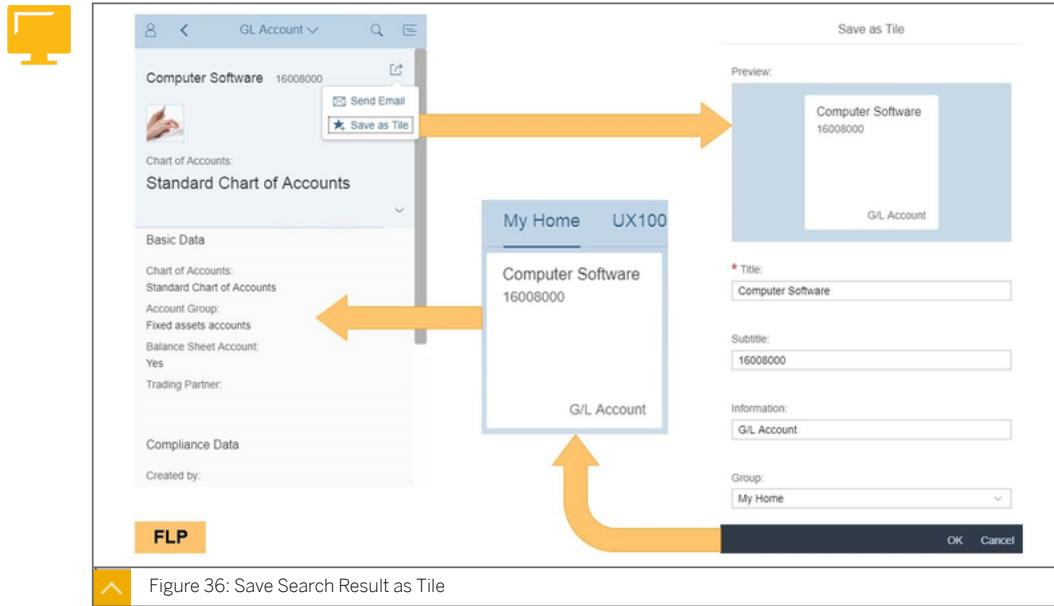
Unit 1: End-User Perspective

The search result offers multiple ways to filter the data. First, the data area can be set by using the anchor bar at the top or selecting a filter option for searching. Second, if an area is set, additional data specific filters appear in the filter options showing a glimpse of the available data. Every change of the filter leads immediately to an adaption of the search result.



Aside from using tiles to start an app, a more direct way to display data in SAP Fiori is to use one of the numerous links available in SAP Fiori search. If you click the header of a search result entry, a fact sheet or object page app visualizing the entry in all its details opens. The visualization of the data object can be adapted, additional information of the data object can be retrieved, and related data objects can be accessed.

In addition, apps related to the data object can be started to perform tasks such as creating, updating, and deleting. These apps need to be part of the user role, but they do not need to be visible in the FLP. It is possible to work with data objects in SAP Fiori without using a single tile. The data leads to the apps.



Tiles are shortcuts to start apps. This includes the result of search requests and all its visualization adaption. All fact sheet and object page apps, as well as many other apps, offer the possibility to save the app status and visualized data as a tile in the FLP for later access. This is part of the personalization of the FLP for the user.



LESSON SUMMARY

You should now be able to:

- Explore SAP Fiori data handling

Unit 1

Learning Assessment - Answers

1. Which dimensions define SAP Fiori?

Choose the correct answers.

- A Concept
- B Design
- C Performance
- D Technology

Correct! The following dimensions define SAP Fiori: Concept, Design, and Technology

2. What are the main principles of SAP Fiori ?

Choose the correct answers.

- A Role-based
- B Adaptive
- C Simple
- D Coherent
- E Delightful

Correct! All of the above are main principles of SAP Fiori.

3. Which clients can be used for SAP Fiori?

Choose the correct answers.

- A Web Browser
- B SAP Logon
- C SAP Fiori Client
- D SAP Easy Access
- E SAP Business Client

Correct! The following clients can be used in SAP Fiori: Web Browser, SAP Fiori Client, and SAP Business Client.

4. Which platforms support the integration of SAP Fiori?

Choose the correct answers.

- A SAP Enterprise Portal
- B SAP Process Integration
- C SAP HANA Enterprise Cloud
- D SAP Cloud Platform Portal
- E SAP Cloud Platform Mobile Services

Correct! The following platforms support the integration of SAP Fiori: SAP Enterprise Portal, SAP Cloud Platform Portal, and SAP Cloud Platform Mobile Services.

5. Which personalization elements can be assigned to user roles?

Choose the correct answers.

- A Fiori Group
- B Fiori Group Tile
- C Fiori Catalog
- D Fiori Catalog Tile

Correct! The following personalization elements can be assigned to user roles: Fiori Group and Fiori Catalog.

6. Which personalization element refers to the application?

Choose the correct answer.

- A Fiori Group
- B Fiori Group Tile
- C Fiori Catalog
- D Fiori Catalog Tile

Correct! The personalization element Fiori Catalog Tile refers to the application.

7. What is meant by the term data-driven navigation?

Data objects offer navigation links to dependent data objects and functions.

**Lesson 1**

Explaining User Interfaces	33
----------------------------	----

Lesson 2

Explaining Data Services	39
--------------------------	----

Lesson 3

Explaining Application Types	47
------------------------------	----

UNIT OBJECTIVES

- Explain user interfaces
- Explain data services
- Explain application types
- Using SAP Fiori App Recommendations

Unit 2

Lesson 1

Explaining User Interfaces



LESSON OBJECTIVES

After completing this lesson, you will be able to:

- Explain user interfaces

UI Technologies

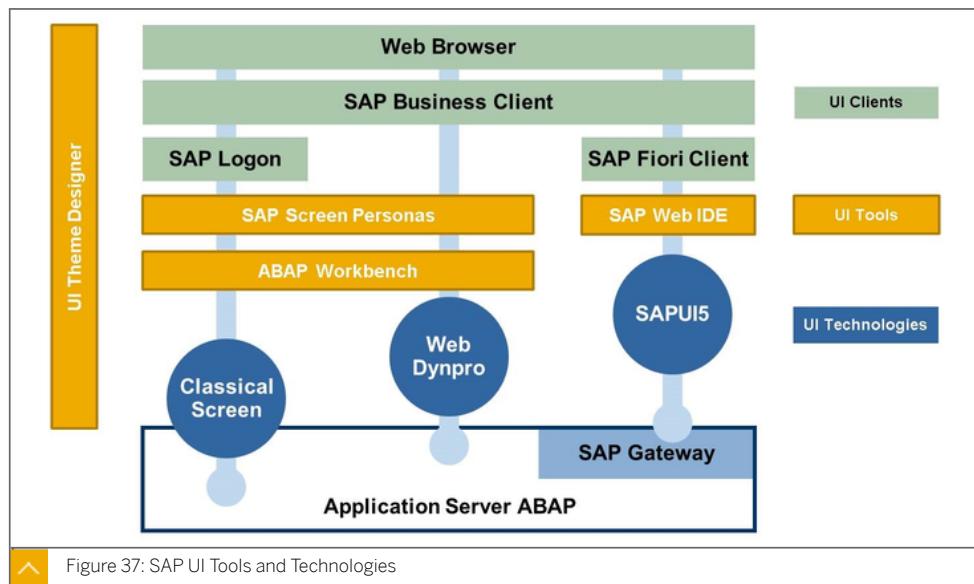


Figure 37: SAP UI Tools and Technologies

There are three main UI technologies used in current SAP applications. Classic screens are developed and run in ABAP. Using the *SAP GUI for HTML*, it is also possible to run classic screens in a browser environment. *Web Dynpro ABAP* is developed in ABAP but uses JavaScript in the browser to communicate with the ABAP system. *SAPUI5* is pure JavaScript and communicates with ABAP using SAP Gateway. There are tools available to adapt the applications to the needs of current end users:

SAP Screen Personas

- Classic screens (including enjoy controls like the *ABAP List Viewer [ALV]*)
- *Web Dynpro* (including *Floorplan Manager [FPM]*)

SAP Web IDE

- *SAPUI5* (including Fiori)

All UI technologies can be consumed using a web browser or the *SAP Business Client*. Classic screens are displayed using SAP GUI for Windows in the *Business Client* as well as SAP Logon. The *SAP Fiori Client* offers a better user experience when consuming SAPUI5 on mobile devices. When it comes to the design of user interface, one tool handles all technologies: the *UI Theme Designer*.



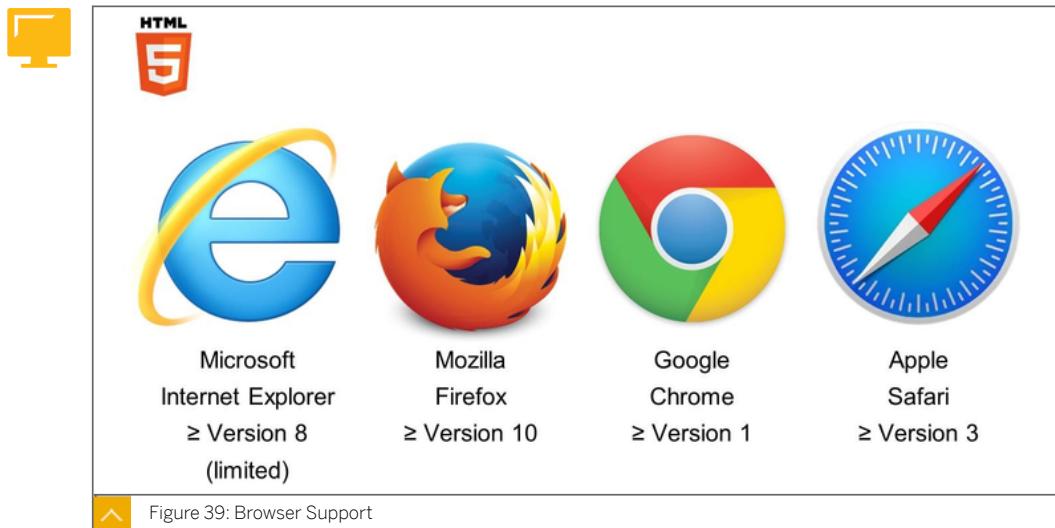
The majority of SAP Fiori apps are web apps built using SAPUI5 as UI technology. SAPUI5 is based on HTML5 and can be consumed on every device using a browser. The recommended development environment for SAPUI5 is the *SAP Web IDE*.

SAP Fiori also supports native apps. These apps are developed in the native programming language used on a device, allowing a better integration. Apple and SAP are cooperating to develop native apps for iOS using *Apple Xcode* as the development environment. The open source language Swift, which was originally created by Apple, serves as programming language (see <https://swift.org>). It is also possible to develop native apps for Android using Java as programming language in the *Android Studio*. In contrast to the cooperation with Apple, apps are neither shipped by Google nor SAP.

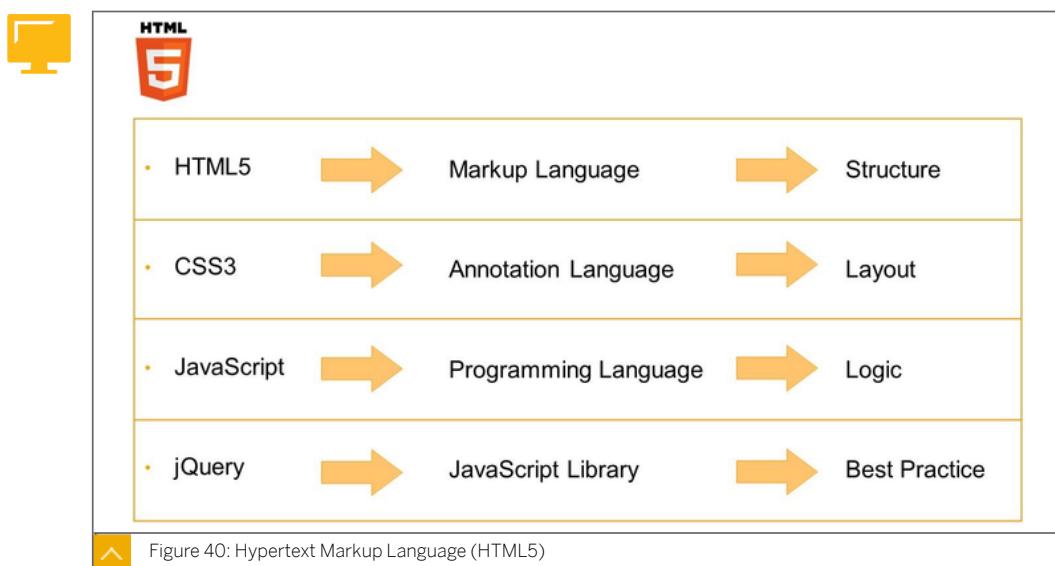
The term Conversational User Experience (CUX) describes digital assistants and chat bots for communicating with enterprise systems. The SAP Conversational Artificial Intelligence (CAI) is a combination of SAP CoPilot and the startup formerly known as Recast.AI.

SAP Conversational AI allows users to smoothly interact with SAP software through one unique conversational interface. The skills of conversational agents are built, trained, generated, and connected using the *Bot Builder*.

SAPUI5



With SAPUI5 as a basis for web apps, browsers need a certain minimum release to be able to understand all coding elements used in HTML5 and JavaScript. Although Microsoft Internet Explorer has officially supported HTML5 since version 8, it is recommended to use version 10 or switch to Microsoft Edge.



Hypertext Markup Language 5 (HTML5) is a markup language used to structure webpages. In combination with Cascading Style Sheets 3 (CSS3) for the layout, these webpages can be visualized using a browser. For dynamic interactions in webpages, the programming language JavaScript is used. JavaScript coding is organized in libraries from which it can be reused in other webpages. jQuery is a well-known library in the area of HTML5 that offers best practices.



- JavaScript-based HTML5 browser rendering library
- Extensible UI component model build on jQuery
- SAP product standard compliant
- Theming support based on CSS3
- Asynchronous JavaScript and XML (AJAX) support

<https://ui5.sap.com>

Figure 41: SAP UI Development Toolkit for HTML5 (SAPUI5)

SAPUI5 builds on top of jQuery and adds additional HTML5 browser rendering libraries. All extensions to jQuery aim to make webpages SAP product standard-compliant in visualizing and handling. The current theme of SAPUI5, which was created in CSS3, is Blue Crystal, but this is to be replaced by Belize. Asynchronous JavaScript and XML (AJAX) was first implemented with *Web Dynpro* at SAP, and it is also used in SAPUI5 to present a native-like handling of web apps. Apps developed with SAPUI5 present a consistent user experience and are responsive across browsers and devices including smartphones, tablets, and desktops. The UI controls automatically adapt themselves to the capabilities of each device.

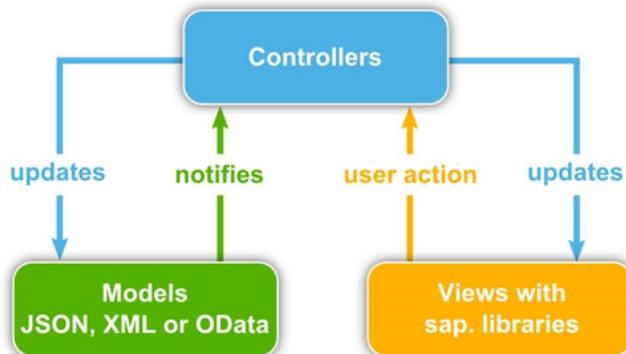


Figure 42: Model View Controller (MVC)

The Model View Controller (MVC) concept separates the tasks in an application into three programmatic elements:

Model

Holds the data and/or connection to the data source organized in JSON or XML connecting to OData

View

Holds the user interface consisting of UI elements organized in libraries

Controller

Holds the logic of the application reacting on messages from models and views and updating these

In SAP Fiori, views are defined using XML. The only Fiori app that uses HTML is the *SAP Fiori launchpad*, which provides a frame for the XML-based views. All controllers are developed using JavaScript and are either bound to a view or standalone to be used by multiple views. Data binding can be used on the views to connect to data in the models.

SAP Web IDE

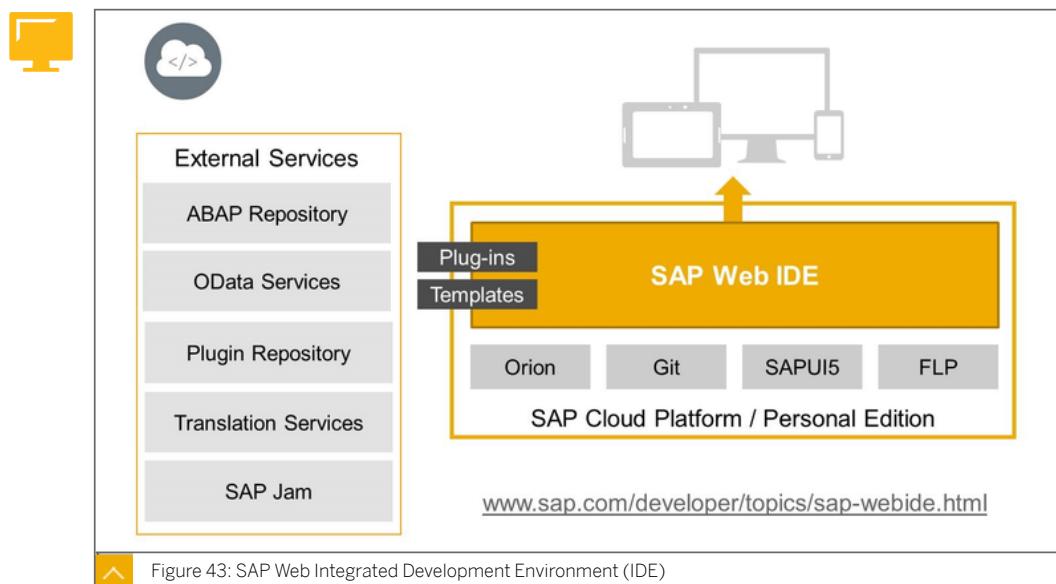
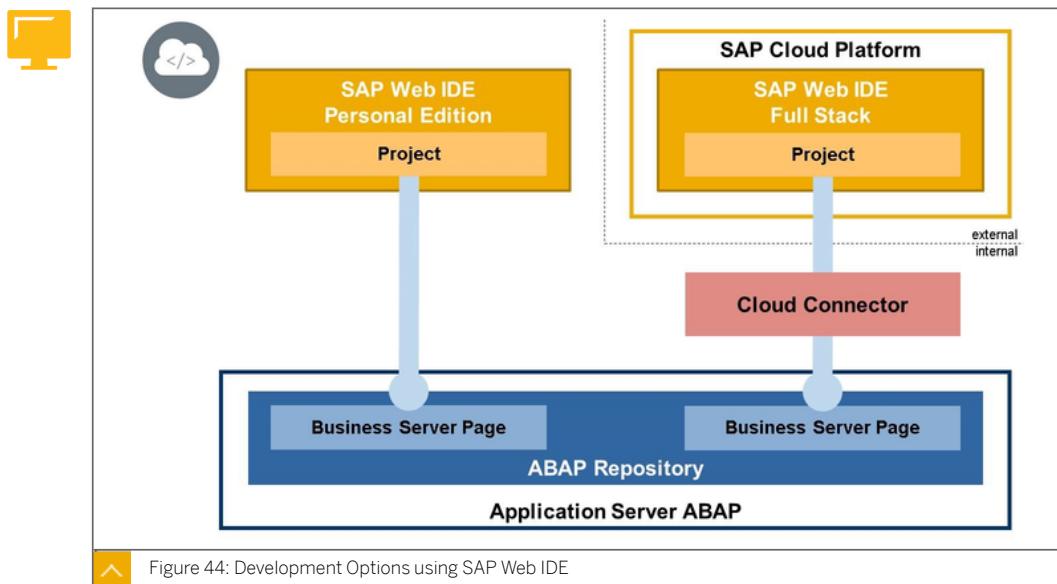


Figure 43: SAP Web Integrated Development Environment (IDE)

SAP Web IDE is hosted on the SAP Cloud Platform (CP) as a subscription for customers or it is available as a standalone local installation. It is based on Orion – an open source project by Eclipse – and SAPUI5. It can integrate with the source management tool Git and has an SAP Fiori *launchpad* for testing. External plugins can be developed and deployed to a plugin repository. Templates are starter-kit projects that can be generated based on user input.

Many additional services are available, especially when using *SAP Web IDE* in the cloud:

- Connection to ABAP repository (cloud and standalone)
- OData service consumption (cloud and standalone)
- Connection to plugin repository (cloud and standalone)
- Integration with translation services (cloud)
- Integration with SAP Jam (cloud)



Developing an SAP Fiori web app means to develop SAPUI5 using the *SAP Web IDE*. A project in the *SAP Web IDE* can connect and, therefore, represents a Fiori app delivered and managed in an Application Server (AS) ABAP as *Business Server Page (BSP)*. The BSP serves as a container for SAPUI5 apps, although BSP was originally developed based on HTML4. The tools in the ABAP workbench have not been updated for this new role of BSP. The complexity of SAPUI5 is better handled in a pure web-based environment.

The *SAP Web IDE* is available as *Personal Edition* for usage inside the customer network, and as *Full Stack* in the SAP Cloud Platform (CP). The Cloud Connector is a standalone software that is available free of charge to connect the services in the SAP CP with the on-premise systems in the customer network. Once installed in the customer network, it establishes a secure SSL Virtual Private Network (VPN) connection to the SAP CP. It is not needed if the *SAP Web IDE Personal Edition* is installed locally.

The *SAP Web IDE Personal Edition* as well as the Cloud Connector are both available for download under <https://tools.hana.ondemand.com/>.



Note:

More information about this topic can be found in UX400 (Developing UIs with SAPUI5):

<https://training.sap.com/course/ux400>



LESSON SUMMARY

You should now be able to:

- Explain user interfaces

Unit 2

Lesson 2

Explaining Data Services

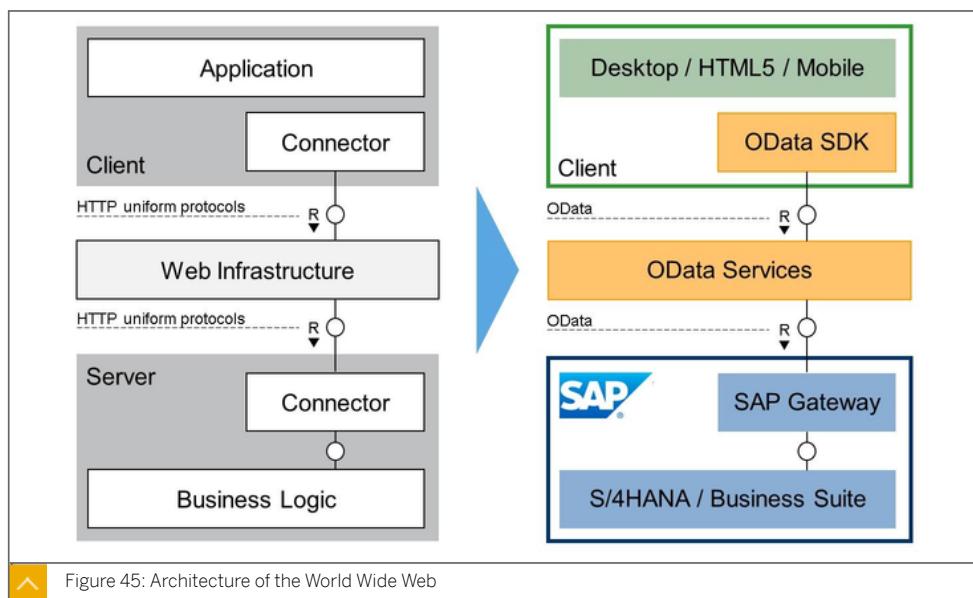


LESSON OBJECTIVES

After completing this lesson, you will be able to:

- Explain data services

OData



One important aspect of the architecture of the World Wide Web is the use of abstract interfaces for component communication. These abstract interfaces are presented as connectors. A client and a server each use a connector component. There is a contract between both connectors that defines the application protocol. It defines the documents, their format, and the behavior. Any protocol can be chosen. By using the connector concept, both client and server are largely independent and exchangeable. Each connector translates the documents exchanged on the communication channel to the internal representations both on the server and on the client side, and vice versa.

The OData protocol defines such a contract by specifying a uniform protocol that has the necessary qualities. For instance, a connector attached to an SAP back-end system translates between ABAP APIs and OData entities. SAP Gateway is such a connector. On the other side, a client connector translates between OData entities and the APIs of the consumer platform. The connector is specified here. As a consequence, any client platform with libraries

Unit 2: Technology

supporting the contracted OData format can communicate with any server supporting the same contract.

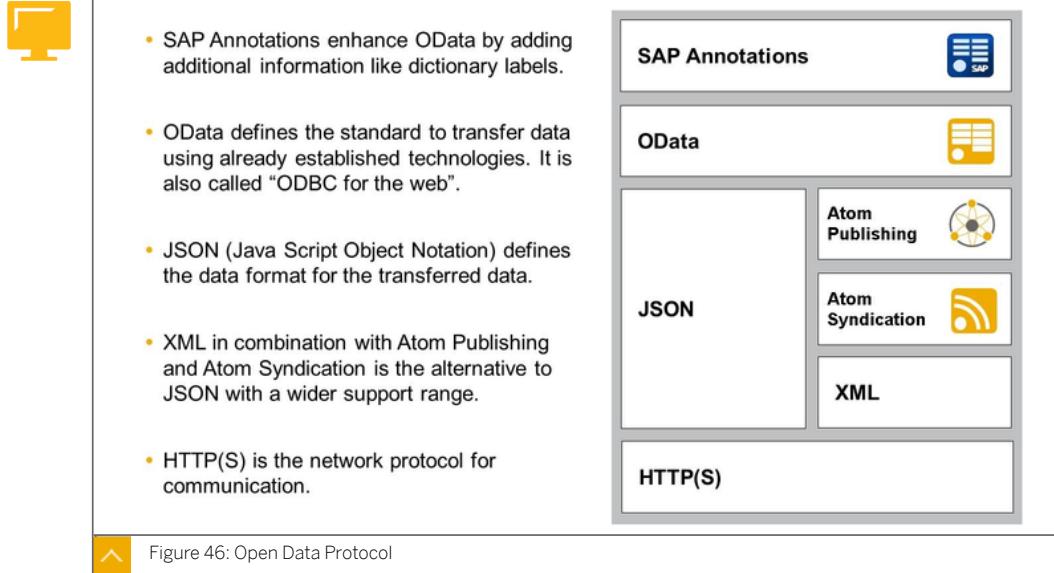


Figure 46: Open Data Protocol

OData is an open standard originally developed by Microsoft but now managed by the Oasis Organization. It is based on the Atom Publishing and Atom Syndication standards, which, in turn, are based on XML and HTTP(S). The objective of the OData protocol is to provide a vendor-neutral, web-based API that fully complies with the design principles of Representational State Transfer (REST) . OData is also extensible. This allows SAP to supplement the data types used by OData with extra information from the ABAP Data Dictionary.

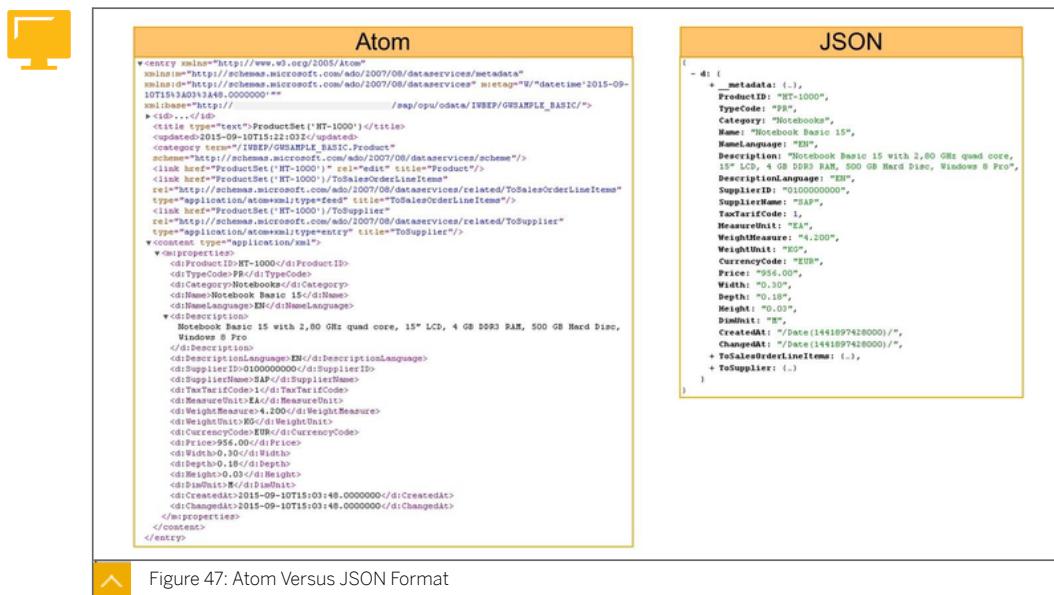


Figure 47: Atom Versus JSON Format

In current real-life applications, JSON (JavaScript Object Notation) is used instead of Atom and XML for structuring data. It needs considerably less meta-information, which reduces the amount of data transferred greatly. Atom and XML, in contrast, are used precisely because of the extensive meta-information when it comes to development.



Operation on resource	HTTP verb
Create	POST
Read	GET
Update	PUT
Delete	DELETE

Figure 48: CRUD Operations

One of the main features of OData is that it uses the existing HTTP verbs GET, POST, PUT, and DELETE at addressable resources identified in the URI. Conceptually, OData is a way of performing database-style create, read, update, and delete operations on resources by using HTTP verbs.



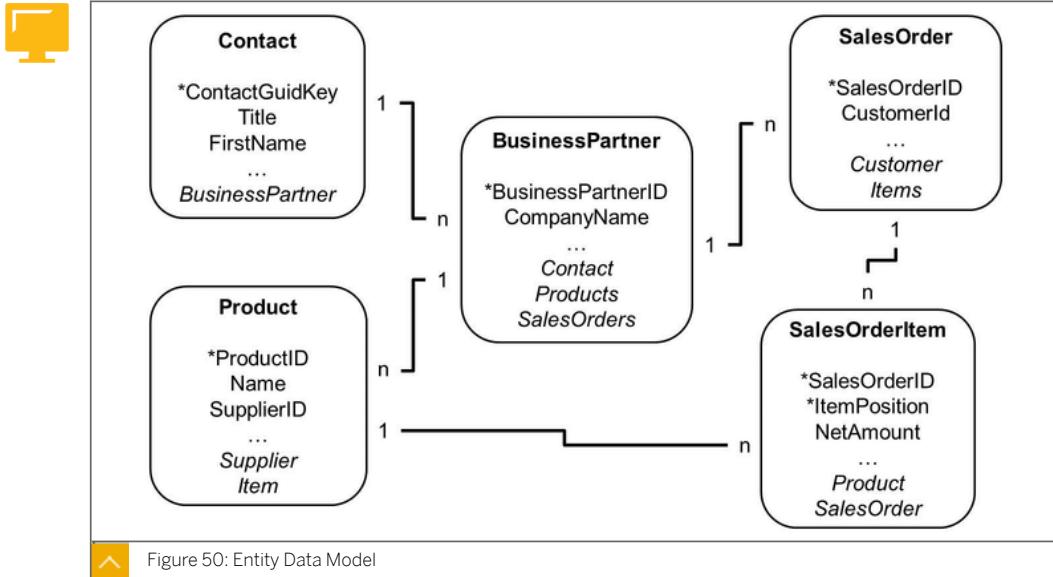
```

<app:service xmlns:app="http://www.w3.org/2007/app" xmlns:atom="http://www.w3.org/2005/Atom"
  xmlns:ns1="http://schemas.microsoft.com/ado/2007/08/dataservices/metadata"
  xmlns:sap="http://www.sap.com/Protocols/SAPData" xml:lang="de"
  xmlns:base="http://
    /sap/opu/odata/IWBEP/GWSAMPLE_BASIC/"/>
  <app:workspace>
    <atom:title type="text">Data</atom:title>
    <app:collection sap:content-version="1" href="BusinessPartnerSet">
      <atom:title type="text">BusinessPartnerSet</atom:title>
      <sap:member-title>BusinessPartner</sap:member-title>
    </app:collection>
    <app:collection sap:content-version="1" href="ProductSet">
      <atom:title type="text">ProductSet</atom:title>
      <sap:member-title>Product</sap:member-title>
    </app:collection>
    <app:collection sap:updatable="false" sap:content-version="1" href="SalesOrderSet">
      <atom:title type="text">SalesOrderSet</atom:title>
      <sap:member-title>SalesOrder</sap:member-title>
    </app:collection>
    <app:collection sap:content-version="1" href="SalesOrderLineItemSet">
      <atom:title type="text">SalesOrderLineItemSet</atom:title>
      <sap:member-title>SalesOrderLineItem</sap:member-title>
    </app:collection>
    <app:collection sap:content-version="1" href="ContactSet">
      <atom:title type="text">ContactSet</atom:title>
      <sap:member-title>Contact</sap:member-title>
    </app:collection>
  </app:workspace>
  <atom:link rel="self"
    href="http://
    /sap/opu/odata/IWBEP/GWSAMPLE_BASIC/"/>
  <atom:link rel="latest-version"
    href="http://
    /sap/opu/odata/IWBEP/GWSAMPLE_BASIC/"/>
</app:service>

```

Figure 49: Service Document

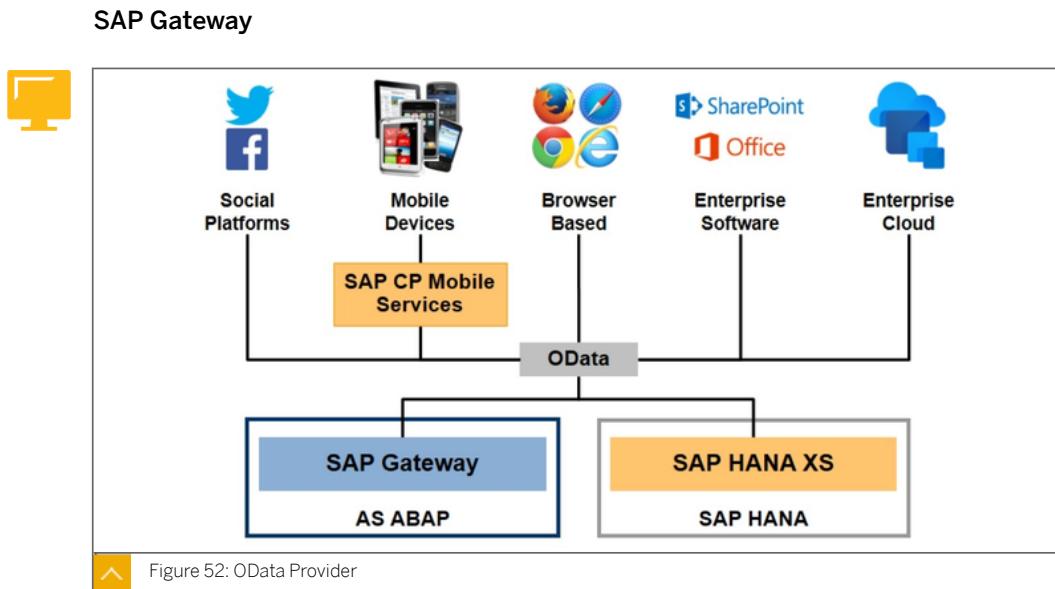
To consume an OData service for read, you just need a browser and the OData service URI. This leads to the service document. To get data from the service, add the name of an entity of the service to the base URI. You get a list of entities of that type, which could be the content of a database table.



The entities of an OData service are defined in an Entity Data Model (EDM). Entity types define properties and navigations to other entity types. These associations define relation constraints based on key properties of the entity types. A navigation property is then used to actively navigate between entities during runtime. For every entity type, at least one entity set is defined. These are shown in the service document and used to request data from the OData service.

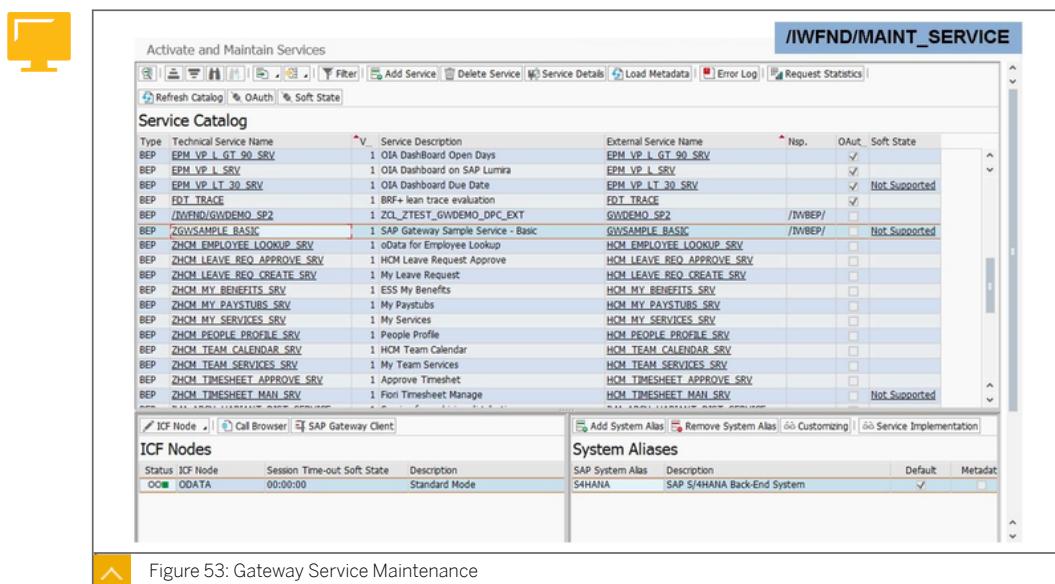


By adding the OData option `$METADATA` to the service root URI, the metadata document of the service is shown. The whole EDM is defined here and available at runtime. Application developers and all the wizards in development environments create their applications based on this information.



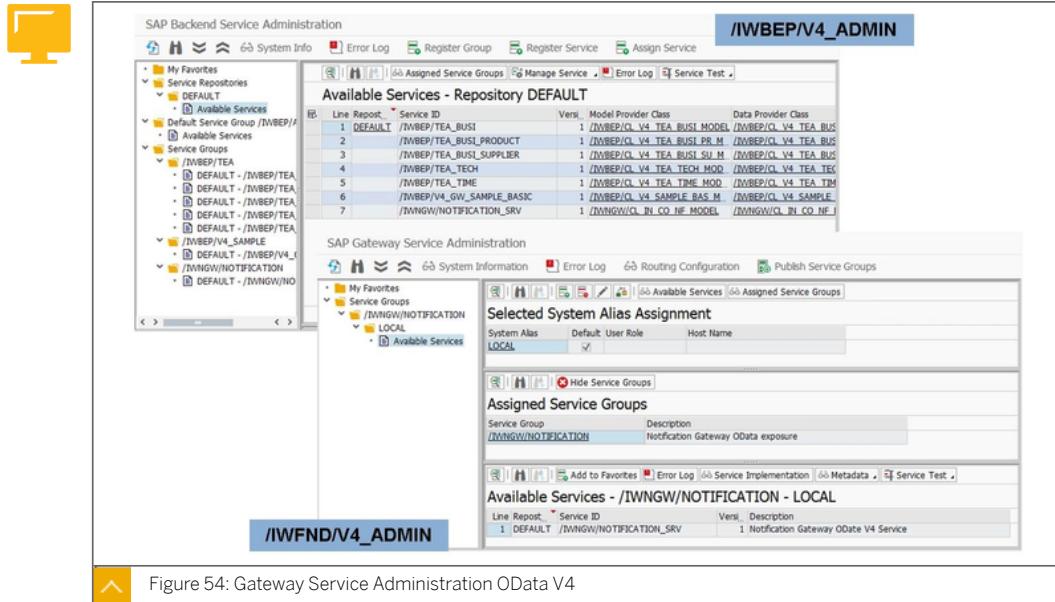
SAP Gateway (formally known as SAP NetWeaver Gateway) provides a single entry point to access business data of ABAP-based systems like the SAP Business Suite or the SAP S/4HANA. The SAP HANA Extended Applications Services (XS) has the same role in SAP HANA. This business data can be shared among multiple environments and platforms. SAP knowledge is not required for the consumption of the data.

Additional servers can be added to the communication path to enhance the possibilities for client and server. For mobile devices, the SAP Cloud Platform (CP) Mobile Services adds additional value to the applications.

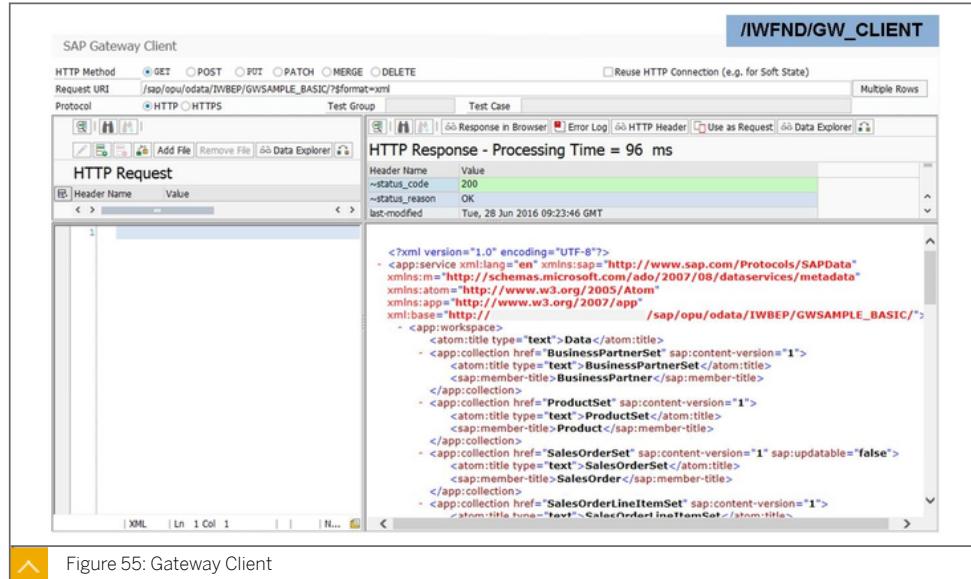


The gateway service maintenance transaction (`/IWFND/MAINT_SERVICE`) offers a list of all registered OData services in the system. Complete maintenance of these services is possible here. The transaction is divided into three areas:

- Service catalog (service name, description, and many additional settings)
- ICF nodes (maintenance of ICF services and testing)
- System aliases (maintenance of connections to backend)

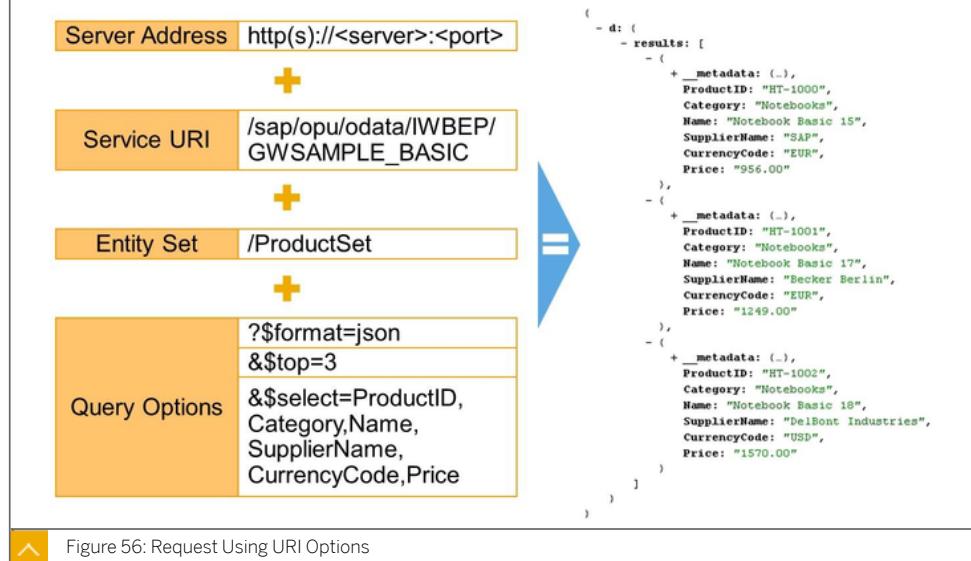


The *Gateway Service Administration* for OData V4 consists of two transactions. Transaction `/IWBEP/V4_ADMIN` defines service groups, which are registered using transaction `/IWFND/V4_ADMIN`. The first productive OData V4 service in SAP Gateway is the notification service for the *Fiori launchpad*.



Using the gateway client transaction (/IWFND/GW_CLIENT), all functionalities of an OData service can be tested. For a read request, it is enough to enter the request URI and execute. For a create, update, and delete request, additional adjustments must be made, for example, a request body filled with data must be created. All adjustments can be saved as test cases for later usage.

Apart from these transactions, there are several more for SAP Gateway. All transactions are connected to each other via buttons and menu entries. Therefore, you can jump from the service maintenance to the gateway client and back.



An OData request consists of a server address, service URI, entity set, and additional query options. There are many query options with many combinations possible. Full documentation is available at <http://www.odata.org>.

Unit 2

Lesson 3

Explaining Application Types



LESSON OBJECTIVES

After completing this lesson, you will be able to:

- Explain application types
- Using SAP Fiori App Recommendations

Overview



Transactional

Task-based Access

Access to tasks like change, create, or entire processes with guided navigation



Any DB

Analytical

Insights

Visual overview of a dedicated topic for further KPI related analyses



SAP HANA

Fact Sheet

Search and Explore

View essential information about objects and contextual navigation between related objects



SAP HANA

Figure 57: SAP Fiori App Types

All SAP Fiori apps utilize the technologies SAPUI5 and SAP Gateway. The three types of SAP Fiori apps are different in terms of their usage of additional technologies:

Transactional Apps

- Usage of ABAP to provide the classic approach for functions of a business system
- Available for S/4HANA and Business Suite on any DB

Analytical Apps

- Usage of analytical capabilities of SAP HANA to provide insights in business data
- Available for S/4HANA and Business Suite on HANA

Fact Sheet Apps

- Usage of Enterprise Search capabilities of SAP HANA to provide search results
- Available for S/4HANA and Business Suite on HANA

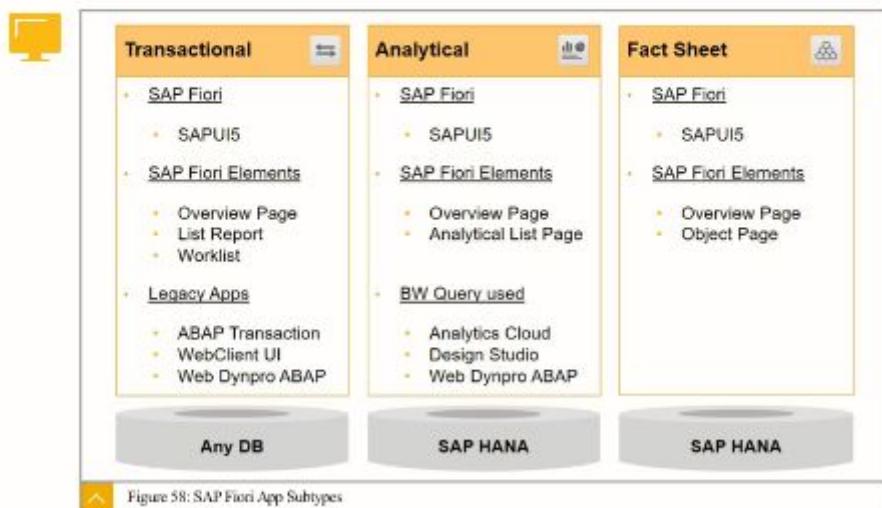


Figure 58: SAP Fiori App Subtypes

SAPUI5 is used in all application types for development. This can be done by implementing JavaScript code directly or by defining metadata, which generates JavaScript code at runtime. These apps are called SAP Fiori elements. The complete UI is controlled by metadata annotations in SAPUI5, SAP Gateway, or CDS views.

List Report

Allows users to view and work with items (objects) organized in list (table) format.

Object Page

Provides functionality to view, edit, and create (business) objects.

Overview Page

Visualizes large amount of data in cards with different formats for different types of content.

Analytical List Page

Identifies relevant areas within data sets or significant single instances using data visualization and business intelligence.

Worklist

Displays a collection of items that a user needs to process.

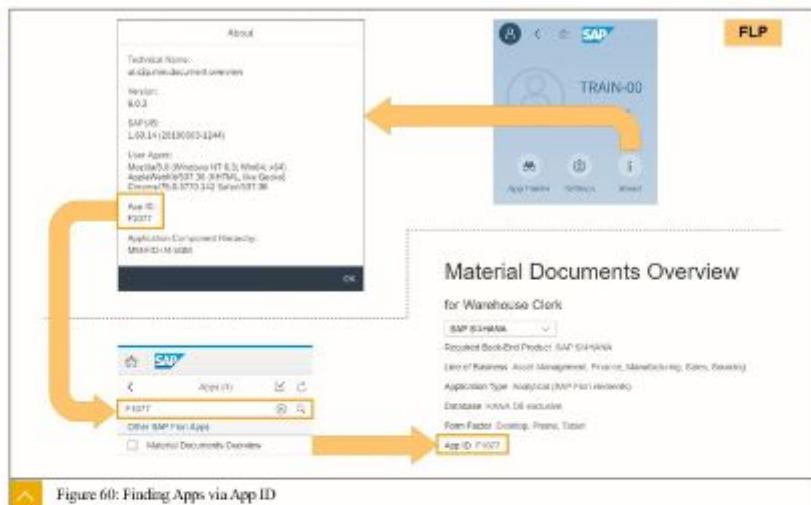
In addition, part of transactional apps are so-called legacy apps. These are ABAP transactions, Web Dynpro ABAP, and WebClient UI Framework applications, which already existed before SAP Fiori was announced. Customers may choose which legacy apps they want to have in their SAP Fiori launchpad using the configuration and customizing of SAP Fiori.

SAP Business Warehouse (BW) queries can be used as a foundation for SAP Fiori applications. These applications are created by using SAP Analytics Cloud, the tool Design Studio, or by developing Web Dynpro ABAP applications.



Figure 59: SAP Fiori Apps Reference Library

SAP Fiori is foremost a collection of apps representing the new user experience of SAP and the face of SAP S/4HANA. SAP Fiori apps can be categorized by line of business, industry, and most important user role, as well as technical foundation. All available apps can be explored using the SAP Fiori apps reference library. See <https://www.sap.com/fiori-apps-library>.



Each SAP Fiori app has a unique App ID. When running the app in the SAP Fiori launchpad, choose About in the Me Area. The following popup shows information about the app including the App ID.

The App ID can be used to search for the documentation of the app in the SAP Fiori reference library and it is visible in the header of the documentation.

 Hint:
The App ID of a transaction is its transaction code.

 **Procurement Overview Page**

for Purchaser

SAP S/4HANA

Required Back-End Product: SAP S/4HANA

Line of Business: Sourcing and Procurement, Supply Chain

Application Type: **Transactional, Analytical (SAP Fiori elements: Overview Page)**

Database: **HANA DB exclusive**

Form Factor: Desktop, Phone, Tablet

App ID: F1990

PRODUCT FEATURES **IMPLEMENTATION INFORMATION**

The Procurement Overview app shows you, at a glance, the most important information and tasks relevant for you right now. The information is displayed on a set of actionable cards. You can therefore focus on the most important tasks, enabling faster decisions and immediate action. The cards show you the most important information ranked according to relevance. For example, on the Monitor Purchase Contracts card, you can see the consumption and expiry date per contract. You can then decide to change a contract by selecting it.

You can use the global filter to filter the entire Procurement Overview by supplier or material group, for example. You then see all urgent contracts, purchase requisitions, or spend information according to the specified filter criteria. For list cards, selecting the header of a card brings you to the app itself, while selecting an item brings you to more detailed item information. For graphical or analytical cards, selecting the card brings you to more detailed analytical information.

 Figure 61: Overview Page Based on SAP Fiori Elements

The overview page is based on SAP Fiori elements technology and uses annotated views of app data. This means that the app content can be tailored to the domain or role. The overview page acts as a UI framework for organizing multiple cards on a single page. Each card can deliver transactional, analytical, or search data. Therefore overview pages are often a mixture of application types.

Transactional

My Quotations

for Field Sales Representative

SAP Business Suite

Required Back-End Product SAP ERP, min. EHP 7

Line of Business Finance, Marketing, Sales, Service, Sourcing and Procurement

Application Type **Transactional (SAP Fiori (SAPUI5))**

Database **Any DB**

App ID F0025

PRODUCT FEATURES **IMPLEMENTATION INFORMATION**

With the transactional app **My Quotations**, you can consult, modify, or generate quotations in the field interactively with customers on desktop, tablet, and mobile platforms. The app also makes it easy to identify quotations that are on the verge of expiring and provides you with a convenient way to manage follow-ups with your customers.

Figure 62: Transactional App Based on SAPUI5

All transactional apps perform transactional tasks, such as creating a leave request for an employee or managing quotations. SAP Fiori transactional apps represent simplified views and interactions with existing business processes and solutions. They generally run on any database except if they are developed and shipped with SAP S/4HANA .

Manage Sales Orders

for Internal Sales Representative

SAP S/4HANA

Required Back-End Product SAP S/4HANA

Line of Business Finance, Marketing, Sales, Service, Sourcing and Procurement

Application Type **Transactional (SAP Fiori elements)**

Database HANA DB exclusive

Form Factor Desktop, Tablet

App ID F1873

PRODUCT FEATURES **IMPLEMENTATION INFORMATION**

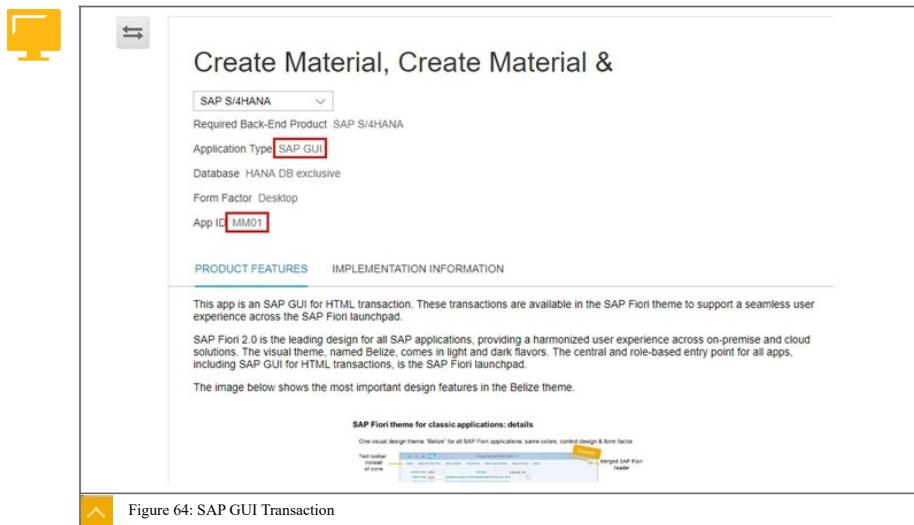
With the **Manage Sales Orders** app, you can search for sales orders according to your filter criteria and display them in **list**. From the list, you can execute different actions, for example, you can reject all items of one or more sales orders. You can also navigate to related apps, for example, to display sales order details or you can display the process flow.

Key Features

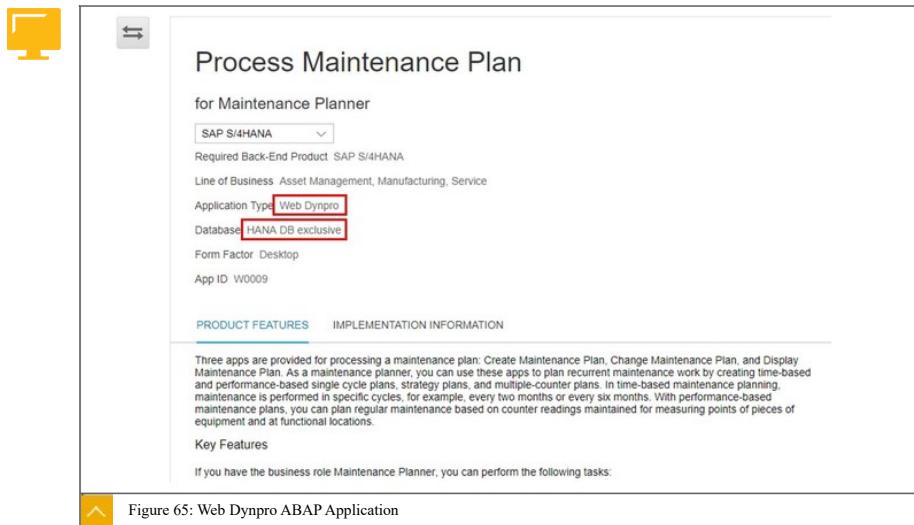
- Search for sales orders using a free text search
- Search for sales orders by entering a value in one or more of the filter fields

Figure 63: List Report Based on SAP Fiori Elements

Transactional apps can be generated based on SAP Fiori elements . These are dynamic SAPUI5 apps controlled by metadata annotations in SAPUI5, SAP Gateway , or CDS views. The most common transactional Fiori element is the list report.



Since the release of SAP S/4HANA 1610, SAP GUI transactions can be found in the SAP Fiori reference library. Due to SAP Fiori 2.0 and the SAP Belize theme, they are an official part of SAP Fiori. The transactions are limited to those targeting end users and those available in SAP S/4HANA 1610 or later.



Like the SAP GUI transactions, transactional Web Dynpro ABAP applications are also part of SAP Fiori 2.0. All transactions and Web Dynpro applications are shown as exclusive to the SAP HANA database in the SAP Fiori reference library. The reason for this is that the documented release is SAP S/4HANA 1610 or later – and this one is exclusive to the SAP HANA database.

Analytical

Material Documents Overview

for Warehouse Clerk

SAP S/4HANA

Required Back-End Product SAP S/4HANA

Line of Business Asset Management, Finance, Manufacturing, Sales, Sourcing and Procurement, Supply Chain

Application Type Analytical (SAP Fiori elements)

Database HANA DB exclusive

Form Factor Desktop, Phone, Tablet

App ID F1077

PRODUCT FEATURES **IMPLEMENTATION INFORMATION**

With this app you can display a list of material document items and display the material document details of a selected material document item. In addition, you can reverse a material document. This app supports you, for example, as a warehouse clerk or an inventory manager in your daily work.

Key Features

- Get an overview of material documents such as goods issue, goods receipt, and material movements with the following information, for example:
- Document date

Figure 66: Analytical App Based on SAPUI5

These apps give a role-based insight into real-time operations of your business by collecting and displaying key figures directly in your browser. They can be SAP Smart Business applications or other analytical, predictive, and planning applications. Analytical apps combine the data and analytical power of SAP HANA with the integration and interface components of SAP Business Suite or SAP S/4HANA. They provide real-time information on large volume data in a simplified front end for enterprise control. With SAP Smart Business, you can closely monitor your most important Key Performance Indicators (KPIs) in real time and react immediately to changes in market conditions or operations.

Cost Centers - Actuals

for Cost Accountant - Overhead

SAP S/4HANA

Required Back-End Product SAP S/4HANA

Line of Business Finance

Application Type Analytical, Design Studio (SAP Fiori, Design Studio)

Database HANA DB exclusive

Form Factor Desktop

App ID F0940A

PRODUCT FEATURES **IMPLEMENTATION INFORMATION**

With all Analytical Apps for Management Accounting you can report on actual and plan data for different entities relevant for management accounting. The name of the app indicates the entity as well as the type of data, for example Cost Centers - Plan/Actual.

Key Features

- Specify the search criteria using variables or filters
- Display the query results in a grid format

Figure 67: Design Studio App

The design studio Fiori apps have nothing to do with the tool design studio available for customers. However, they use a similar technology in the Business Warehouse area. They are pre-generated for customers without the use of KPIs.

Design studio apps can be generated using SAPUI5 or Web Dynpro / Floorplan Manager (FPM) as UI technology.

Fact Sheet

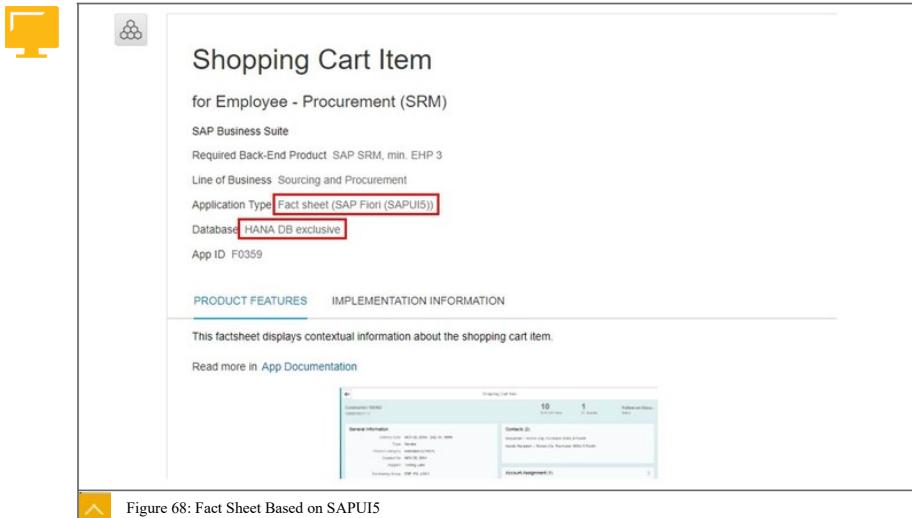
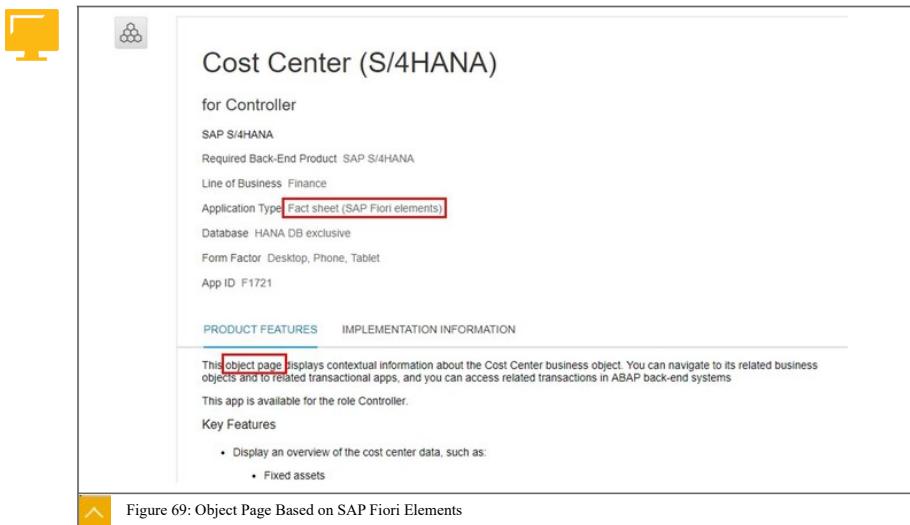


Figure 68: Fact Sheet Based on SAPUI5

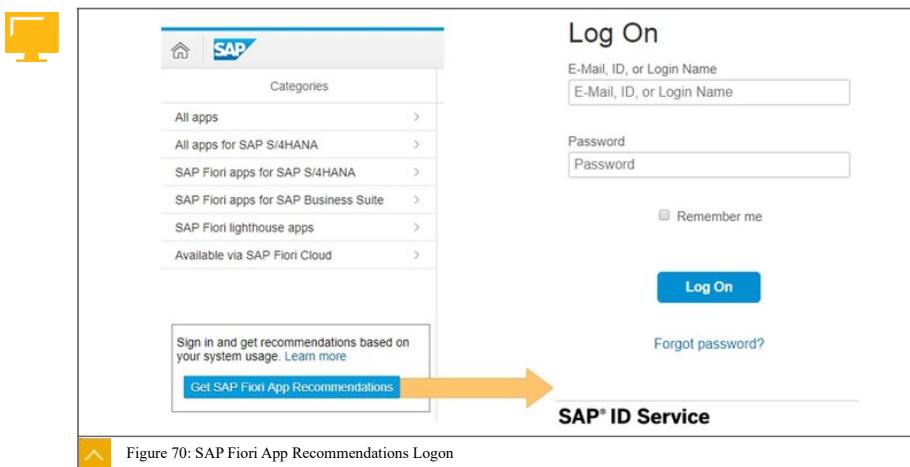
These apps display contextual information and key facts about central objects used in business operations. Fact sheets are designed to be intuitive and harmonized. From a fact sheet area (tile), you can drill down into its details. You can easily navigate from one fact sheet to its related fact sheets. For example, you can navigate from a document to the related business partner or the master data.

From fact sheets, you can start transactions by navigating to transactional apps, or by accessing the back-end system directly. For example, from a document fact sheet, you can access the back-end system to display document details or edit the document in SAP GUI or Web Dynpro .



Object pages can change or create the data found via search directly in the app without opening other applications. The changes are performed in place. Fact sheet and object pages do not have tiles delivered by SAP. That is a task for customizing or configuration, or for the user when saving their search result as a tile.

Use SAP Fiori App Recommendations



The SAP Fiori reference library offers recommendations for SAP Fiori apps by analyzing the usage of an already existing system landscape. The analysis can be started on the main page by choosing **Get SAP Fiori App Recommendations**. A logon with an SAP user is mandatory.

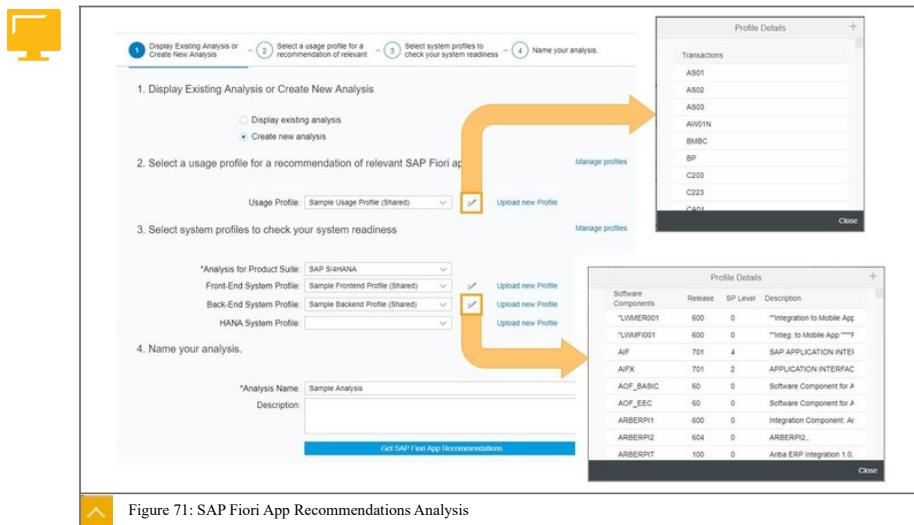


Figure 71: SAP Fiori App Recommendations Analysis

The analysis needs a usage and some system profiles as input:

Usage Profile

A usage profile is a list of the most used transactions in a system. These can be collected based on feedback of end users or by tracing the used transactions in the system, for example, by using the Workload Monitor (ST03).

System Profile

A system profile is a list of installed software components in an SAP system. These can be collected using the system information of a system or by using functions of the SAP Solution Manager, for example, the Landscape Management Database (LMDB)

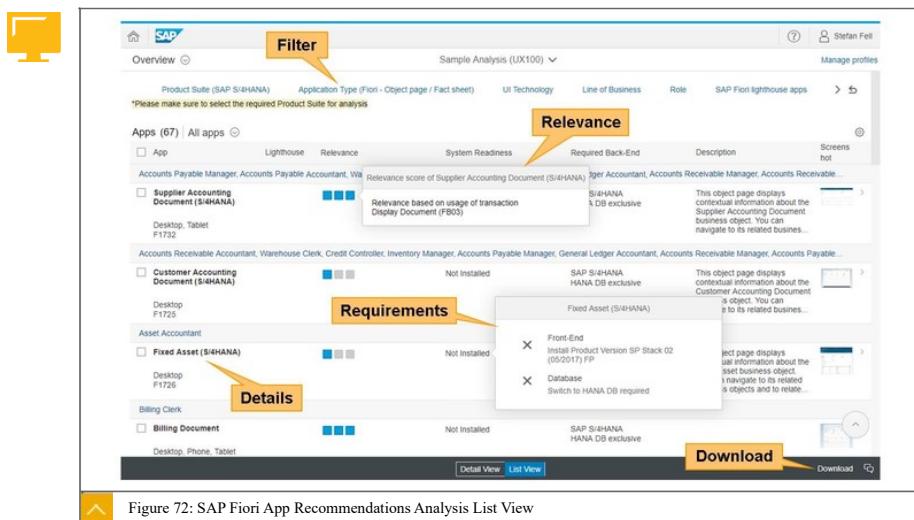


Figure 72: SAP Fiori App Recommendations Analysis List View

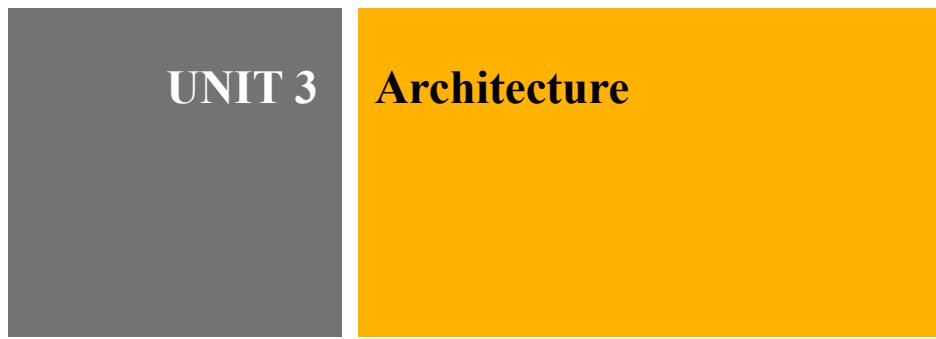
The analysis results in a list of apps with information about their relevance and system requirements depending on the usage and system profiles. The list can be filtered in many ways, for example, by application types, line of business, or roles. Details of each app can be accessed by just clicking on one entry or you can switch from the list view to the detail view, which jumps to a filtered view of the SAP Fiori reference library .



LESSON SUMMARY

You should now be able to:

- Explain application types
- Using SAP Fiori App Recommendations



UNIT 3

Architecture

Lesson 1

Examining SAP Fiori for ABAP Platform	63
---------------------------------------	----

Lesson 2

Examining SAP Fiori for SAP Business Suite on HANA	72
--	----

Lesson 3

Examining SAP Fiori for SAP S/4HANA	82
-------------------------------------	----

UNIT OBJECTIVES

- Examine SAP Fiori for ABAP Platform
- Examine SAP Fiori for SAP Business Suite on HANA
- Examine SAP Fiori for SAP S/4HANA

Unit 3

Lesson 1

Examining SAP Fiori for ABAP Platform



LESSON OBJECTIVES

After completing this lesson, you will be able to:

- Examine SAP Fiori for ABAP Platform

Landscape of SAP Fiori for ABAP Platform

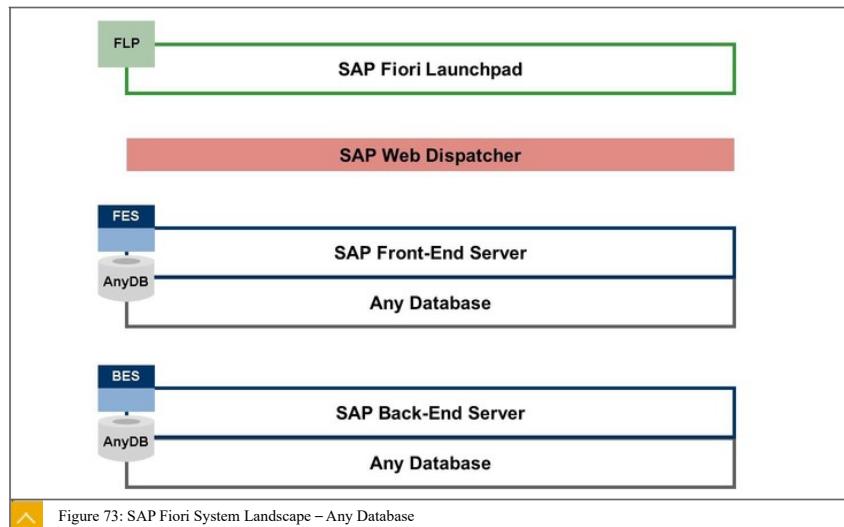


Figure 73: SAP Fiori System Landscape – Any Database

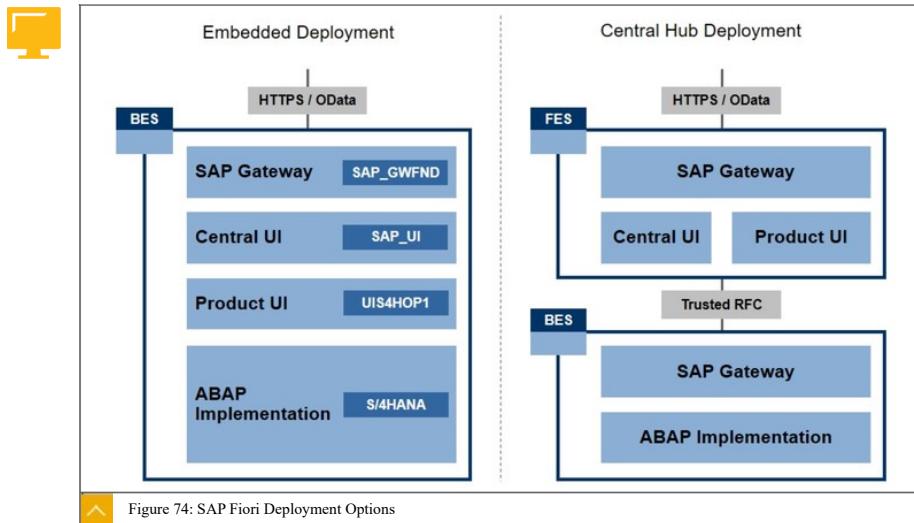
The system landscape for SAP Fiori consists mainly of an SAP Front-End Server (FES) and an SAP Back-End Server (BES). These are system roles of an Application Server (AS ABAP or ABAP Platform) in this landscape. The back-end server is an SAP Business Suite or an SAP S/4HANA holding applications and data of any area based on an AS ABAP 7.40 or higher. The front-end server is a basic AS ABAP 7.40 or higher without any business products installed. Both systems can run on any database.



Note:
From release 7.53, Application Server ABAP is renamed ABAP Platform. The reason was the availability of ABAP in SAP Cloud Platform.

Although the SAP Fiori launchpad (FLP) running in any client is able to communicate with the FES directly, it is recommended to use an SAP Web Dispatcher or another reverse proxy.

between FLP and FES in external facing scenarios and also internally. The reason is that for some features of the FLP multiple systems have to be reached, which is forbidden in web communication with good reason. A reverse proxy offers a single point of communication holding routing rules to forward requests to the correct target systems.



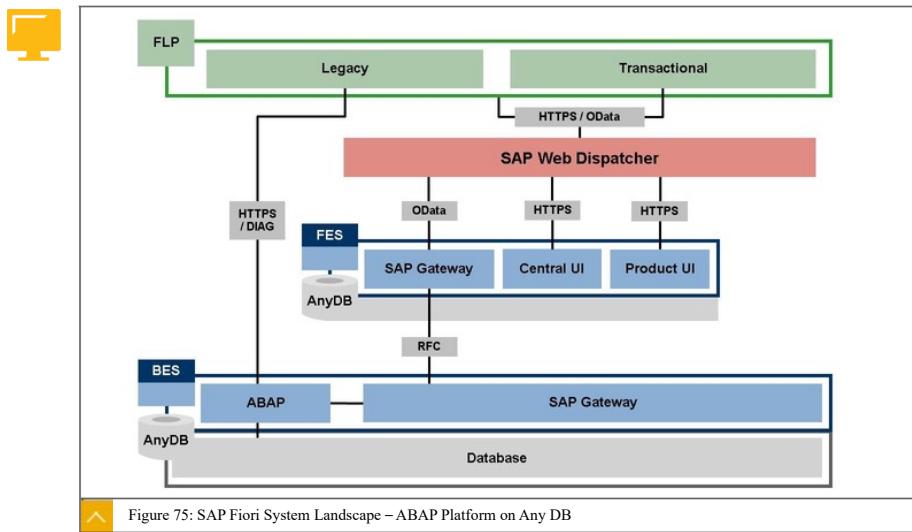
In some scenarios, it is not necessary to operate separated front-end and back-end servers. It can even be counterproductive to do so. Therefore, it is possible to deploy all components of SAP Fiori in one system, in an embedded deployment.

Embedded Deployment	Central Hub Deployment
The tasks of the FES for providing SAP Fiori are embedded in the BES. That means there is only one system.	The FES and BES are two separated systems splitting the tasks in providing SAP Fiori.

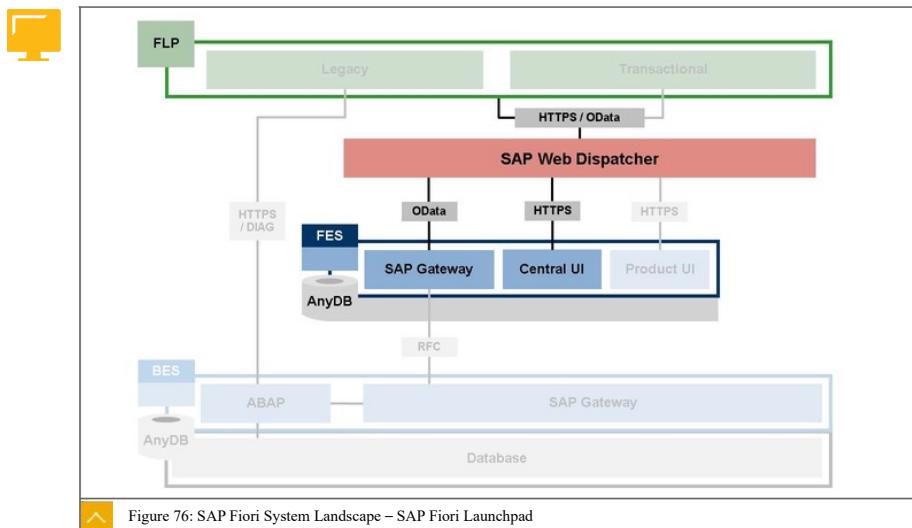
SAP Gateway, the only component in FES and BES, is the only part where you have to check carefully, which settings are in which system. As soon as there are multiple systems that want to provide Fiori apps, the FES has many benefits in organizational and security terms.



Note:
The recommended deployment option for SAP Business Suite is the embedded scenario.



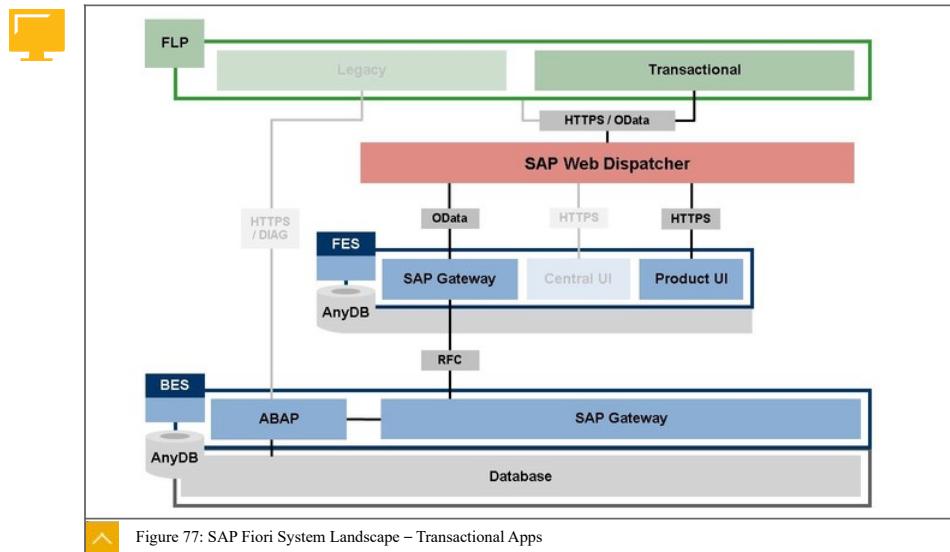
The Fiori system landscape for a BES on any database supports transactional apps based on SAPUI5 and legacy apps based on Web Dynpro and ABAP transactions. Mandatory components of the FES are the SAP Gateway for OData communication, the central UI for general Fiori functions, and product UIs for the apps. The SAP Gateway and ABAP coding for the logic of the apps are also in the BES.



To be able to use any SAP Fiori app, a running FLP is needed. The FLP is part of the central UI including all views, functions, and tools for personalization, customizing, and configuration of the contents. When thinking about personalization, this is performed by the user in a running

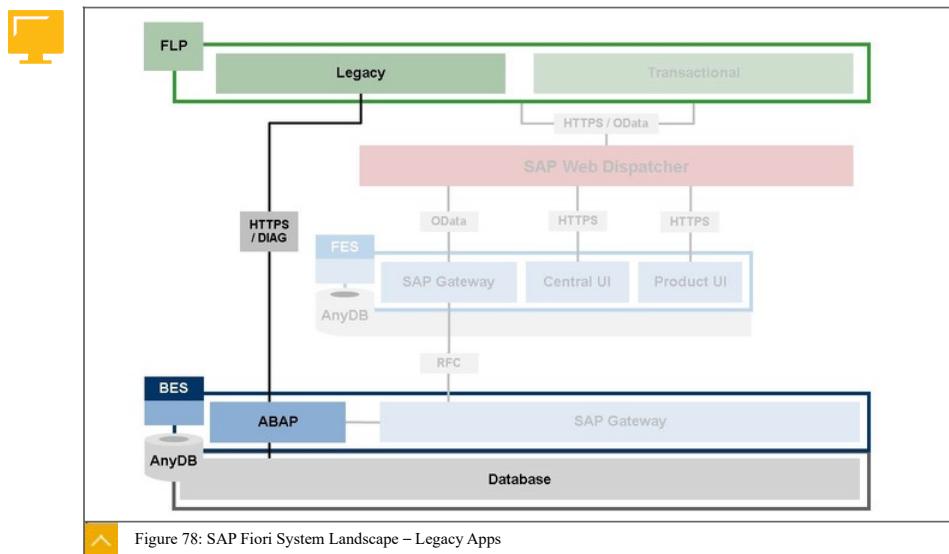
FLP. Therefore, the FLP needs to read and write data from and to the FES using OData , not only for personalization, but also in other areas.

There is no communication to the BES by the FLP. All general settings for SAP Fiori are saved in the FES to be independent from the BES. Although this is not visible in the diagram, having multiple back ends is usual for many customers. This means that having only one place (system) for SAP Fiori settings is a huge benefit.



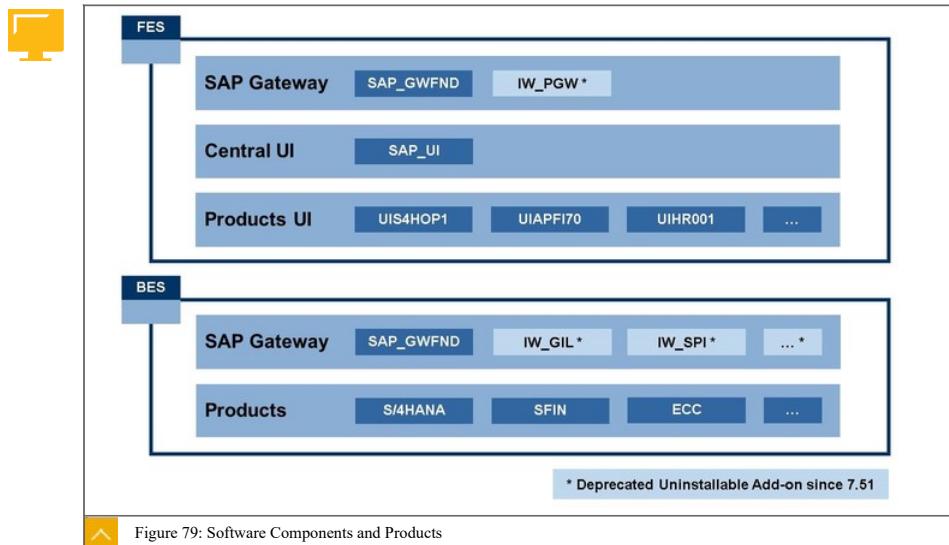
Disregarding the need for a running FLP, transactional apps consist of SAPUI5 apps and SAP Gateway services. SAPUI5 apps are saved in the ABAP repository as BSPs and delivered using product-specific UI add-ons. These are installed on the FES. The SAP Gateway services are delivered as part of updates of the BES solution and are written in ABAP.

The registration of the SAP Gateway services is performed on the FES by the customer. So, when transactional apps need data, an OData request is performed via the SAP Web Dispatcher to the registration of the SAP Gateway service on the FES. From there, the SAP Gateway framework creates an RFC request to the implementation of the SAP Gateway service on the BES. In the implementation, ABAP coding is used to access the data in the database.



Legacy apps are part of the BES solution and have no interaction with the FES besides the FLP itself. The UI and data directly originates from the ABAP coding on the BES. The only task of the FES is to provide the connection information to the BES when a navigation request — like clicking a tile — is initiated in the FLP. This connection information has to be set in the FES by an administrator.

Installation of Transactional Apps



Looking more closely at software components and products, SAP Gateway consists mainly of the software component SAP_GWFND in the FES and BES. This is part of every AS ABAP

since 7.40. Up to 7.50, additional features of SAP Gateway may be needed, depending on the apps. When using SAP Workflow in Fiori 1.0, the add-on IW_PGW is needed in the FES. When using data models of the Generic Interaction Layer (GENIL), the add-on IW_GIL is needed in the BES. When using Service Provider Infrastructure (SPI), IW_SPI is needed, and so on. All these SAP Gateway add-ons are deprecated since 7.51 and Fiori 2.0 and can be uninstalled. Either the functionality was integrated in SAP_GWFND (for example, SAP Workflow) or it is no longer needed in SAP S/4HANA 1610.

The central UI consists of the software component SAP_UI, which is part of every AS ABAP since 7.40 and holds all UI technologies in the system, not just Fiori.

The product-specific UI add-ons hold not one but all SAPUI5 apps of one solution area. The type and the release should fit to the solution implemented in the BES.

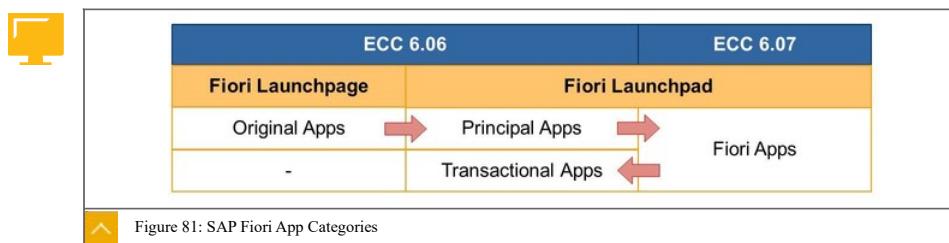


Front-End Components		My Quotations [F0025]
Product Version	UI FOR EH7 FOR SAP ERP 6.0 SAP Fiori 1.0 for SAP ERP	
Support Package Stack	10 (07/2016)	
Software Component Version	UIEAAP01 100 - SPS 0010	
	UI FOR EH7 FOR SAP ERP 6.0 - SPS 10 (07/2016) is an Add On to SAP FIORI FRONT-END SERVER 2.0 - SPS 01 (12/2015) or SAP FIORI FRONT-END SERVER 3.0 - SPS Initial Shipment Stack or SAP FIORI FRONT-END SERVER 4.0 - SPS Initial Shipment Stack or SAP GATEWAY 2.0 - SPS 10 (03/2015) or SAP NETWEAVER 7.4 - SPS 04 (09/2013)	
Prerequisite for installation		
Back-End Components (ABAP)		
Product Version	EHP7 FOR SAP ERP 6.0 SAP enhancement package 7 for SAP ERP 6.0 or EHP8 FOR SAP ERP 6.0 SAP enhancement package 8 for SAP ERP 6.0	
Support Package Stack	12 (07/2016)	
Software Component Version	03 (08/2016) FP SAP_APPL 617 - SPS 0012	
	SAP_APPL 618 - SPS 0003	



Figure 80: SAP Fiori App Installation

In this example of the Fiori app reference library, you see the product UI needed to be installed on the FES and the update of the solution was needed on the BES. The product UI is an add-on to SAP Gateway and SAP NetWeaver, consisting not only of this app, but all apps for ERP 6.0 EhP7. The update of the BES is the Enhancement Package (EhP) 7 for ERP. Nearly all Fiori applications are installed in such a way - with the exception of 25 apps.



In 2013, the first 25 Fiori apps were developed for ECC 6.06. These so-called original apps are completely different in their architecture compared to current Fiori apps. They are stand-alone transactional apps, which can be embedded in the Fiori launchpage. The launchpage is

simply a webpage offering links visualized as tiles to start the original apps. No customizing or any other adaptation is possible. The original apps as well as the Fiori launchpage are deprecated and no longer available.

With ECC 6.07, the Fiori apps as we know them currently were introduced, including the Fiori launchpad as the successor of the launchpage. As a first step, the original apps were rewritten for the ECC 6.06 from HTML-based to XML-based apps and integrated in the Fiori launchpad. They were then up-ported in a second step to ECC 6.07.

Simultaneously, the additional created transactional apps in ECC 6.07 were down-ported to ECC 6.06. Due to the missing support for SAP HANA in ECC 6.06, analytical and fact sheet apps could not be down-ported.

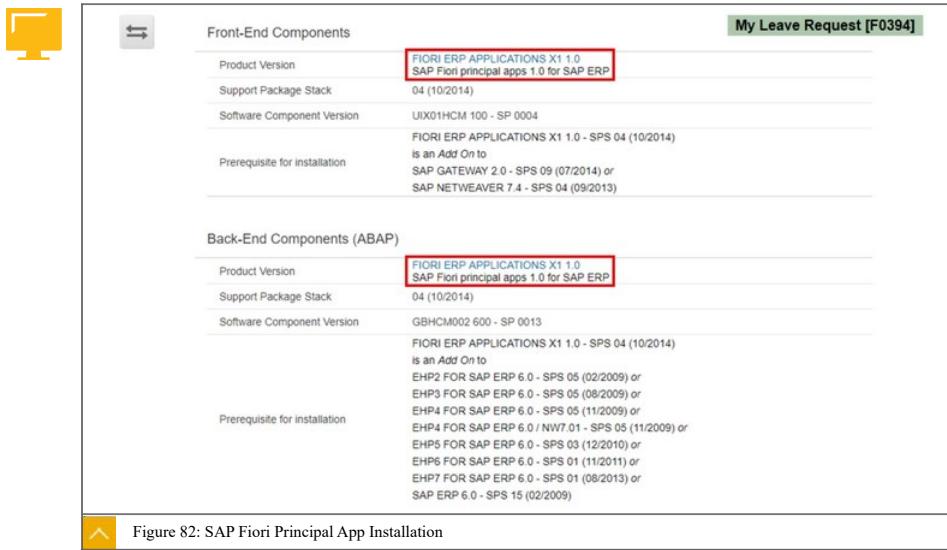
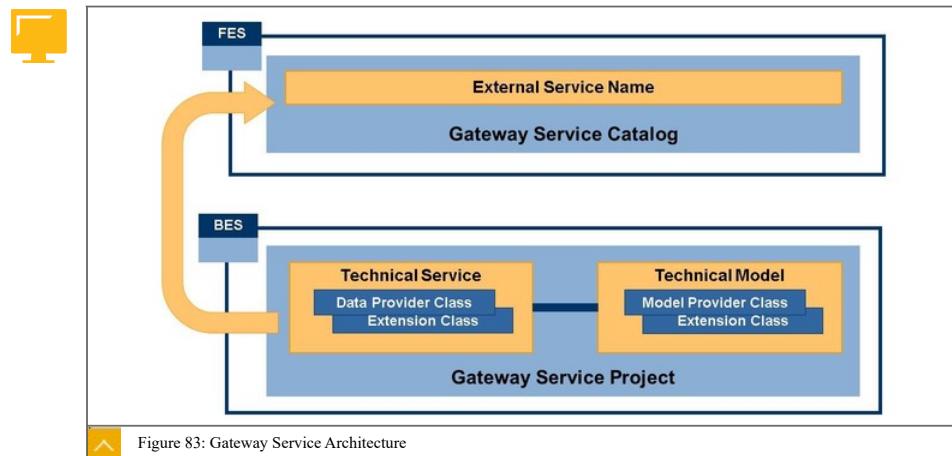


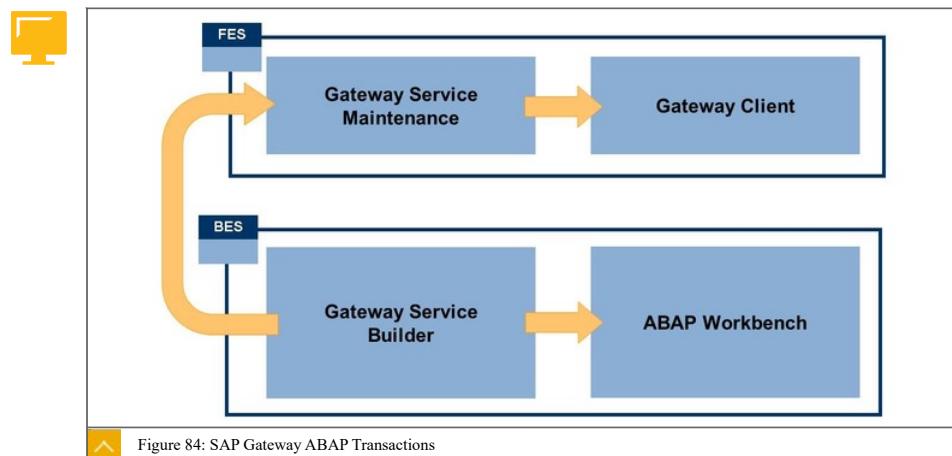
Figure 82: SAP Fiori Principal App Installation

In this example of the Fiori app reference library, you see one of the principal apps is still available. Due to their creation history, there is one add-on for all principal apps for ERP for the FES and the BES. It is important to mention that the implementation of the principal apps is not delivered as part of a regular update of the back-end solution. Principal apps still exist because of compatibility reasons. All principal apps are also available in a new version as part of all Fiori apps. Most of them have a version 2 in the title or have been renamed.

Transactional Applications based on SAP Gateway



A Gateway service consists of ABAP classes and customizing elements in the FES, as well as a service catalog entry in the BES. The ABAP classes separate into two model provider classes (MPC) that provide the coding for the service metadata and two data provider classes (DPC) that provide the coding to handle service requests. The model and data provider classes are connected via customizing. The external service name defined in the customizing is then used to register the service in the service catalog on the FES.



All SAP Gateway transactions are connected to each other via forward navigation. The following are the most important transactions for SAP Gateway :

Gateway Service Maintenance (/IWFND/MAINT_SERVICE)

Central Gateway service management in the front end

Gateway Client (/IWFND/GW_CLIENT)

Gateway service test environment in the front end

Gateway Service Builder (SEGW)

Central Gateway development environment in the back end

ABAP Workbench (SE80)

Central ABAP development environment in the back end

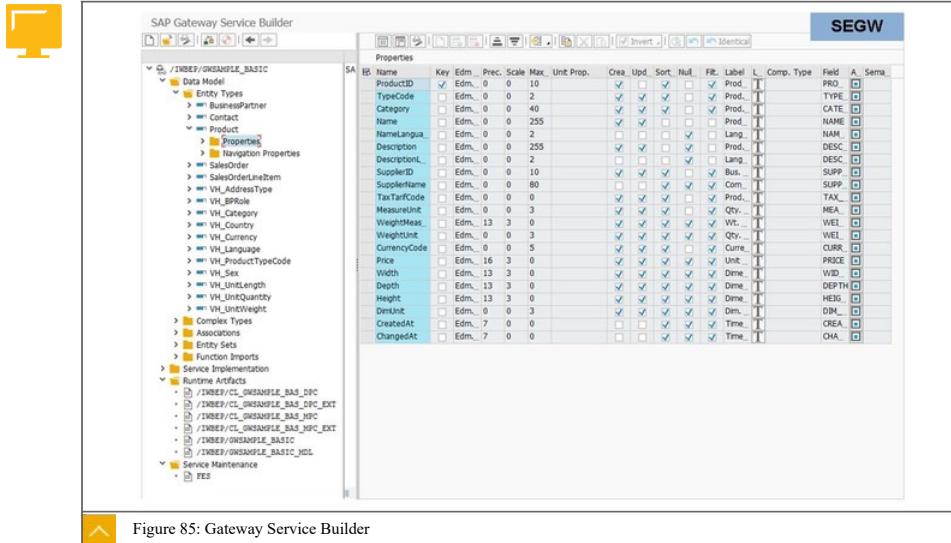


Figure 85: Gateway Service Builder

The service builder uses projects to bundle all artifacts of a service in one central place, which helps to organize the service development and modeling process. Since projects consolidate all related data, developers can easily work on multiple projects in parallel and reuse data between projects before generating and activating the actual service.



LESSON SUMMARY

You should now be able to:

- Examine SAP Fiori for ABAP Platform

Unit 3

Lesson 2

Examining SAP Fiori for SAP Business Suite on HANA

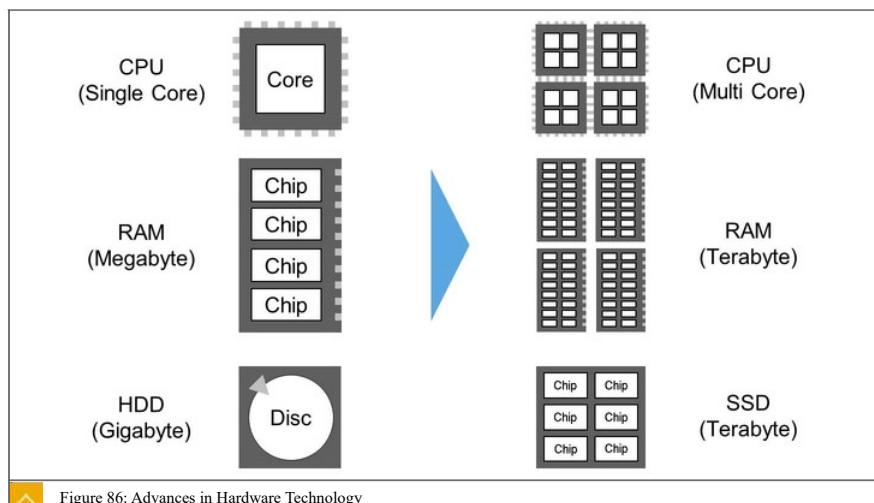


LESSON OBJECTIVES

After completing this lesson, you will be able to:

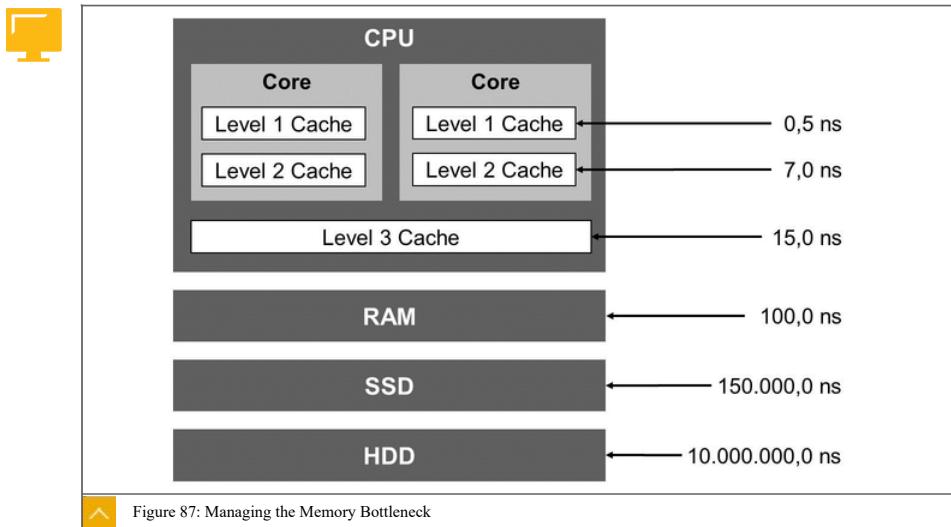
- Examine SAP Fiori for SAP Business Suite on HANA

SAP HANA



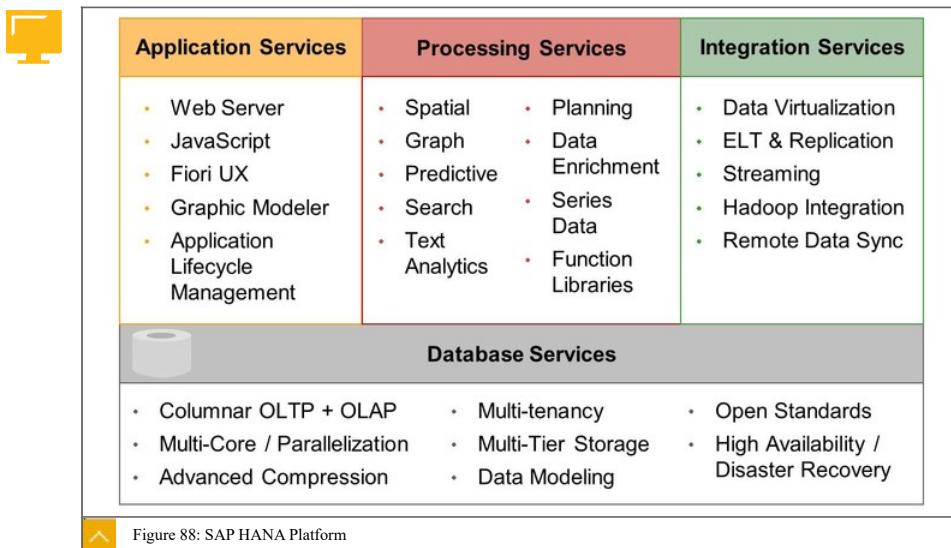
Historically, the amount of memory was limited because of its high costs. This caused a serious bottleneck in the data flow from disk to the Central Processing Unit (CPU). Today, memory is no longer a problem. SAP HANA runs on hardware with many terabytes of memory. With so much memory available, entire databases of large organizations can be stored completely inside memory. Without the need for mechanical spinning disks, data becomes accessible instantly without waiting time.

In addition to huge memory, the processors continue to improve at a phenomenal rate. There are currently high speed multi-core processors that can take on multiple complex tasks and process them in parallel. This means response times for even complex analytical tasks such as predictive analysis can be carried out in real time.



Modern processors have the ability to manage on-board memory, so-called CPU cache. The new generation of CPUs manage on board cache in hierarchical layers. A nanosecond is one billionth of a second. One nanosecond is to one second, as one second is to 31.71 years. So, a nanosecond is fast, and it is the measure of how processor speeds need to perform for modern application in a leading business operating in the digital world.

From the very beginning of SAP HANA development in 2008, SAP worked closely with CPU manufacturers in co-development projects to understand how to exploit all the power from their next generation processors. In particular, we needed to understand how data moves from memory to core, so we could code in exactly the right way to get the most out of the processors.



SAP HANA is a platform that is built on modern hardware. The SAP HANA platform combines all key components, such as database, data processing, app development, lifecycle management, and data integration, that can be used to power any application (both SAP and non-SAP).

Application Services

SAP HANA provides many application services. This means applications can be built in a two-tier model rather than a three-tier model. SAP HANA can handle the business logic as well as the database services. It provides a full development environment with a productivity tool supplied in the box. Everything the developer needs at design and runtime is there.

Processing Services

SAP HANA can handle many new types of data. This includes text, spatial, graph, and more. But it is not enough to simply store these new data types to be able to build applications that can process and integrate this data with traditional data types such as business transactions. SAP HANA provides native in-memory engines that process any type of data in real time.

Integration Services

SAP HANA has multiple data consumption options built in. It can analyze continual streaming data, read data remotely in any data source, and read big data stores like devices in the area of Internet of Things (IoT). SAP HANA has built in Extraction, Transformation, and Loading (ETL) capabilities so that separate software is no longer needed to clean, enrich, and profile data from any source.

Database Services

SAP HANA is a full in-memory column and row store database that can support both Online Transactional Processing (OLTP) and Online Analytical Processing (OLAP) requirements, and it is built to run on high-end hardware. It stores data optimally using automatic compression and can manage data on different storage tiers to support data ageing strategies. It has built-in high availability functions that keep the database running and ensure mission critical applications are never down.

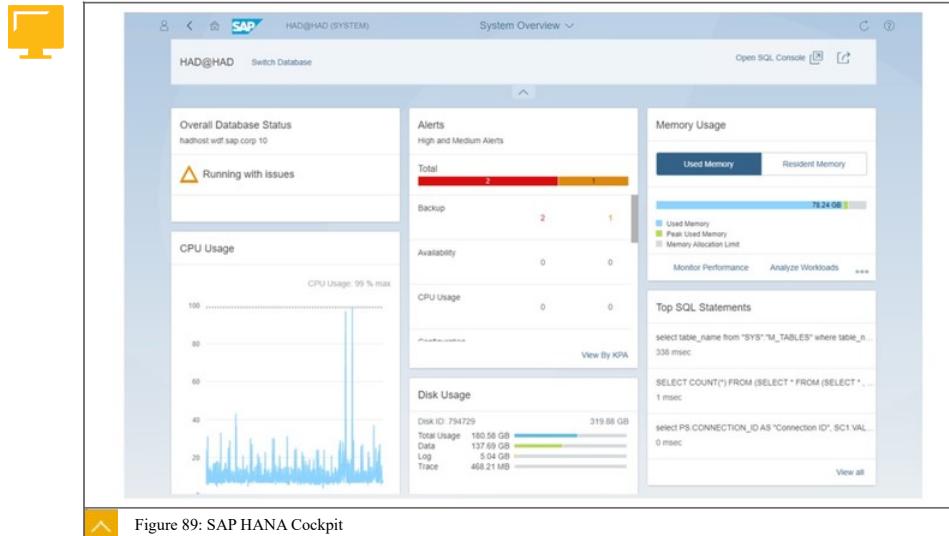


Figure 89: SAP HANA Cockpit

The SAP HANA cockpit provides tools for the administration and monitoring of SAP HANA databases (resources), and for development capabilities through the SAP HANA database explorer. Administrators can use the SAP HANA cockpit to start and stop services, to monitor the system, to configure system settings, and to manage users and authorizations.

When you first launch the SAP HANA cockpit, you can see system and tenant databases and you can drill down to perform in-depth monitoring on an individual system or tenant. To see alerts and other data for this individual resource, you need to enter database user credentials.

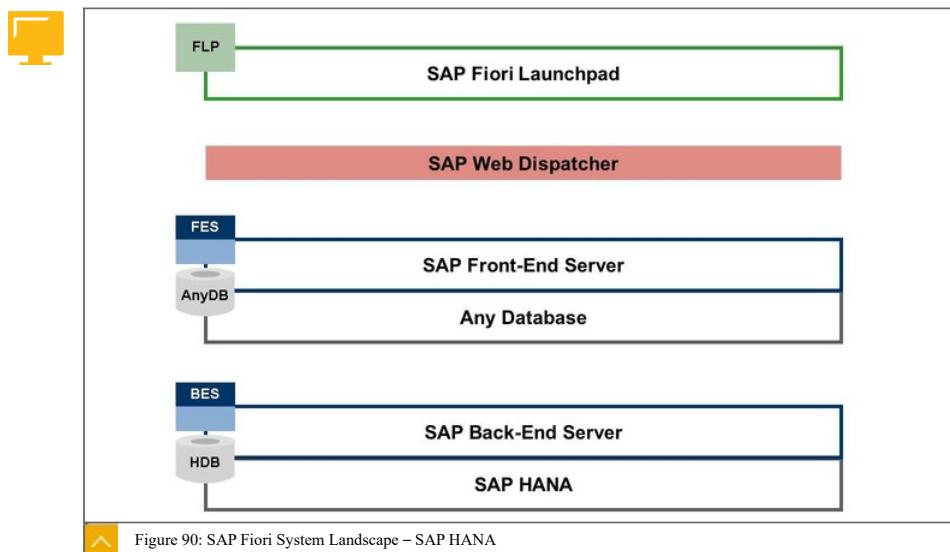
The SAP HANA database explorer is integrated into the SAP HANA cockpit. It allows you to query information about the database using SQL. You can also view information about your database's catalog objects and look into the database diagnostic files.



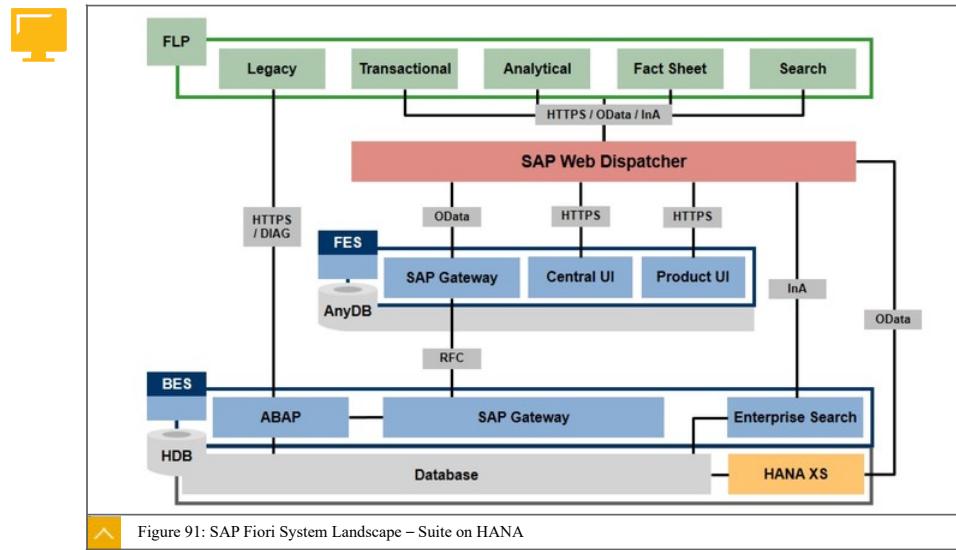
Note:

SAP HANA cockpit runs on an SAP HANA Express database, which is included in the installation. The SAP HANA cockpit can also be installed in an existing SAP HANA system in a separate tenant database (shared database).

Landscape of SAP Fiori for SAP Business Suite on HANA

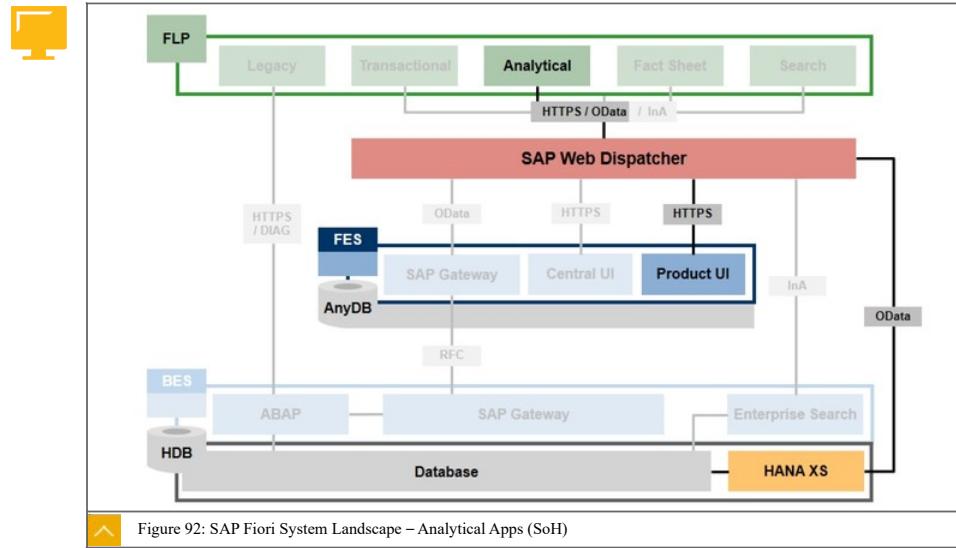


The general system roles in the SAP Fiori system landscape do not change when moving to SAP HANA. The SAP Fiori launchpad, running in a client, still connects to the FES via the SAP Web Dispatcher. What changes are the capabilities of the BES, which is now an SAP Business Suite on HANA.



Besides the legacy and transactional apps, with a Suite on HANA (SoH) , analytical and fact sheet apps are now available together with the SAP Fiori search on back-end data. The Web Dispatcher becomes more important because analytical apps get their data directly from the SAP HANA extended application services (XS) , and the fact sheet apps and the SAP Fiori search get their data from the SAP HANA enterprise search in the BES. Following the three application types, transactional, analytical, and fact sheet, there are now three system architecture types. The first one, transactional, stays the same as for any DB.

SAP HANA Extended Application Services



In the second architecture type, the UI for analytical apps resides on the FES as for all other SAPUI5-based apps. The data is still delivered to the client via OData but no longer originates in ABAP as an SAP Gateway service. The origin is now the SAP HANA extended application services (XS), which offers .xsodata-files based on data views in SAP HANA. The ABAP part of the BES does nothing when a request from an analytical app is directed to the SAP HANA XS. The application services in the SAP HANA XS work independently including logon and user management.

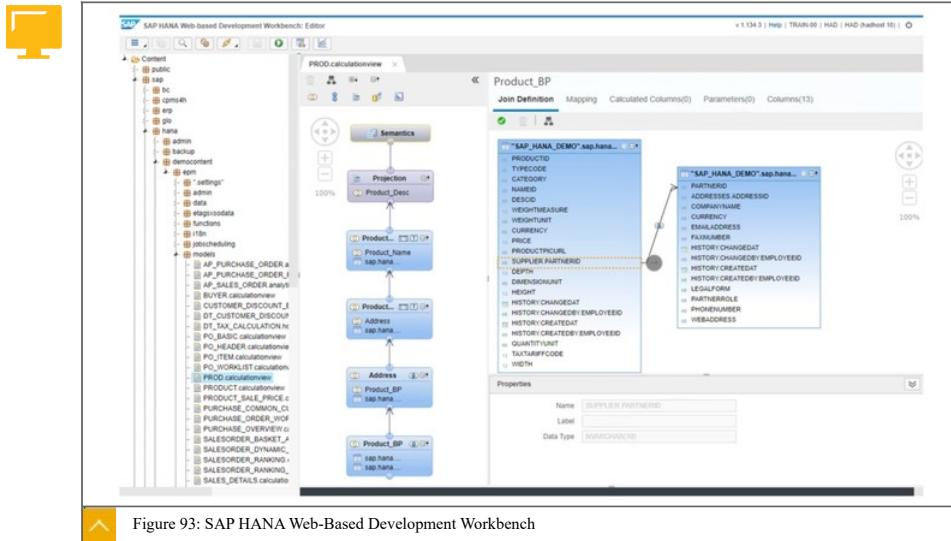


Figure 93: SAP HANA Web-Based Development Workbench

SAP HANA XS provides the SAP HANA Web-based Development Workbench that you can use to build and test development artifacts in the SAP HANA environment. It provides the following tools:

Catalog

Create, edit, execute, and manage SAP HANA DB SQL catalog artifacts.

Editor

Create, edit, execute, debug, and manage SAP HANA repository artifacts.

Security

Create users, create roles, assign objects, and manage security.

Traces

Set trace levels, view and download traces for SAP HANA applications.

The SAP HANA Web-based Development Workbench allows you to develop entire applications in a Web browser without having to install any development tools and is therefore a quick and easy alternative to the SAP HANA studio for developing native applications for SAP HANA XS. It is accessible via the following URL:

<http://<HANAsystem>:<port>/sap/hana/ide>



Note:

The SAP Web IDE for SAP HANA is another tool for developing native SAP HANA applications. It is designed to support developers who use SAP HANA extended application services, advanced model (XS Advanced), which has to be installed separately.



My Quotation Pipeline [F0028]

The app consists of front-end components (such as the user interfaces) and back-end components (such as the OData service). The back-end and front-end components are delivered with separate products and have to be installed in a system landscape that is enabled for SAP Fiori.

HANA XS Components

Product Version	SAP SMART BUSINESS FOR ERP 1.0
Support Package Stack	SAP Smart Business 1.0 for SAP ERP 05 (01/2015)
Product Instance(s)	Quotation Apps
Software Component Version	HCO_HBA_A_QTPL_S 100
Prerequisite for installation	SAP SMART BUSINESS FOR ERP 1.0 - SPS 05 (01/2015) is an Add On to SAP ANALYTICS FOUNDATION 1.0 - SPS 02 (01/2015)

Configuration

The following sections list app-specific data required to configure the app:

OData Services

oData Service	Package
qtpl_xsodata	sap_hba_apps_qtpl_s.odata

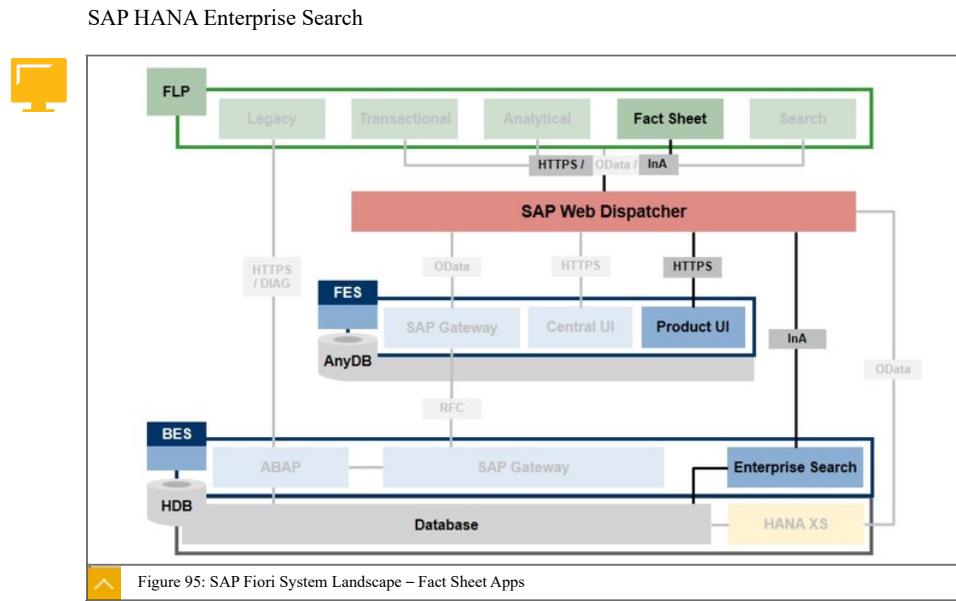
SAP HANA Roles

Role	Package
sap_hba_apps_qtpl_s.roles:qtpl	sap_hba_apps_qtpl_s.roles

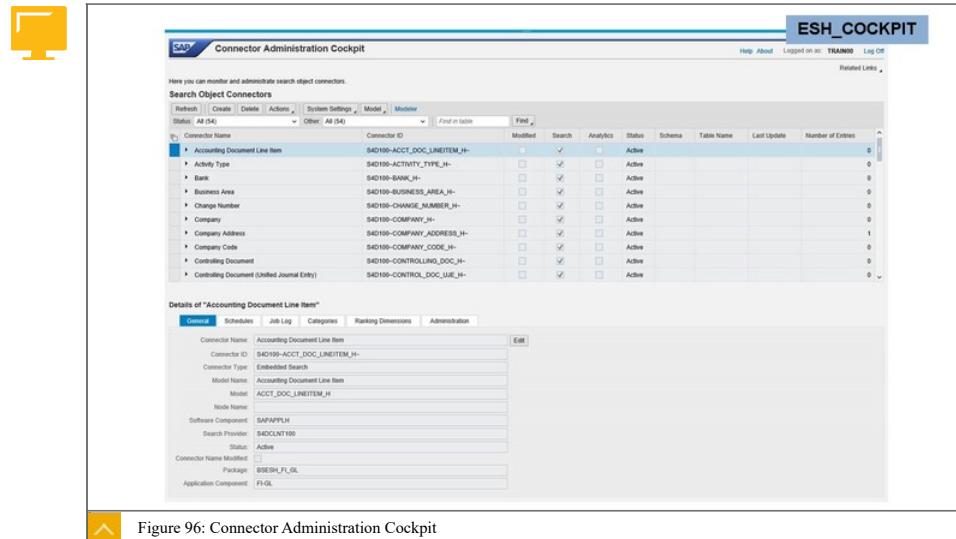


Figure 94: SAP Analytical App based on SAP HANA XS

In this example from the Fiori app reference library, you see HANA XS components are part of the installation. These have to be installed in SAP HANA. In addition, an OData service based on an .xsodata-file and a section showing SAP HANA roles to access the data is under configuration. Remember that the SAP HANA XS is independent of ABAP, including users and authorizations.



In the third architecture type, again, the UI for the fact sheet apps are on the FES. The data here originates from the Enterprise Search in the BES. The protocol used for network communication is the SAP proprietary Information Access (InA) protocol, which has some parallels to OData but is specialized for search requests and responses. The original SAP NetWeaver enterprise search was connected to SAP Search and Classification (TREX). The SAP HANA enterprise search is the functional successor, using SAP HANA as search engine.



Fact sheet apps use search models developed for SAP HANA enterprise search. These search models use search capabilities of SAP HANA and, therefore, do not run on a TREX. The administration and handling is mainly the same when using SAP HANA or TREX. All transactions and applications work for both search engines. The following are some important ones:

Connector Administration Cockpit (ESH_COCKPIT)

Connector administration cockpit for connectors of search models to the search engine

Search and Analytics Modeler (ESH_MODELER)

Modeler for search and analytic for managing search object connector models

Enterprise Search Test (ESH_TEST_SEARCH)

Test environment for enterprise search checking consistency of connectors and search results

Search and Operational Analytics Implementation Guide (ESH_IMG)

Area of the Implementation Guide (IMG) containing administration and configuration of the enterprise search

Search models are shipped by SAP and are the basis for search connectors used by apps.

Search connectors are indexed (generated) once the search models are available in a system. Therefore, the search connector ID contains the system and client in which it was generated.

Bill of Material [F0640]

Back-End Components (ABAP)

Product Version	EHP7 FOR SAP ERP 6.0
Support Package Stack	SAP enhancement package 7 for SAP ERP 6.0
Product Instance(s)	09 (07/2015)
Software Component Version	Master Data Governance
	SAP_APPL_617

Configuration

The following sections list app-specific data required to configure the app:

OData Services

The following OData services must be activated on the front-end server. Users require PFCG authorization for the front-end and back-end systems.

oData Service	Version	Back-End Authorization Role (PFCG)
CB_BILLOFMATERIAL_SRV	001	SAP_PLM_BILLOFMATERIAL_APP

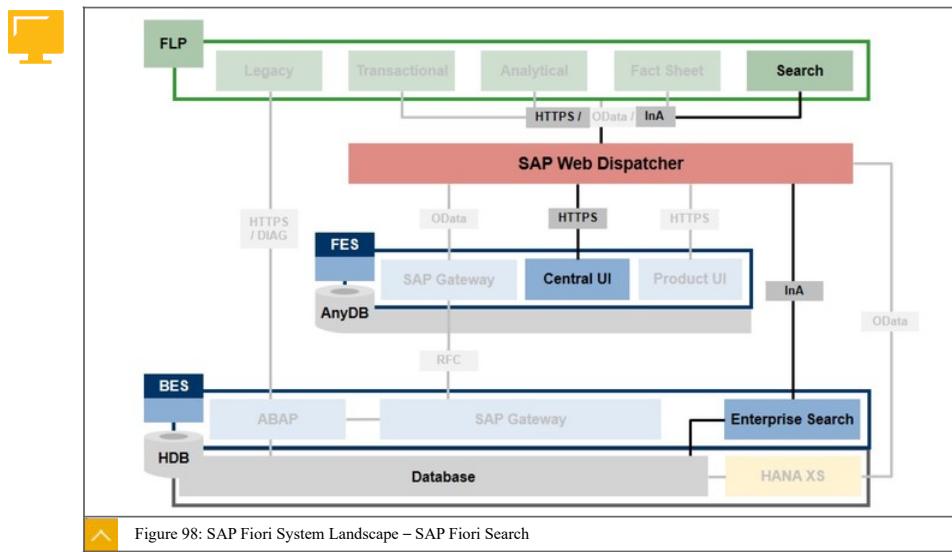
Search Connector

The following search connectors need to be activated in the back-end system:

Search Component (Software Component)	Application Component	Search Model
SAPAPPL	PLM-FIO-BOM	MBOM_APPL_MODEL_H

Figure 97: SAP Fact Sheet App Based on Enterprise Search

In this example from the SAP Fiori apps reference library, you see back-end components are also available for transactional apps. From an installation perspective, it does not make a difference whether you install transactional or fact sheet apps. The differences come with configuration. Here, besides existing authorizations to access data in the database, additional authorizations are needed for users to access data coming via the SAP HANA enterprise search. Finally, the section Search Connector lists the search components and search models needed for the app. It can also be the case that multiple search models are needed for one app.



When users want to search for data in the SAP Fiori launchpad, the SAP Fiori search accesses the Enterprise Search in the same way as fact sheet apps. A request using the InA protocol is sent to the BES and performed by the SAP HANA enterprise search and its search models on SAP HANA.

LESSON SUMMARY

You should now be able to:

- Examine SAP Fiori for SAP Business Suite on HANA

Unit 3

Lesson 3

Examining SAP Fiori for SAP S/4HANA



LESSON OBJECTIVES

After completing this lesson, you will be able to:

- Examine SAP Fiori for SAP S/4HANA

SAP S/4HANA

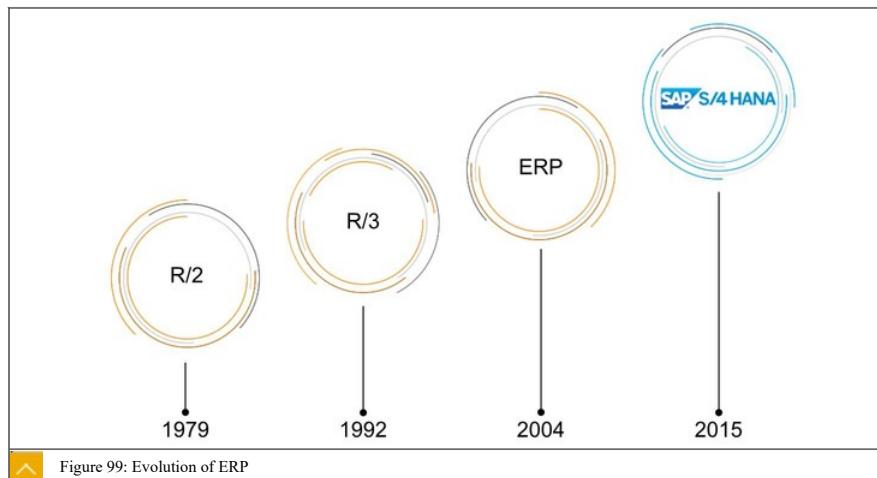
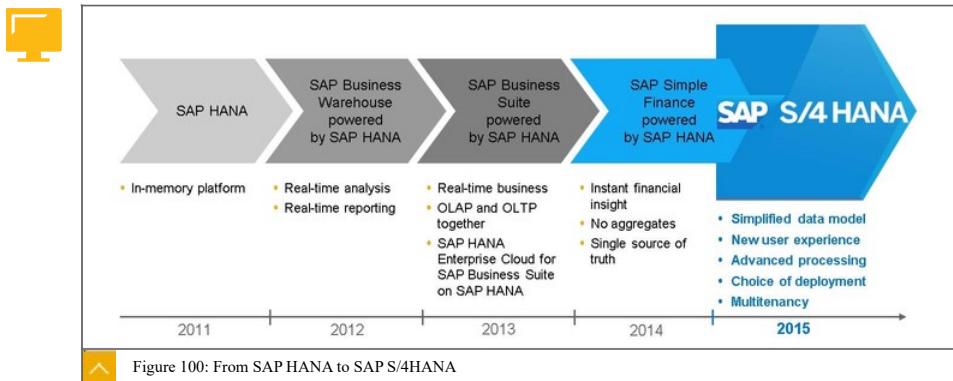


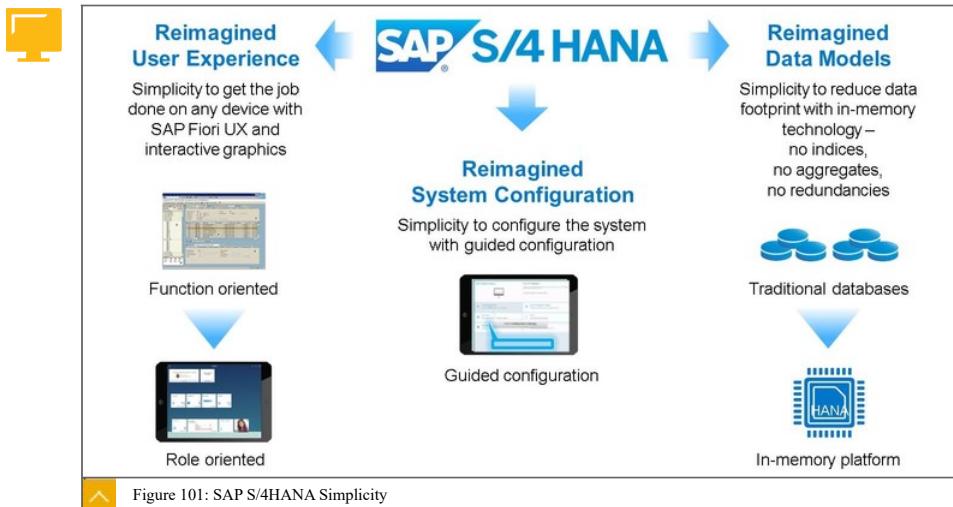
Figure 99: Evolution of ERP

In SAP R/2, SAP started to deliver screens to specialists in order to enter information into the system. These users were experts in their fields, highly-skilled, and well-trained to use the system. Today, even the operation of an MRP Run is more like controlling a machine or using an interface to a business function than an intuitive and self-explaining UI.

With SAP S/4HANA, SAP is executing the vision to empower every customer employee to use SAP business software. The "S" in S/4HANA stands for Simple, but it also stands for Suite. The "4" stands for fourth generation. The complete name is SAP Business Suite 4 for SAP HANA (S/4HANA).



SAP S/4HANA is the next-generation business suite to help lines of business and industries run simple, with all that only SAP HANA can do. SAP S/4HANA combines the most recent innovations (SAP HANA platform and SAP Fiori UX) with over 40 years of experience in mastering complex industry challenges in a new suite that caters to a digital, networked economy. The first part of S/4HANA that was available was SAP Simple Finance, the first optimized solution.

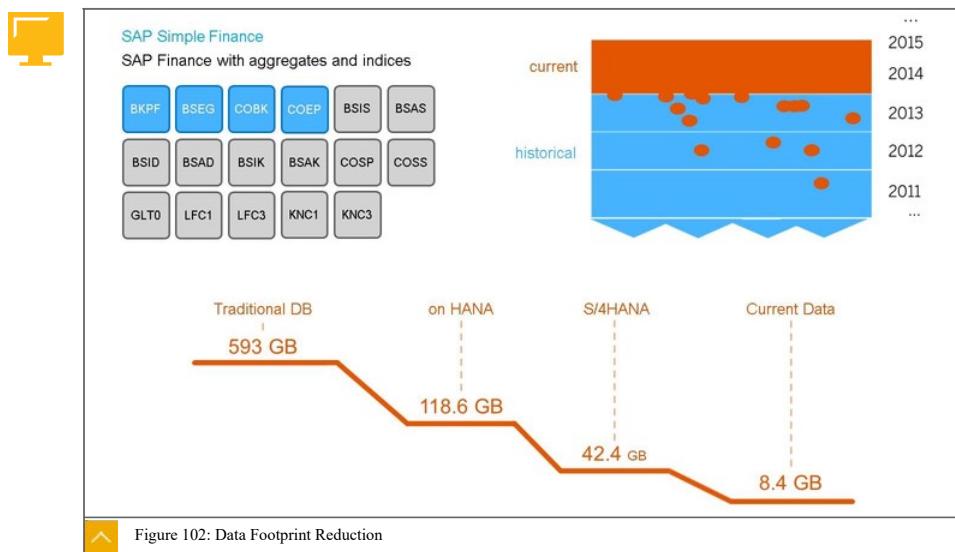


From an IT value perspective, SAP S/4HANA creates unique opportunities to dramatically simplify the landscape and reduce TCO, with SAP HANA as the great simplifier. Business users can leverage a simple and role-based user experience based on modern design principles, minimizing training efforts while increasing productivity. Guided configurations ease the creation of settings in areas from system administration to application customizing. Enterprises can now significantly reduce their data footprint and work with larger data sets in one system to reduce hardware and operational costs and save time.

SAP S/4HANA Key Facts

The following are key facts about SAP S/4HANA :

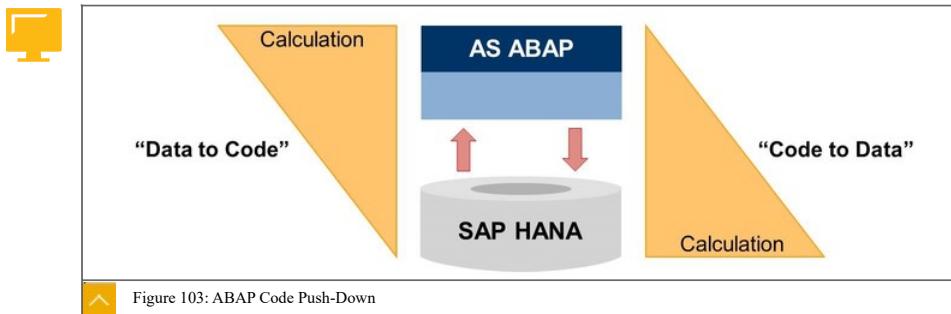
- Faster analytics and reporting
- Fewer process steps
- ERP, CRM, SRM, SCM, and PLM co-deployed
- All data: social, text, geo, graph, and processing
- No locking, parallelism
- Actual data (25%) and historical (75%)
- Smaller total data footprint
- SAP Fiori UX for any device



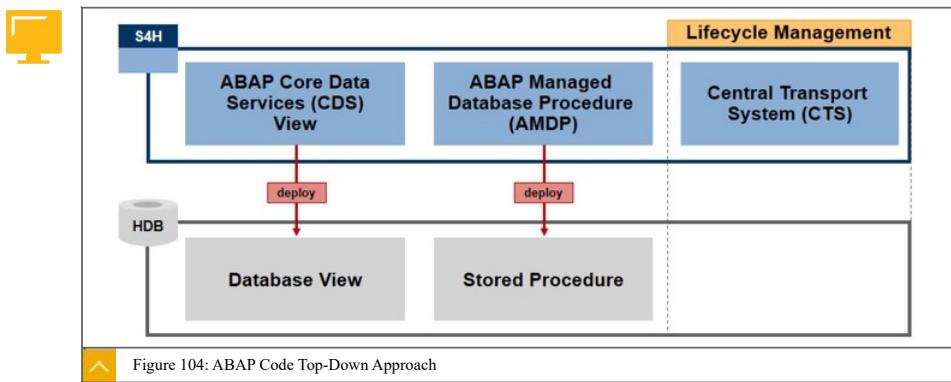
Tables in SAP HANA are automatically compressed so it is possible to store huge amounts of data in a small space. Further data footprint reductions are achieved by removing unnecessary tables and indexes, which were needed in classic databases to achieve response times in a meaningful runtime. These aggregates and redundancies are no longer needed in the architecture of SAP HANA, so they were removed in the new development of solutions. In SAP Simple Finance, that means a reduction of the number of database tables from 17 to just 4.

The last step is to reduce the in-memory data footprint by implementing data-ageing strategies. SAP HANA can split data across an in-memory tier (referred to as HOT storage) and a disk-based tier (referred to as WARM storage). A developer does not need to know where the data is physically located; this is handled internally by SAP HANA. The benefit is that data that is used less frequently can be moved automatically from HOT to WARM storage, so memory is not filled with data that is less useful. However, this data is still available whenever it is needed.

Development Paradigm Shift



SAP HANA is more than just a database. SAP HANA can perform calculations on a data level much faster than any previous technology - including ABAP programs. In the SAP Business Suite, the data was brought to the code, which means it was read from the database and processed in the AS ABAP. With SAP HANA, it is more efficient to bring the code to the data, meaning to push down calculations from AS ABAP in SAP HANA and only transfer the results back.



Moving calculations to the database to benefit from its features is not new to SAP HANA. Stored procedures could be used previously for calculations in many databases. However, they were rarely used in ABAP because they only run in the database they are developed in and are not transportable. With SAP S/4HANA, the ABAP Repository was extended with two new database objects: ABAP Core Data Services (CDS) Views and ABAP Managed Database Procedures (AMDP). CDS Views are deployed as HANA views in the HANA Database (HDB), and the AMDPs are deployed as stored procedures. This is comparable to deploying a transparent table of the ABAP repository as a database table in any DB.

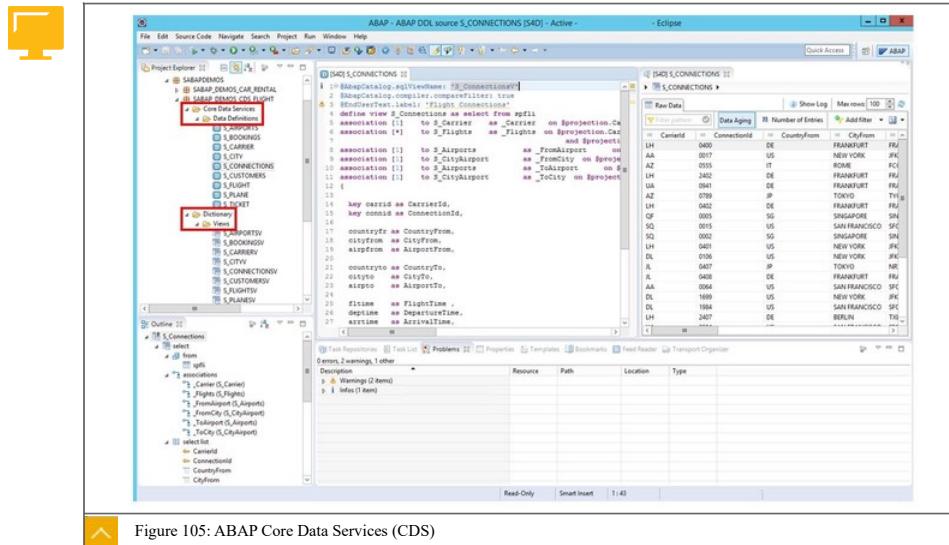
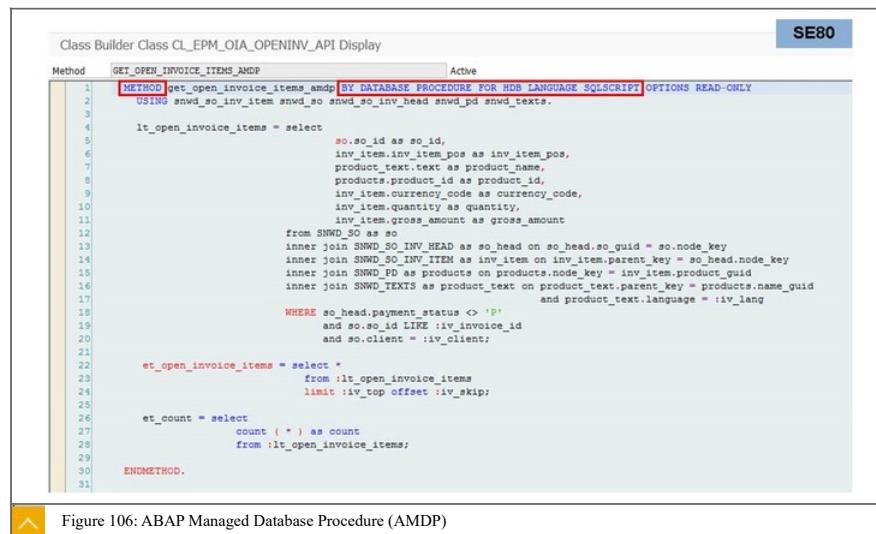


Figure 105: ABAP Core Data Services (CDS)

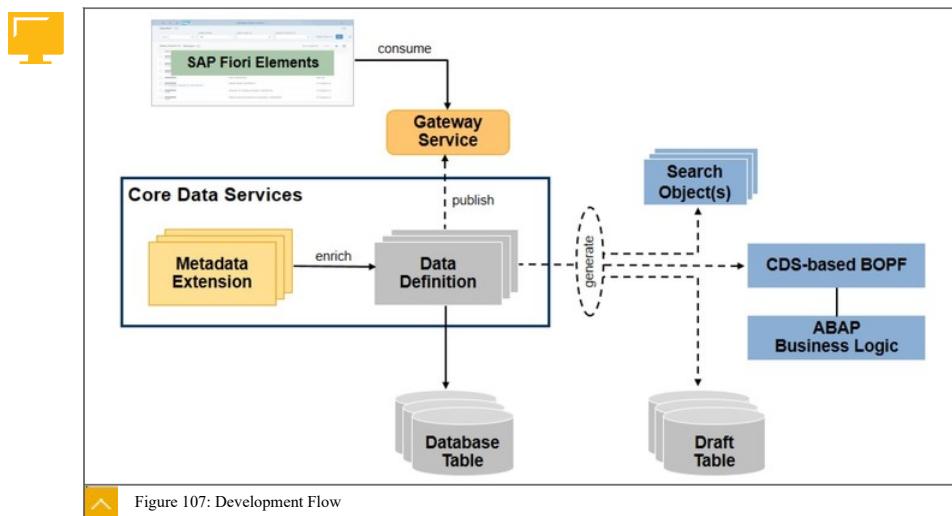
On one hand, ABAP CDS Views are classic database views that are compatible with any DB. On the other hand, they include side elements such as annotations, which can only be interpreted by a HANA database. Although CDS Views are stored in the ABAP repository, they can only be viewed using transaction SE80 in the SAP GUI. CDS Views are developed using ABAP Development Tools (ADT) in Eclipse.



AMDPs can be developed using the SE80 in SAP GUI and the ADT. An AMDP is a method of a global class containing SQLScript as the programming language. The set of SQL extensions for the SAP HANA database that allow developers to push data intensive logic into the database is called SQLScript. Conceptually, SQLScript is related to stored procedures as defined in the SQL standard, but SQLScript is designed to provide superior optimization.

possibilities. SQLScript should be used in cases where CDS Views are not sufficient, for example, for complex selections or calculations.

AMDPs can be identified by the syntax `BY DATABASE PROCEDURE FOR HDB LANGUAGE SQLSCRIPT` behind the name of the method. Classes containing AMDPs implement the marker interface `IF_AMDP_MARKER_HDB`.



Since SAP S/4HANA 1610, the complete development flow is moving in the direction of metadata-driven development with CDS Views in the center. The pure data definition for accessing the database tables is enriched by metadata extensions consisting of annotations, which can be part of the CDS View or organized in separate files. These annotations can access search objects, call CDS-based BOPF (Business Object Processing Framework) for CRUD support, or enable the draft concept in Fiori apps, for example.

For the UI part, annotations are used to publish the Gateway service to be consumed by apps based on SAPUI5. If using SAP Fiori elements, the structure and behavior of the UI is completely managed by the metadata extensions. This development flow is called ABAP Programming Model for SAP Fiori.

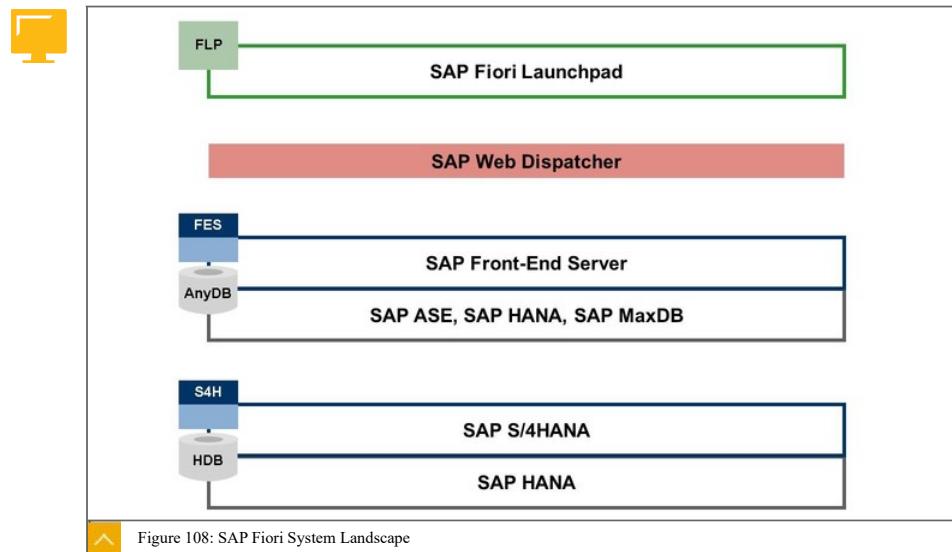


Note:

More information about this topic can be found in S4DEV (Introduction to Application Programming on SAP S/4HANA):

<https://training.sap.com/course/s4dev>

Landscape of SAP Fiori for SAP S/4HANA



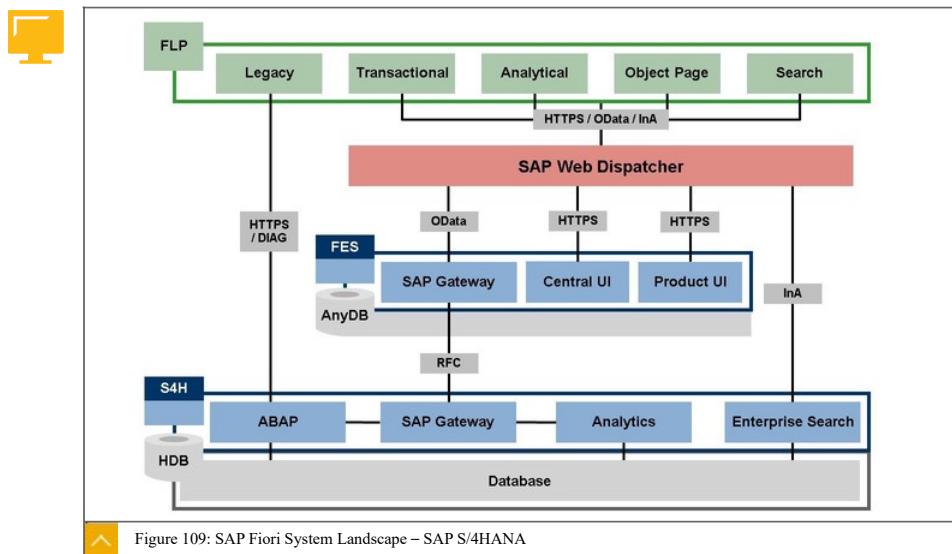
The general system roles in the SAP Fiori system landscape do not change when moving to SAP S/4HANA. The Fiori launchpad running in a client still connects to the FES via the SAP Web Dispatcher. What changes are the capabilities of the BES, which now has optimized coding for SAP HANA as an SAP S/4HANA. General information about the FES can be found in the following SAP notes:

- SAP Fiori Front-End Server 2.0 — General Information: SAP Note [2219596](#)
- SAP Fiori Front-End Server 3.0 — General Information: SAP Note [2355644](#)
- SAP Fiori Front-End Server 4.0 — General Information: SAP Note [2484979](#)
- SAP Fiori Front-End Server 5.0 — General Information: SAP Note [2618605](#)
- SAP Fiori Front-End Server 6.0 — General Information: SAP Note [2775163](#)



Note:

The recommended deployment option for SAP S/4HANA is very dependent on the customer scenario. Therefore SAP provides the following document for this topic: <https://www.sap.com/documents/2018/02/f0148939-f27c-0010-82c7-ed471af511fa.html>



All application types available for SAP S/4HANA (S4H) are also available for SAP S/4HANA (S4H). The important difference is that there is only one architecture type for all application types. All applications act like transactional apps communicating via SAP Gateway over the FES now an S4H. Only the legacy apps and the Fiori search connect directly to the S4H.

Some may propose to remove the SAP Web Dispatcher and route the InA requests via an ICF-handler in the FES. However, this has the following two major drawbacks and is not recommended:

Bad Performance

The ICF handler needs more resources compared to a routing rule.

Bad Security

The FES cannot verify the request and response, which is part of the SAP Product standard requirements.

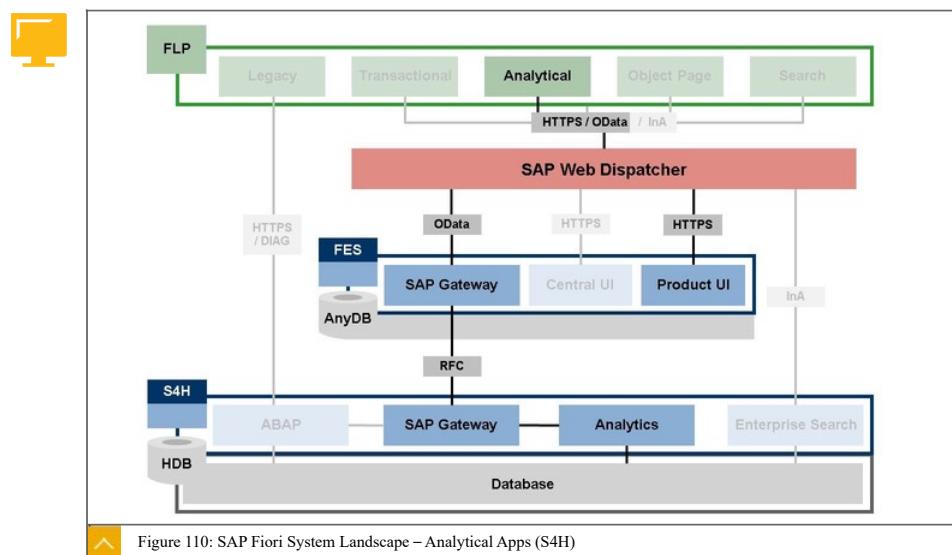


Figure 110: SAP Fiori System Landscape – Analytical Apps (S4H)

CDS Views and AMDPs are or will be used in every Fiori application in S4H. Analytical apps implement CDS Views combined with SAP S/4HANA Embedded Analytics, which originated from SAP Business Warehouse (BW). There is no longer a need to access data views via the HANA XS Engine, having now CDS Views as part of ABAP integrated with an analytics engine.

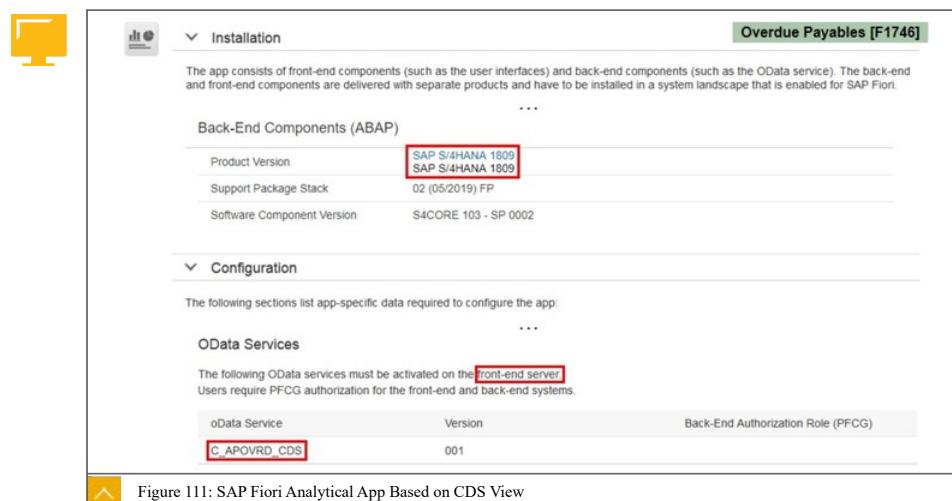
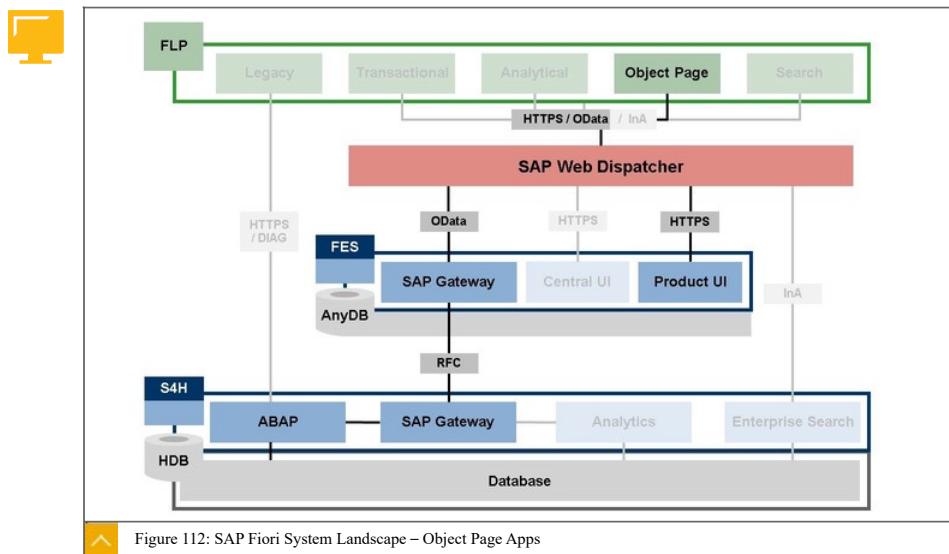


Figure 111: SAP Fiori Analytical App Based on CDS View

In this example from the Fiori app reference library, you see the SAP S/4HANA 1610 as a back-end component and no entry for HANA XS Engine. Instead, there is just an OData service in the FES containing the shortage CDS in its name.



The successor of the fact sheet called **object page** also uses CDS Views and AMDPs and utilizes the search capabilities of the HDB directly. Therefore, the **Enterprise Search** is no longer used here. Again, there is only one architecture type in **S/4HANA**. There is no longer a need to access search models in the BES via the protocol InA - except for the **Fiori search**. That is the main reason why the Enterprise Search is still used in **SAP S/4HANA**.



Note:

Another reason for the above is that not all fact sheets in the S4H already use CDS Views. There are still fact sheets originating from SoH available that use the Enterprise Search in S4H.

Supplier [F0354]

Installation

The app consists of front-end components (such as the user interfaces) and back-end components (such as the OData service). The back-end and front-end components are delivered with separate products and have to be installed in a system landscape that is enabled for SAP Fiori.

Back-End Components (ABAP)

Product Version	SAP S/4HANA 1809
Support Package Stack	SAP S/4HANA 1809
Software Component Version	02 (05/2019) FP
	S4CORE 103 - SP 0002

Configuration

The following sections list app-specific data required to configure the app:

OData Services

The following OData services must be activated on the **front-end server**. Users require PFCG authorization for the front-end and back-end systems.

oData Service	Version	Back-End Authorization Role (PFCG)
C_SUPPLIER_FS_SRV	0001	

Figure 113: SAP Fiori Object Page App Based on CDS View

UNIT 4

Configuration

Lesson 1

Creating SAP Fiori Groups	98
---------------------------	----

Lesson 2

Managing SAP Fiori Catalogs	103
-----------------------------	-----

Lesson 3

Creating SAP Fiori Catalogs	110
-----------------------------	-----

Lesson 4

Creating LPD_CUST Target Mappings	116
-----------------------------------	-----

Lesson 5

Creating SAPUI5 Target Mappings	120
---------------------------------	-----

Lesson 6

Creating Web Dynpro Target Mappings	122
-------------------------------------	-----

Lesson 7

Creating Transaction Target Mappings	125
--------------------------------------	-----

Lesson 8

Creating Back-End Catalogs	128
----------------------------	-----

Lesson 9

Configuring SAP Fiori Launchpad	131
---------------------------------	-----

Lesson 10

Configuring Analytical Applications	137
-------------------------------------	-----

UNIT OBJECTIVES

- Create SAP Fiori groups
- Manage SAP Fiori catalogs
- Create SAP Fiori catalogs
- Create LPD_CUST target mappings
- Create SAPUI5 target mappings
- Create Web Dynpro target mappings
- Create transaction target mappings
- Create back-end catalogs
- Configure SAP Fiori launchpad
- Enable SAP Fiori Smart Business
- Model SAP Fiori KPI Apps

Unit 4

Lesson 1

Creating SAP Fiori Groups

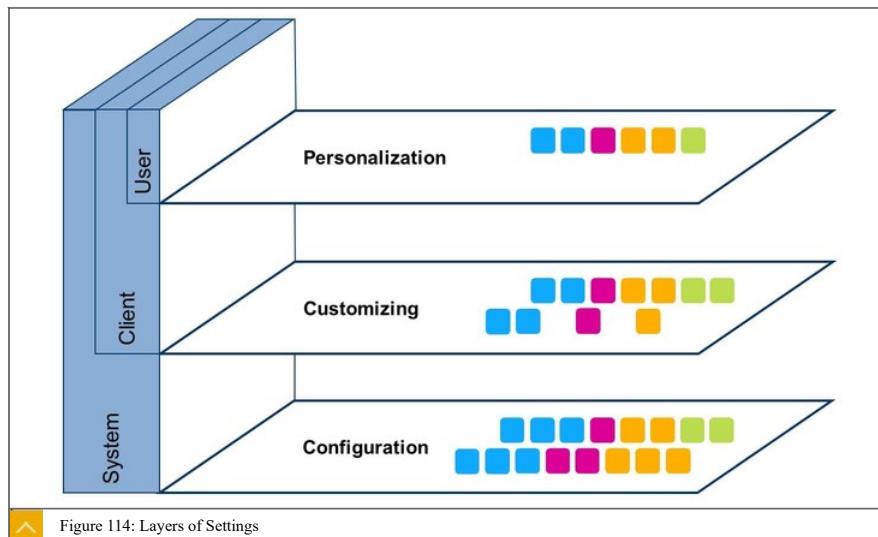


LESSON OBJECTIVES

After completing this lesson, you will be able to:

- Create SAP Fiori groups

SAP Fiori Launchpad Designer



The settings for the Fiori launchpad are organized in the three layers configuration, customizing, and personalization. The basic layer configuration consists of settings valid for the whole system. The layer customizing builds on the configuration and fine tunes the settings per client. End users can then personalize these settings in their personalization layer.

Every layer can only change the elements made available by their sublevel or reduce elements for usage in the next layer. So users can only personalize those apps made available in the customizing and originally provided by the configuration layer.

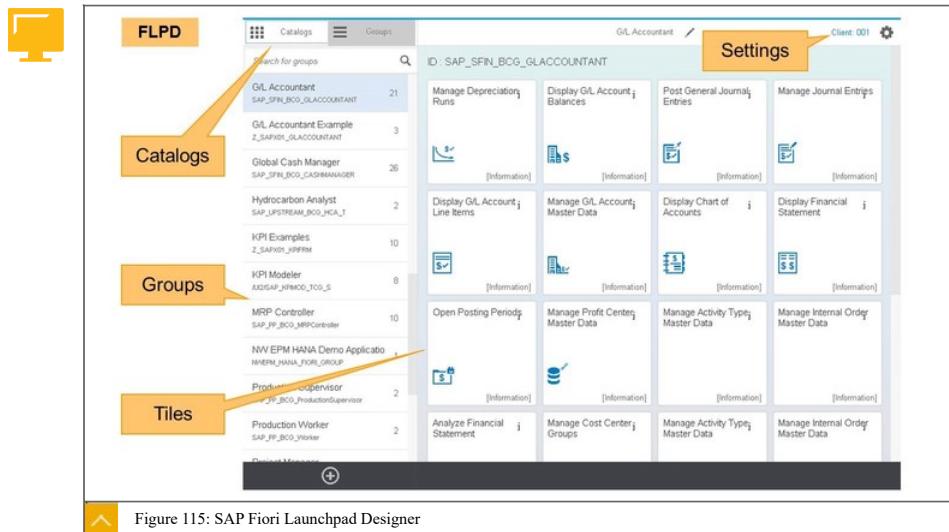
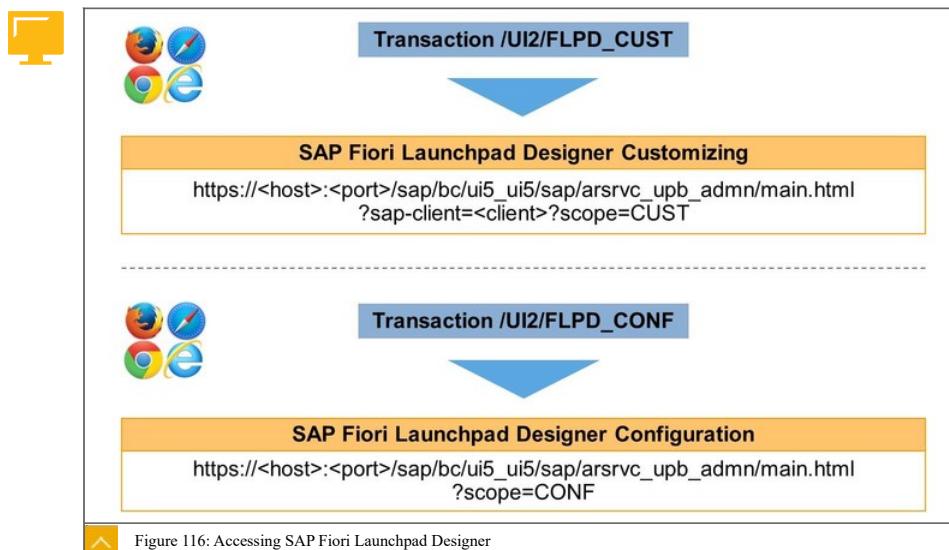


Figure 115: SAP Fiori Launchpad Designer

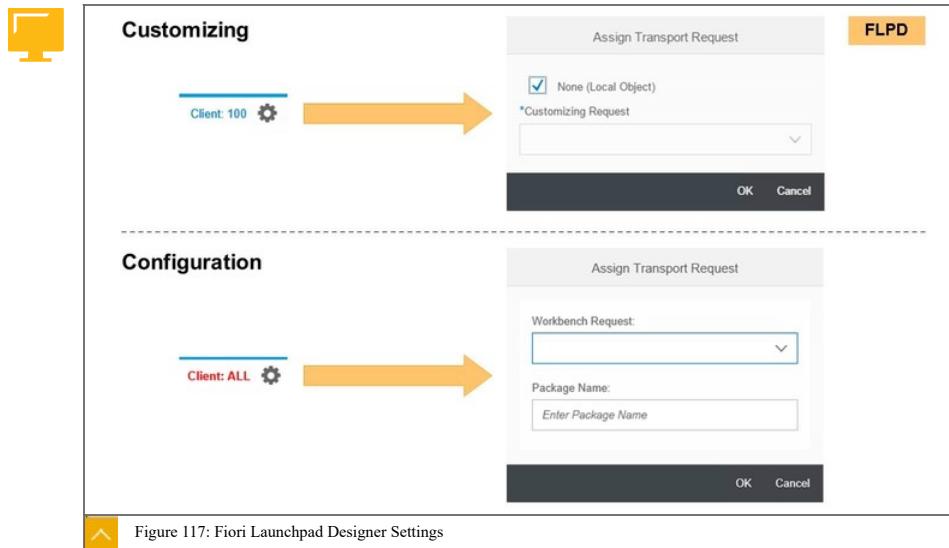
The personalization is performed by each user in the SAP Fiori launchpad. For customizing and configuration, the tool SAP Fiori Launchpad Designer (FLPD) offers an environment for both layers. It is an SAPUI5 application following the Fiori design guidelines, but not an SAP Fiori app documented in the SAP Fiori app reference library. It is part of the central UI of every AS ABAP since 7.40, as well as the SAP Fiori launchpad.



Depending on the host and port as well as on the client for customizing, the FLPD can be called using the following URI:

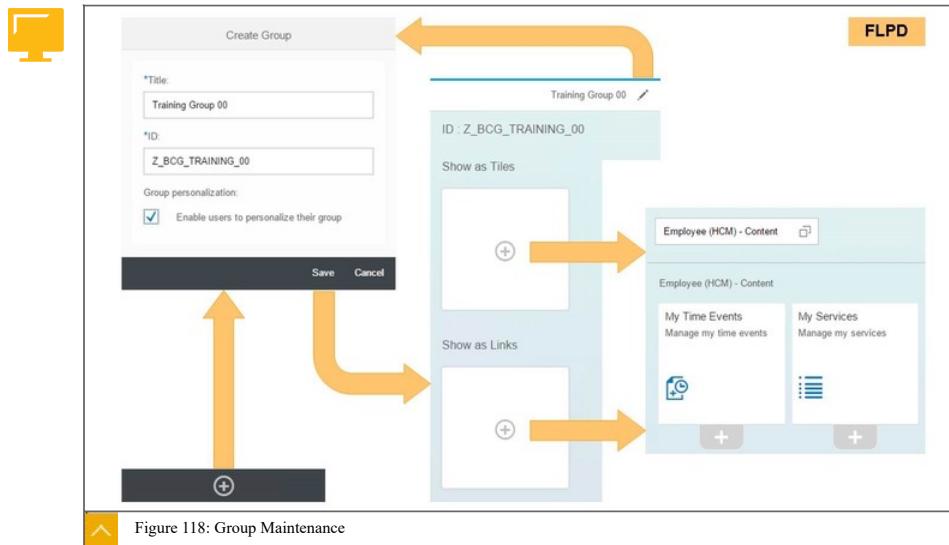
- https://<host>:<port>/sap/bc/ui5/ui5/sap/arsrv_upb_admn/main.html?sap-client=<client>?scope=CUST
- https://<host>:<port>/sap/bc/ui5/ui5/sap/arsrv_upb_admn/main.html?scope=CONF

The FLPD offers the same options for customizing per client and configuration per system. It is all about handling catalogs, groups, and tiles.



When using the FLPD for customizing, changes performed can be saved as local object for the user doing them (default setting) or a customizing request can be used. When using the FLPD for configuration, changes performed have to be saved in a certain development package and attached to a workbench request. Settings are accessible for customizing and configuration in the upper right corner of the FLPD.

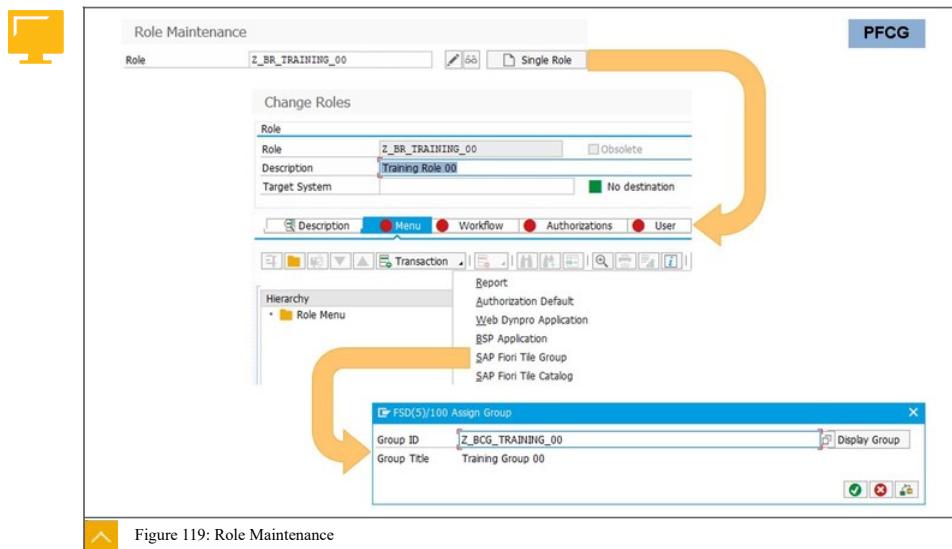
Group and Role Maintenance



Groups can be created by clicking the plus symbol in the lower left corner of the group overview. This opens a dialog box for entering a group title and technical ID and specifying if users are allowed to change the group using personalization. The group title and group personalization can be changed at a later point via the pencil next to the heading of the group, but not the group ID. Groups delivered by SAP follow the naming schema SAP_<area>_BCG_<topic>. Groups created by customers should start with **Z** and contain the shortage **BCG** (Business Catalog Group).



Caution:
The group ID has to be written in capital letters.



Groups can be added to user roles using the role maintenance transaction PF03. In the Menu tab, use the small triangle beside the Transaction button (or what was last added to the menu) and choose SAP Fiori Tile Group. In the dialog box that appears, enter the group ID and continue. The prerequisite for the user seeing the tiles of the group in the FLP is that the catalogs containing the tile definitions are also part of their user master record.



Hint:

If a newly created group is not available in the dialog box, restart the

PF03.



LESSON SUMMARY

You should now be able to:

- Create SAP Fiori groups

Unit 4

Lesson 2

Managing SAP Fiori Catalogs

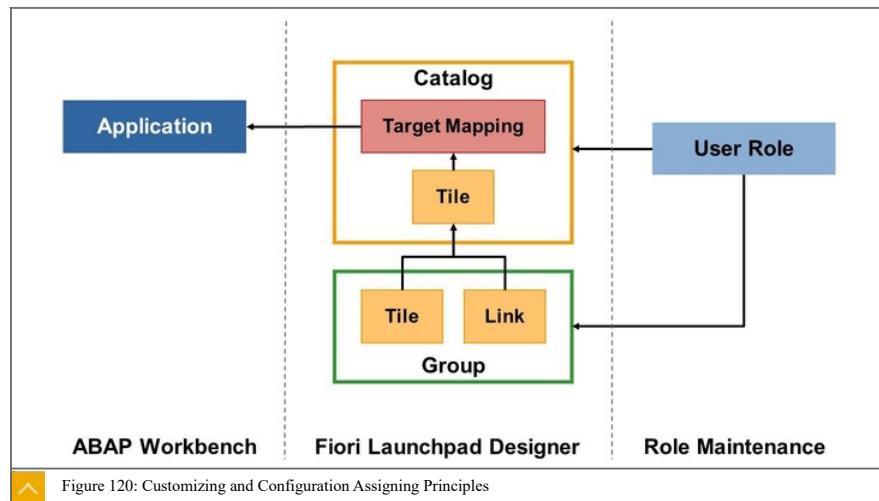


LESSON OBJECTIVES

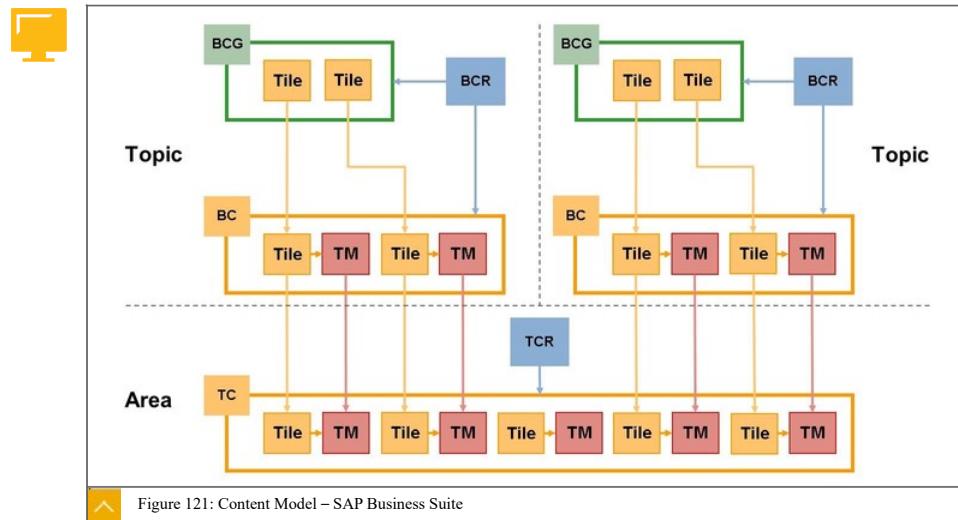
After completing this lesson, you will be able to:

- Manage SAP Fiori catalogs

Content Models



In personalization, groups and catalogs just visualize tiles or links for calling an application. In customizing and configuration, catalogs define target mappings holding all the information needed to actually start an application. All this information is accessible using the SAP Fiori launchpad designer .



There are two content models for SAP Fiori available. The first one is valid for SAP Business Suite on HANA as well as any other database. The following elements can be distinguished:

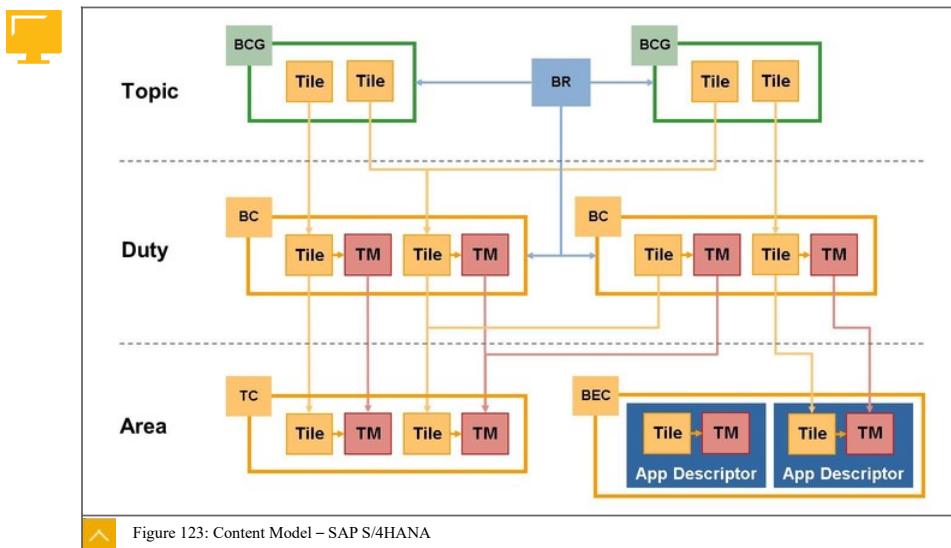
Element	Short	Naming Schema	Description
Business Catalog Role	BCR	SAP_<area>_BCR_<role>_T	Role for topic
Business Catalog Group	BCG	SAP_<area>_BCG_<role>_T	Tiles for topic
Business Catalog	BC	SAP_<area>_BC_<role>_T	Tiles for topic
Technical Catalog Role	TCR	SAP_<area>_TCR_T	Role for area
Technical Catalog	TC	SAP_<area>_TC_T	Tiles for area

Technical catalogs (TC) cut tiles and target mappings by technology per solution area, for example, fact sheet, analytical, and transactional apps. For each technical catalog (TC), a technical catalog role (TCR) exists.

Business catalogs (BC) reference tiles and target mappings of technical catalogs (TC) are per solution topic. Each business catalog (BC) has one business catalog group (BCG) and one business catalog role (BCR), which also includes the business catalog group (BCG).

Figure 122: Content Model – SAP Business Suite Example

In this example from the SAP Fiori app reference library, you see content model elements under Technical Configuration, starting with TC and TCR and continuing with BC, BCG, and BCR. Business elements all have the end user role in their name. The shortage makes the difference.



The second one is valid for SAP S/4HANA.

The following elements can be distinguished:

Element	Short	Naming Schema	Description
Business Role	BR	SAP_BR_<role>	Role for topic
Business Catalog Group	BCG	SAP_BCG_<topic>	Tiles for subtopic
Business Catalog	BC	SAP_BC_<topic>	Tiles for duty
Technical Catalog	TC	SAP_TC_<area>_<subarea>_COMMON	Tiles for area
Back-End Catalog	BEC	SAP_TC_<area>_<subarea>_BE_APPS	Legacy app descriptors for area

Technical catalogs (TC) continue to cut tiles and target mappings by solution area, but not by technology. Legacy apps are now available in back-end catalogs (BEC), which contain app descriptors. These are assignments of tiles and target mappings to transaction codes or Web Dynpro ABAP applications. Technical catalog roles (TCR) no longer exist.

Business catalogs (BC) reference tiles and target mappings of technical catalogs (TC) and back-end catalogs (BEC) according to segregation of duty. Business catalog groups (BCG) now contain apps from various business catalogs (BC), making a subtopic.

Business roles (BR) replace business catalog roles (BCR), combining multiple business catalogs (BC) and business catalog groups (BCG) in one topic.

In this example from the SAP Fiori app reference library, you see content model elements under Technical Configuration, starting with TC and continuing with BC, BCG, and BR. BC and BCR have the name of the topic in common, whereas the BR contains the end user role name.

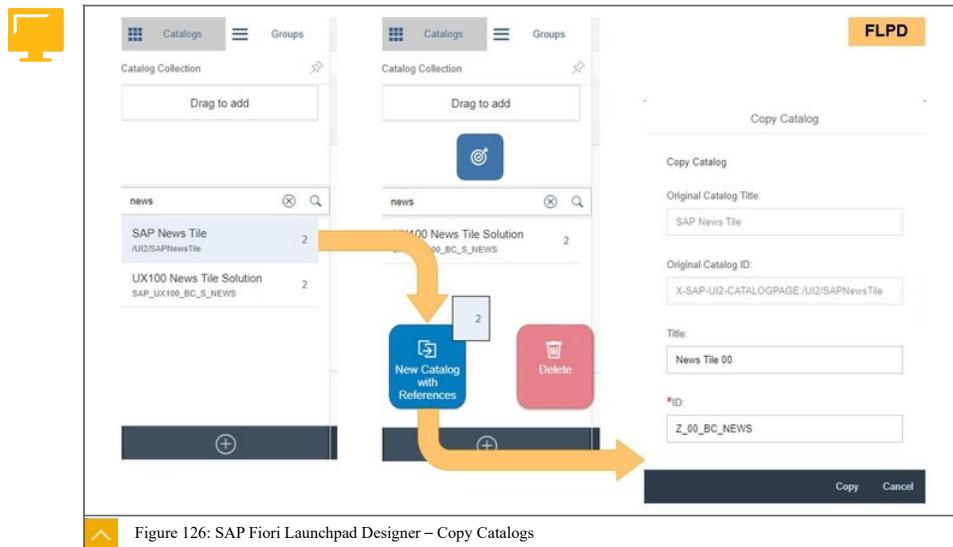
Catalog Management

Figure 125: Remote Catalogs

Back-end catalogs are available in the Back-End Server (BES). In a system scenario where there is a separated Front-End Server (FES), these are extracted by an administrator from the BESs to the FES. In the Fiori launchpad designer on the FES, they are visible as remote catalogs. A remote catalog is used in other systems as the source system they are defined in and can only be changed in the source system.

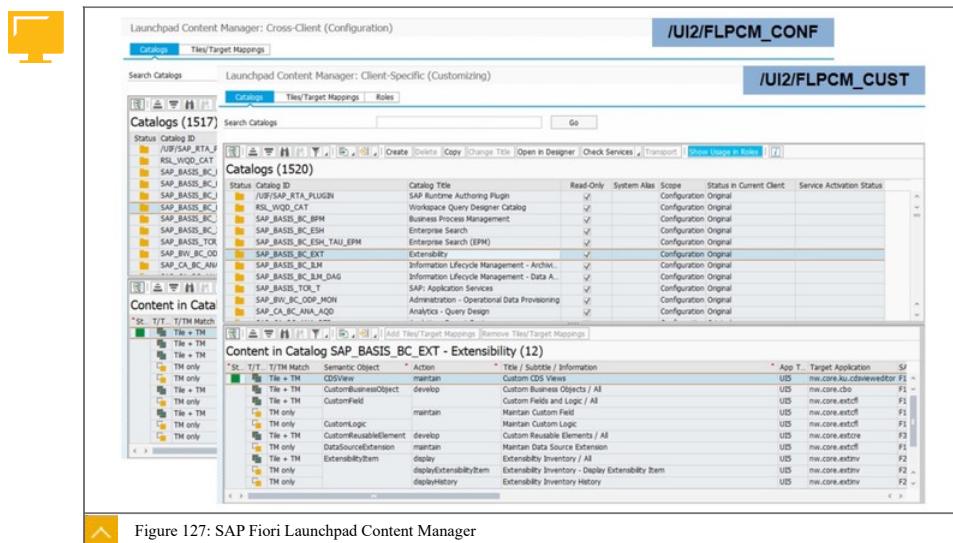
Remote catalogs can be used to load additional tiles from a remote system, but not for adding additional applications and target mappings to a Fiori installation. This means that a remote catalog can only provide new tile configurations, but not new application implementations. The application code has to be installed in the system to which the target mapping is pointing.

The remote catalog feature was mainly intended to support integration of smart business content from an SAP HANA XS Engine. In this scenario, the code for the smart business app has to be installed on the FES. However, the KPI tile definitions are created using the KPI modeler on the SAP HANA XS Engine. Another use case is the integration of SAP Jam tiles from the SAP Cloud Platform or having multiple FESs pointing to one central FES with custom catalogs.



Catalogs can be referenced in the SAP Fiori launchpad designer to create a copy of the catalog consisting of references to tiles and target mappings of the source catalog. It is a good starting point when defining own catalogs to start with referencing a catalog shipped by SAP. If SAP ships a new version of a catalog, all references immediately use the new version.

SAP Fiori Launchpad Content Manager

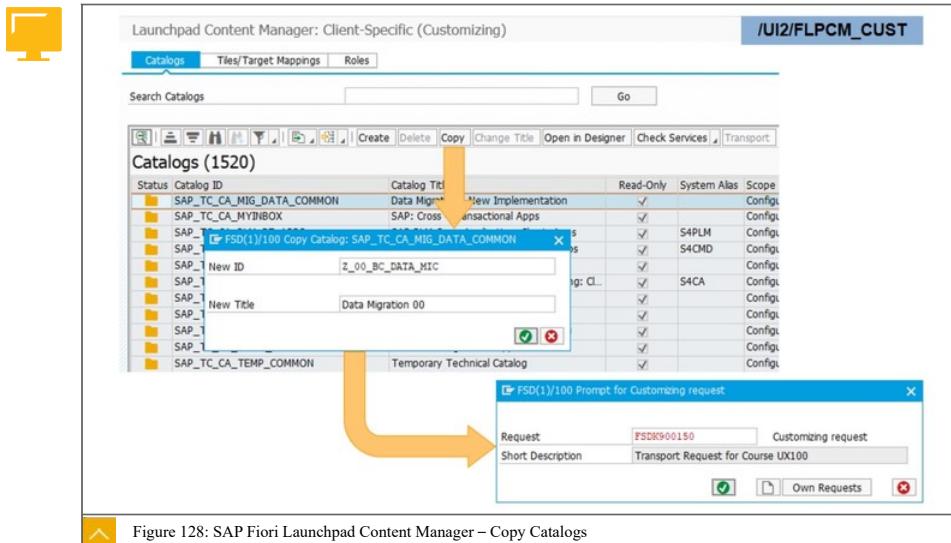


In SAP S/4HANA 1809 SPS03, the SAP Fiori launchpad content manager was introduced to ease mass-administration of catalogs. You can show the catalogs and their content, the tiles and target mappings for all catalogs, or roles and their catalogs. There are easy ways to copy complete catalogs or reference certain tiles and target mappings in other catalogs. Like the

SAP Fiori launchpad designer , it is possible to use the SAP Fiori launchpad content manager for configuration (transaction /UI2/FLPCM_CONF) or customizing (transaction /UI2/FLPCM_CUST).

Whereas the SAP Fiori launchpad designer focuses on single catalogs or groups, the SAP Fiori launchpad content manager has all catalogs in a system or client in its focus. The administrative work around catalogs, such as transport or role-assignment, is quite easy compared to the SAP Fiori launchpad designer . However, it is not possible to create groups, tiles, or target mappings. The target group for the SAP Fiori launchpad content manager is administrators, while the target group for SAP Fiori launchpad designer is developers.

Element	SAP Fiori Launchpad Content Manager	SAP Fiori Launchpad Designer
Catalog	Create/Copy/Check/Transport	Create/Copy
Group	-	Create
Tile	Reference	Create/Reference
Target Mapping	Reference	Create/Reference



Both tools can also be used in combination. For example, after copying a catalog in the SAP Fiori launchpad content manager , the new catalog can be opened directly in the SAP Fiori launchpad designer in order to adapt the tiles and target mappings. Combined with the possibility to also jump from the SAP Fiori launchpad content manager to the Role Maintenance (PFCG), the SAP Fiori launchpad content manager acts as the center for configuring SAP Fiori catalogs.



LESSON SUMMARY

You should now be able to:

- Manage SAP Fiori catalogs

Unit 4

Lesson 3

Creating SAP Fiori Catalogs

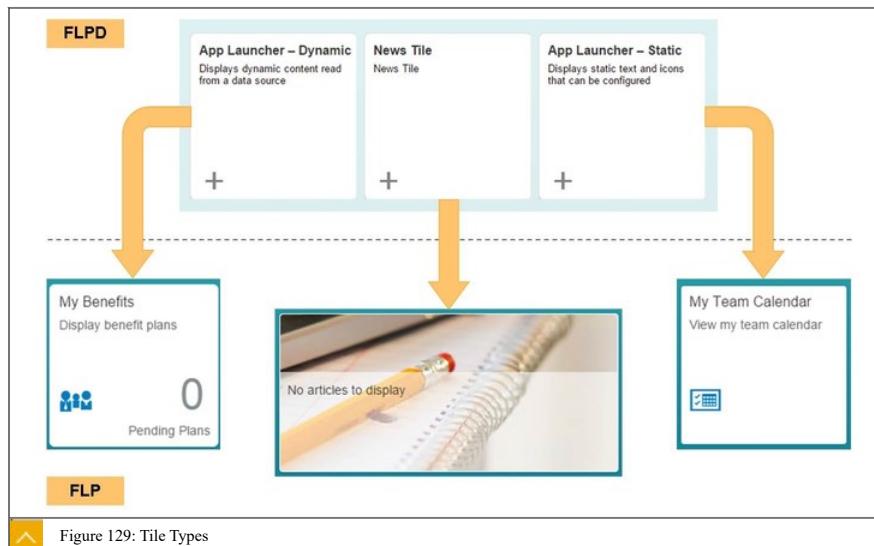


LESSON OBJECTIVES

After completing this lesson, you will be able to:

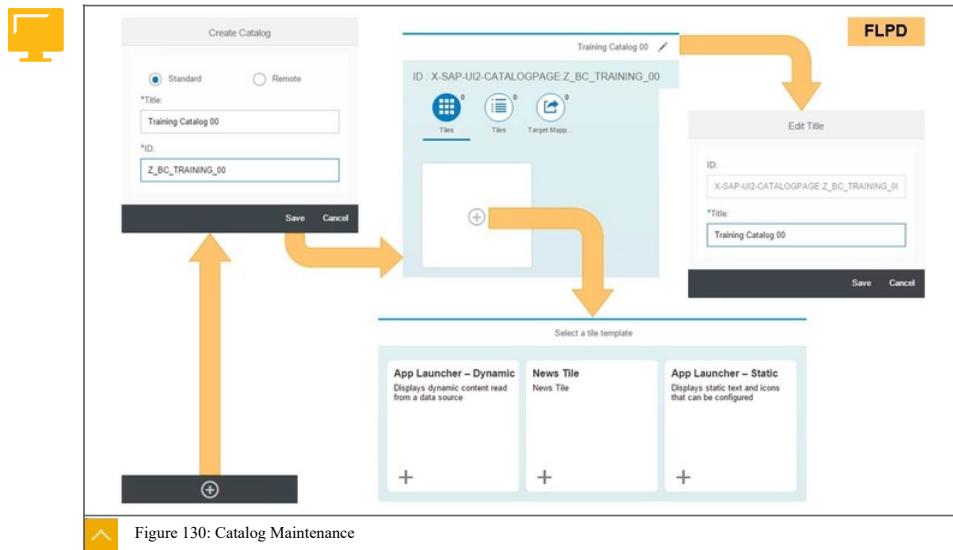
- Create SAP Fiori catalogs

Catalog Maintenance



There are three different types of tile in SAP Fiori :

- The App Launcher - Static or just static tile consists of a title and subtitle, an icon, keywords for search, and general information.
- The App Launcher - Dynamic or just dynamic tile has, in addition, the possibility to show a dynamic number. This number originates from an OData service request, providing a natural number like a number of data sets. The refresh interval of the number can also be set.
- The News tile has the special purpose of showing news from Really Simple Syndication (RSS) feeds. It is twice as large as the other two tiles and shows a preview of the feeds listed in its tile definition. When you choose the tile, the news app starts showing all elements of the underlying RSS feeds.

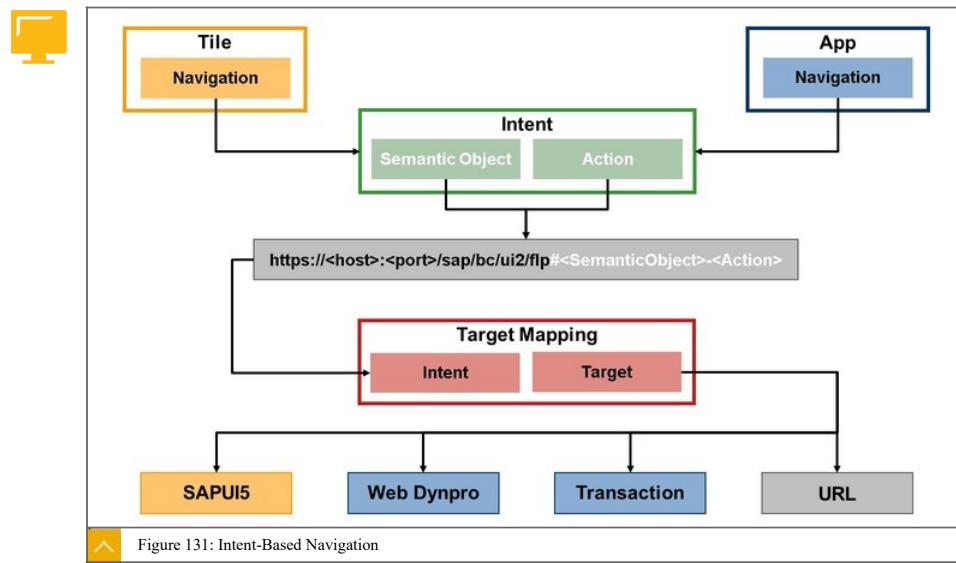


Catalogs can be created by choosing the plus symbol in the lower left corner of the catalog overview. This opens a dialog box for entering the catalog title and technical ID and specifying if the catalog should be pulled from another system as a remote catalog. The catalog title can be changed at a later point via the pencil next to the heading of the catalog, but not the catalog ID. Catalogs created by customers should start with **Z** and contain the shortage BC (Business Catalog) .



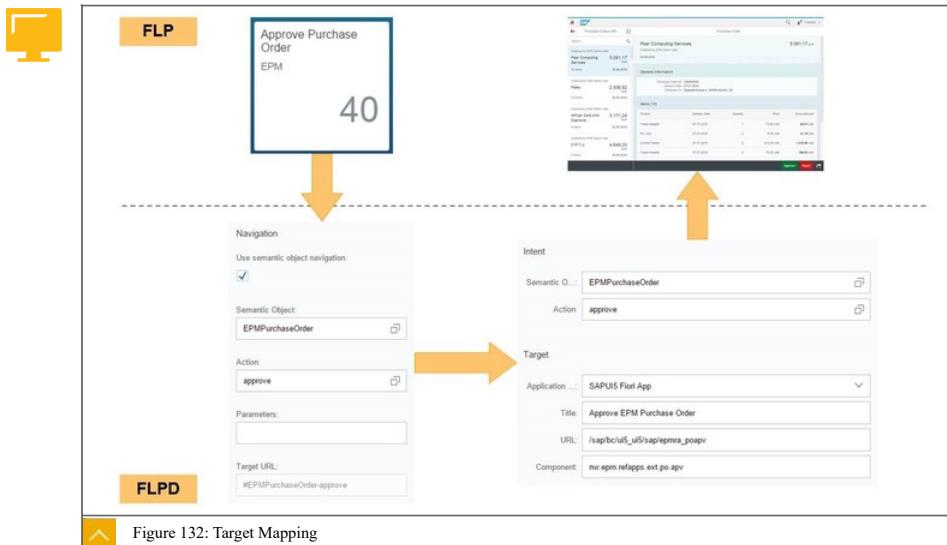
Caution:
The catalog ID has to be written in capital letters.

Intent-Based Navigation



The most important part of a tile is its navigation information, which contains an intent that connects to a target mapping. An intent is short for intention and consists of a semantic object like a sales order and an action like display. If the user wants to see the details of a sales order, their intention is to display a sales order. Fiori adds this intention behind the FLP URL separated by a # to call a target mapping having this intent defined. By that it is possible to call an application using HTTP without showing the real URL of the application, but, instead, mapping the request to an internally-defined target.

Intent-based navigation is highly flexible. It is not important how the intent is created. It can be a tile in the FLP or any other function in any other app the user is working with. Apps just need to start a navigation using an intent and the target mapping available for the current user is called. Which intent is called depends on the user role of the user starting the navigation. That means that the same intent can lead to different results for different users.



Target mappings contain the information about which app to start, with what parameters, and on what device types. They are identified by its intent, the combination of semantic object and action. If the user chooses a tile, the intent-based navigation defined in the tile is started. The suitable target mapping then calls the target app that is defined.

Figure 133 illustrates the list of semantic objects delivered by SAP. The table is titled '/UI2/SEMOBJ_SAP' and lists various semantic objects along with their semantic object name, application component, and description. For example, APARToleranceGroup is mapped to FIR and represents a tolerance group for customers and suppliers.

Semantic Object	Semantic Object Name	Applic. Component	Semantic Object Description
APARToleranceGroup	APAR Tolerance Group	FIR	Tolerance Group for Customers and Suppliers
ATPRequirement	ATP Requirement	SCM-APO-ATP	A requirement that is derived from a sales order, out
AcademicCourse	Academic Course	IS-PS-CA	A basic unit of academic content where students ca
AccessAlert	Access Alert	GRC-AC	Access control alert
AccessControlRole	Access Control Role	GRC-AC	Access Control Role
AccessRequest	Access Request	GRC-AC	Access Request
AccessRisk	Access Risk	GRC-AC	Access Risk
AccessUser	Access User	GRC-AC	Access User
Account	Account	CRM-HD-BP	Organizations or firms, but also individuals or groups
AccountingClerk	Accounting Clerk	FIR	Accounting Clerk
AccountingDocument	Accounting Document	FIR	A representation of changes to values of general led
AccountingDocumentItem	Accounting Document Item	FIR	A value-based change within an accounting docume
AccountingJournalList	Accounting Journal List	FI-F10-GL	A list of journal entries which represent changes in
AccrualCustomerSettlement	Accruals Customer Settlement	LO-AB	The act of reversing (in a customer contract) the ac
AccrualVendorSettlement	Accruals Vendor Settlement	LO-AB	In a vendor contract settlement, the actual recordin
Action	Action	GRC-AC	Actions include transactions or other activities monit
ActivityType	Activity Type	CRM-BTX-ACT	Activity type for confirmation
ActivityTypeGroup	Activity Type Group	FIR	Organizational unit used to combine a group of activ
Address	Address	SD	An address is a collection of information, presented i
AddressScreeningList	Address Screening List	GRC-FRA	A list of people or organizations to be kept under su
AdhocAnalysis	Adhoc Analysis	CA	Analyze and explore data freely in the HANA System
AdvancedStatusActivation	Advanced Status Activation	CEC-MKT-CEI	In this activity, you carry out the following steps: ACT

A semantic object represents a business entity such as a customer, a sales order, or a product. Using semantic objects bundles applications that reflect a specific scenario. They allow referring to business entities in a standardized way, abstracting from concrete implementations. A list of all semantic objects delivered by SAP can be accessed using the transaction `/UI2/SEMOBJ_SAP`. If customers want to define their own semantic objects, they can do so using the transaction `/UI2/SEMOBJ`. By defining an entry with the same key, customers can overwrite the attributes of a semantic object delivered by SAP.

References in Catalogs

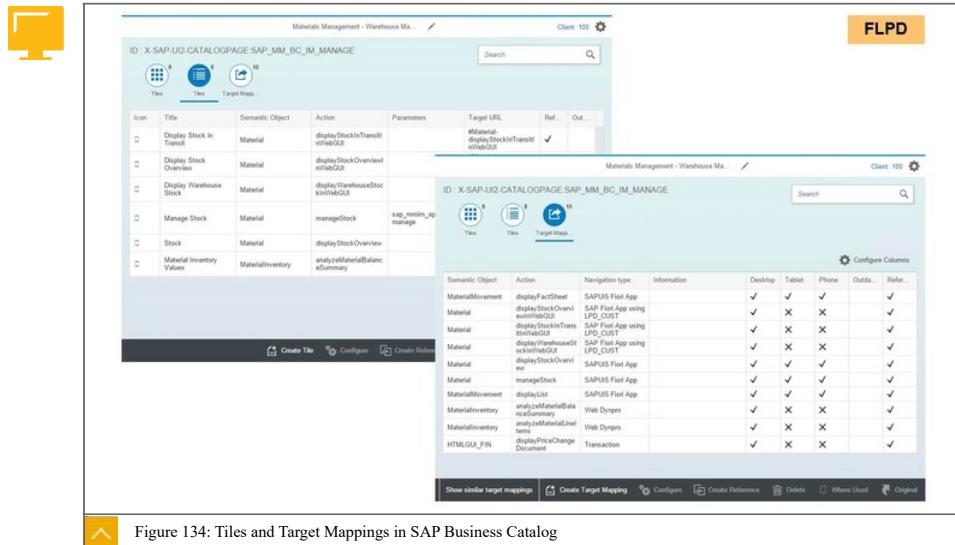


Figure 134: Tiles and Target Mappings in SAP Business Catalog

SAP delivers numerous catalogs containing many tiles and target mappings. All definitions are tested and distinguish, for example, the different device types the app is working in or include dynamic information in the tiles. All of these can also be referenced in other catalogs such as a new one created by a customer.

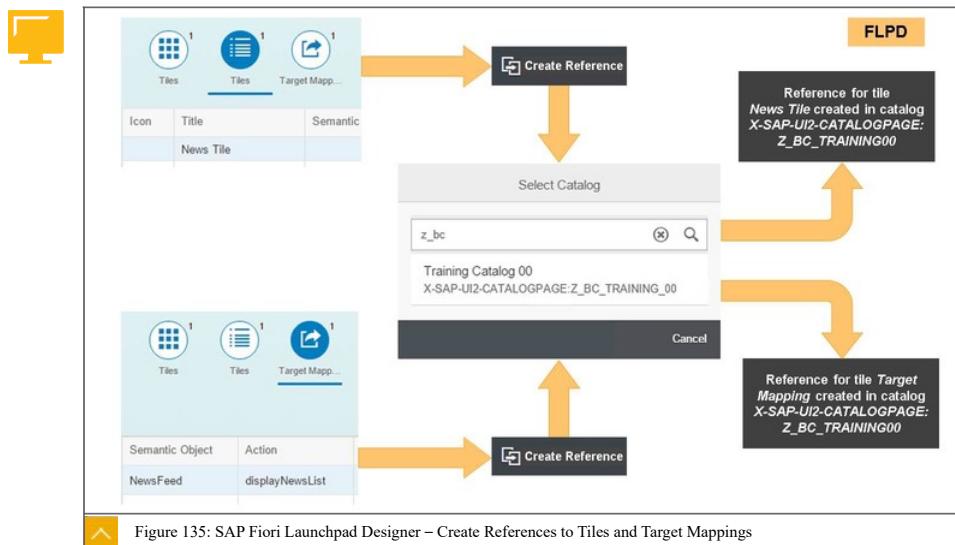
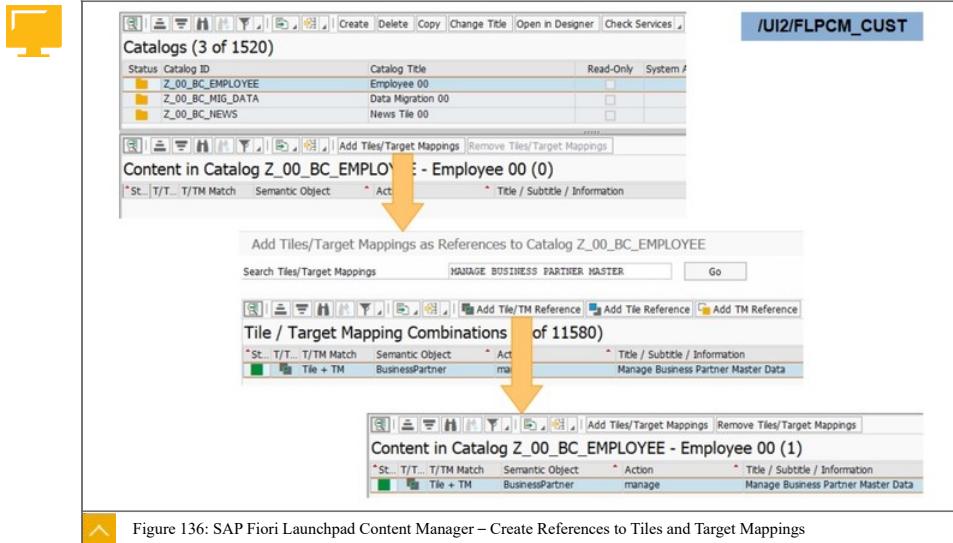


Figure 135: SAP Fiori Launchpad Designer – Create References to Tiles and Target Mappings

To create a reference of a tile or a target mapping in the SAP Fiori launchpad designer, select it in the source catalog and choose Create Reference in the lower part of the screen. A dialog

box then appears with a request for the target catalog. This is followed by a success message if all works correctly.



To create a reference of a tile and target mapping in the SAP Fiori launchpad content manager, choose **Add Tiles/Target Mappings** in the target catalog. In the next screen, search for the app you want to reference in all catalogs of the system. You can create a reference just for the tile, the target mapping, or both.



LESSON SUMMARY

You should now be able to:

- Create SAP Fiori catalogs

Unit 4

Lesson 4

Creating LPD_CUST Target Mappings



LESSON OBJECTIVES

After completing this lesson, you will be able to:

- Create LPD_CUST target mappings

LPD_CUST Mapping Elements

Component	Technical Name	Path to ICF Node	SAP UI5 Component
SAP UI5 Application	MDG_APPROVE_CR	/sap/bc/ui5_ui5/sap/mdg_appr ove_cr	fcg.mdg.approvecr

Technical Catalog	SAP_MDG_TC_T
Technical Catalog Description	Master Data Governance
PFCG Role for Technical Catalog	SAP_MDG_TCR_T
LPD_CUST Role	UIMDG001
LPD_CUST Instance	TRANSACTIONAL
SAPUI5 Application	MDG_APPROVE_CR

Figure 137: Technical Configuration Using LPD_CUST

In this example from the Fiori app reference library, you see the technical configuration using the transaction LPD_CUST. This transaction was the first way to map an intent-based navigation to the implementation of an app. You see the semantic object and action forming the intent, the role and instance of the mapping in LPD_CUST, and the name of the SAPUI5 application.

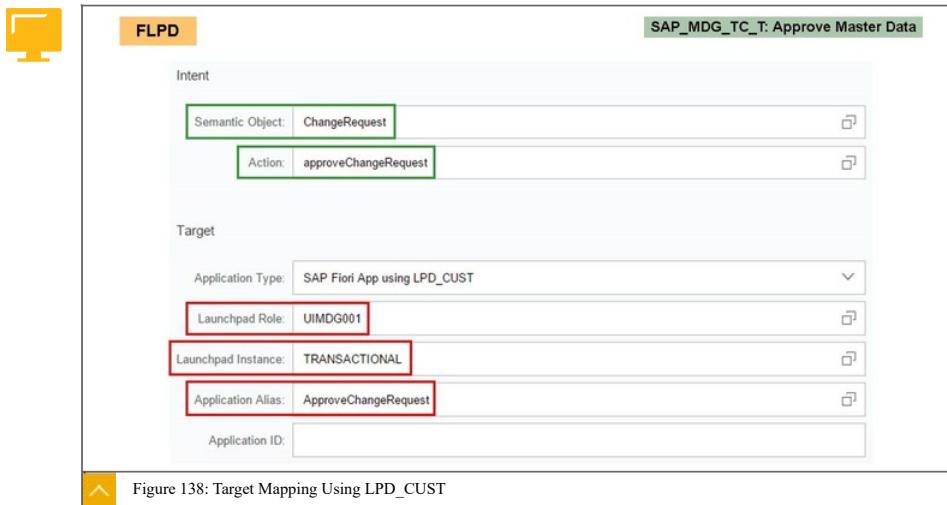


Figure 138: Target Mapping Using LPD_CUST

The intent can be found in the target mapping. As target, the application type SAP Fiori App using LPD_CUST is set. What happens when a navigation request occurs is not visible in the target mapping. That is defined in the LPD_CUST following the launchpad role, instance, and application alias.

Overview of Launchpads													
Code	Instance	Description	Re...	Em...	FP...	Change	Deploy	Delete	Copy	Transp.	Alias M...	Transf.	User Data
UIMP_FILE_R	UIMP_FILE_R	Local Reporting Renovation	/	/	/	/	/	/	/	/	/	/	SAP 07.11.2012
UIMP_LR_DP	UIMP_LR_DP	Local Reporting Renovation	/	/	/	/	/	/	/	/	/	/	SAP 07.11.2012
UIMP_REN_LR	UIMP_REN_LR	Local Reporting Renovation	/	/	/	/	/	/	/	/	/	/	SAP 07.11.2012
LUMIRA	ANALYTICS	Lumira Analytics	/	/	/	/	/	/	/	/	/	/	SAP 13.05.2014
MISMATCH_	MISMATCH_INSTANCE	Customization for the error application	/	/	/	/	/	/	/	/	/	/	SAP 16.09.2010
SPRUTBL	NEWSITEM	News	/	/	/	/	/	/	/	/	/	/	SAP 11.03.2014
UIMAPF170	FACTSHEETS	SAP Fiori Launchpad for Fact Sheets	/	/	/	/	/	/	/	/	/	/	SAP 01.04.2015
UIMAPF170	TRANSACTIONAL_CM	SAP Fiori Launchpad for CM-Transactions	/	/	/	/	/	/	/	/	/	/	SAP 19.11.2015
UIMAPF170	TRANSACTIONAL_CO	SAP Fiori Launchpad for CO-Transactions	/	/	/	/	/	/	/	/	/	/	SAP 19.11.2015
UIMAPF170	TRANSACTIONAL_FI	SAP Fiori Launchpad for FI-Transactions	/	/	/	/	/	/	/	/	/	/	SAP 19.11.2015
UIMAPF170	TRANSACTIONAL_FO	SAP Fiori Launchpad for FO-Transactions	/	/	/	/	/	/	/	/	/	/	SAP 19.11.2015
UIMAPF170	TRANSACTIONAL_GLO	SAP Fiori Launchpad for Globalization	/	/	/	/	/	/	/	/	/	/	SAP 19.11.2015
UICORE	CA	Applications - CA	/	/	/	/	/	/	/	/	/	/	SAP 20.11.2015
UICORE	CMD	Applications - Central Master Data	/	/	/	/	/	/	/	/	/	/	SAP 20.11.2015
UICORE	EAM	Applications - EAM	/	/	/	/	/	/	/	/	/	/	SAP 20.11.2015
UICORE	GTS	Applications - Global Trade Services	/	/	/	/	/	/	/	/	/	/	SAP 19.11.2015
UICORE	LO	Applications - Settlement Management	/	/	/	/	/	/	/	/	/	/	SAP 19.11.2015
UICORE	PLM	Applications - PLM	/	/	/	/	/	/	/	/	/	/	SAP 19.11.2015
UICORE	PRC	Applications - Procurement	/	/	/	/	/	/	/	/	/	/	SAP 19.11.2015
UICORE	PS	Applications - PS	/	/	/	/	/	/	/	/	/	/	SAP 19.11.2015
UICORE	QM	Applications - Quality Management	/	/	/	/	/	/	/	/	/	/	SAP 19.11.2015
UICORE	SCM	Applications - SCM	/	/	/	/	/	/	/	/	/	/	SAP 19.11.2015
UICORE	SD	Applications - SD	/	/	/	/	/	/	/	/	/	/	SAP 19.11.2015
UIMR001	TRANSACTIONAL	Launchpad for HCM Fiori Transactional ap	/	/	/	/	/	/	/	/	/	/	SAP 27.05.2015
UIML001	DAAG_APPS	Data Aging Application	/	/	/	/	/	/	/	/	/	/	SAP 23.05.2014
UIML001	TRANSACTIONAL	ILM Data Archiving Job Mon Ipd	/	/	/	/	/	/	/	/	/	/	SAP 01.04.2014
UIMDC001	TRANSACTIONAL	Master Data Consolidation Launchpad	/	/	/	/	/	/	/	/	/	/	SAP 02.09.2014
UIMDG001	TRANSACTIONAL	HDI Fiori Launchpad	/	/	/	/	/	/	/	/	/	/	SAP 21.10.2015

Figure 139: Launchpads in LPD_CUST

Starting LPD_CUST displays a list of launchpad roles and instances. These are containers of links to applications of a certain solution area.



Caution:

The LPD_CUST is not only used for SAP Fiori but also for other technologies and solution areas.

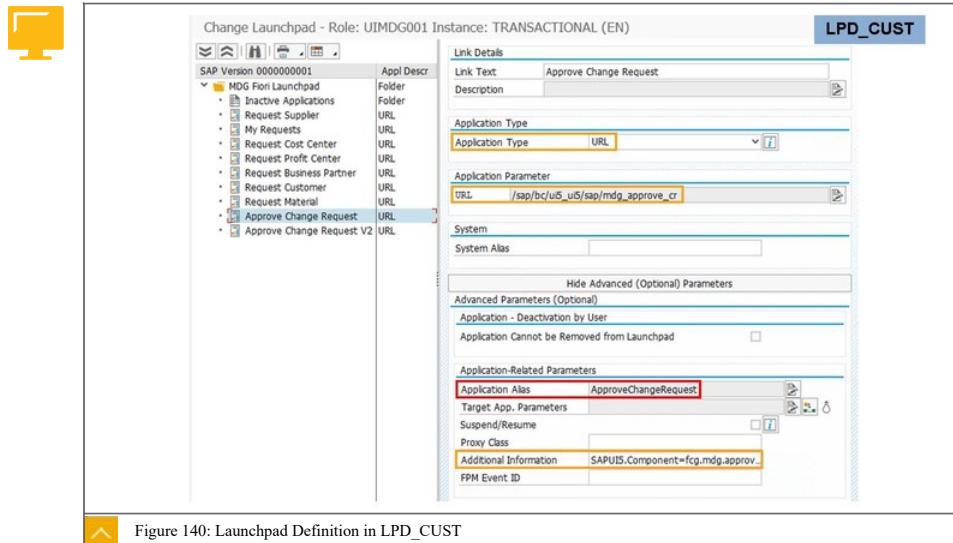


Figure 140: Launchpad Definition in LPD_CUST

Each link has an application type and, depending on the type, several additional parameters. The application type of SAPUI5 is URL. The additional parameters are the URL of the ICF node and the path to the component for the SAPUI5 app, as well as the application alias connecting to the target mapping in the FLPD.



Note:

There are also many other possible application types in LPD_CUST but only URL for SAPUI5, Web Dynpro, and transactions are valid for SAP Fiori.

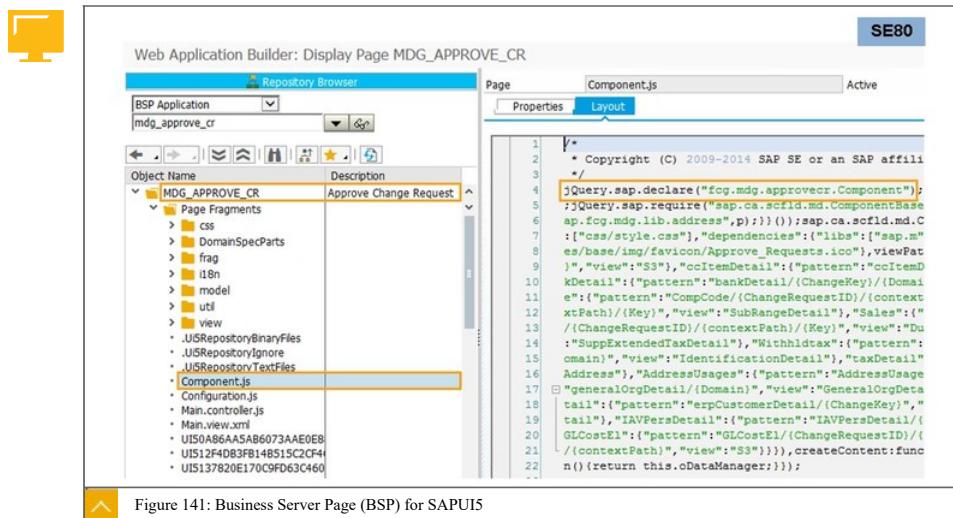


Figure 141: Business Server Page (BSP) for SAPUI5

An SAPUI5 app is saved as a Business Server Page (BSP) in the FES. The central file of an SAP Fiori app based on SAPUI5 is `Component.js`. To start the app, the path to this file must be visible in the statement `jQuery.sap.declare`.



LESSON SUMMARY

You should now be able to:

- Create LPD_CUST target mappings

Unit 4

Lesson 5

Creating SAPUI5 Target Mappings



LESSON OBJECTIVES

After completing this lesson, you will be able to:

- Create SAPUI5 target mappings

SAPUI5 Mapping Elements



The screenshot shows the SAP Fiori app configuration interface for the 'Post General Journal Entries [F0718]' application. The 'Configuration' tab is selected. The 'SAPUI5 Application' section lists the ICF nodes required for the application. The table shows the following data:

Component	Technical Name	Path to ICF Node	SAP UI5 Component
SAP UI5 Application	FIN_GLDOCPOST	/sap/bc/ui5_ui5/sap/fin_gldocpost	fin.gl.documentpost
	FIN_GLDOCDISP *	/sap/bc/ui5_ui5/sap/fin_gldocdisp	fin.gl.documentdisplay
	FIN_GL_POST_LIB *	/sap/bc/ui5_ui5/sap/fin_gl_postlib	sap.fin.gl.lib.posting
	FIN_LIB *	/sap/bc/ui5_ui5/sap/fin_lib	

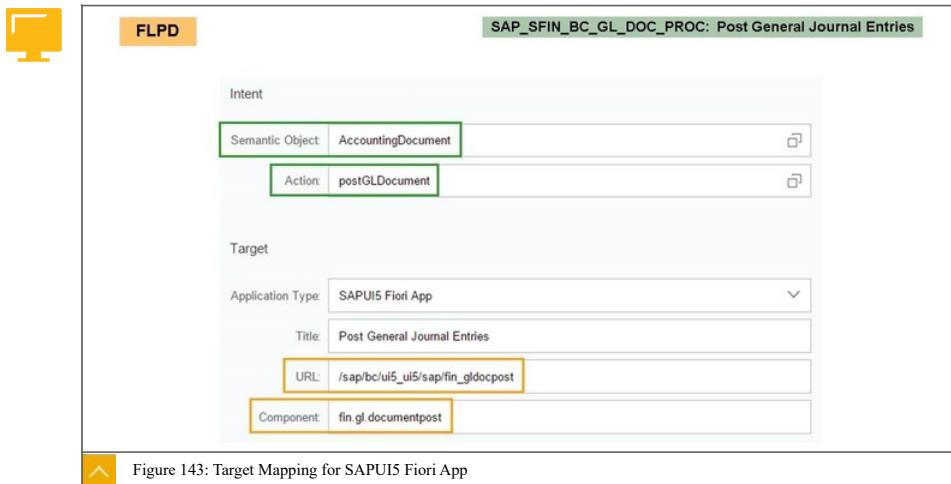
* Added automatically due to dependencies

The 'Technical Configuration' section shows the semantic objects and actions for the application:

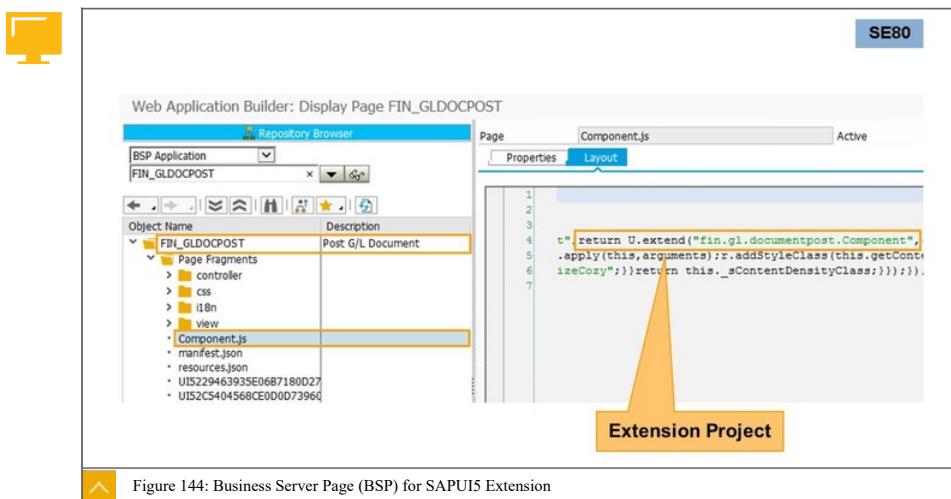
- Technical Catalog: SAP_TC_FIN_ACC_COMMON
- Technical Catalog Description: SAP
- SAPUI5 Application: FIN_GLDOCPOST

Figure 142: Technical Configuration for SAPUI5 App

In this example from the Fiori app reference library, you see the semantic object and action forming the intent and the name of the SAPUI5 application. The transaction LPD_CUST is not mentioned.



Again, the intent can be found in the target mapping. As target the application type SAP Fiori App is set. So, instead of using LPD_CUST, the URL and path to the component of the SAPUI5 application is part of the target mapping in the FLPD.



In this example, the BSP holding the SAPUI5 app has less files than usual. This is a sign for an SAPUI5 extension, which only consists of the changes to the original app. Checking the content of the Component.js, there is no statement `jQuery.sap.declare`, but, instead, `return U.extend` holding the path to the Component.js.



LESSON SUMMARY

You should now be able to:

- Create SAPUI5 target mappings

Unit 4

Lesson 6

Creating Web Dynpro Target Mappings



LESSON OBJECTIVES

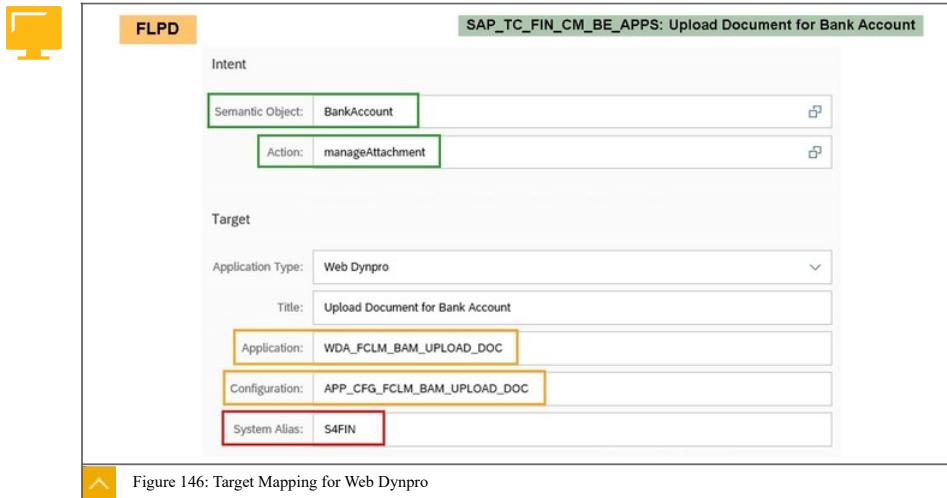
After completing this lesson, you will be able to:

- Create Web Dynpro target mappings

Web Dynpro Mapping Elements

Figure 145: Technical Configuration for Web Dynpro

In this example from the Fiori app reference library, you see the system alias for the Web Dynpro application and configuration as well as the semantic object and action of the target mapping forming the intent.

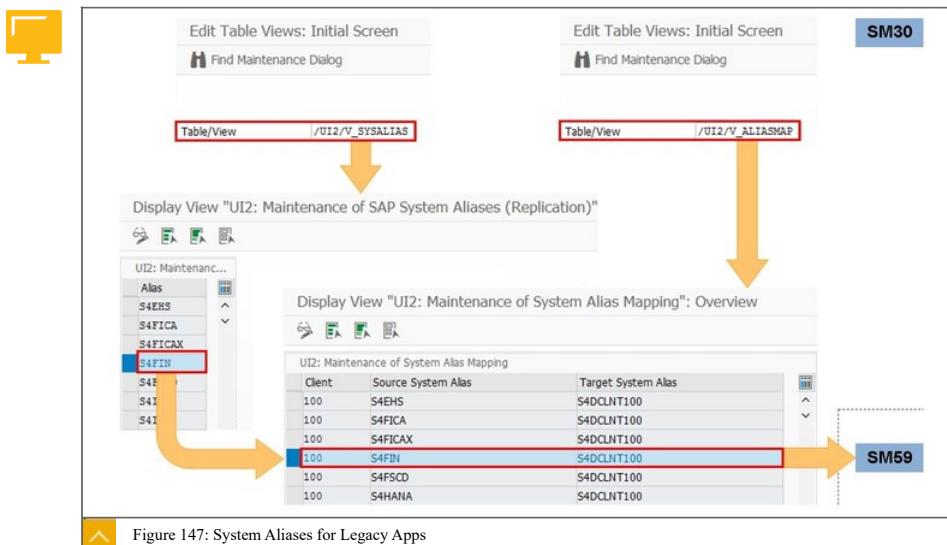


Again, the intent can be found in the target mapping. As target the application type Web Dynpro is set. The Web Dynpro application and configuration is visible in the target mapping as well as the system alias pointing to the system holding the application.

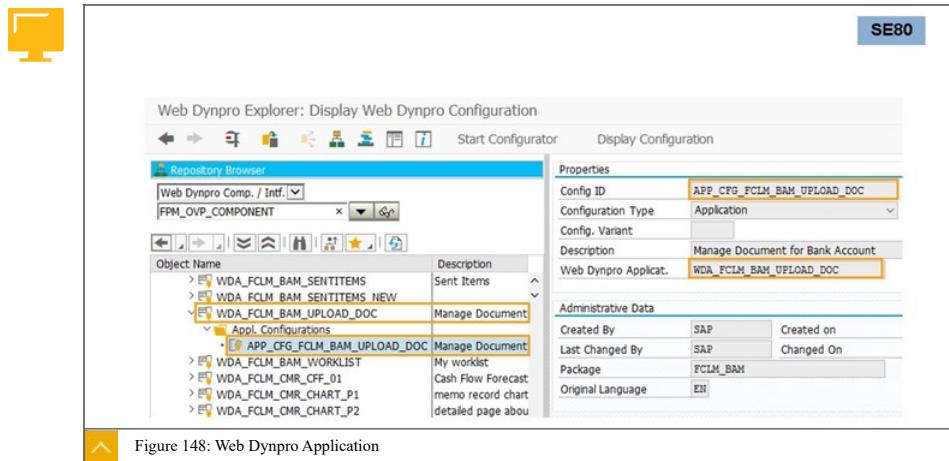


Caution:

This system alias has nothing to do with the system alias used with SAP Gateway. The one referred to here is only used for legacy apps to identify the BES.



System aliases for legacy apps are delivered by SAP and distinguished by solution area. They are viewable in the maintenance view `/UI2/V_SYSALIAS`. For each one an RFC (`<alias>_RFC`) and HTTPS (`<alias>_HTTPS`) destination should be created in the `SM59`. If several aliases point to the same system like an `SAP S/4HANA`, the maintenance view `/UI2/V_ALIASMAP` can be used to map these aliases to a target system alias, which can be defined freely.



Following the system alias leads to the system holding the application. Many Web Dynpro applications are Floorplan Manager (FPM) applications that need an application configuration as well to be able to start the app. The configuration is also visible in the target mapping.



LESSON SUMMARY

You should now be able to:

- Create Web Dynpro target mappings

Unit 4

Lesson 7

Creating Transaction Target Mappings



LESSON OBJECTIVES

After completing this lesson, you will be able to:

- Create transaction target mappings

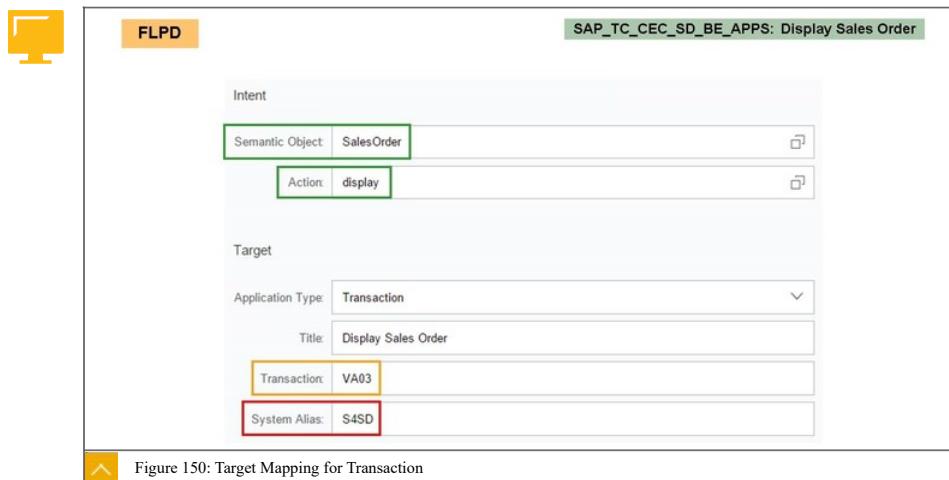
Transaction Mapping Elements



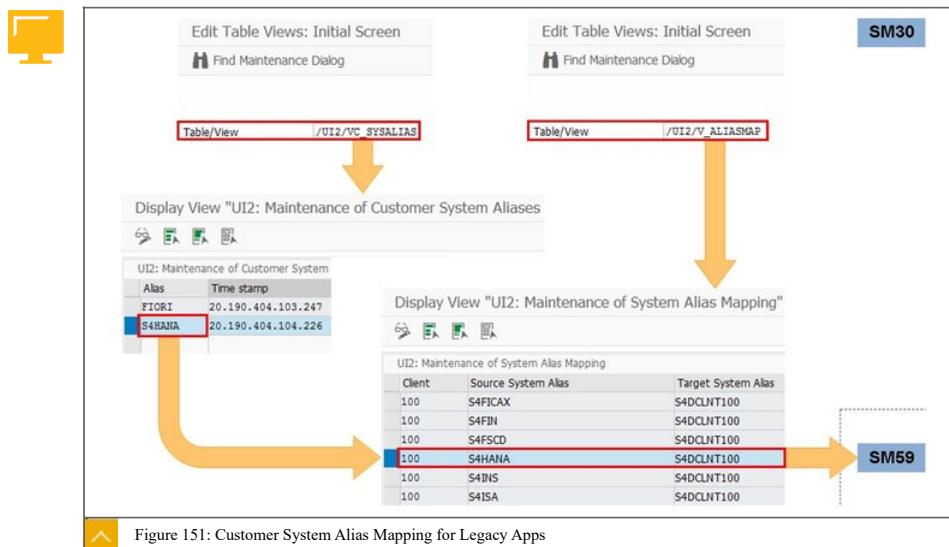
Semantic Object	Semantic Action	Parameter Key	Parameter Value
SalesOrder	display	sap-system	S4OIL
SalesOrder	display	sap-system	S4SD
SalesOrderWithoutCharge	display	DynproOkCode	ENT2

Figure 149: Technical Configuration for Transaction

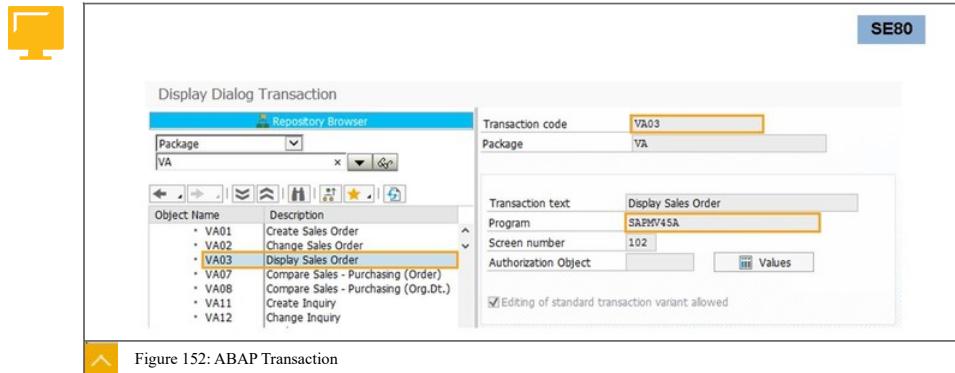
In this example from the Fiori app reference library, you see the system aliases for a transaction code as well as the semantic objects and actions of the target mappings forming the intents. Because of the function-oriented style of transactions, they are often used in many areas and therefore have many target mappings and even system aliases.



Again, the intent can be found in the target mapping. As target, the application type Transaction is set. The transaction as well as the system alias are visible in the target mapping.



Transactions are — like Web Dynpro applications — legacy apps and, therefore, use the same system aliases viewable in the maintenance view /UI2/V_SYSALIAS. If customers want to define their own aliases, they can define them in the maintenance view /UI2/VC_SYSALIAS. These can also be mapped to a target system alias using the maintenance view /UI2/V_ALIASMAP. For each target system alias, an RFC (<alias>_RFC) and HTTPS (<alias>_HTTPS) destination should be created in the SM59.



Following the system alias leads to the system holding the application. For transactions, this is the transaction code and the program to which it is pointing.



LESSON SUMMARY

You should now be able to:

- Create transaction target mappings

Unit 4

Lesson 8

Creating Back-End Catalogs



LESSON OBJECTIVES

After completing this lesson, you will be able to:

- Create back-end catalogs

Back-End Catalogs



Figure 153: Mass Maintenance for App Descriptors

For SAP Business Suite , it is only possible to map an ABAP transaction to an intent using `LPD_CUST`. In SAP S/4HANA , back-end catalogs are available holding app descriptors pointing to legacy apps of the type Web Dynpro ABAP and ABAP transaction. Back-end catalogs are shipped by SAP as part of updates of back-end systems.

Back-end catalogs can be examined in the BES using the Web Dynpro application `SUI_TM_MM_APP` or transaction `MM_APP`. Each catalog provides a list of semantic objects and actions with the corresponding legacy apps. The transaction `/UI2/APPDESC_GET` in the FES is used to extract back-end catalogs from the back-end servers as remote technical catalogs to the front-end server. They can be used there like any other catalog in the SAP Fiori Launchpad Designer .

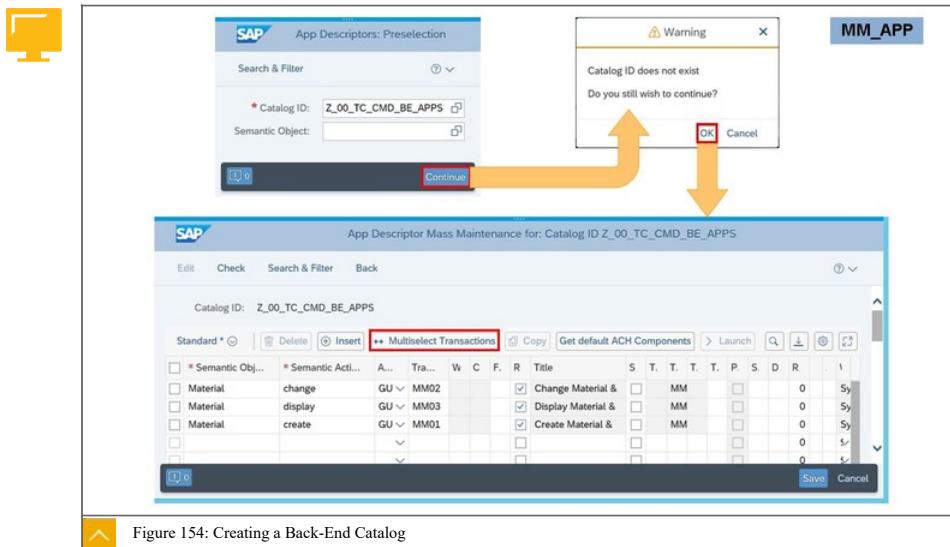


Figure 154: Creating a Back-End Catalog

Customers can create own back-end catalogs using transaction MM_APP, too. When entering a catalog in the Catalog ID field not available in the system, a pop-up appears asking if this back-end catalog should be created. The catalog can then be filled with Web Dynpro applications and transactions assigning semantic objects and actions.

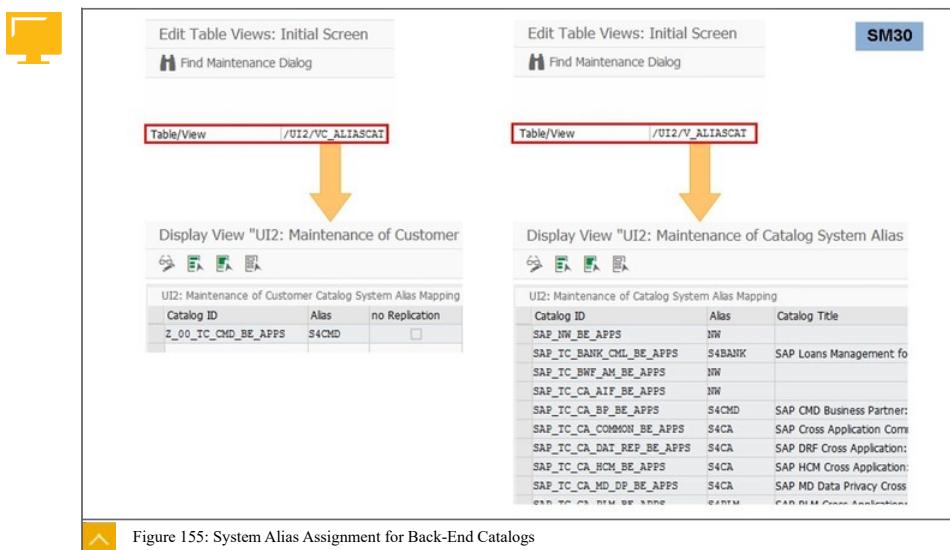


Figure 155: System Alias Assignment for Back-End Catalogs

Each back-end catalog needs a system alias for legacy apps assigned. It is needed for the replication to the front-end server. System alias assignments for back-end catalogs delivered by SAP can be found in the maintenance view /UI2/V_ALIASCAT. Customers can assign system aliases to their back-end catalogs in the maintenance view /UI2/VC_ALIASCAT.

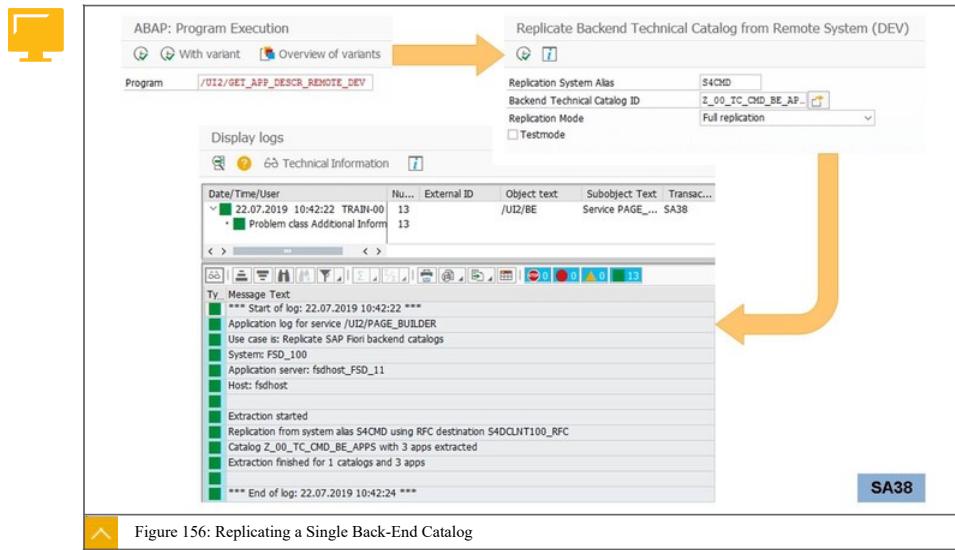


Figure 156: Replicating a Single Back-End Catalog

To replicate a single back-end catalog to the front-end server, SAP provides the /UI2/GET_APP_DESCR_REMOTE_DEV report. After entering a system alias and the name of at least one catalog, you can choose between a full or delta replication. A Testmode allows a test run before actually replicating anything.



LESSON SUMMARY

You should now be able to:

- Create back-end catalogs

Unit 4

Lesson 9

Configuring SAP Fiori Launchpad



LESSON OBJECTIVES

After completing this lesson, you will be able to:

- Configure SAP Fiori launchpad

Maintenance

The screenshot shows the SAP Fiori Implementation Guide (IMG) structure. The main menu includes 'Existing BC Sets', 'BC Sets for Activity', and 'Activated E'. The 'Structure' section is expanded, showing 'UI Technologies' and 'SAP Fiori'. Under 'SAP Fiori', 'Initial Setup' is highlighted with a red box. 'Initial Setup' contains 'Prerequisites' (with 'Schedule SAPUI5 Application Index Calculation' highlighted with a red box), 'Connection Settings (Front-End Server to ABAP Back-End Server)' (with 'Define Trust for SAP Business Systems' highlighted with a red box), 'Initial Launchpad Configuration' (with 'Service Maintenance of SAP Gateway' highlighted with a red box), 'SAP Web Dispatcher' (with 'Configure SAP Web Dispatcher' highlighted with a red box), and 'Configuring SAP Fiori Search' (with 'Setup Feedback Service' highlighted with a red box). To the right, 'Configuring Launchpad Content' is highlighted with a red box. It contains 'Adding Apps to SAP Fiori Launchpad' (with 'Navigation Concept' highlighted with a red box), 'Configure Target Mappings and Tiles' (with 'SAP Fiori Launchpad Designer (Cross-Client)' and 'SAP Fiori Launchpad Designer (Current Client)' highlighted with red boxes), 'Edit Legacy Navigation Target Configurations in LPD_CUST', 'Prepare Backend Catalogs for Reuse' (with 'Establish RFC and HTTP Connection' highlighted with a red box), 'Mass Maintenance Tool for App Descriptors' (with 'Extract Transactions and Web Dynpro Applications' highlighted with a red box), 'Define Semantic Objects for Navigation' (with 'Configure Authorization Roles' highlighted with a red box), 'Data Administration' (with 'Invalidate Caches' highlighted with a red box), 'Create Cleanup Jobs' (with 'Support Tools' highlighted with a red box), and 'Support Tools' (with 'SAP Fiori Launchpad Content Check Report', 'SAP Fiori Launchpad Intent Analysis', 'SAP UI2 Application Log', 'SAP Gateway Application Log', and 'SAP Gateway Error Log' highlighted with red boxes). The top right corner of the screenshot shows the transaction code '/UI2/CUST'.

Figure 157: SAP Fiori Implementation Guide

SAP Fiori is part of the SAP Reference Implementation Guide (IMG). Transaction /UI2/CUST can be used to access only the UI-relevant parts of the IMG. The main parts for SAP Fiori are as follows:

- Initial Setup for administrators
- Configuring Launchpad Content for administrators and consultants
- Data Administration for administrators and developers
- Support Tools for administrators, consultants, and developers

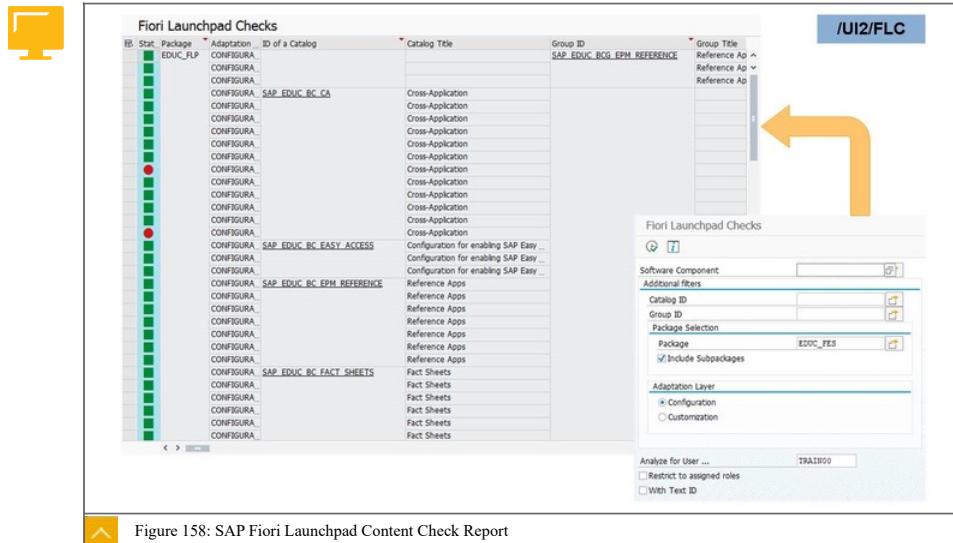


Figure 158: SAP Fiori Launchpad Content Check Report

One important support tool for the SAP Fiori launchpad configuration is transaction **/UI2/FLC**. It checks the consistency of delivered and customized catalogs and groups for configuration and customizing. It quickly identifies problems in target mappings and tiles concerning elements of the intent-based navigation and parameters.

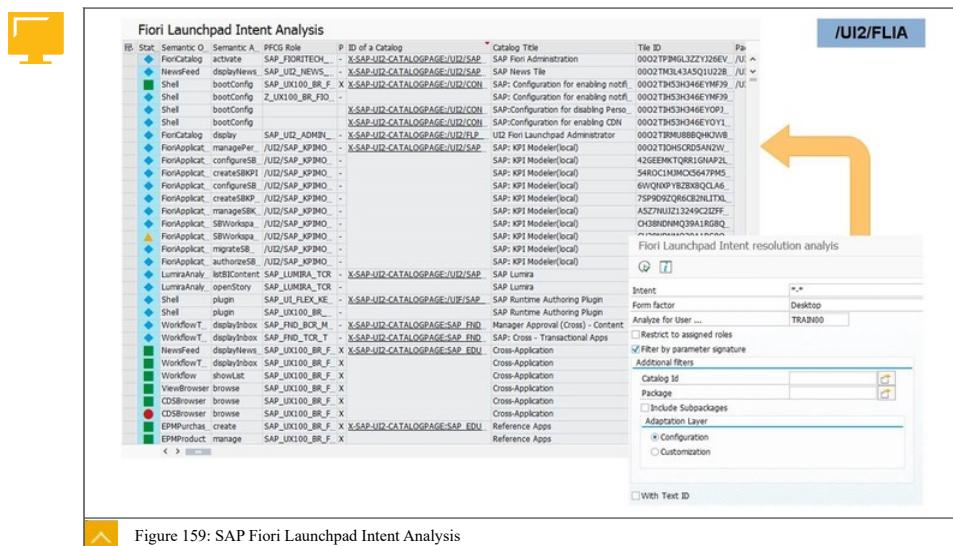
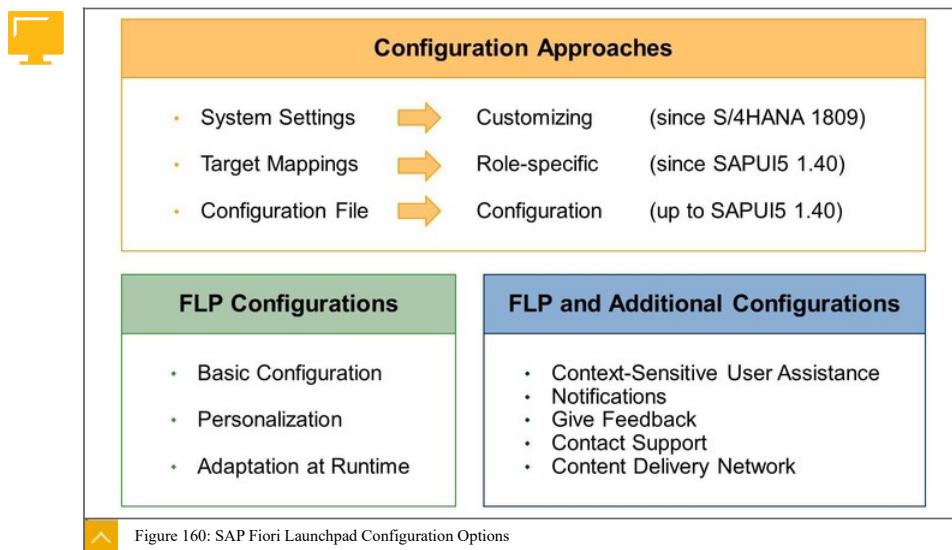


Figure 159: SAP Fiori Launchpad Intent Analysis

Transaction **/UI2/FLIA** also shows errors and problems in target mappings, but it goes deeper. It offers a full intent analysis for semantic objects and actions per role and user. Through this duplicated intents pointing to different targets can also be found. A full intent resolution analysis takes some time, depending on the number of semantic objects and

actions in the system. Therefore, it is recommended to restrict the analysis to the assigned roles of a user.

Configuration Options



There are plenty of configuration options for the FLP. Depending on the release of the software component SAP_UI on the FES and/or the release of the BES, several basic configurations just for the FLP and advanced configurations needing additional technical settings elsewhere are possible. There are three possible approaches to configure the FLP:

System settings

Define system settings using the transaction `/UI2/FLP_CUS_CONF` is recommended since SAP S/4HANA 1809. These settings are valid for all users in one system client.

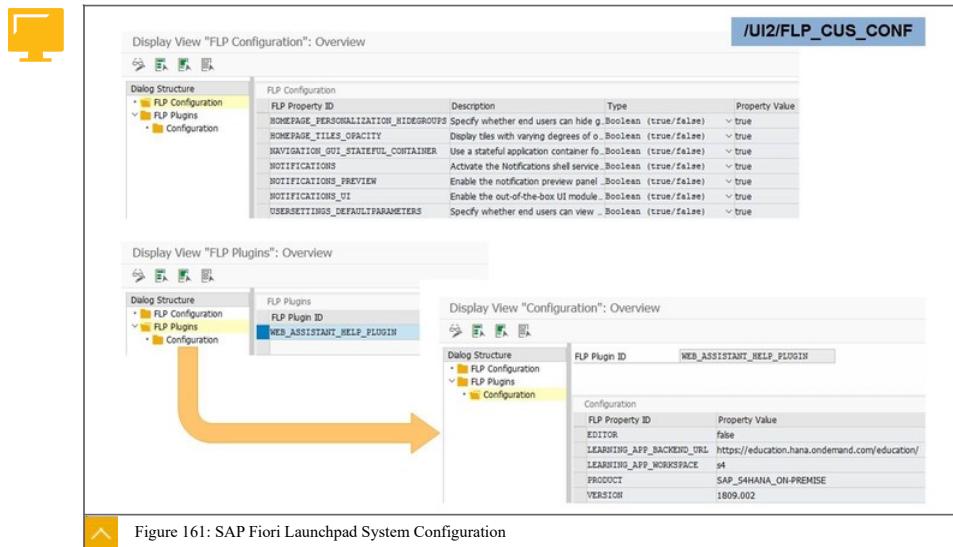
Target Mapping

Defining target mappings in catalogs is the recommended approach since SAPUI5 1.40. These catalogs can then be assigned to users via roles as usual.

Configuration File

Creating a configuration file is recommended up to SAPUI5 1.40. In this file, URL parameters are defined to enable or disable features when starting the FLP.

To configure the additional settings needed for context-sensitive user assistance, notifications, give feedback, contact support, and Content Delivery Network (CDN), consult the SAP online documentation for your system release or ask your administrator.



The transaction `/UI2/FLP_CUS_CONF` is available since SAP S/4HANA 1809. It allows central customizing of the SAP Fiori launchpad (FLP) for all users in one client. The value help for properties makes it easy to get an overview over possible settings. A full release-dependent documentation of all settings can be found in the SAP Fiori launchpad documentation on <https://help.sap.com/>.

Beside general settings, it is also possible to add plugins to the FLP and configure them. One example is the Web assistant plugin providing the context-sensitive user assistance in Fiori apps.

Note:

The transaction `/UI2/FLP_SYS_CONF` allows the configuration of the FLP for all clients. But these settings are saved in the SAP namespace and therefore count as modification including all drawbacks.



FLPD

Intent	Target
Semantic Object: Shell	Application Type: SAPUI5 Fiori App
Action: bootConfig	Title: Configuration for enabling FLP features
	URL:
	ID: sap ushell renderers fiori2

General

Information:

Device Types: Desktop Tablet Phone

Parameters:

Name	Mandatory	Value	Is Regular Expression	Default Value	Target Name
renders/fiori2/componentData/config/applications/Shell-home/enableHideGroups	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	true	
renders/fiori2/componentData/config/applications/Shell-home/enableTilesOpacity	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	true	
renders/fiori2/componentData/configenableUserDefaultParameters	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	true	
renders/fiori2/componentData/configanimationMode	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	full	
renders/fiori2/componentData/config/title	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	UX100	

Allow additional parameters

Add **Delete**

 Figure 162: Custom Catalog for Basic FLP Configuration

Target mappings for configuring the FLP can be created in any catalog, although it is recommended to use a separate one without any tiles or target mappings for apps. The semantic object for the FLP is **Shell**, followed by an action depending on the FLP part. The application type is **SAPUI5 Fiori App**, followed by a component ID for the technical component. The settings themselves are created using parameters of the target mapping and depend on the system release.



FLPD

Catalog: /UI2/CONFIG_PERS_OFF

Intent	Target
Semantic Object: Shell	Application Type: SAPUI5 Fiori App
Action: bootConfig	Title: Configuration for disabling Personalization
	URL:
	ID: sap ushell config personalization

General

Information:

Device Types: Desktop Tablet Phone

Parameters:

Name	Mandatory	Value	Is Regular Expression	Default Value	Target Name
renders/fiori2/componentData/configenablePersonalization	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	false	
renders/fiori2/componentData/configenableTagFiltering	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	false	
renders/fiori2/componentData/configenableSetTheme	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	false	
renders/fiori2/componentData/configenableAccessibility	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	true	
renders/fiori2/componentData/configenableSendFeedback	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	false	

Allow additional parameters

Add **Delete**

 Figure 163: Configuration for Disabling Personalization

For some configuration areas, SAP delivers predefined catalogs that already consist of target mappings with parameters. These catalogs can be assigned directly to roles to enable a certain feature set, or they can be copied in the customer namespace to change the default values to meet someone's needs.



LESSON SUMMARY

You should now be able to:

- Configure SAP Fiori launchpad

Unit 4

Lesson 10

Configuring Analytical Applications



LESSON OBJECTIVES

After completing this lesson, you will be able to:

- Enable SAP Fiori Smart Business
- Model SAP Fiori KPI Apps

SAP Fiori Smart Business

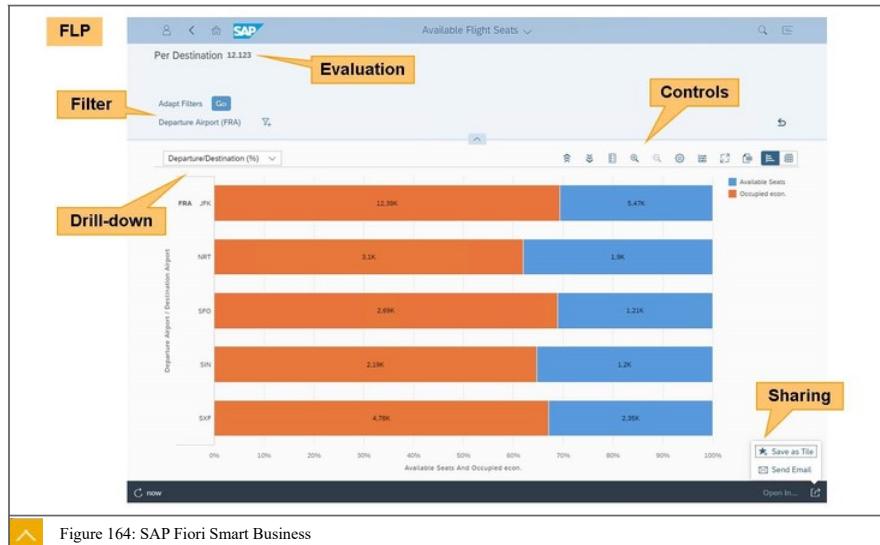
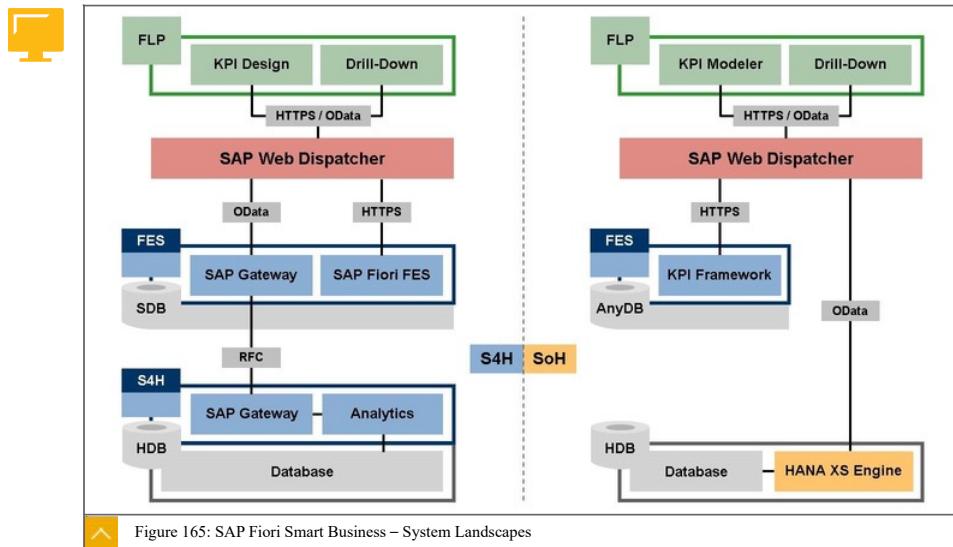


Figure 164: SAP Fiori Smart Business

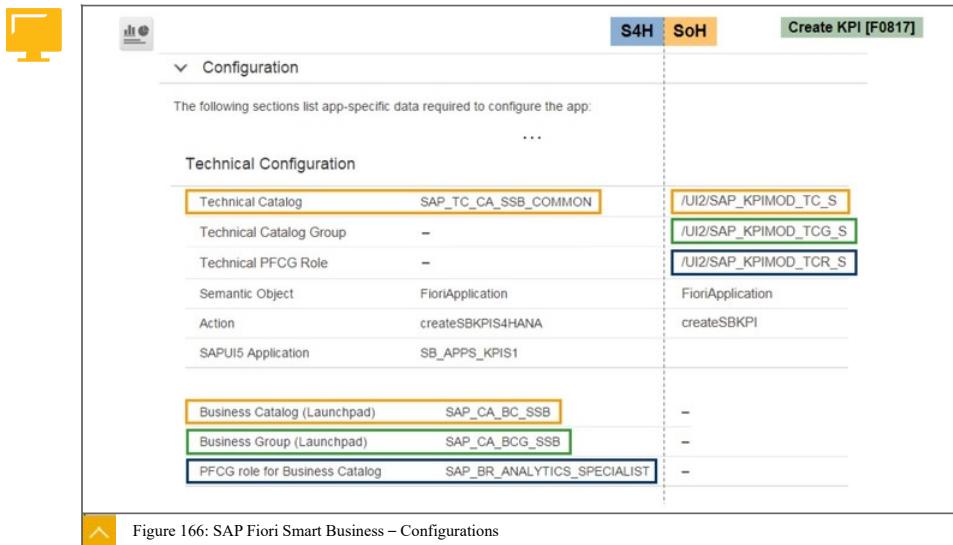
SAP Fiori Smart Business enables customers to create their own analytical apps based on Key Performance Indicators (KPIs). Using these selection criteria for business data, users can visualize an evaluation using a drill-down, control the visualization of the data, apply filters, export the data to Microsoft Excel, and share everything as a tile or email.



SAP Smart Business is available for SAP S/4HANA (S4H) and SAP Business Suite on HANA (SoH). For both solutions, the tool for creating KPIs is part of the Front-End Server (FES) and available as apps in a catalog for the SAP Fiori launchpad. The drill-down UI is also part of the FES for S4H and SoH.

In S4H, the tool to create KPIs is called KPI Design. It is part of the software component SAP Fiori Front-End Server and uses SAP Gateway services to communicate with the SAP S/4HANA Embedded Analytics. Beside the initial configuration of the Embedded Analytics, the handling of the KPI Design apps is comparable with every other Fiori app.

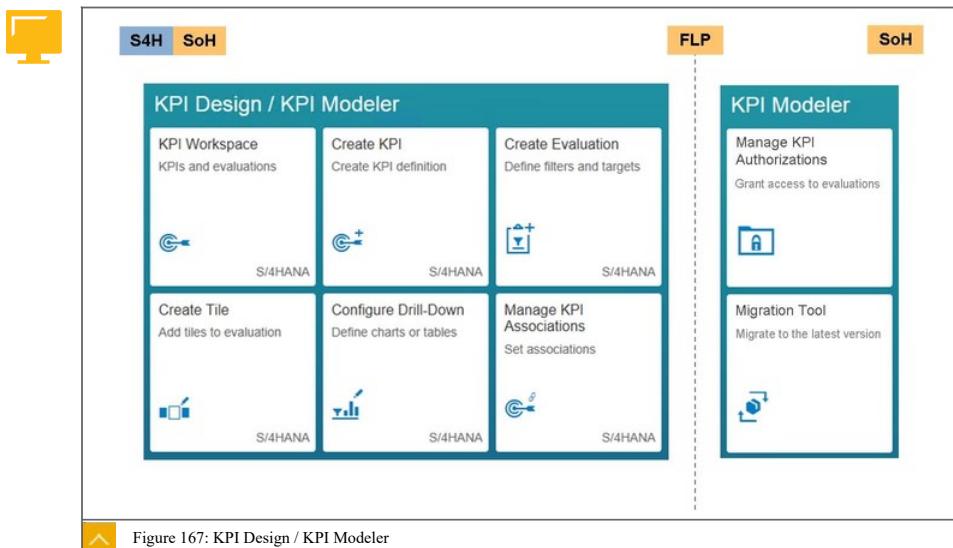
In SoH, the tool to create KPIs is called KPI Modeler. It is part of the KPI Framework, which is available as SAP Analytics Foundation for the FES. In contrast to S4H, the KPI Modeler and drill-down configurations communicate with SAP HANA XS OData services as part of the SAP HANA XS Engine. Therefore, an SAP Web Dispatcher is mandatory, providing routing rules as well as additional users and authorizations in SAP HANA.



The screenshot shows the SAP Fiori Smart Business - Configurations interface. At the top, there are tabs for S4H (selected) and SoH, and a button to 'Create KPI [F0817]'. Below the tabs, there is a section titled 'Configuration' with a note: 'The following sections list app-specific data required to configure the app:'. Under 'Technical Configuration', there are two main sections: 'Technical Catalog' and 'Business Catalog (Launchpad)'. The 'Technical Catalog' section includes fields for Technical Catalog (SAP_TC_CA_SSB_COMMON), Technical Catalog Group (empty), Technical PFCG Role (empty), Semantic Object (FioriApplication), Action (createSBKPI4HANA), and SAPUI5 Application (SB_APPS_KPI1). The 'Business Catalog (Launchpad)' section includes fields for Business Catalog (SAP_CA_BC_SSB), Business Group (SAP_CA_BCG_SSB), and PFCG role for Business Catalog (SAP_BR_ANALYTICS_SPECIALIST). The 'Create KPI [F0817]' button is located in the top right corner of the configuration area.

In this example from the [Fiori app reference library](#), you see the technical configuration of a SAP Smart Business app in S4H alongside the technical configuration of the same app in SoH. It is important to remember that S4H and SoH have different SAP Fiori content models.

Key Performance Indicator



The screenshot shows the KPI Design / KPI Modeler interface. At the top, there are tabs for S4H (selected) and SoH, and a tab for FLP (Fiori Launchpad). Below the tabs, there is a central 'KPI Workspace' section with sub-sections: 'KPIs and evaluations' (with a 'Create KPI' button), 'Create Evaluation' (with a 'Create KPI definition' button), 'Create Tile' (with a 'Add tiles to evaluation' button), 'Configure Drill-Down' (with a 'Define charts or tables' button), and 'Manage KPI Associations' (with a 'Set associations' button). To the right, there is a 'KPI Modeler' section with sub-sections: 'Manage KPI Authorizations' (with a 'Grant access to evaluations' button) and 'Migration Tool' (with a 'Migrate to the latest version' button). The 'Create KPI' and 'Create Evaluation' buttons are specifically marked as S/4HANA.

KPI Design and KPI Modeler follow the same process and handle similar elements to provide analytical data. For both, the central app is called [KPI Workspace](#). This is used to manage all KPIs and all its sub elements in the system. In addition, five connected apps exist to create

everything needed for SAP Smart Business . During the creation of a KPI, you jump from one app to another in the following order:

1. Create KPI
2. Create Evaluation
3. Create Tile
4. Configure Drill-Down
5. Manage KPI Associations (optional)

In an S4H environment, authorizations for the KPIs are handled using Fiori catalogs. In an SoH environment, additional authorizations in SAP HANA are needed. These are maintained using the Manage KPI Authorizations app of the KPI Modeler . Lastly, the KPI Modeler offers the Migration Tool app, which is only needed when working with the SAP HANA XS Engine , if there is an update.



Note:

The first version of the KPI Modeler was not integrated in the FLP as apps in a catalog. Instead, it used an own URL to the FES (http://<host>:<port>/sap/bc/ui5/ui5/sap/ca_kpi/modeler). The corresponding add-on UISKPI01 is no longer supported.

The screenshot shows the SAP Fiori Launchpad (FLP) interface. A yellow arrow points from the FLP icon in the top left to the 'Analytical Queries' catalog. Another yellow arrow points from the catalog to the 'View Details' screen of a specific query. The 'View Details' screen shows the following details for the query 'C_UX100FLIGHTBYAIRPORTQRY':

View Details

C_UX100FLIGHTBYAIRPORTQRY
UX100: Flights by Airport Query

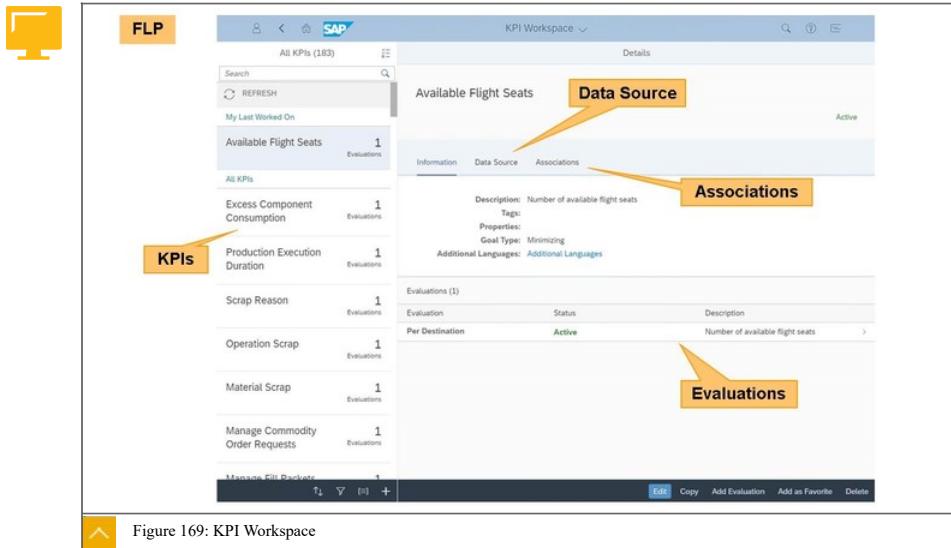
Application Component: CA
Description: Cross-Application Components
Tags:

Definition **Annotation**

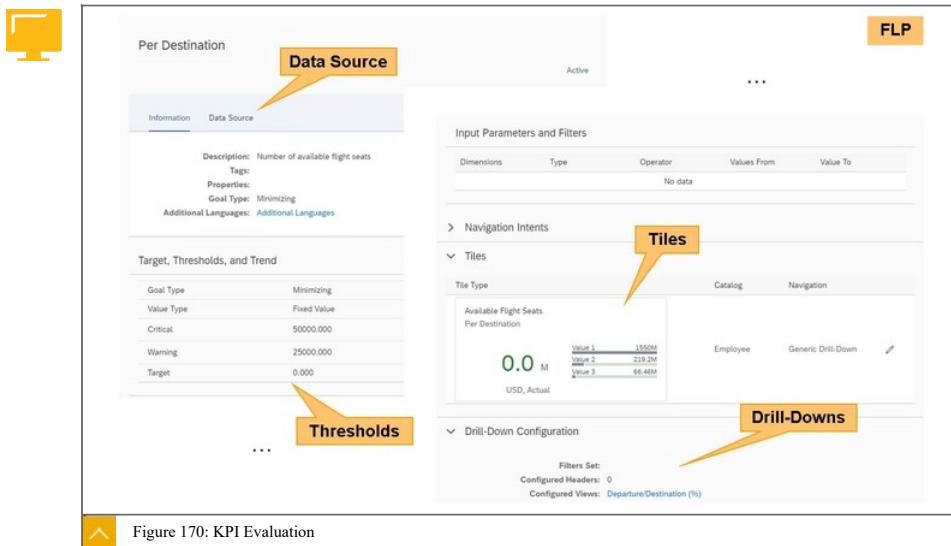
ABAPCATALOG_SQVIEWNAME: 'UX100FLIGHTBYAIRPO'
ANALYTICS_QUERY: true
ENCUSERTEXTLABEL: 'UX100: Flights by Airport Query'
ODATA_PUBLISH: true
VDM_VIEWTYPE: #CONSUMPTION

Figure 168: Query Browser

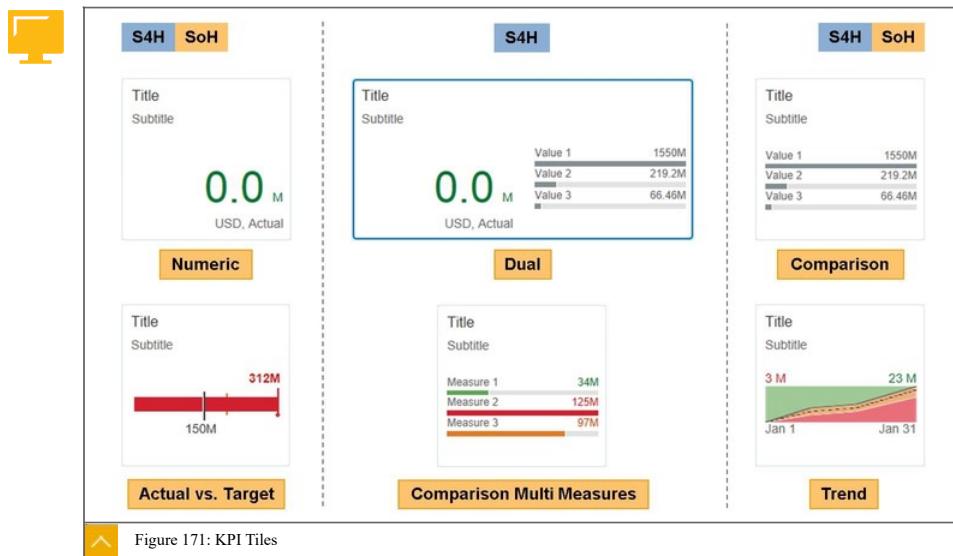
In SAP S/4HANA, an additional app called Query Browser is available. It shows CDS views, which are defined as analytics queries and, therefore, act as a source for KPIs. The app has its own business catalog called SAP_CA_BC_VDM.



The KPI Workspace shows a list of all KPIs in the system and their details such as data source, associations to other KPIs, and, most importantly, evaluations. KPIs can be edited, copied, or deleted, and the evaluations can be opened or added.



A KPI evaluation consists of a data source, thresholds, tiles, and drill-downs. The data source is an OData service (SAP Gateway for S4H and SAP HANA XS OData for SoH) offering all data needed for the KPI. Threshold, target, and trend values add the semantic to the data, which leads to the coloring of the numbers and graphs. The tiles offer an overview and short status of the data before the app is opened using graphs and numbers. Drill-downs are responsible for the visualization of the data.



There are four different tiles available in SoH and at least two additional ones in S4H. They are defined as part of an evaluation and saved in a Fiori catalog. Tiles consist of a configuration about how to visualize the data on the tile and the navigation target, what can be a self-defined target mapping to start a specific app or the generic drill-down.

- To navigate to the generic drill-down app in SAP Business Suite, the user needs the target mapping with semantic object * and action `analyzeSBKPIDetails`. It is part of every catalog containing analytical tiles by SAP and in the `SAP_CA_BC_SSB` business catalog.
- To navigate to the generic drill-down app in SAP S/4HANA 1511, the user needs the target mapping with semantic object * and action `analyzeSBKPIDetailsS4HANA`. It is part of every catalog containing analytical tiles by SAP and in the `SAP_CA_BC_SSB` business catalog.
- To navigate to the generic drill-down app since SAP S/4HANA 1610, the user needs the target mapping automatically generated when creating a tile for the KPI. It is saved in the same catalog as the tile.



Note:

The tiles had different names in the first version of the KPI Modeler. Numeric was called Point Value, Comparison was called Deviation, and Actual vs. Target was called Contribution. Only Trend did not change.

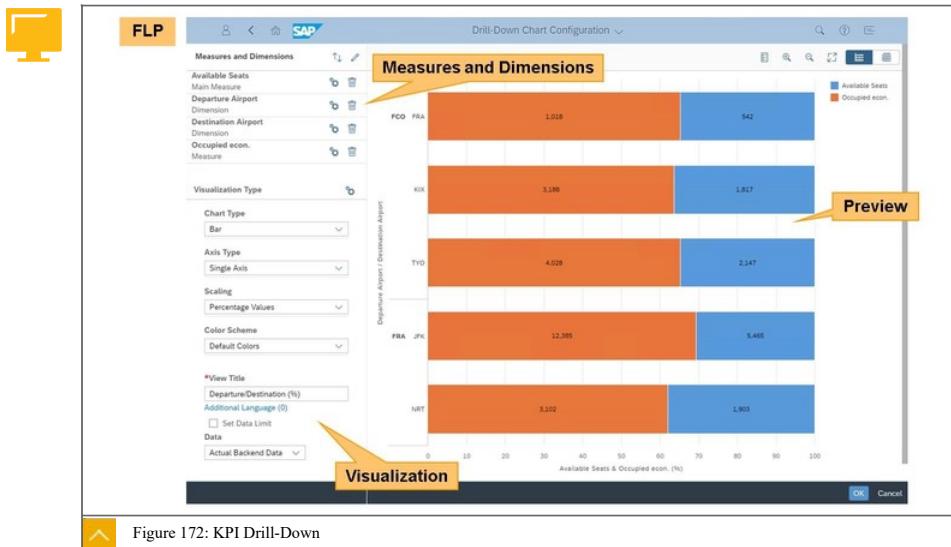


Figure 172: KPI Drill-Down

The generic KPI drill-down enables customers to visualize their business data in an app without the need for any development. Measures and dimensions define what data is used and in which way, and the visualization defines how the data is presented using diagrams and tables. The KPI drill-down configuration offers the possibility to use dummy data or actual back-end data including a preview.

SAP Fiori

Creating the drill-down is the last step in the process of creating a KPI. The only thing remaining is to add the catalog and all the KPI tiles it contains to a user role.



LESSON SUMMARY

You should now be able to:

- Enable SAP Fiori Smart Business
- Model SAP Fiori KPI Apps

11. What does a KPI consist of?

Choose the correct answers.

A Evaluation

B Authentication

C Drill-down

D Tile

E Indicator

Unit 4

Learning Assessment - Answers

1. Which layers of settings are available for SAP Fiori?

Choose the correct answers.

A Configuration

B Personalization

C Customizing

D Internationalization

Correct! The following layers of settings are available for SAP Fiori: Configuration, Personalization, and Customizing.

2. Which tool provides the definition of Fiori tiles?

SAP Fiori launchpad designer

3. Which element groups Fiori apps based on business area?

Technical Catalog (TC)

4. Which element groups Fiori apps based on business topic?

Business Catalog Group (BCG)

5. What is meant by the term legacy app descriptor?

Assignment of tiles and target mappings to transaction codes or Web Dynpro applications

6. For intent-based navigation, what does an intent consist of?

Choose the correct answers.

A Target mapping

B Semantic object

C Action

D ICF path

Correct! For intent-based navigation, an intent consist of: Semantic Object and Action.

7. What can be used as target in a target mapping?

Choose the correct answers.

A Web Dynpro application

B Generic URL

C ABAP Transaction

D SAPUI5 application

Correct! All of the above can be used as target in a target mapping.

8. Which tool is used in a Suite on HANA environment to model KPI?

KPI Modeler

9. Which tool is used in an S/4HANA environment to model KPIs?

KPI Design

10. Which ABAP transactions can be used to analyze the intents used in target mappings?

Choose the correct answers.

A /UI2/FSAC

B /UI2/FLIA

C /UI2/FLC

D /UI2/FLP

Correct! ABAP transactions /UI2/FLIA and UI2/FLC can be used to analyze the intents used in target mappings.

11. What does a KPI consist of?

Choose the correct answers.

A Evaluation

B Authentication

C Drill-down

D Tile

E Indicator

Correct! A KPI consist of: Evaluation, Drill-down, and Tile.

UNIT 5

Adaptation

Lesson 1

Using the UI Theme Designer	151
-----------------------------	-----

Lesson 2

Using SAP Screen Personas	156
---------------------------	-----

Lesson 3

Using SAP Fiori Runtime Authoring	165
-----------------------------------	-----

Lesson 4

Extending SAP Fiori Applications	168
----------------------------------	-----

UNIT OBJECTIVES

- Use the UI Theme Designer
- Use SAP screen personas
- Use SAP Fiori runtime authoring
- Extend SAP Fiori applications
- Extend SAP Fiori launchpad

Unit 5

Lesson 1

Using the UI Theme Designer

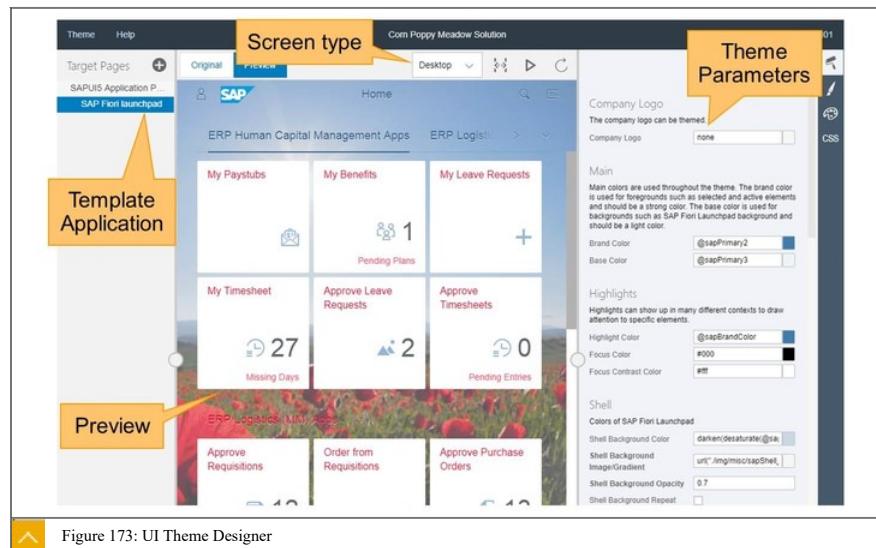


LESSON OBJECTIVES

After completing this lesson, you will be able to:

- Use the UI Theme Designer

Adaption of Themes



The UI Theme Designer is a browser-based, graphical WYSIWYG (what you see is what you get) editor. The built-in preview page not only allows you to see the effect of design changes instantly, it can also emulate different screen sizes and orientations on the fly. The tool can be used for quick, simple design changes up to a full implementation of a corporate design.

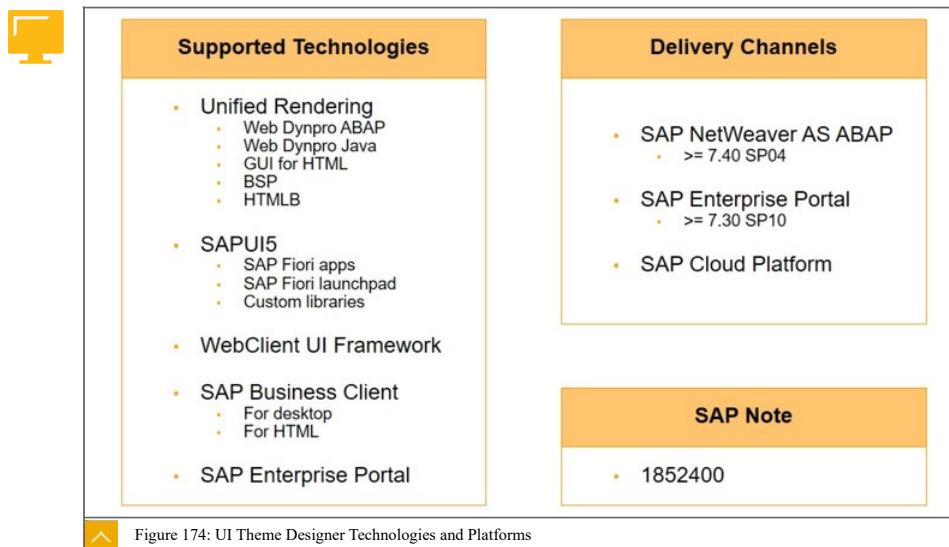


Figure 174: UI Theme Designer Technologies and Platforms

The UI Theme Designer supports all web-based SAP applications and clients based on HTML4 and HTML5. It is available in three channels:

SAP NetWeaver Application Server ABAP

Transaction /UI5/THEME_DESIGNER

SAP Enterprise Portal

Content Administration → Portal Display → Portal Themes → UI Theme Designer

SAP Cloud Platform (Trial)

<https://themedesigner-themedesigner.dispatcher.hanatrial.ondemand.com/>

Prerequisites, installation, and configuration are available in the

SAP Note 1852400.

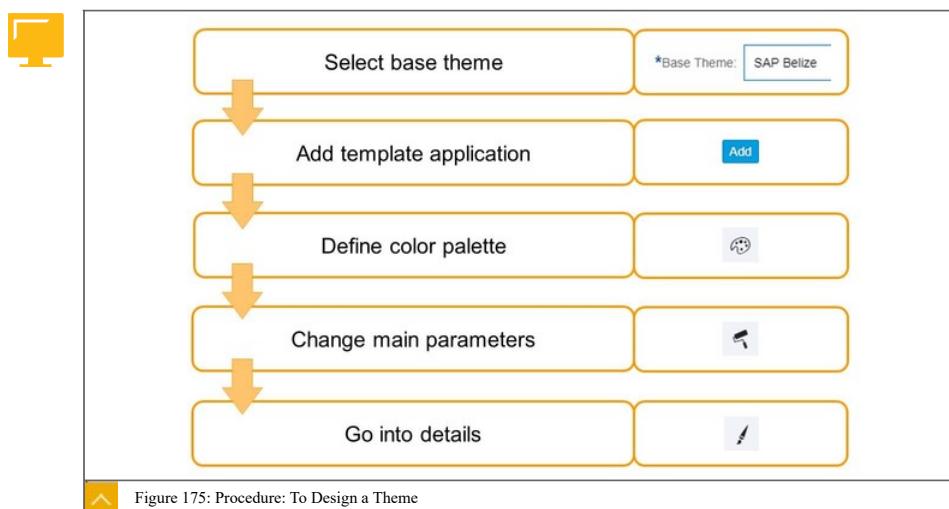
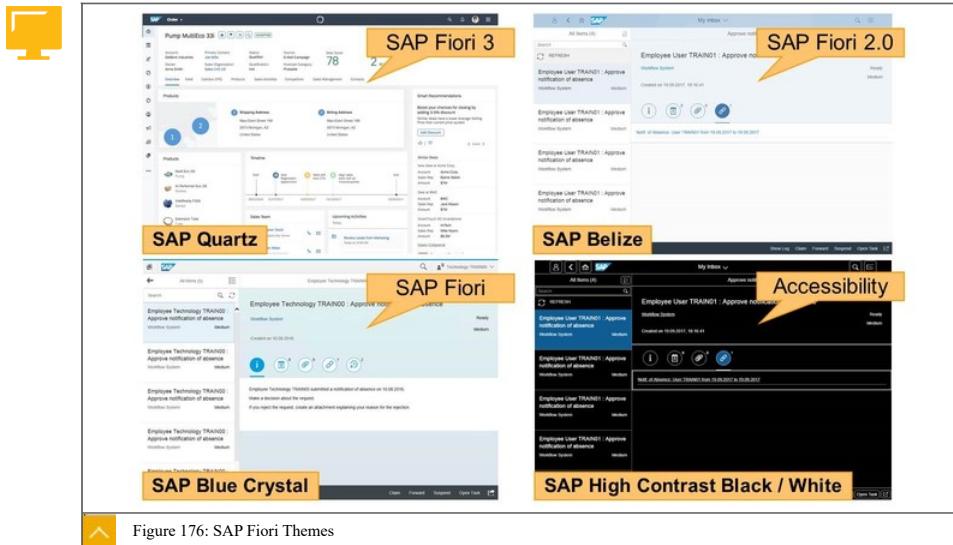


Figure 175: Procedure: To Design a Theme

All themes have a base theme (CSS files). For SAP Fiori and SAPUI5, this is SAP Blue Crystal. After choosing a template application to visualize the changes, the new theme is built upon a set of theme-related (specific style) CSS files. The theme generator merges the base theme with the specific style into the final theme.

SAPUI5 uses LESS to handle the CSS parameters and allows some additional features. LESS can be considered as a preprocessor that results in the final version of the CSS. Find out more at <http://lesscss.org/>.



The available base themes depend on the release of the system providing the UI theme designer. SAP has delivered the following themes for SAPUI5:

SAP Quartz

- Theme released for SAP Fiori 3
- Initial shipment in SAP Cloud Platform 1904 / SAPUI5 1.65
- Uses own typeface "72"

SAP Belize

- Theme released for SAP Fiori 2.0
- Initial shipment in software component SAP_UI 7.51 / SAPUI5 1.44
- Uses typefaces Arial Regular and Bold

SAP Blue Crystal

- Theme released for SAP Fiori
- Initial shipment in software component SAP_UI 7.40 / SAPUI5 1.28
- Uses typefaces Arial Regular and Bold

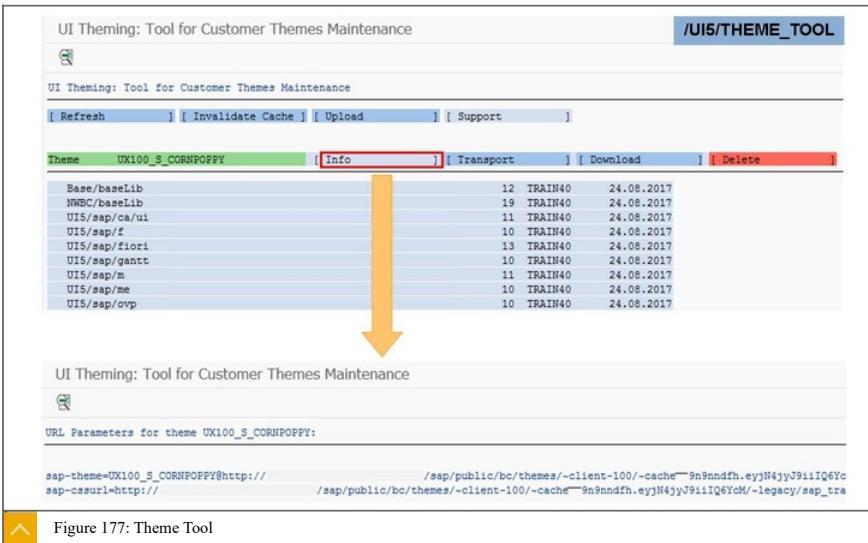
SAP High Contrast Black / White

- Theme released for accessibility purpose
- Initial shipment in software component SAP_UI 7.40 / SAPUI5 1.28
- Reworked in software component SAP_UI 7.51 for SAP Fiori 2.0

SAP Gold Reflection

- Theme released for non-Fiori standalone apps
- Initial shipment in SAPUI5 1.0
- Not working with SAP Fiori
- Deprecated

Maintenance of Themes



The screenshot shows the SAP UI5/Theme_Tool interface. The top part is a table listing themes, with a red box highlighting the 'Info' button for the theme 'UX100_S_CORNPOPPY'. A large orange arrow points downwards from this row to the bottom part of the interface. The bottom part shows URL parameters for the selected theme, including 'sap-theme=UX100_S_CORNPOPPY@http://sap-csurl=http://'. The URL is partially cut off at the end.

Theme	UX100_S_CORNPOPPY	[Info]	[Transport]	[Download]	[Delete]
Base/baseLib		12	TRAIN40	24.08.2017	
NGSC/baseLib		19	TRAIN40	24.08.2017	
UI5/sap/ca/ui		11	TRAIN40	24.08.2017	
UI5/sap/f		10	TRAIN40	24.08.2017	
UI5/sap/fiori		13	TRAIN40	24.08.2017	
UI5/sap/gantt		10	TRAIN40	24.08.2017	
UI5/sap/m		11	TRAIN40	24.08.2017	
UI5/sap/me		10	TRAIN40	24.08.2017	
UI5/sap/ovp		10	TRAIN40	24.08.2017	

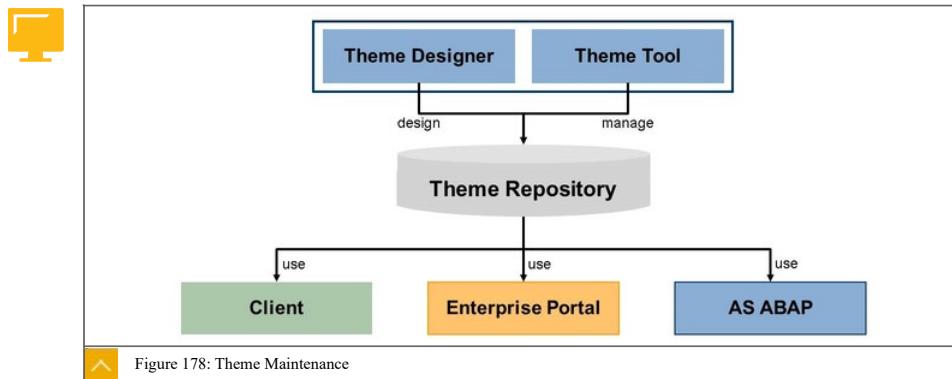
UI Theming: Tool for Customer Themes Maintenance

UI Theming: Tool for Customer Themes Maintenance

URL Parameters for theme UX100_S_CORNPOPPY:

sap-theme=UX100_S_CORNPOPPY@http://sap-csurl=http://sap/public/bc/themes/-client-100/-cache="9n9nndfh.eyJN4jyJ9iiIQ6Yc/sap/public/bc/themes/-client-100/-cache="9n9nndfh.eyJN4jyJ9iiIQ6YcM/-legacy/sap_tr...

The Theme Tool manages the themes in the theme repository of an SAP NetWeaver Application Server ABAP. Themes can be viewed, imported, exported, and deleted, as well as transported to the follow-up system. Here is also the easiest way to get a URI pointing to a certain theme. The tool can be started using the following transaction: /UI5/THEME_TOOL.



By adding the URI parameter `sap-theme=<theme-id>@<theme-root>` pointing to the theme repository to the application URI, the theme is used as the design for the application. This is available to all clients but stays visible in the URI.

The SAP Enterprise Portal and the SAP NetWeaver Application Server (NW AS) ABAP can also set central parameters for their applications to use a theme of the theme repository. No URI parameter is used and so no theme reference is visible. The central parameter for the SAP Fiori launchpad can be set in the transaction `/UI2/NWBC_CFG_CUST` and consists of the following:

- Filter = **SAP_FLP**
- Parameter Name = **THEME**
- Value = **<theme-name>**



LESSON SUMMARY

You should now be able to:

- Use the UI Theme Designer

Unit 5

Lesson 2

Using SAP Screen Personas



LESSON OBJECTIVES

After completing this lesson, you will be able to:

- Use SAP screen personas

Potential of SAP Screen Personas

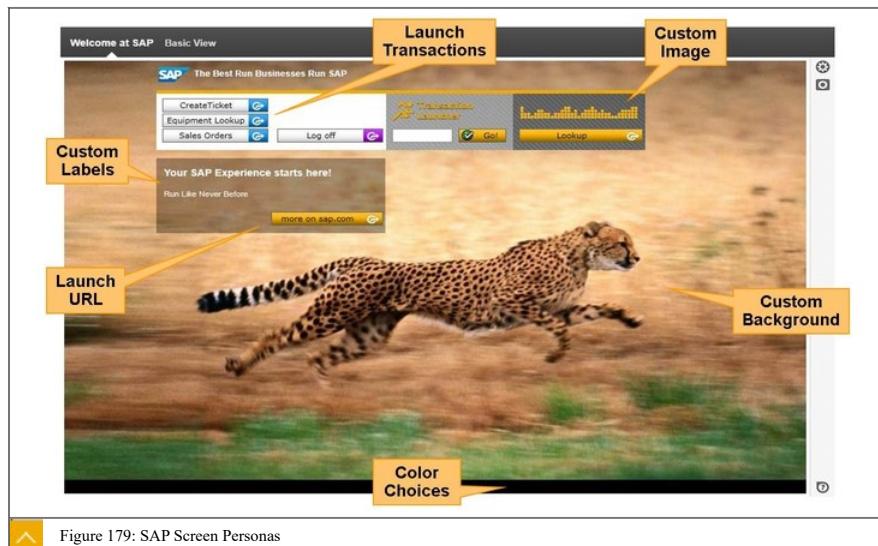


Figure 179: SAP Screen Personas

The SAP Screen Personas product allows companies to improve user productivity through personalization of their SAP GUI Dynpro and Web Dynpro screens. By providing the right information to the right users in the right context, they can get more done in less time with greater accuracy. With more intuitive SAP screens, organizations can enhance user satisfaction, minimize training time, and spend less money on screen modifications, as changes to the user experience are made without programming a single line of code.

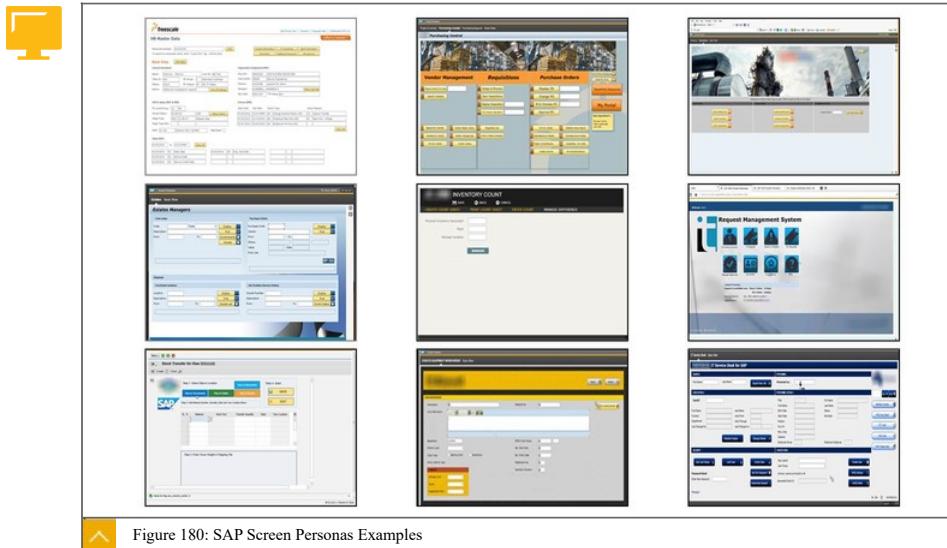


Figure 180: SAP Screen Personas Examples

Customers use SAP Screen Personas in many ways, across industries and lines of business. The common element is a desire to simplify the way people interact with SAP. These are all examples of screens adapted by customers using SAP Screen Personas. Some organizations choose to match their corporate colors or integrate simplified SAP screens into their corporate portal. A collection of flavors is available in the SAP Screen Personas Flavor Gallery: <http://link.personas.help/SAPScreenPersonasFlavorGallery>

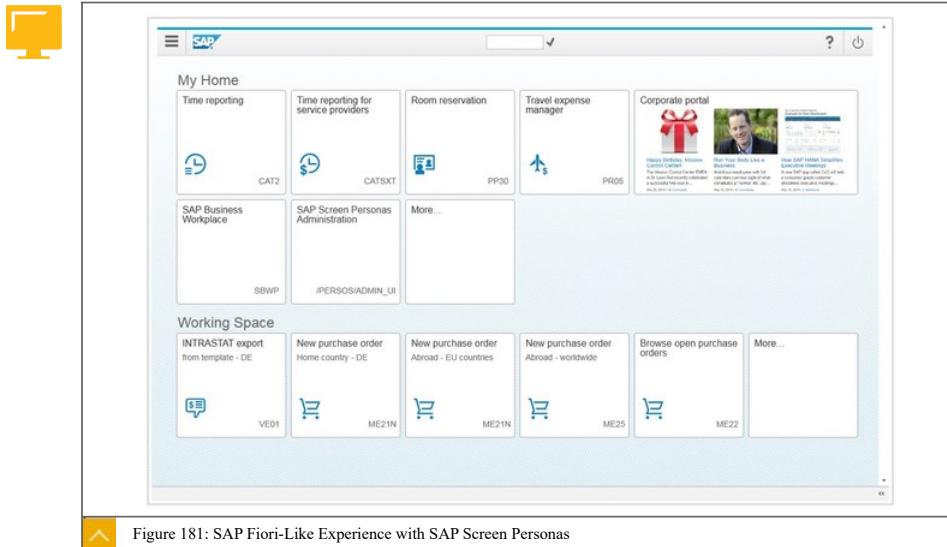
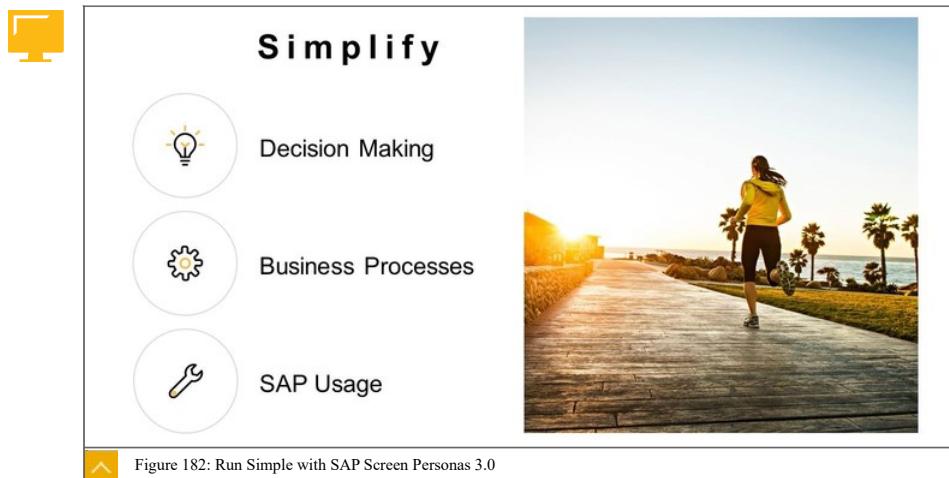


Figure 181: SAP Fiori-Like Experience with SAP Screen Personas

Using SAP Screen Personas, you have the capability to modify any SAP GUI interface into a design much like SAP Fiori. At first glance, this looks like a Fiori screen, but it was built using SAP Screen Personas. You can also use SAP Fiori launchpad or simply a web browser to

launch SAP Screen Personas transactions. The SAP Screen Personas Fiori style guide is available at <http://www.sapscreenpersonas.com>.



SAP Screen Personas is a key part of the SAP Run Simple approach. Customers that use this product to simplify their screens benefit from improved decision making, streamlined business processes, and a more consistent UX across their screens. It helps to do the following:

Simplify decision making

- Increase usage of SAP as a system of record so real-time information can drive decisions.
- Accelerate adoption through more intuitive user interaction.
- Make SAP more user-friendly by eliminating the need to memorize transactions codes.

Simplify business processes

- Streamline business processes into fewer screens.
- Consolidate multiple transactions, screens, or tabs into one simple screen.
- Automate repetitive actions or navigation into a single keystroke.

Simplify SAP usage

- Standardize one consistent user experience/theme for all SAP transactions.
- Reduce training requirements through intuitive interfaces and navigation.
- Embed training content into screens, including video tutorials, notes, and roll-over help.

Features of SAP Screen Personas

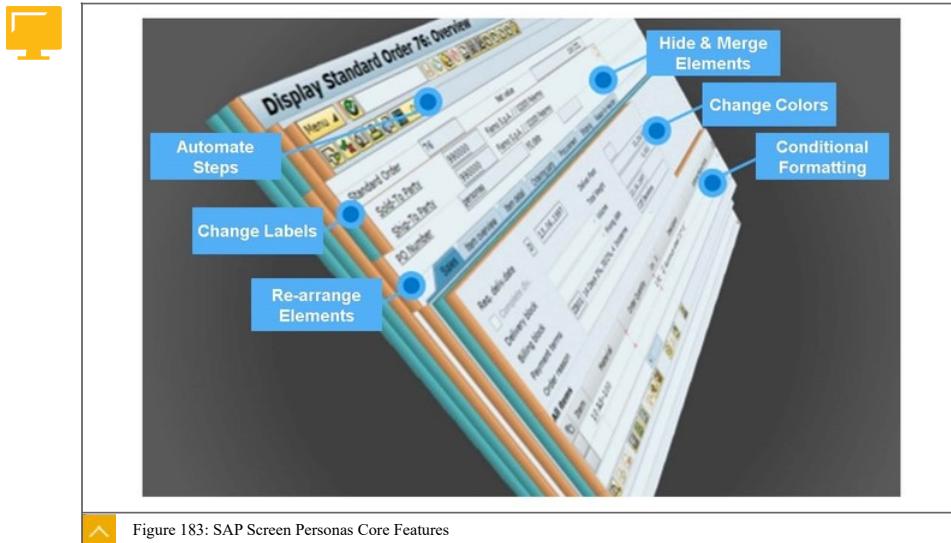
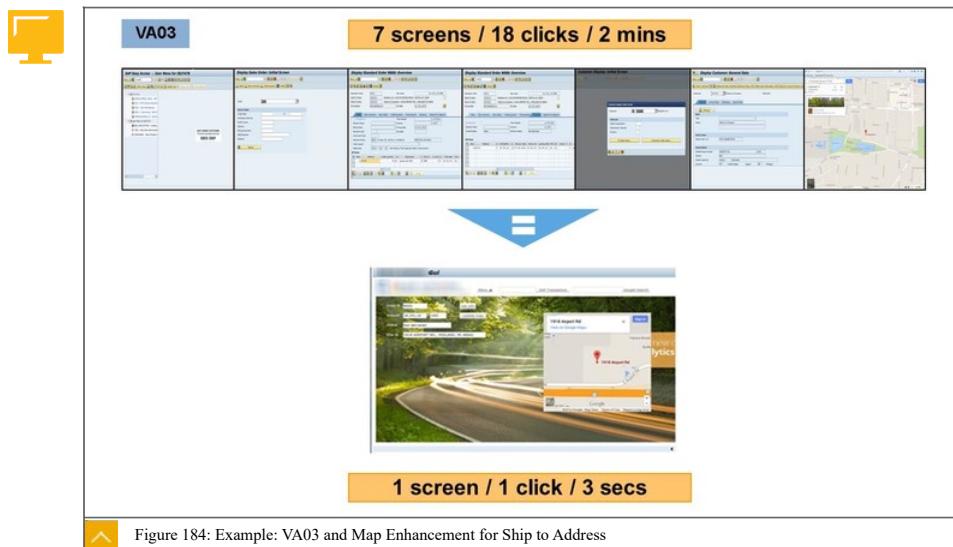


Figure 183: SAP Screen Personas Core Features

SAP Screen Personas is available for AS ABAP as an add-on. It creates an additional HTML5 layer on top of the normal UI and saves it in the ABAP system. Using this additional layer, customers can add, rename, delete, or re-arrange every element on the screen. There are a wide range of formatting options available to change colors, formatting, styles, fonts, and pictures. By using JavaScript, macros can be created to automate steps like filling fields and navigating to screens.

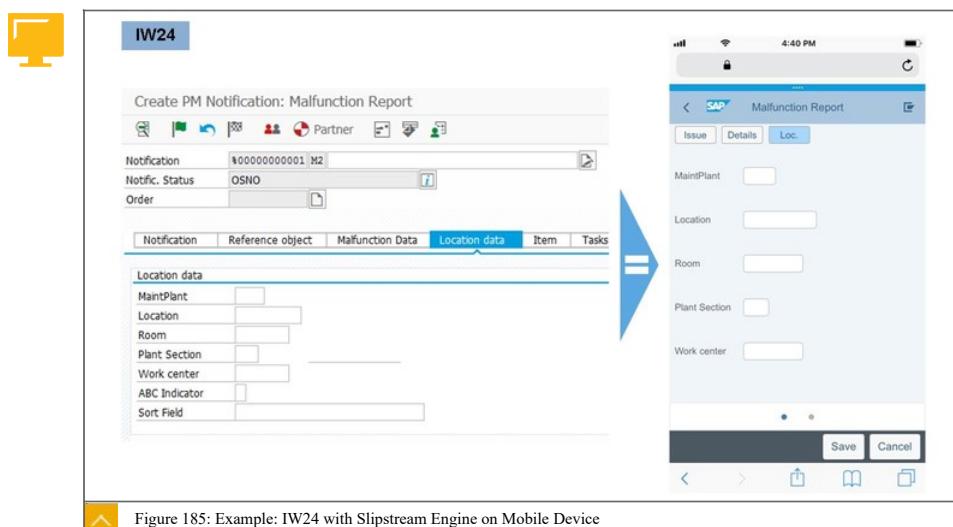
The following clients are supported for SAP Screen Personas :

- SAP Business Client
- SAP GUI for HTML
- SAP GUI for Windows (see SAP Note [2080071](#))
- SAP GUI for Java (see SAP Note [2505697](#))



This is an example of looking up a sales order in VA03 and then creating a map for the ship-to location. Using SAP GUI, this takes 7 screens and 18 clicks, including going to Google to create the map. The entire process takes a skilled user more than two minutes.

With SAP Screen Personas, you can simplify everything down to a single click on a single screen that takes a few seconds to render. This is the type of productivity savings that customers see when using SAP Screen Personas.

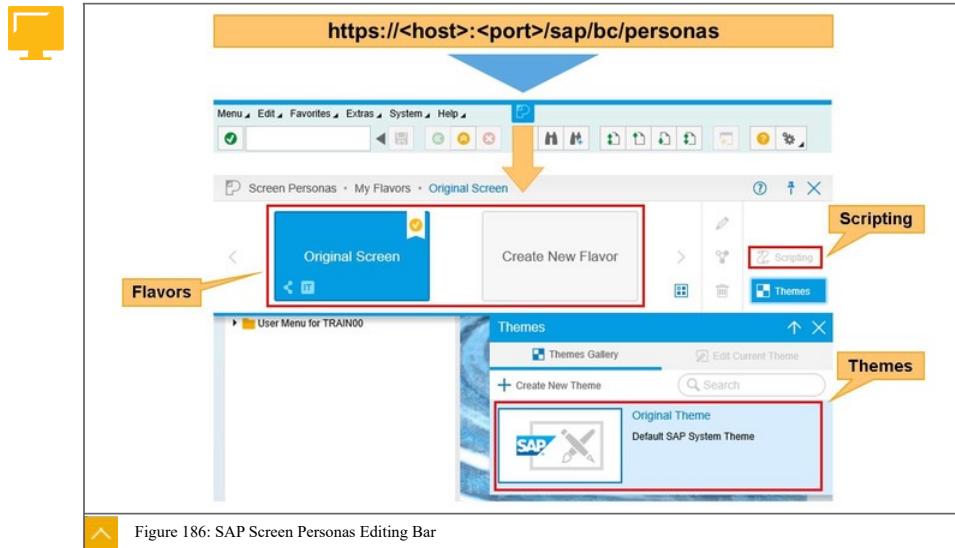


Slipstream Engine provides an alternative browser-based rendering approach for SAP GUI screens. It is a component of SAP Screen Personas introduced with version 3.0 SP06 and allows to render flavors on mobile devices such as tablets or smart phones. It is an SAPUI5

application that runs in a browser and obtains screen definitions from the Internet Transaction Server (ITS).

Building mobile flavors follows the same process as building desktop flavors. But it is recommended to use the SAP Screen Personas editor running in Slipstream Engine available since version 3.0 SP07, not the editor in SAP GUI for HTML.

For more information about Slipstream Engine, see SAP Note [2557076](#).



SAP Screen Personas can be accessed using the ICF service /SAP/BC/PERSONAS, which starts SAP GUI for HTML including the Screen Personas editing bar. By clicking on the blue P at the top of the screen (which appears when you hover the mouse), the editing bar opens. Here, you can select or create flavors to change the UI elements on the current screen, themes to change colors and styles, and scripting for automation.

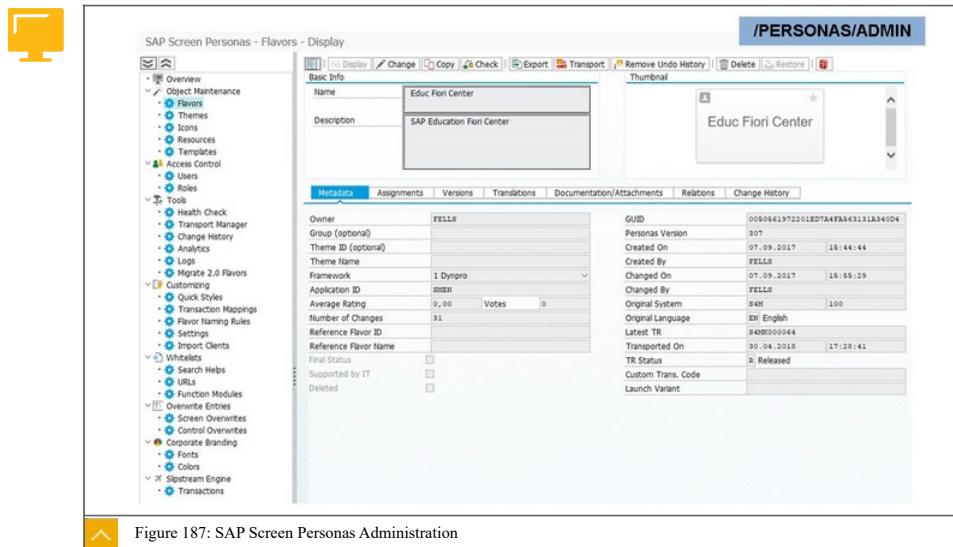


Figure 187: SAP Screen Personas Administration

There are several transactions available to administer SAP Screen Personas, but the central transaction is /PERSONAS/ADMIN, which offers a dashboard to access all other transactions. It is divided into six sections:

- Flavor Maintenance
- Theme Maintenance
- Resources Maintenance
- User and Role Maintenance
- Icon Maintenance
- Additional Administrative Tasks

There are many options available for administrators, such as assignment of flavors and themes to users or roles, up-loading and downloading of resources, or transporting elements to other systems.

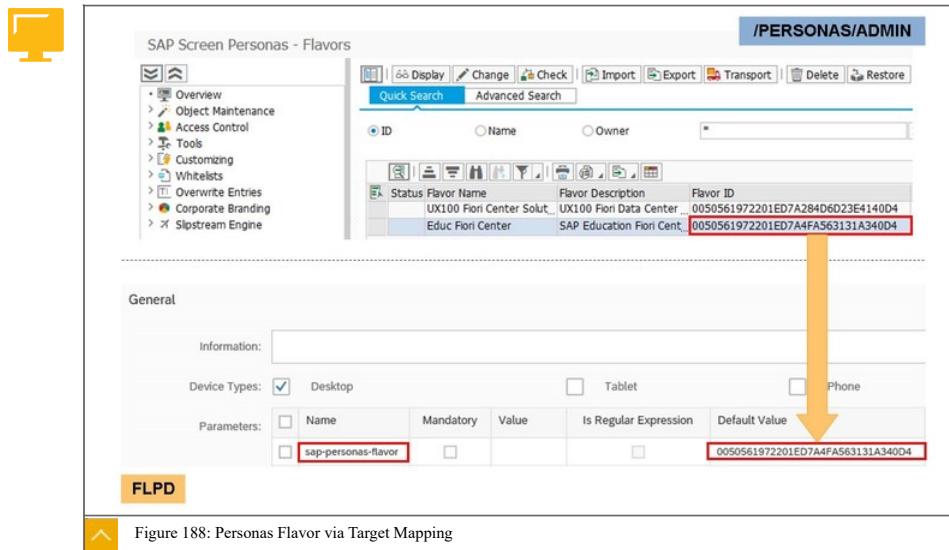


Figure 188: Personas Flavor via Target Mapping

Flavors can also be assigned to target mappings for Web Dynpro ABAP and SAP GUI transactions using the parameter `sap-personas-flavor`. The flavor ID for the default value of the parameter is available in the flavor manager or in transaction `/PERSONAS/ADMIN`. The flavor is then used every time for every user starting the app via this target mapping.

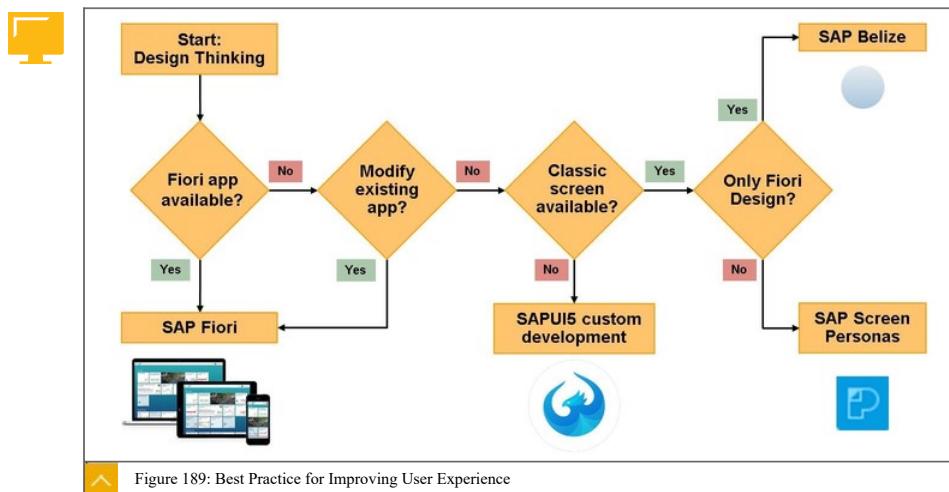


Figure 189: Best Practice for Improving User Experience

Flavors or themes are not shipped by SAP. SAP Screen Personas is a tool that allows customers to create their own user experience. The question is when customers should invest in creating flavors or themes. This best practice decision map can help customers to decide this.



Note:

More information about this topic can be found in UX300 (SAP Screen Personas Development):

<https://training.sap.com/course/ux300>



LESSON SUMMARY

You should now be able to:

- Use SAP screen personas

Unit 5

Lesson 3

Using SAP Fiori Runtime Authoring



LESSON OBJECTIVES

After completing this lesson, you will be able to:

- Use SAP Fiori runtime authoring

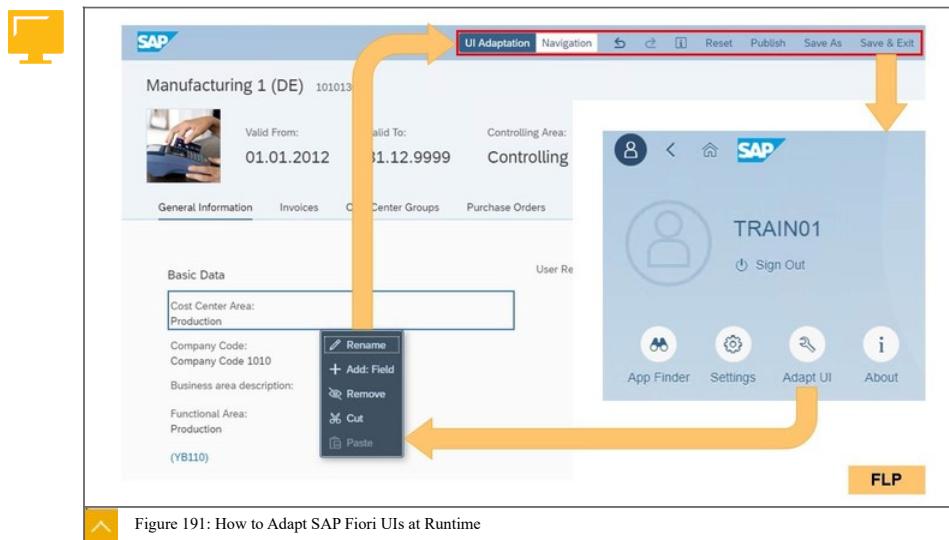
Adaptation at Runtime



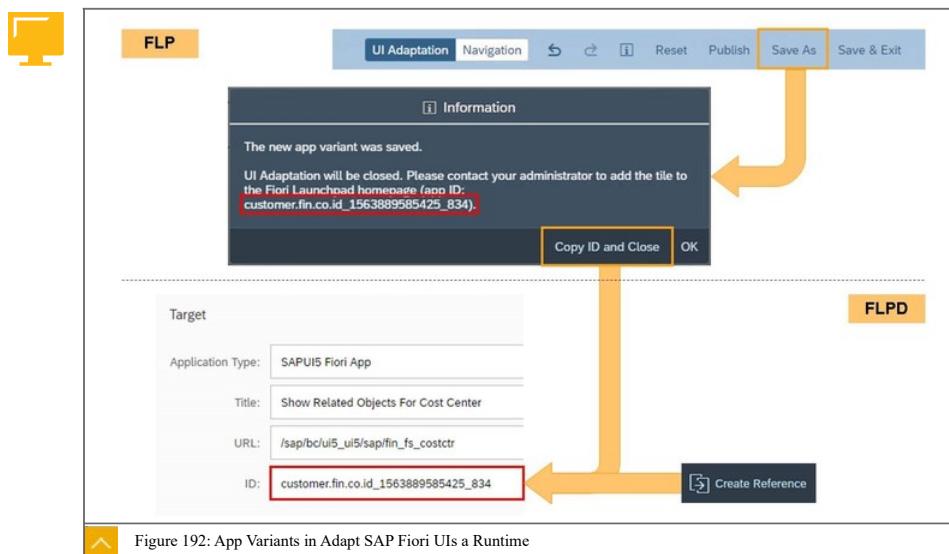
Figure 190: Adapt SAP Fiori UIs at Runtime

With UI adaptation at runtime (RTA), SAP Fiori apps based on SAPUI5 that support RTA can be changed directly and intuitively in the Fiori launchpad without developing any source code. Changes done via RTA apply to the entire app and, therefore, to all users of the app system-wide. All changes to an app can be undone or even discarded to reset the UI to the default app. Apps based on SAP Fiori elements support RTA by default.

SAPUI5 flexibility demo apps are available for testing in the SAPUI5 homepage: <https://ui5.sap.com/#/demoapps>



The UI adaptation mode can be accessed via the Adapt UI button in the Me Area for the running app. This displays a new header with actions around RTA. By clicking UI elements in the app, a dialog box appears with the possible changes. Single changes can be undone one by one, all changes can be reset, or the changes can be published for all users in the system and for transporting the changes to a follow-up system.



By choosing Save As, the changes can be saved as app variant. It is possible to save multiple variants of one application and also of other variants. Each variant has an app ID, which can be used in the SAP Fiori launchpad designer to create a target mapping starting the app with this

variant. The target mapping can be created by referencing the original target mapping of the app and exchanging the ID with the app ID of the app variant.

Runtime Authoring Plugin

Figure 193: SAP Runtime Authoring Plugin

To be able to adapt Fiori apps at runtime, the user needs a target mapping to the runtime authoring plugin. No additional settings are needed to activate the **Adapt UI** button in the **Me Area**. SAP ships the `/UIF/SAP_RTA_PLUGIN` catalog and the `SAP_UI_FLEX_KEY_USER` role with a preconfigured target mapping.



LESSON SUMMARY

You should now be able to:

- Use SAP Fiori runtime authoring

Unit 5

Lesson 4

Extending SAP Fiori Applications

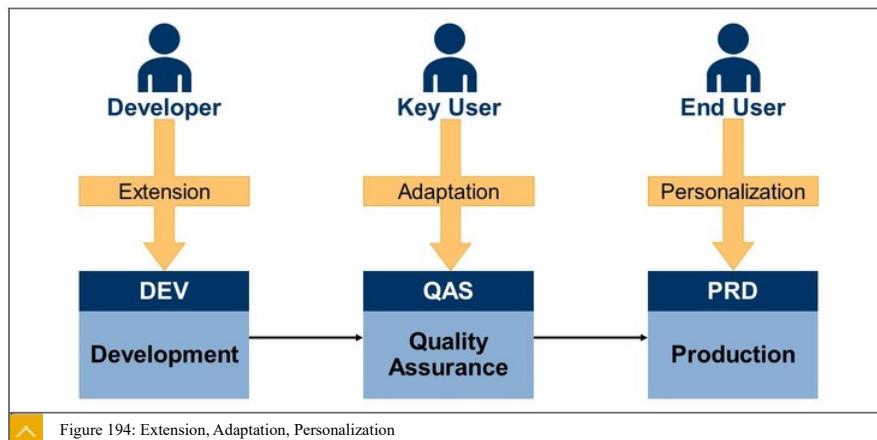


LESSON OBJECTIVES

After completing this lesson, you will be able to:

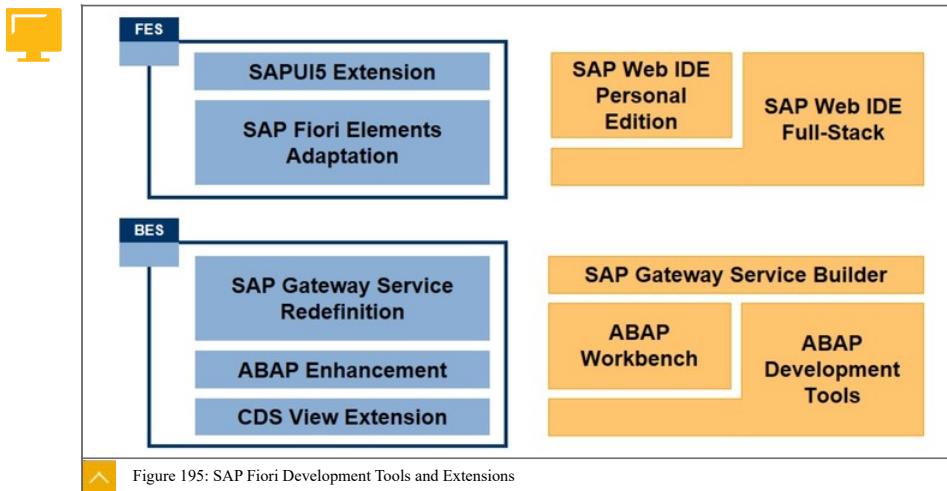
- Extend SAP Fiori applications
- Extend SAP Fiori launchpad

SAP Fiori Extension Options



In a three-system landscape, changes to SAP Fiori applications are possible in every system but performed by different users with different goals:

- The end user performs personalization in the SAP Fiori launchpad with a limited tool set. The changes are only valid for him or herself.
- The key user performs adaptation switching to the adaption mode in the SAP Fiori launchpad. The tool set is limited to the features implemented in the application. The changes are valid for a group of users or up to all users in a system.
- The developer performs extension using a wide range of tools and frameworks. Depending on the technology, there are nearly no limitations for changing apps. The changes are valid for all users in a system.



Each technology defines its own naming when talking about extensibility. Technologies used in the front-end server providing the user interface of an app are:

- SAPUI5 → Extension
- SAP Fiori Elements → Adaptation

SAP Web IDE is the only tool for extending SAPUI5 and SAP Fiori elements but with restrictions for the second one when using the Personal Edition depending on the release.

Technologies used in SAP Fiori in the back-end server providing the logic of an app are:

- SAP Gateway → Redefinition
- ABAP → Enhancement
- CDS View → Extension

The ABAP Development Tools (ADT) are used for all technologies, the ABAP Workbench (SE80) is used with ABAP and SAP Gateway, and the SAP Gateway Service Builder (SEGW) as its name says.

CDS View Extension

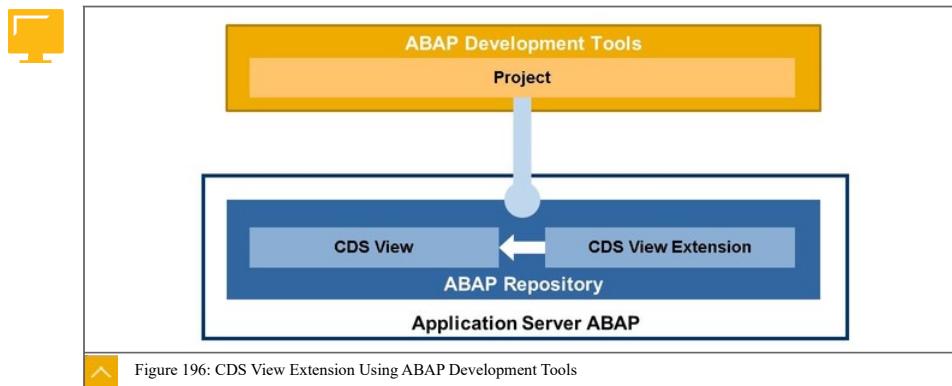


Figure 196: CDS View Extension Using ABAP Development Tools

The ABAP Development Tools (ADT) define a project for connecting to the repository of an ABAP system. A CDS view is saved as part of a data definition in the ABAP repository. To extend a CDS view, a CDS view extension needs to be created in the customer namespace referencing the original CDS view.

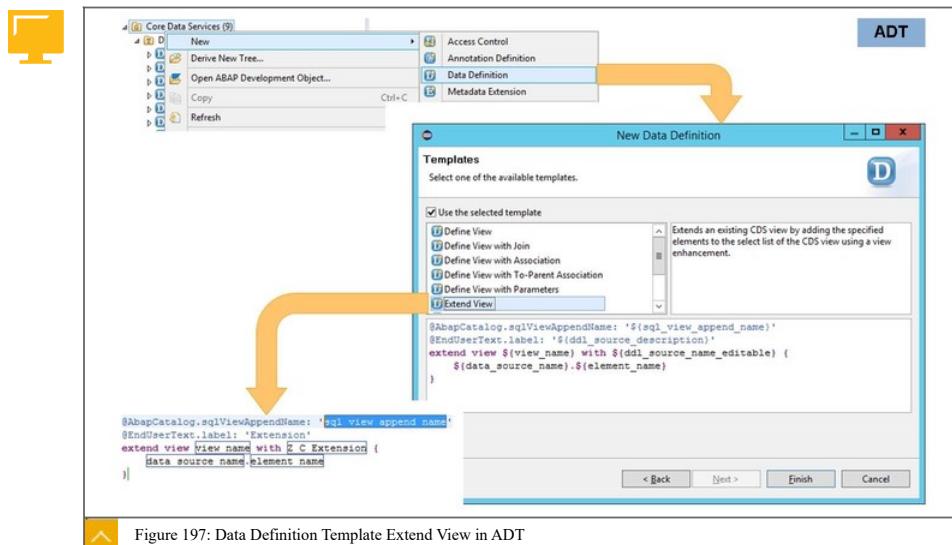


Figure 197: Data Definition Template Extend View in ADT

There is not a special button in ADT to create an extension but a template for creating a data definition. Inside of the data definition, the CDS View extension refers to the original CDS view using the syntax `extend view <view_name> with <extension>`.

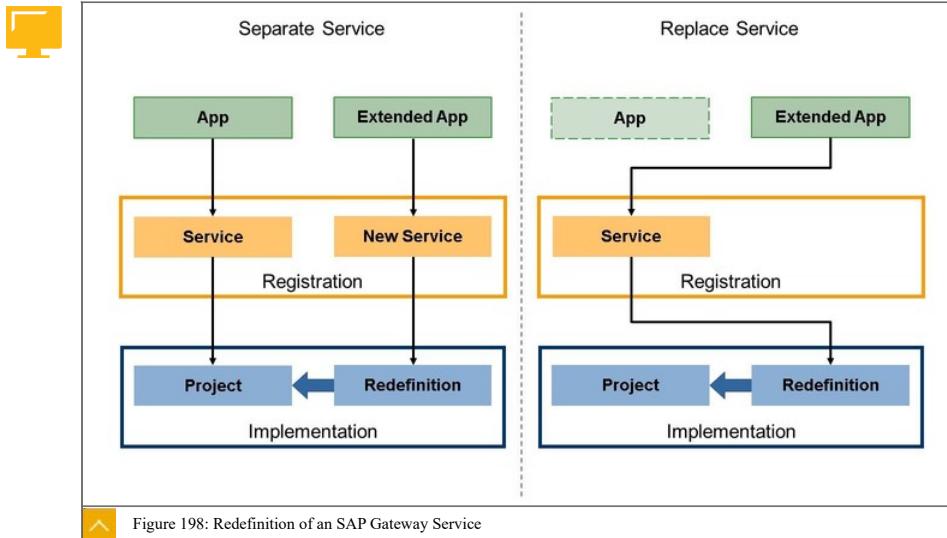


Note:

More information about this topic can be found in S4D430 (Building Views in ABAP Core Data Services (CDS)):

<https://training.sap.com/course/s4d430>

SAP Gateway Service Redefinition

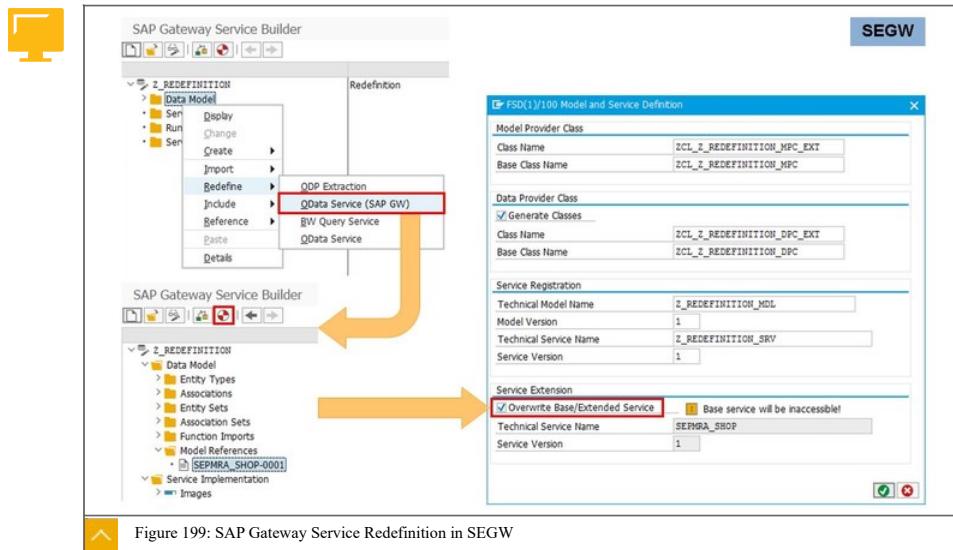


Each gateway service is organized in a gateway project and developed using ABAP. The enhancement framework is used to enhance services delivered by SAP on the ABAP level. The counterpart on project level is called redefinition. A redefined project references all of the original project and adds or subtracts elements of the model and implementation.

When registering the redefinition, customers have two options:

1. Register a new service and preserve the original one.
2. Replace the original service.

The first option allows you to continue using the original app and use a new extended app in parallel with it. The second option makes it impossible to use the original app anymore, because it involves replacing the project behind the original service with the redefinition. Both are valid options, depending on the use case.



After creating a new model in the SAP Gateway Service Builder (SEGW), it is possible to use the **Redefine** option in the menu of the data model to reference the model of the original service. When generating the runtime objects the first time, a checkbox offers the possibility to overwrite the original base service with the new extended one.



Caution:

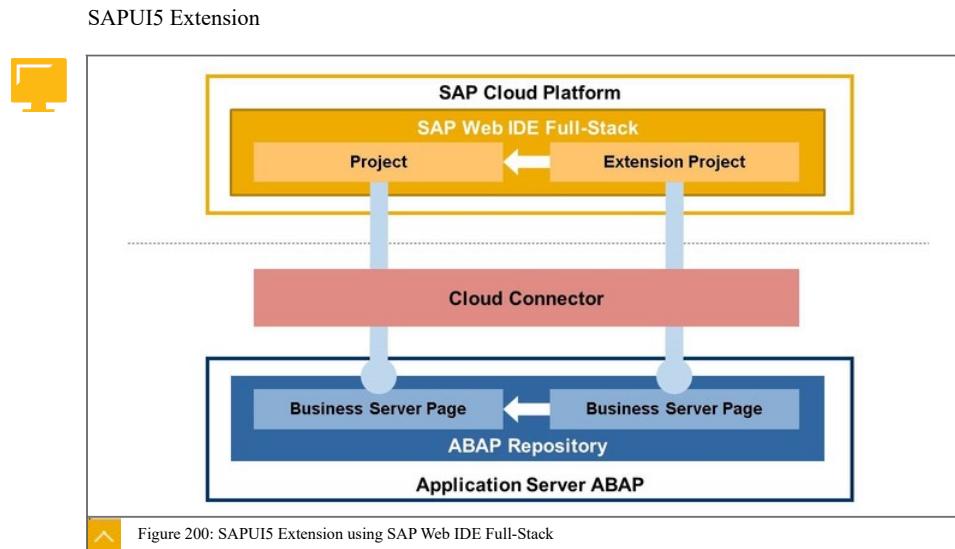
Be mindful when choosing to overwrite the original service. This cannot be easily undone.



Note:

More information about this topic can be found in GW100 (SAP Gateway – Building OData Services):

<https://training.sap.com/course/gw100>

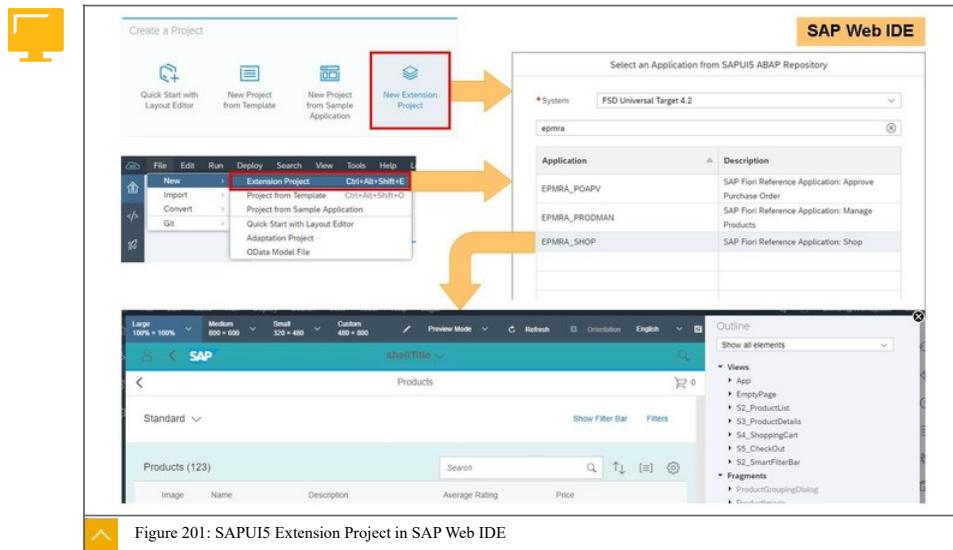


SAP Web IDE in all versions offers a complete tool-set for extending SAPUI5 applications. The Full-Stack in the SAP Cloud Platform offers in addition plugins or extensions, which can be helpful beyond the pure extension. A project in the SAP Web IDE is a local copy of the files inside of a Business Server Page (BSP) in an ABAP system.



Note:
The Cloud Connector is only needed when using SAP Web IDE Full-Stack .

SAPUI5 builds on jQuery and its feature of being extensible. That means that SAPUI5 inherits this feature. Customers are able to extend every SAPUI5 app, including every SAP Fiori web app. The process involves downloading the original app as a project in the SAP Web IDE, creating an extension project referencing the original one, programming the differences, and uploading the extended app as a new BSP in the ABAP repository. When calling the extended app, anything that has not been changed is processed by the referenced original app, and all elements that have been changed are processed by the extended app.



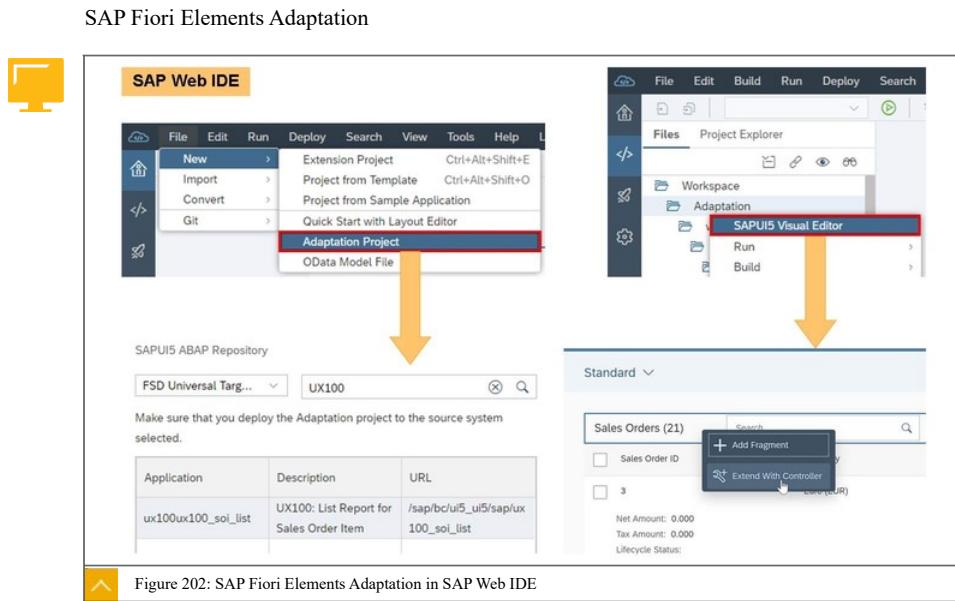
SAP Web IDE offers a template to create an extension project. This template connects to the ABAP system and downloads all information needed for the extension to a new project. As a WYSIWYG (What You See Is What You Get) editor, the Extensibility Pane then allows plenty of changes. SAP delivers prepared extension points and hook methods to ease the extension of the UI for certain purposes. Extension points for each app are documented in the SAP Fiori reference library .



Note:

More information about this topic can be found in UX410 (Developing SAP Fiori UIs):

<https://training.sap.com/course/ux410>

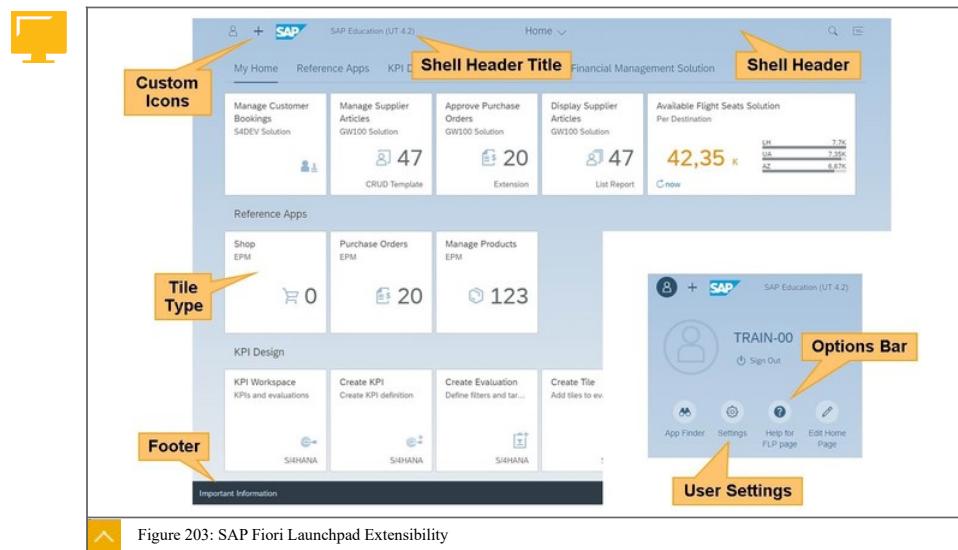


Similar to SAPUI extensions, all tools needed to adapt an SAP Fiori elements app are included in the SAP Web IDE and all extensions are saved in an own project deployed as a BSP in the ABAP system. The difference is that you do not develop JavaScript but you define metadata to create or change SAP Fiori elements. Therefore there are different tools available.

Creating an adaptation project is done by using the corresponding wizard referencing the original app in an ABAP system. In fact, you create a new variant of an existing application. The app variant refers to the original application but contains a separate set of changes created in the adaptation project. Also, an application ID is defined for the variant to enable a separate registration in the SAP Fiori launchpad.

The SAPUI5 Visual Editor is then again a WYSIWYG editor to preview the application, make changes to the properties of controls, add code fragments, extend with controller, and see a list of changes that had been made.

SAP Fiori Launchpad Plugins



The SAP Fiori launchpad (FLP) can be extended in its functionality by using plugins. Adapting the shell header, adding a footer, defining additional buttons, or creating tile types are just some examples. Plugins have the following characteristics:

- Loaded automatically when FLP starts
- Implemented as SAPUI5 component
- Implemented in a platform-independent way
- Configured client-wide or assigned to roles

Developing a plugin involves the following steps:

1. Implementing the plugin
2. Activating and configuring the plugin

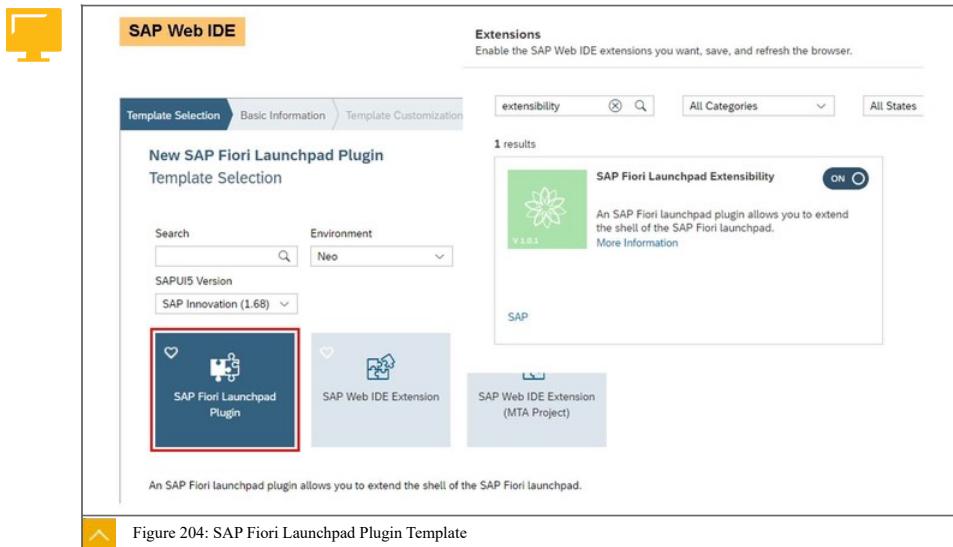


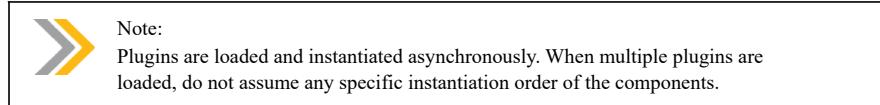
Figure 204: SAP Fiori Launchpad Plugin Template

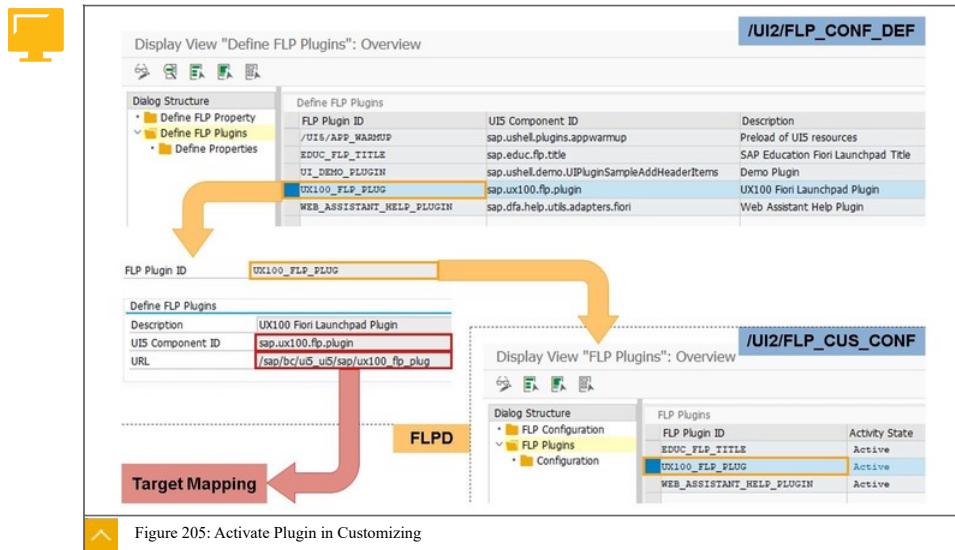
The SAP Web IDE Full-Stack in the SAP Cloud Platform offers an extension called SAP Fiori Launchpad Extensibility. The main component of this extension is a template to create an SAP Fiori Launchpad Plugin. This template generates the frame for developing plugins and some example code, too.

There are no views in a plugin in general. All changes to the FLP are defined by accessing the renderer of the FLP in the `Component.js` file. This enables the FLP to load and instantiate the plugins automatically during start-up. A code changing the shell header looks like this:

```
...
init: function () {
  var rendererPromise = this._getRenderer();

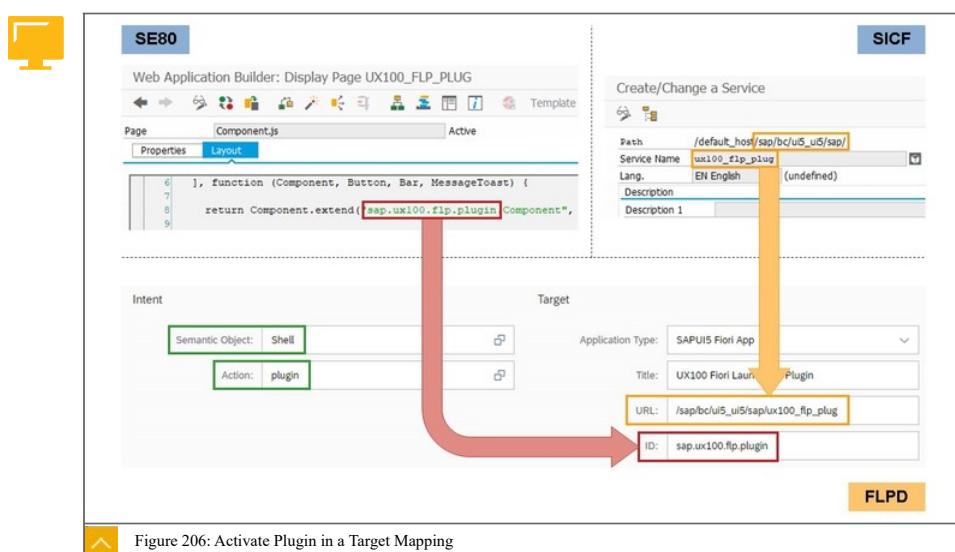
  /**
   * Set header title
   */
  rendererPromise.then(function (oRenderer) {
    oRenderer.setHeaderTitle("SAP Education (UT 4.2)");
  });
},
...
...
```





Every plugin needs to be defined in transaction `/UI2/FLP_CONF_DEF` before it can be used in the FLP. The plugin needs an FLP Plugin ID as well as the component ID and URL (ICF path) like other SAPUI5 applications.

After the plugin is defined, it can be activated in transaction `/UI2/FLP_CUS_CONF` for all users in the system client. Plugins have an own branch in the structure of the transaction. This allows the user to add additional configuration to the activation, which can be used in the plugin during runtime.



Like other FLP configurations, it is also possible to activate and assign the plugin to roles using a target mapping in a catalog. The component ID and URL can be copied from

transaction `/UI2/FLP_CONF_DEF` or by accessing the source code and the ICF path directly.
The intent has to consist of `Shell` as semantic object and `plugin` as action.



LESSON SUMMARY

You should now be able to:

- Extend SAP Fiori applications
- Extend SAP Fiori launchpad

Unit 5

Learning Assessment - Answers

1. Which system environments offer the UI theme designer?

Choose the correct answers.

- A SAP Enterprise Portal
- B SAP NetWeaver Application Server ABAP
- C SAP Cloud Platform
- D SAP NetWeaver Application Server Java
- E SAP Business Intelligence

Correct! The UI theme designer offers the following system environments: SAP Enterprise Portal, SAP NetWeaver Application Server ABAP, and SAP Cloud Platform.

2. Which design template is used for SAP Fiori 2.0?

SAP Belize

3. Where are the themes saved for the Fiori launchpad?

Theme repository in front-end server

4. What is SAP Screen Personas?

A web-based layer for personalization of SAP GUI Dynpro and Web Dynpro ABAP

5. Which UI clients does SAP Screen Personas support?

Choose the correct answers.

A SAP Business Client

B SAP Fiori Client

C SAP GUI for Windows

D SAP GUI for HTML

E SAP GUI for Java

Correct! SAP Screen Personas supports the following UI clients: SAP Business Client, SAP GUI for Windows, SAP GUI for HTML, and SAP GUI for Java.

6. What is created with SAP Screen Personas to change the structure of classical screens?

Flavors

7. How do you enter the UI adaptation mode of SAP Fiori apps?

Start app → Me Area → Adapt UI

8. How can you enable adaptation at runtime for SAP Fiori apps?

Choose the correct answers.

A Target mapping to runtime authoring plugin

B Fiori catalog SAP_CA_BC_SSB

C User role SAP_UI_FLEX_KEY_USER

D Fiori catalog /UIF/SAP_RTA_PLUGIN

E User role SAP_S_RFCACL

Correct! You can enable adaptation at runtime for SAP Fiori apps by: Target mapping to runtime authoring plugin, by User role SAP_UI_FLEX_KEY_USER, and by Fiori catalog /UIF/SAP_RTA_PLUGIN.

9. Which terms are correct when talking about extensibility for SAP applications in a front-end server?

Choose the correct answers.

- A** SAP Gateway Redefinition
- B** CDS View Extension
- C** SAPUI5 Extension
- D** ABAP Enhancement
- E** SAP Fiori Elements Adaptation

Correct! When talking about extensibility for SAP applications in a front-end server, the correct terms are: SAPUI5 Extension and SAP Fiori Elements Adaptation.

UNIT 6

Mobility

Lesson 1

Examining SAP Fiori Mobile

186

Lesson 2

Examining SAP Fiori for iOS

193

UNIT OBJECTIVES

- Examine SAP Fiori Mobile
- Examine SAP Fiori for iOS

Unit 6

Lesson 1

Examining SAP Fiori Mobile



LESSON OBJECTIVES

After completing this lesson, you will be able to:

- Examine SAP Fiori Mobile

Consumption Options

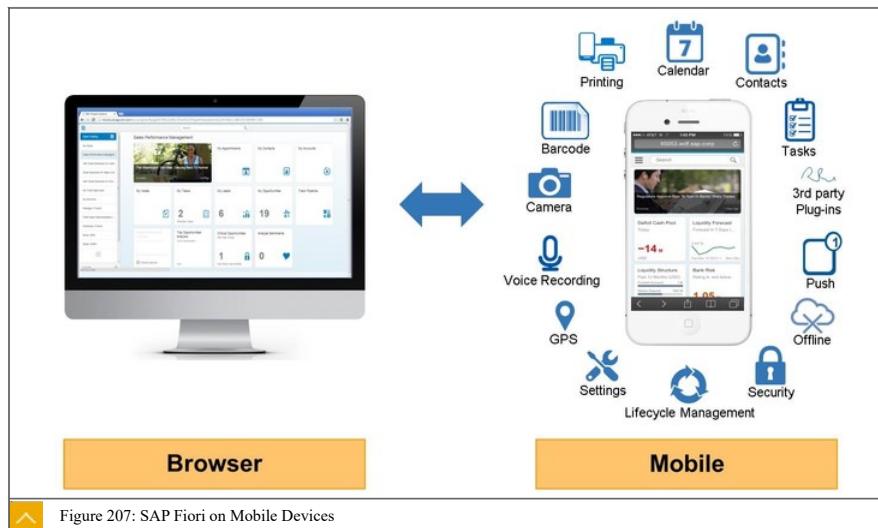
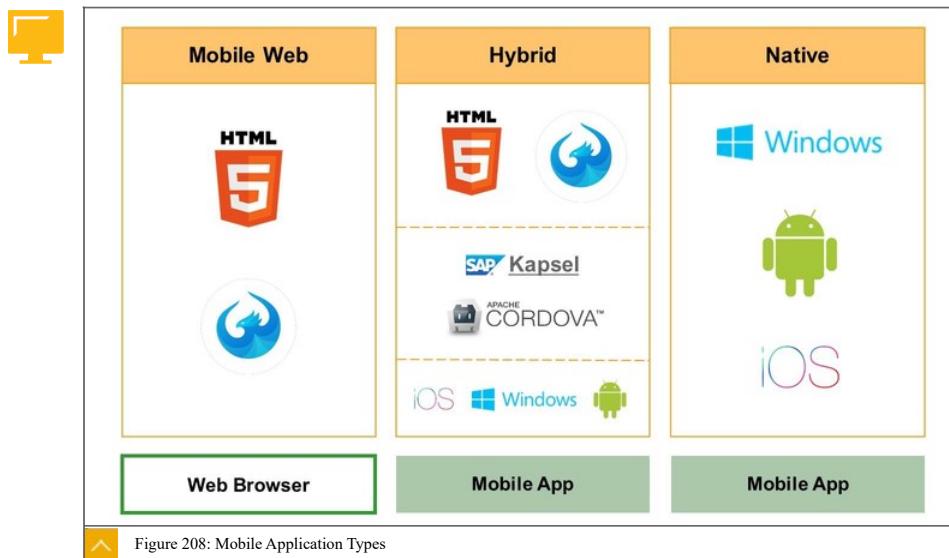


Figure 207: SAP Fiori on Mobile Devices

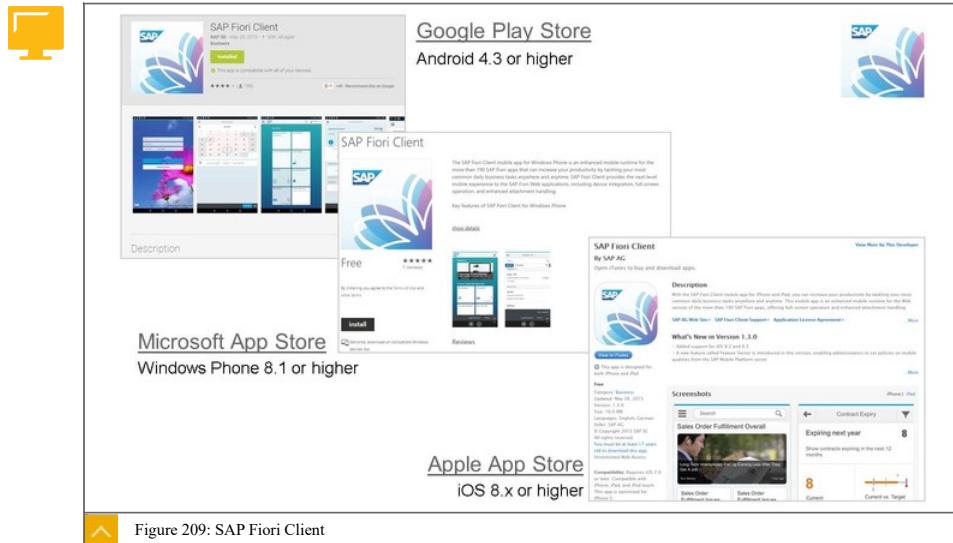
Using SAP Fiori in a mobile environment involves more than just opening the Fiori launchpad in the browser. A mobile device has plenty of features that benefit the functionality of apps. Users are accustomed to accessing contacts, calendars, and tasks or using geo-location, camera, and voice recording in mobile apps. In addition, topics like lifecycle management, security, or app settings can get complicated, and even dangerous, if they are not handled properly.



Before going through the details of SAP Fiori Mobile, it is important to distinguish the three general mobile application types. Mobile web is an app developed using HTML5 to adapt to the browser environment used for running the app. Beside the development of the app, no further tasks are needed. Mobile web browsers should be able to display a mobile web app. SAP Fiori can be used out-of-the-box as a mobile web app.

Hybrid apps are mobile web apps running in a native container app. Such a container app provides browser technology to run the app enhanced by additional features such as better integration in the native runtime environment of the mobile device. A well-known container app is Apache Cordova, which SAP uses as a foundation of its own container app SAP Kapsel, which is specialized for SAPUI5 mobile web apps.

Lastly, native apps are apps developed for a certain runtime environment using its native development environment. They can only run in this one environment, for example, Microsoft Windows Phone, Google Android, or Apple iOS. Their lack of portability is a weakness compared to mobile web apps, but their strength lies in the performance and the possibility for deep integration in the native environment using features not available with web technologies.



The SAP Fiori Client is a runtime container for SAP Fiori launchpad based on SAP Kapsel. It provides HTML5 and CSS 3.0 browser support for iOS, Android, and Windows Phone. It is completely free, so you can just go to the store of your native environment and install it. The central setting is the URL of the Fiori launchpad that should be displayed.

Note the following information:

Google Play Store:

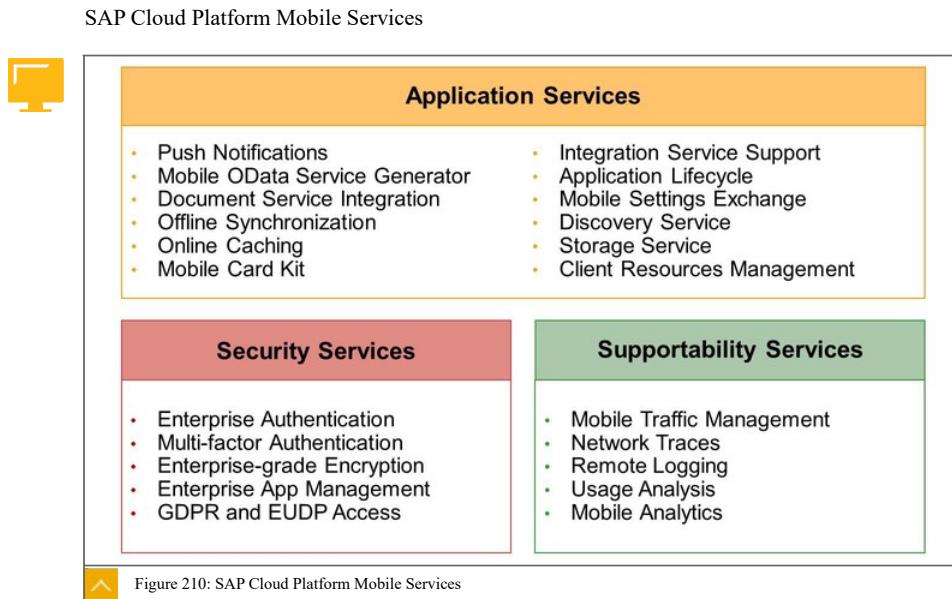
<https://play.google.com/store/apps/details?id=com.sap.fiori.client>

Microsoft App Store:

<http://apps.microsoft.com/windows/de-de/app/sap-fiori-client/1e835454-bb59-4078-ad31-f3e4719baba9>

Apple App Store:

<https://itunes.apple.com/de/app/sap-fiori-client/id824997258?mt=8>



The SAP Cloud Platform (CP) Mobile Services serves as a middleware to enhance communication for mobile applications and provide additional features in apps. Such a middleware is called a Mobile Application Development Platform (MADP), which means that developing and providing apps is a major part. The main app offered and enhanced for SAP Fiori is the SAP Fiori Client.

The service areas of SAP CP Mobile Services distinguish between services for security, supportability, and applications.

Application

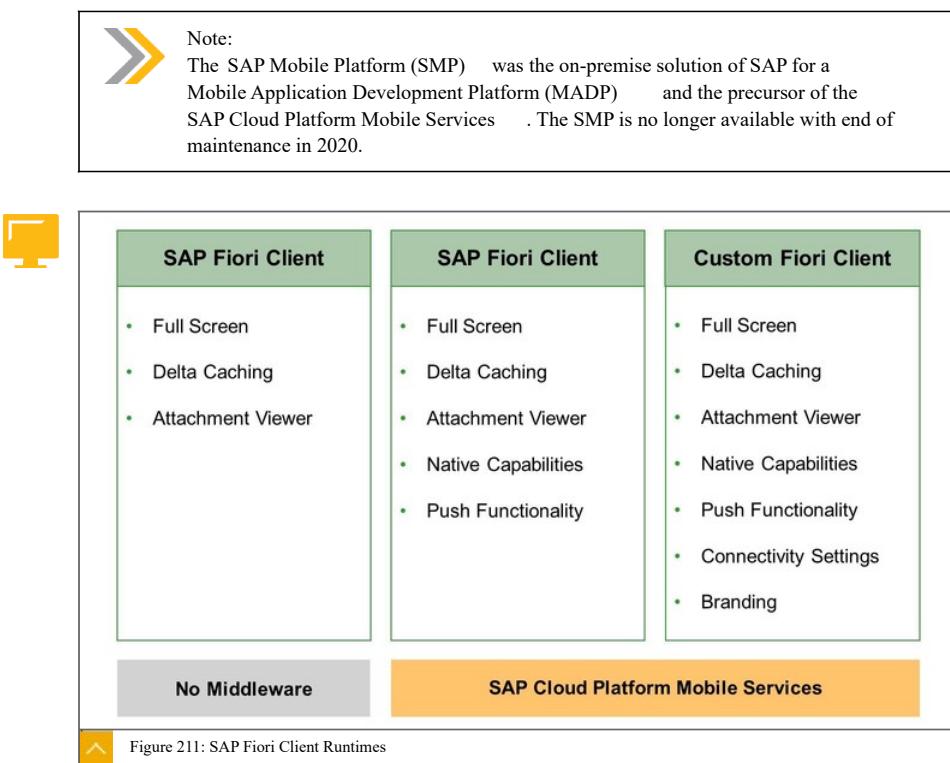
For SAP Fiori applications, push notifications, offline synchronization, and online caching are major benefits when using on mobile devices. The SAP Mobile Cards are an app type of their own, which can be combined with the SAP Fiori launchpad. All other application services come in handy for certain apps or scenarios.

Security

In the security area, the SAP Fiori Client benefits mainly of the various authentication possibilities and enterprise-grade encryption. For GDPR (General Data Protection Regulation) and EUDP (European Data Protection Board) access, the back-end server is the central component for SAP Fiori.

Supportability

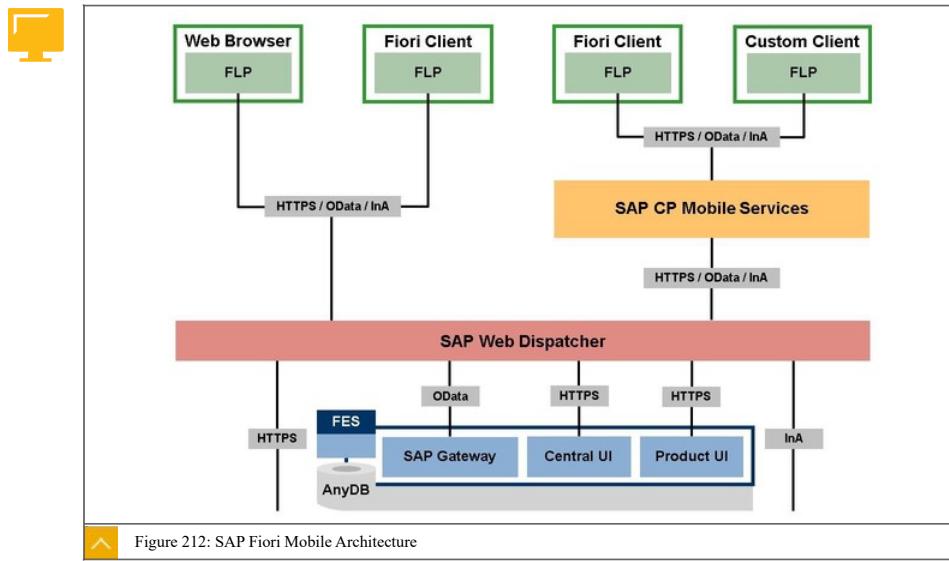
In the supportability area, the mobile traffic management and network traces are important for a reliable operation of Fiori applications. The analytical features are useful on a more generic level.



By accessing the Fiori launchpad using the Fiori client, users benefit from a full-screen display of the FLP, the support of app-specific, on-device, persistent caching strategy, and an attachment and document viewer for PDF, Microsoft Word, PNG, and so on. The viewer is launched in full-screen mode on top of the browser frame and can be closed manually.

When connecting the Fiori Client to SAP CP Mobile Services, logon can be handled more easily using SSO (Single Sign-On). Native device capabilities are supported (camera, geo-location, barcode, printing, contacts, calendar) as well as third-party certificate providers and mobile device management (MDM). In addition, some Fiori apps can now harness push and synchronization functionality of the SAP CP Mobile Services.

It is also possible to adapt the Fiori Client to the needs of a customer. Using the native development environment of the mobile device, customers can extend the Fiori Client implementing a certain branding like a corporate design as well as predefined connectivity settings. That also allows the usage of multiple Fiori Clients on one device. To provide such a custom Fiori Client and the features inside to the users device, application services of the SAP CP Mobile Services are needed.



The system landscape of the Fiori mobile architecture offers four mobile use cases for apps:

1. The web browser on the mobile device accesses the FLP via the SAP Web Dispatcher like a desktop browser.
2. The Fiori Client on the mobile device accesses the FLP directly via the SAP Web Dispatcher.
3. The Fiori Client on the mobile device accesses the SAP CP Mobile Services, which forwards the request to the FLP or provides a service directly.
4. A custom Fiori Client on the mobile device accesses the SAP CP Mobile Services, which forwards the request to the FLP or provides a service directly.

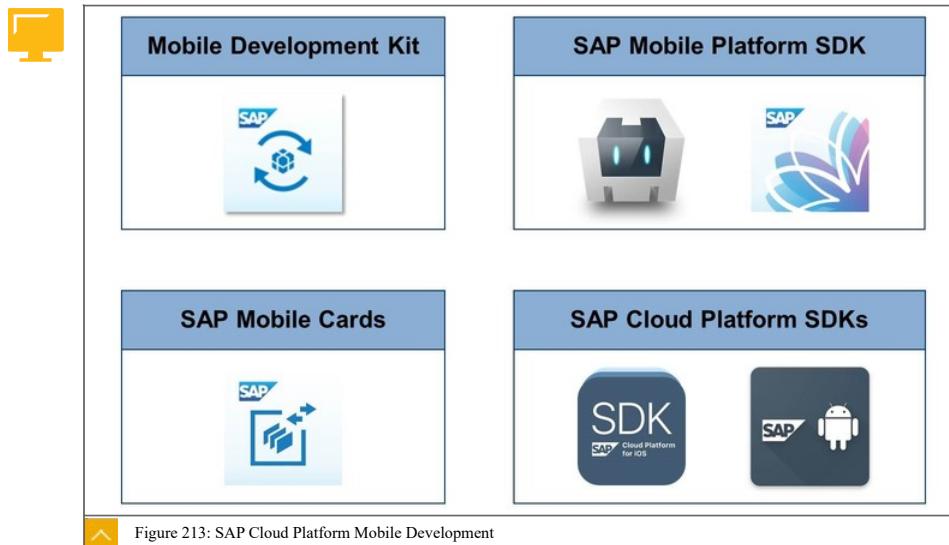


Figure 213: SAP Cloud Platform Mobile Development

The SAP CP Mobile Services offers four areas for developing mobile apps. Most of them are somehow connected to SAP Fiori:

Mobile Development Kit

The Mobile Development Kit enables developers to quickly build mobile apps with a native look and feel in a visual and declarative way for Apple iOS and Google Android. It is a metadata-driven model for mobile app development integrated in the SAP Web IDE. It offers a code-free tool and a WYSIWYG editor as well as app templates including support for offline, push, geo-location and more.

SAP Mobile Platform SDK

The SAP Mobile Platform SDK aims for developing hybrid apps combining the benefits of web and native apps. It is based on Apache Cordova and integrated in the SAP Web IDE including a WYSIWIG editor. The SAP Fiori Client is the specialized web container based on Apache Cordova for an SAP Fiori launchpad.

SAP Mobile Cards

The SAP Mobile Cards are consumer grade "wallet" or "passbook" style apps. These are very simple apps like workflow approvals or bits of valuable data. Cards can be stored offline, auto refreshed, subscribed to, and maintained as a single business content repository on Apple iOS, Google Android, and Tencent WeChat devices.

SAP Cloud Platform SDKs

The SAP Cloud Platform SDKs offer an efficient way to develop native SAP Fiori apps for Apple iOS and Google Android. For both platforms, design guides and tools integrated in the native development environments Apple Xcode and Android Studio are available. High performance, the full set of native features, and the native look and feel of each platform are some of the major benefits.



LESSON SUMMARY

You should now be able to:

- Examine SAP Fiori Mobile

Unit 6

Lesson 2

Examining SAP Fiori for iOS



LESSON OBJECTIVES

After completing this lesson, you will be able to:

- Examine SAP Fiori for iOS

Native SAP Fiori Apps

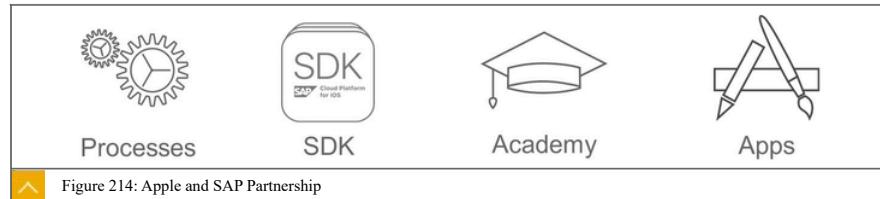


Figure 214: Apple and SAP Partnership

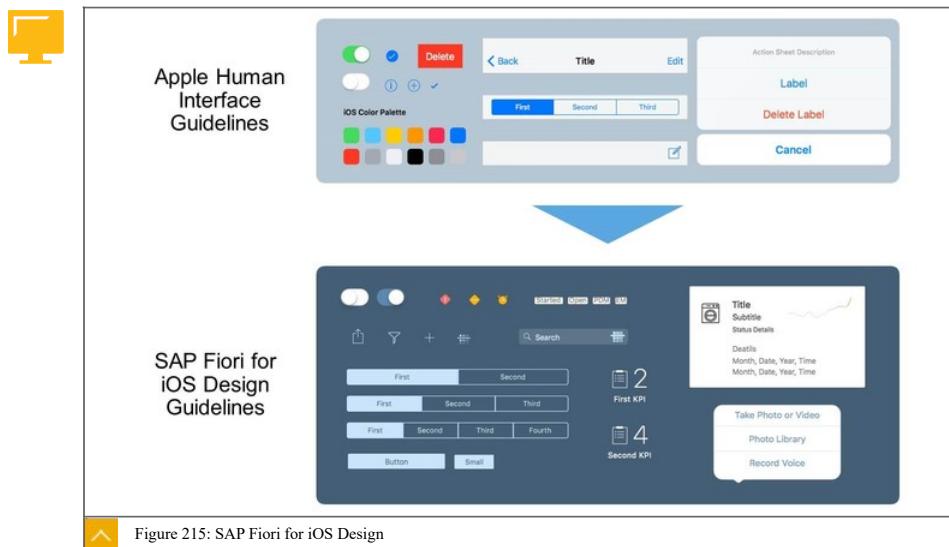
The Apple and SAP partnership for SAP Fiori for iOS consists of four key areas that are focused on helping customers accelerate their mobilization of business-critical workflows, changing how people do their jobs, and increasing productivity and efficiency. This involves the following:

- Redesign of business-critical workflows and processes to have mobility at their core
- Delivery of tools to accelerate the development of transformative apps
- Creation of a training academy to educate the SAP developer community on the new tools and iOS
- Co-development of user-centric apps for industry to extend the reimaged workflows



Note:

The Google and SAP partnership for SAP Fiori for Android only consists of design guidelines and SDK (Software Development Kit).

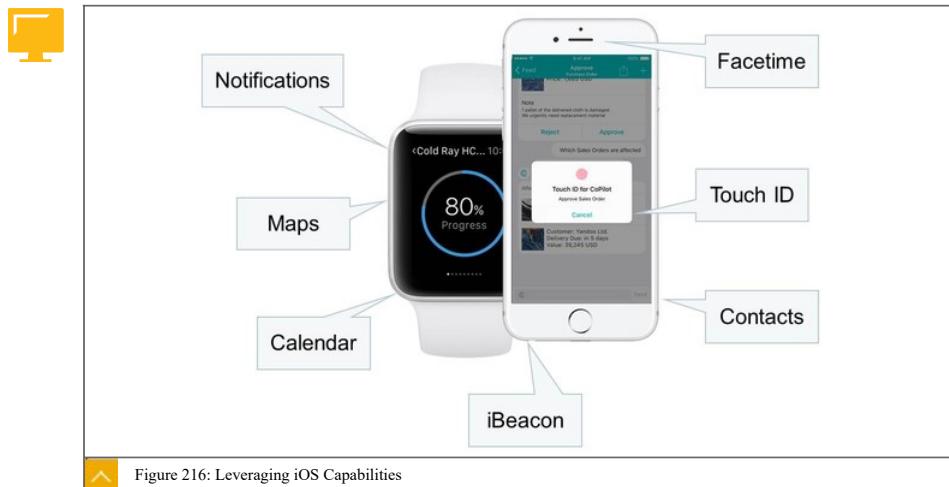


The SAP Fiori for iOS design guidelines are an enterprise-grade design for the iOS interface based on the Apple Human Interface Guidelines (HIG). The basic patterns and controls defined in HIG are enhanced with new patterns and frameworks, as well as new reusable UI elements and floorplans. The complete design guidelines are available at <https://experience.sap.com/fiori-design-ios/>.



Note:

The design guidelines for Android are available at <https://experience.sap.com/fiori-design-android/>.



One major benefit of native mobile apps is the possible leveraging of all native capabilities:

Touch ID

Provides secure biometric login to enterprise apps or to legitimize actions like payments or approvals

iBeacon

Provides location-specific information like info on assets close-by or warnings

Maps

Presents location and destination guidance

Notifications

Pushes important information such as schedule changes, updates, or warnings to end users immediately

Facetime

Allows ad-hoc communication, for example, between maintenance technicians and technical back-office experts

Calendar

Create or change meeting request directly in app

Contacts

Display linked contacts to data sets and start communication



Note:

There are comparable native features available for SAP Fiori for Android .



SAP Project Companion for Consultants

By SAP SE

This app is only available on the App Store for iOS devices.



Description

With the SAP Project Companion for Consultants mobile app for iPhone, you can easily get on top of day-to-day activities in your projects on your customer projects anywhere and anytime. This app allows contributors to facilitate collaboration with project managers and team members, stay on track with your actions and topics, and help the team to focus on priority issues relevant to the project.

[SAP SE Web Site](#) • [SAP Project Companion for Consultants Support](#) • [Application license Agreement](#) • [More](#)

What's New in Version 1.1.2

With the latest version of SAP Project Companion for Consultants, contributors can now delegate work packages to managers in work packages delegated to them. By taking responsibility for a work package, managers can contribute to project management and execution.

iPhone Screenshots



SAP Project Companion for Managers

Apple App Store

iOS 10.x or higher

View More by This Developer

What's New in Version 1.1.2

With the latest version of SAP Project Companion for Managers, you can delegate your work packages to team members to reduce your load and accelerate project execution.

[SAP SE Web Site](#) • [SAP Project Companion for Managers Support](#) • [Application license Agreement](#) • [More](#)

iPhone Screenshots



Apple App Store

iOS 10.x or higher

Figure 217: SAP Project Companion

The first enterprise app for iOS is the SAP Project Companion , which is available in one flavor for consultants and managers each. It digitizes the processes that a project manager needs to

deal with; provides an overview of the key data of the project; displays notes and relevant information on tasks; reaches the team members, boosting project-related communication; and decreases the time unplanned distractions can cause.

It is completely integrated with SAP Cloud Platform (CP) and leverages Touch ID to access the mobile and security services in SAP CP.

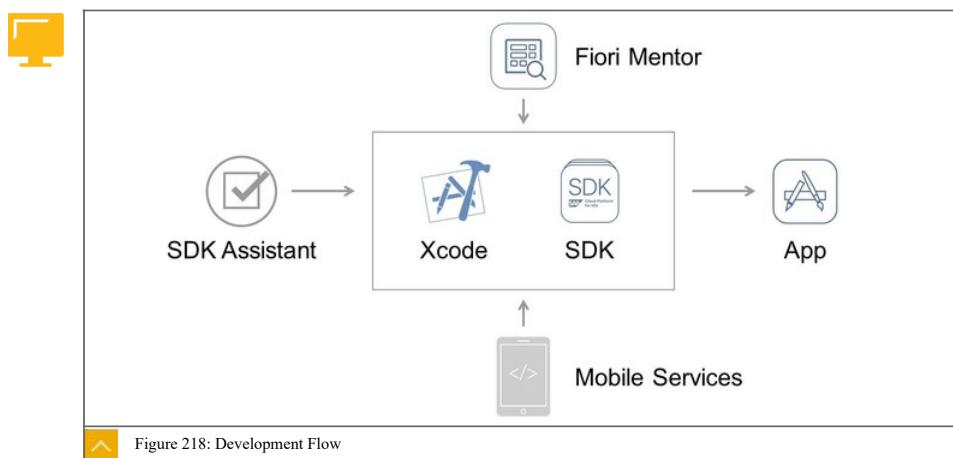
SAP Project Companion for Consultants in Apple App Store:

<https://itunes.apple.com/de/app/sap-project-companion-for-consultants/id1183388005?mt=8>

SAP Project Companion for Managers in Apple App Store:

<https://itunes.apple.com/de/app/sap-project-companion-for-managers/id1183390731?mt=8>

Native SAP Fiori Development



Building apps for SAP Fiori for iOS starts with the SAP Cloud Platform SDK Assistant, which generates projects and code to ease integration into the SAP Cloud Platform (CP). These projects are used by Xcode and the SAP Cloud Platform SDK for iOS to create native apps. The SDK is a fully Swift-based API accelerating enterprise app development in Apple Xcode.

Developers can use the SAP Fiori Mentor app for iPad to select and configure reusable Fiori UI library graphics elements in order to create the app UI and generate the code used to create those components. They can also use the power of SAP Cloud Platform Mobile Services to accelerate developing analytics, push notifications, do online and offline syncing, and more.

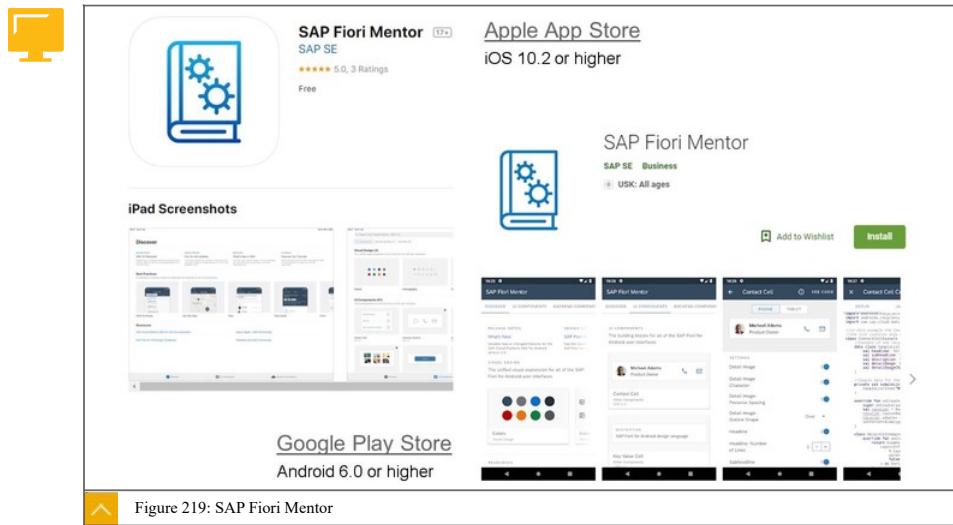


Figure 219: SAP Fiori Mentor

The SAP Fiori Mentor app is the central access point for developers to get information about the SAP partnership with Apple and Google, design guidelines, SDK documentation, and the SAP development environment. The goals are as follows:

- Help the developer get started quickly.
- Access all available online resources.
- Preview all available UI and foundation components.
- Explore sample Fiori component styles and explicitly configure all possible variations.
- Generate ready-made code snippets incorporating selected settings.

SAP Fiori Mentor in Apple App Store:

<https://itunes.apple.com/de/app/sap-fiori-mentor/id1215284965?mt=8>

SAP Fiori Mentor in Google Play Store:

<https://play.google.com/store/apps/details?id=com.sap.mobile.mentor.android>

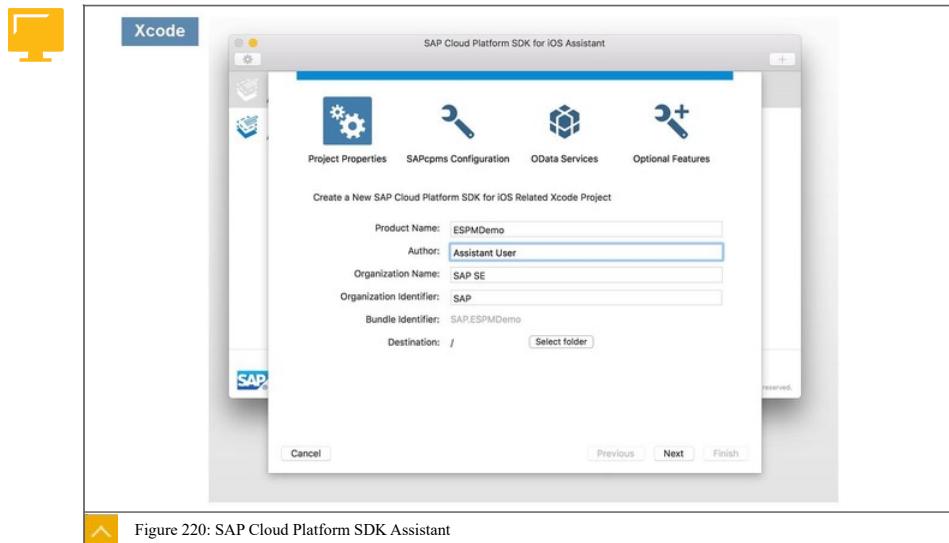


Figure 220: SAP Cloud Platform SDK Assistant

The SAP Cloud Platform SDK Assistant is a plugin for Apple Xcode and creates a ready to run Xcode project for iPhone/iPad. It generates object-oriented Swift proxy classes for OData services, eliminates exposure to low-level APIs, and generates SAP CP Mobile Service configuration.



Note:

The pendant for the Google Android Studio is called SAP Cloud Platform SDK Wizard.

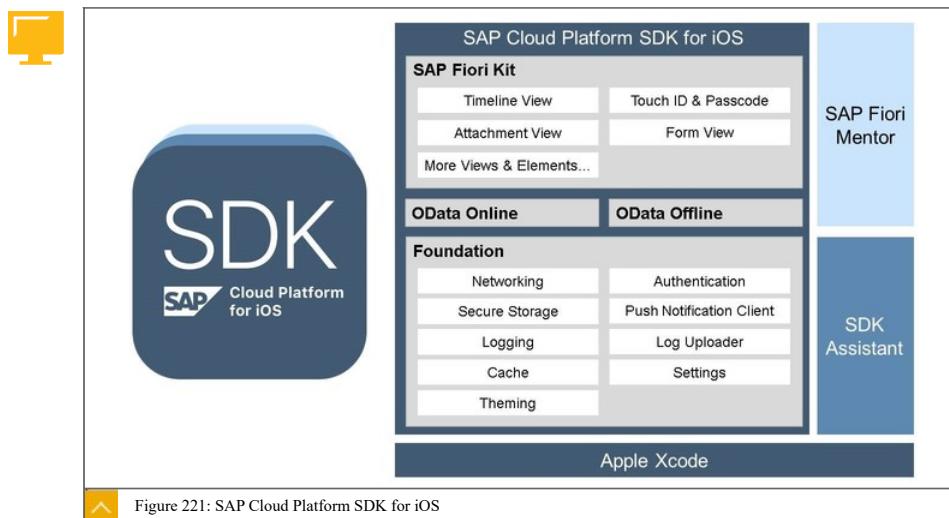
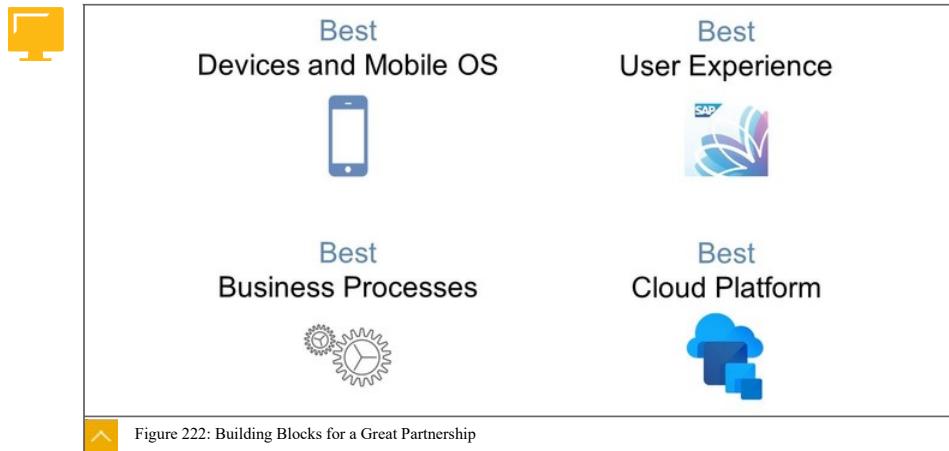


Figure 221: SAP Cloud Platform SDK for iOS

The SAP Cloud Platform SDK is the core in terms of developing SAP Fiori for iOS apps. It is a native UI layer that implements the Fiori design language with UI reuse components. It rapidly generates scaffolding for enterprise apps and utilizes mobile services API for online and offline data access, security, and more. It is also an integral part of SAP CP, providing governance, security, supportability, and integration services.



The building blocks for the great partnership SAP Fiori for iOS are that the best devices and mobile OS are combined with the best business processes, which are combined with the best user experience, which is combined with the best cloud platform:

Best Devices and Mobile OS

- Integrated hardware and software, designed and built by Apple
- Powerful, streamlined deployment and management capabilities
- Built-in security and privacy, protecting the user without compromising their experience

Best Business Processes

- Leveraging large customer base and expertise in 25 industries and 11 lines of business
- Total end-to-end process perspective, from mobile workers through to the back end
- Providing new, innovative capabilities such as SAP S/4HANA to enable business transformation

Best User Experience

- Apple iOS with excellent user satisfaction ratings and many differentiating capabilities
- Apple and SAP collaborating closely to create tools and methodologies for user-centric development
- SAP Fiori for iOS for an intuitive experience with enterprise grade capabilities

Best Cloud Platform

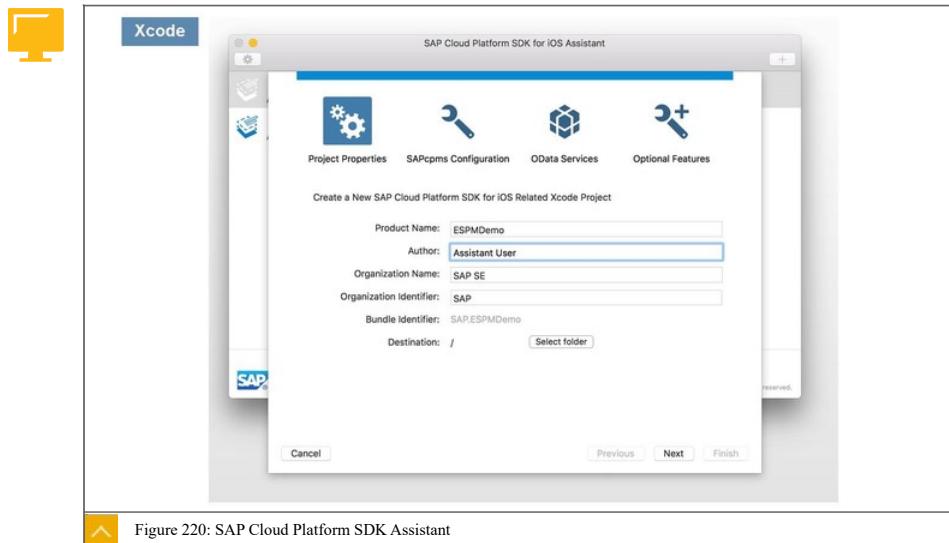


Figure 220: SAP Cloud Platform SDK Assistant

The SAP Cloud Platform SDK Assistant is a plugin for Apple Xcode and creates a ready to run Xcode project for iPhone/iPad. It generates object-oriented Swift proxy classes for OData services, eliminates exposure to low-level APIs, and generates SAP CP Mobile Service configuration.



Note:

The pendant for the Google Android Studio is called SAP Cloud Platform SDK Wizard.

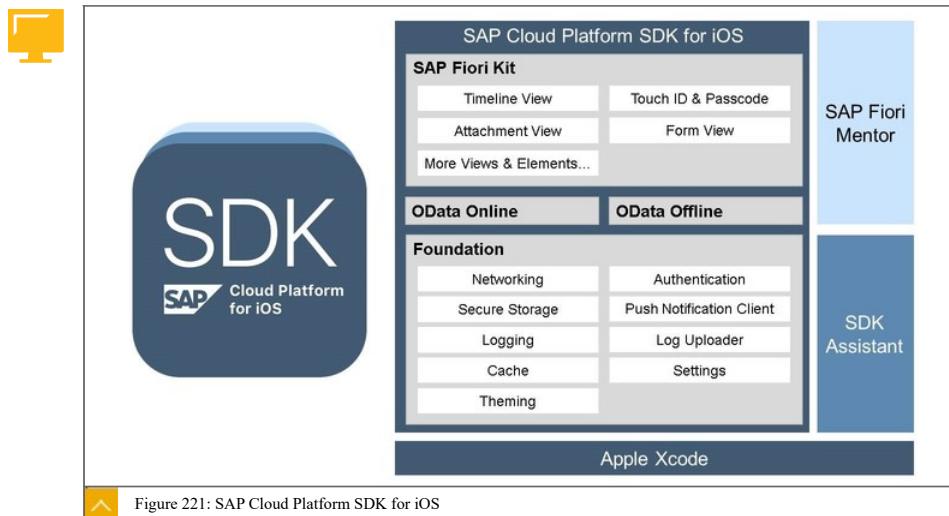
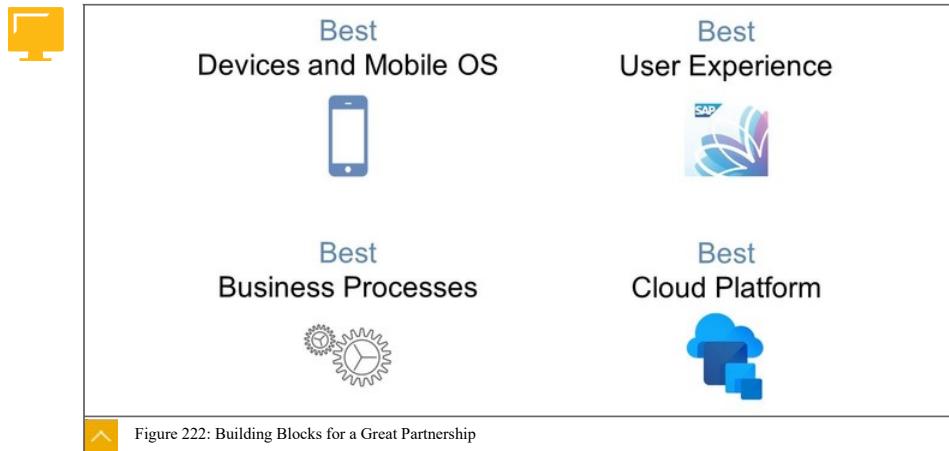


Figure 221: SAP Cloud Platform SDK for iOS

The SAP Cloud Platform SDK is the core in terms of developing SAP Fiori for iOS apps. It is a native UI layer that implements the Fiori design language with UI reuse components. It rapidly generates scaffolding for enterprise apps and utilizes mobile services API for online and offline data access, security, and more. It is also an integral part of SAP CP, providing governance, security, supportability, and integration services.



The building blocks for the great partnership SAP Fiori for iOS are that the best devices and mobile OS are combined with the best business processes, which are combined with the best user experience, which is combined with the best cloud platform:

Best Devices and Mobile OS

- Integrated hardware and software, designed and built by Apple
- Powerful, streamlined deployment and management capabilities
- Built-in security and privacy, protecting the user without compromising their experience

Best Business Processes

- Leveraging large customer base and expertise in 25 industries and 11 lines of business
- Total end-to-end process perspective, from mobile workers through to the back end
- Providing new, innovative capabilities such as SAP S/4HANA to enable business transformation

Best User Experience

- Apple iOS with excellent user satisfaction ratings and many differentiating capabilities
- Apple and SAP collaborating closely to create tools and methodologies for user-centric development
- SAP Fiori for iOS for an intuitive experience with enterprise grade capabilities

Best Cloud Platform

5. What is SAP Cloud Platform SDK Assistant?

Unit 6

Learning Assessment - Answers

1. Which benefits are offered by the SAP Fiori Client alone over a Web browser?

Choose the correct answers.

A Full screen

B Offline data

C Delta caching

D Push notifications

E Attachment viewer

Correct! The following benefits are offered by the SAP Fiori Client alone over a Web browser: Full Screen, Delta Caching, and Attachment Viewer.

2. What can be used as middleware to improve SAP Fiori for mobile?

Choose the correct answer.

A SAP Unwired Platform (SUP)

B SAP Mobile Platform (SMP)

C SAP Cloud Platform Mobile Services

Correct! To improve SAP Fiori for mobile SAP Cloud Platform Mobile Services can be used as middleware.

3. What are the SAP Fiori for iOS design guidelines based on?

[Apple Human Interface Guidelines \(HIG\)](#)

4. Which native capabilities of a mobile device can only be leveraged when using SAP Fiori for iOS?

Choose the correct answers.

A Touch ID

B Camera

C Facetime

D iBeacon

E Geolocation

Correct! The following native capabilities of a mobile device can only be leveraged when using SAP Fiori for iOS: Touch ID, Facetime, and iBeacon.

5. What is SAP Cloud Platform SDK Assistant?

[Plugin for Apple Xcode to create ready-to-run Xcode projects for SAP Fiori](#)

UNIT 7

Integration

Lesson 1

Integrating SAP Workflow in SAP Fiori

206

Lesson 2

Integrating SAP Fiori in SAP Enterprise Portal

212

Lesson 3

Exploring SAP Cloud Platform

219

UNIT OBJECTIVES

- Integrate SAP Workflow in SAP Fiori
- Integrate SAP Fiori in SAP Enterprise Portal
- Explore SAP Cloud Platform

Unit 7

Lesson 1

Integrating SAP Workflow in SAP Fiori



LESSON OBJECTIVES

After completing this lesson, you will be able to:

- Integrate SAP Workflow in SAP Fiori

SAP Workflow Apps

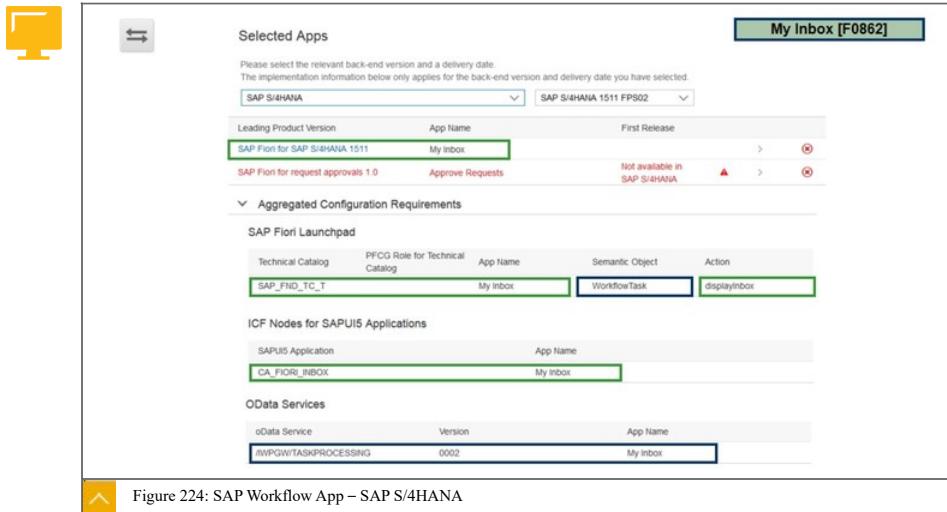


The screenshot shows the SAP Workflow Apps interface within SAP Business Suite. The top navigation bar includes 'Selected Apps', 'My Inbox [F0862]', and 'Approve Requests [F0392]'. The 'Selected Apps' section shows a table with 'SAP Business Suite' selected for delivery date 'Q2/2016'. The 'Fiori Launchpad' section displays two rows of app configurations: 'SAP Fiori for request approvals 1.0' (Approve Requests) and 'SAP Fiori for request approvals 2.0' (My Inbox). The 'ICF Nodes for SAPUI5 Applications' section shows 'CA_ALL_APV' (Approve Requests) and 'CA_FIORI_INBOX' (My Inbox). The 'OData Services' section lists two entries: 'IWPGW/TASKPROCESSING' (Approve Requests) and 'IWPGW/TASKPROCESSING' (My Inbox).

Figure 223: SAP Workflow Apps – SAP Business Suite

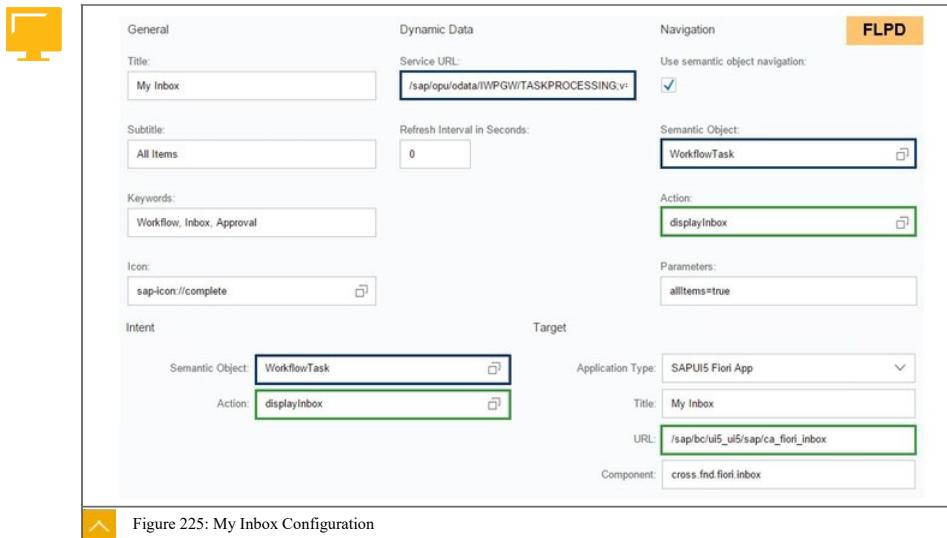
One of the first integrations of SAP Fiori was with SAP Workflow. There are plenty of apps available that are connected in some way to workflows running in an AS ABAP. These can range from a simple UI for approving steps of workflows to complex integrations including attachments. Aside from these apps that were developed for certain workflows delivered by SAP, there are also two generic apps available for custom workflows. The first one is Approve Requests, which was delivered as a principal app right from the beginning. However, instead of getting an up-port to ECC 6.07 like the other principal apps in 2015, it was replaced by the My Inbox app.

The My Inbox app offers more features such as the ability to also handle workflows of SAP Process Orchestration (PO), and uses an updated version of the central workflow Gateway service /IWPGW/TASKPROCESSING. This Gateway service was already providing SAP Workflow data before SAP Fiori, and can be installed using the IW_PGW add-on. During 2015, both apps were offered in parallel, but, currently, there is no further development ongoing for Approve Requests.



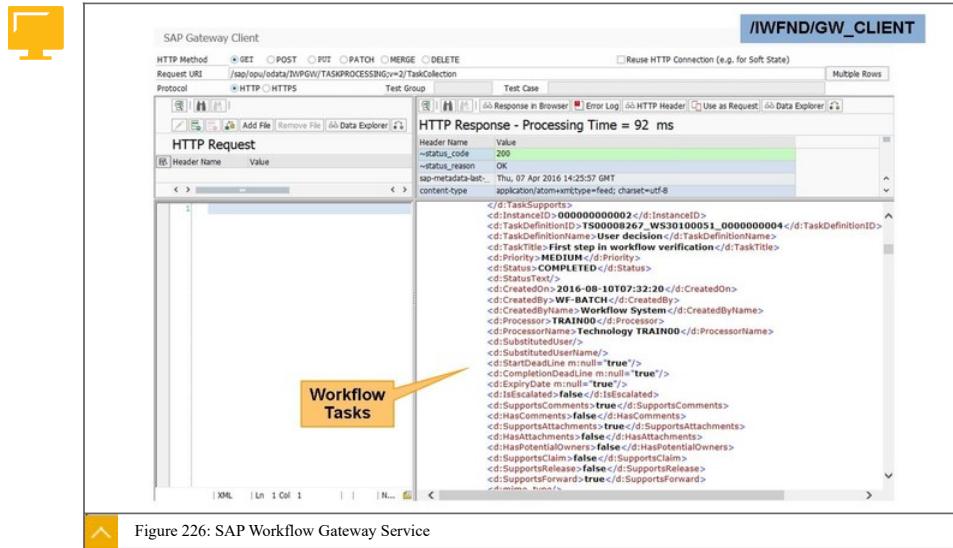
In SAP S/4HANA, only the **My Inbox** app is available as a generic workflow app. The technical foundation is the same as for SAP Business Suite, except that it is an integral part of S/4HANA instead of an add-on like before. The same applies for the **Gateway** workflow service, which is now also part of the software component **SAP Fiori Front-End Server 2.0**, like the **My Inbox** app.

My Inbox

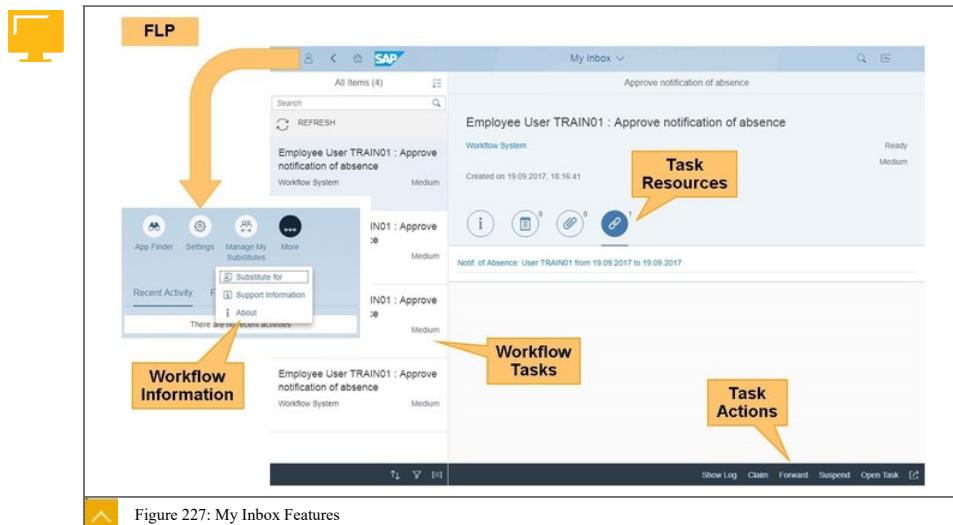


In AS ABAP 7.40 and 7.50, the **My Inbox** app is only part of the **SAP_FND_TC_T** technical catalog. Since AS ABAP 7.51, it is part of the **SAP_TC_CA_MYINBOX** technical catalog and the **SAP_CA_BC_MYINBOX** business catalog contained in the **SAP_BR_MANAGER** business role. If customers want to configure it to a certain workflow, it is recommended to reference it in a

new business catalog. If the configuration is not adapted to a certain workflow, the app shows all the open workflow tasks of the user running the app. To reduce the number of displayed workflow tasks, workflow scenarios can be defined in SAP Workflow to combine tasks of one or several connected workflows.



In contrast to other Gateway services, the implementation of the Gateway workflow service is part of the FES. The reason is that by running the OData logic on the FES, back ends just need to provide the workflow data via RFC using function modules. It is not necessary to update the system providing workflows, which greatly increases the possible sources for workflows. The Gateway workflow service takes all information about workflow tasks and provides it to the My Inbox app or any other OData client.



The My Inbox app is built using the master-detail design. On the left, the user sees a list of all their tasks, and, in the main area, they see the details of the task selected. The app is capable of displaying all workflow tasks of a user with the role of the **Universal Work List (UWL)** for Fiori.

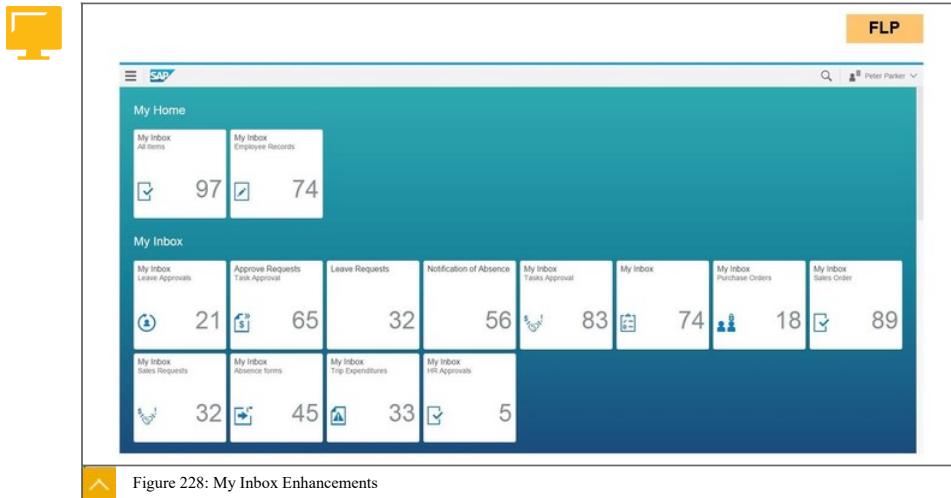


Figure 228: My Inbox Enhancements

By using SAPUI5 extensions, it is possible to enhance the features of the app, such as showing document attachments of the workflow task or adapting the buttons at the bottom, to reflect the possible decisions of the workflow.

There are several **Business Add-ins (BAdI)** available in ABAP to implement the logic behind SAPUI5 extensions like the following:

/IWWRK/ES_WF_WI_BEFORE_UPD_IB

Map decision to outcome container value

/IWPGW/ES_TGW_TASK_QUERY

Change subject in query result

/IWPGW/ES_TGW_TASK_DATA

Modify task title and task description

/IWPGW/ES_TGW_USER_DETAIL

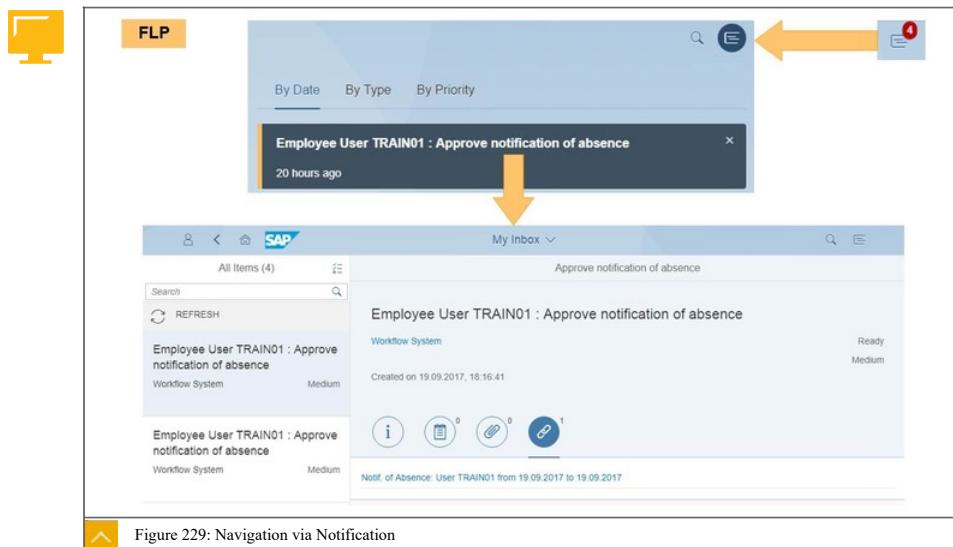
Modify user details

Note:

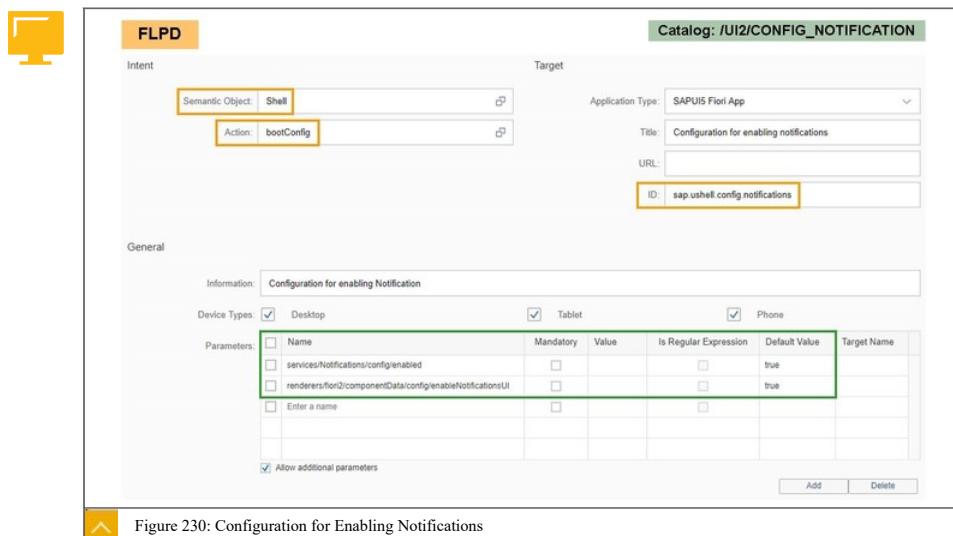
More information about this topic can be found in BIT602 (SAP Workflow with SAP Fiori):

<https://training.sap.com/course/bit602>

Push Notifications



In SAP Fiori 2.0, workflow tasks can be shown as notifications in the Notifications Center. Users who are allowed to receive notifications have an additional button in the upper right, which shows a number of unread notifications. Clicking on a notification in the Notifications Center starts a navigation to the mapped application. For SAP Workflow tasks, this opens the My Inbox app by default.



To activate the Notifications Center in the FLP, the user needs a corresponding target mapping for notifications available in the /UI2/CONFIG_NOTIFICATION catalog. In addition, the BES needs to be configured to send notifications for workflow tasks, and the FES needs to be configured to receive notifications from the BES. This has to be done once by an administrator for both systems to activate the notification channel in general. The required

steps for this can be found in the SAP Reference Implementation Guide (transaction SPRO) at SAP NetWeaver → Notification Channel .



LESSON SUMMARY

You should now be able to:

- Integrate SAP Workflow in SAP Fiori

Unit 7

Lesson 2

Integrating SAP Fiori in SAP Enterprise Portal

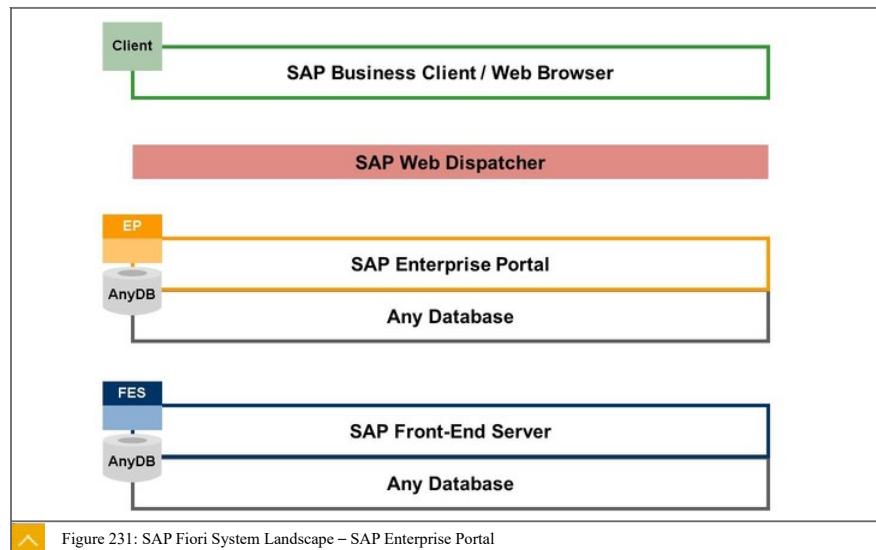


LESSON OBJECTIVES

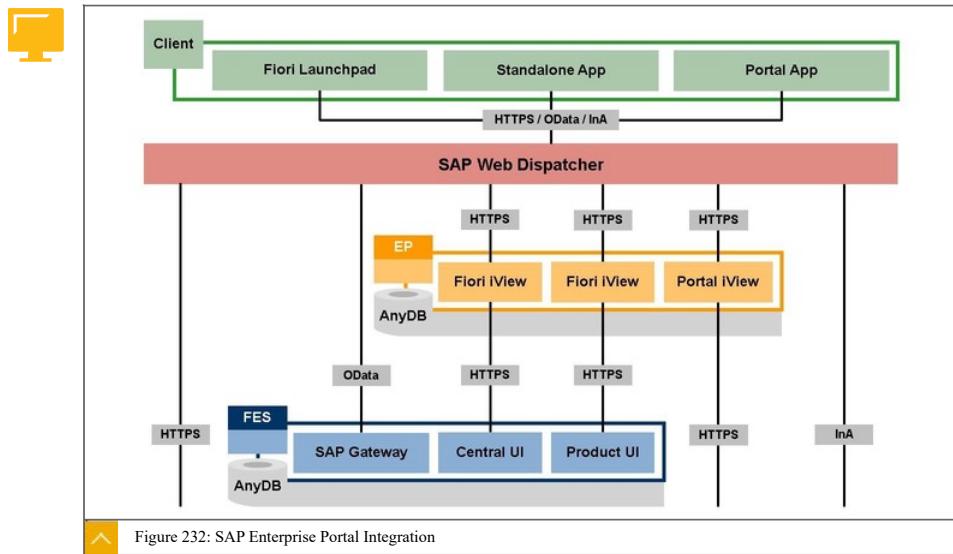
After completing this lesson, you will be able to:

- Integrate SAP Fiori in SAP Enterprise Portal

Landscape for SAP Fiori with SAP Enterprise Portal

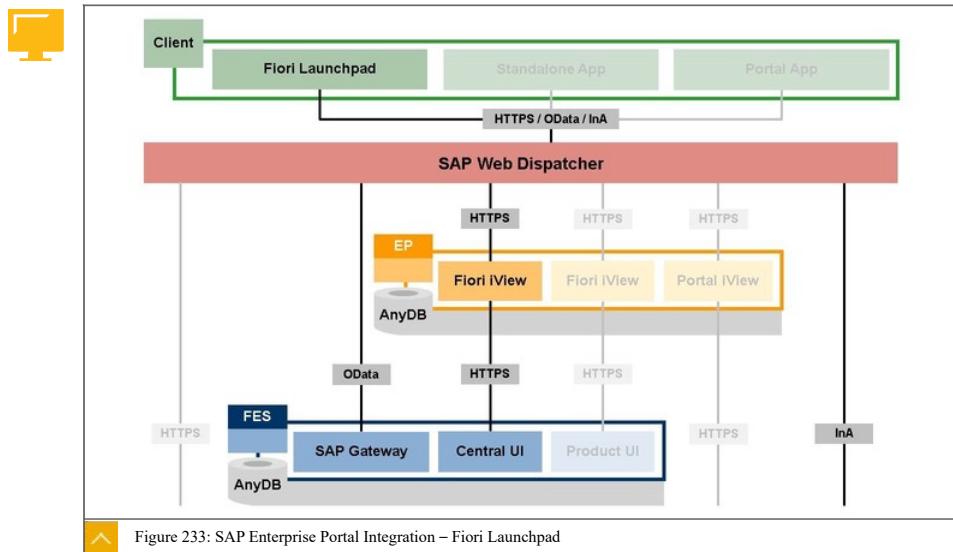


The SAP Fiori launchpad can be embedded in the SAP Enterprise Portal (EP). From a system landscape point of view, the Enterprise Portal sits in between the FES and the SAP Web Dispatcher. Clients for the EP are the SAP Business Client (BC) or a web browser. For the purposes of simplification, the BES is omitted in the figures included, but it is still the source of all data for Fiori apps.



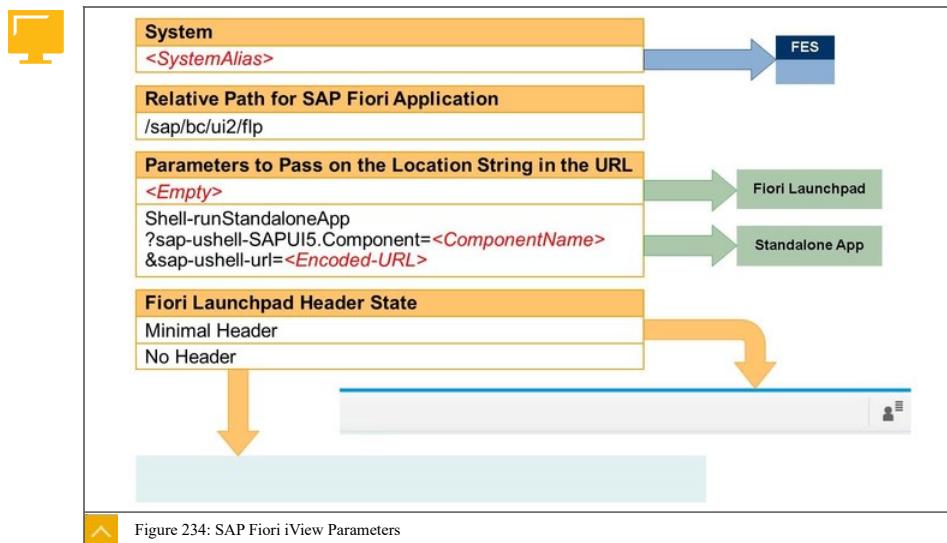
Apps running in the client can be distinguished as follows: the Fiori launchpad, a Fiori standalone app, or any other portal app. The EP concentrates on HTTPS communication around web pages. It does nothing with OData and InA requests. These are still routed to the FES and BES by the Web Dispatcher. The FLP and Fiori standalone apps can be embedded in the EP by using Fiori iViews.

SAP Fiori iView Mode



When using the FLP in the client, OData and InA requests are routed to the FES and BES but the UI is embedded in a Fiori iView. The iView contains parameters to access the central UI components of the FES. These parameters are as follows:

- System
- Relative Path for SAP Fiori Application
- Parameters to Pass on the Location String in the URL
- Fiori Launchpad Header State

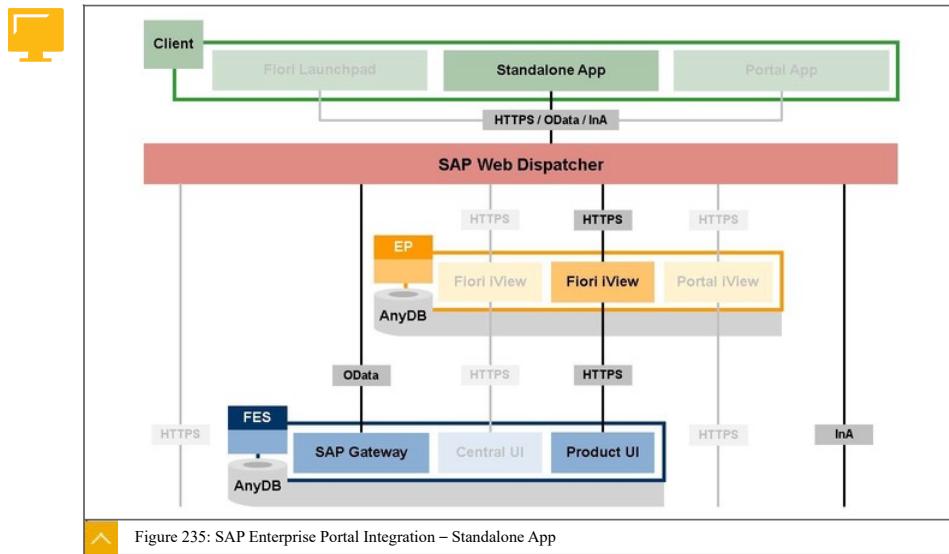


The System parameter contains the system alias pointing to the FES. The Relative Path for SAP Fiori Application contains the path to the FLP.

If the Parameters to Pass on the Location String in the URL are empty, the FLP is directly embedded in the iView, showing the tiles configured on the FES for the user. By creating a target mapping following the rules for Fiori standalone, a Fiori app can be called directly irrespective of whether the user has configured it on the FES.

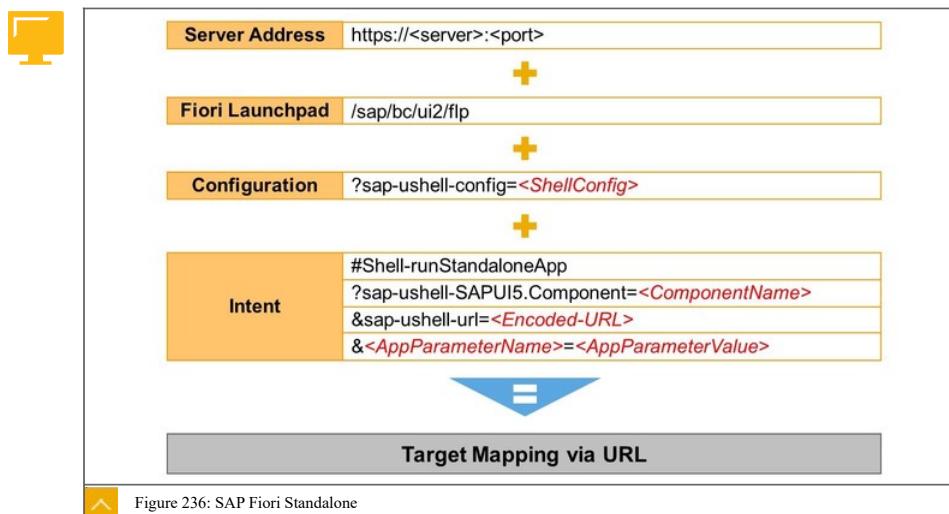
The Fiori Launchpad Header State can be set to **Minimal Header**, which just shows a button with user information, or **No Header**, which shows the Fiori app without any FLP frame.

SAP Fiori Standalone Mode



It is possible to embed a single Fiori app in the EP without starting the FLP first and especially without any customizing or configuration in the FLPD. The EP takes over the role of the FLPD to provide a target mapping for starting a Fiori app. This target mapping is created in the URL.

Note:
Only applications built without an app descriptor file (manifest.json) are supporting the standalone mode.

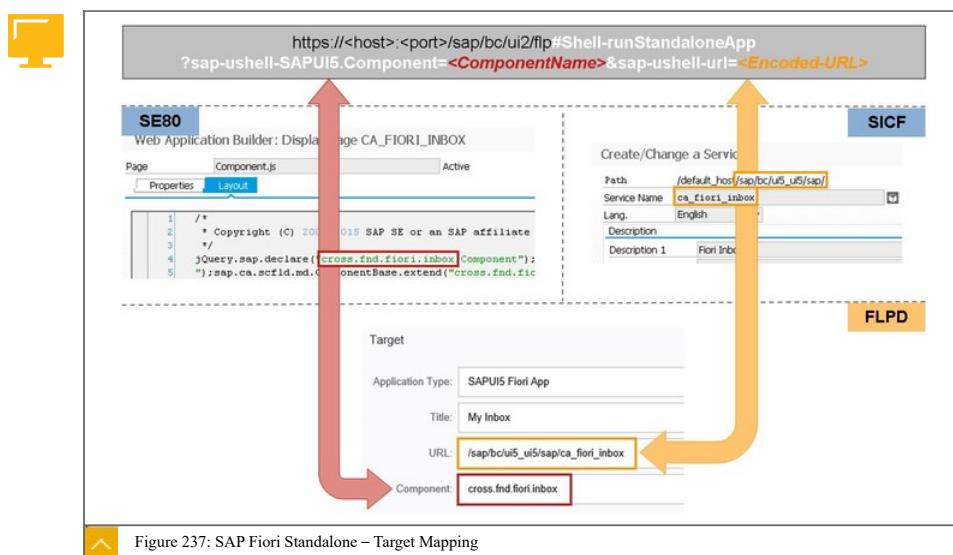


Calling a Fiori app in standalone mode can be done by building a URL containing all information that is normally defined as a target mapping in a Fiori catalog, that is, the SAPUI5 component, the ICF node, and optional parameters of the app on the FES. These are set as URL parameters of the `Shell-runStandaloneApp` option separated by a # from the other URL parts. Any additional configurations for the app like ABAP client or shell configuration are added in front of the #.



Note:

Specifying the path to the application component as a URL parameter does not allow you to use the proper cache busting token for the application resources. Thus, updating the application on the server may require clients to update their caches.

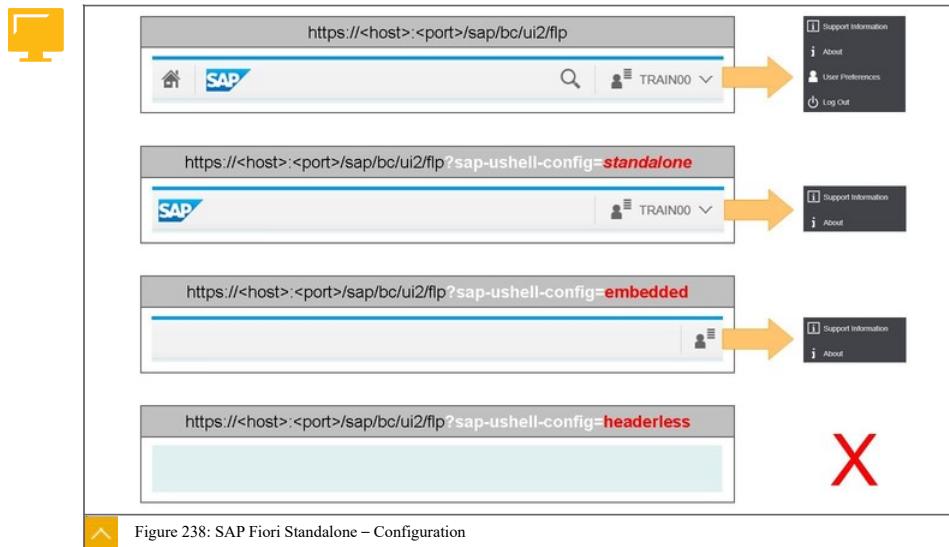


The elements of the target mapping can be taken from the system getting the path for the SAPUI5 component from the BSP application using SE80 and the path of the ICF node from the app using SICF. If the app is delivered by SAP, another option is to search for the suitable target mapping in the Fiori catalog defined in the Fiori app reference library .



Note:

Depending on the browser, certain characters in the URL such as the / have to be encoded to work properly.



When calling a Fiori app standalone, it is possible to configure the appearance and functions of the FLP header bar by using the `sap-ushell-config` URL option. Without any configuration, the full FLP header is shown without any restrictions. Note the following:

standalone

The home and search buttons disappear, as well as the user preferences and log out entries in the user menu.

embedded

The logo and the user name disappear, but the user menu still has the same options as for standalone. Embedded matches the `Minimal Header` value for the Fiori Launchpad Header State parameter of a Fiori iView.

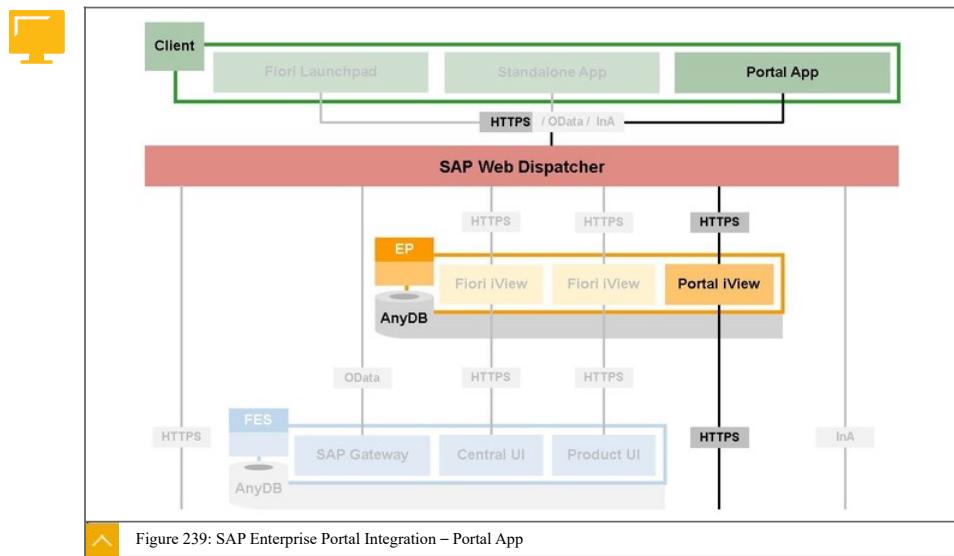
headerless

The header bar disappears completely showing only the app. Headerless matches the `No Header` value for the Fiori Launchpad Header State parameter of a Fiori iView.



Note:
With headerless, all functions of the app integrated in the user menu are also gone.

SAP Enterprise Portal iView



Another added value of using an SAP Enterprise Portal combined with SAP Fiori is that all other portal iViews can be combined with Fiori. Particularly if there is already an EP available in the customer landscape, all configured iViews and other technologies can be combined with Fiori. In addition, Web Dynpro and ABAP transactions can be made available for users using portal configurations instead of Fiori legacy apps.

A list of SAP Fiori apps released by SAP to run in standalone and/or in Portal iView mode is available in the SAP Note 2103156 – LAUNCHING FIORI APPS IN STANDALONE MODE.



LESSON SUMMARY

You should now be able to:

- Integrate SAP Fiori in SAP Enterprise Portal

Unit 7

Lesson 3

Exploring SAP Cloud Platform

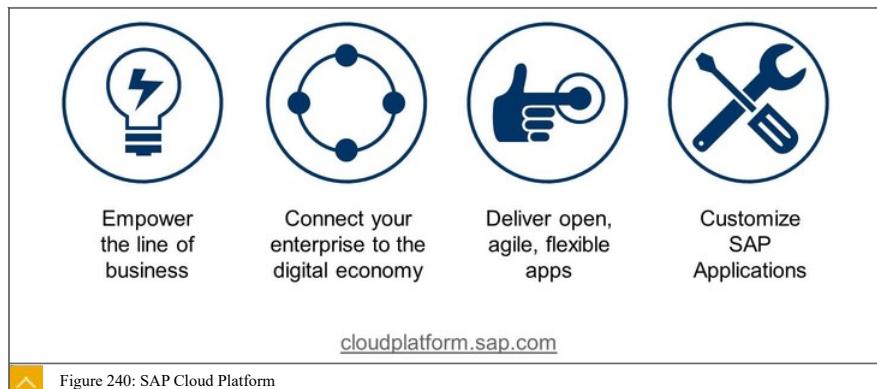


LESSON OBJECTIVES

After completing this lesson, you will be able to:

- Explore SAP Cloud Platform

Services for SAP Fiori



SAP Cloud Platform (CP) is an open platform-as-a-service (PaaS) that provides unique in-memory database and application services. It empowers all lines of businesses and connects enterprises to the digital economy using open APIs and standards. The SAP CP delivers a variety of open, agile, flexible, and customizable apps, as well as an environment to develop and operate own apps.

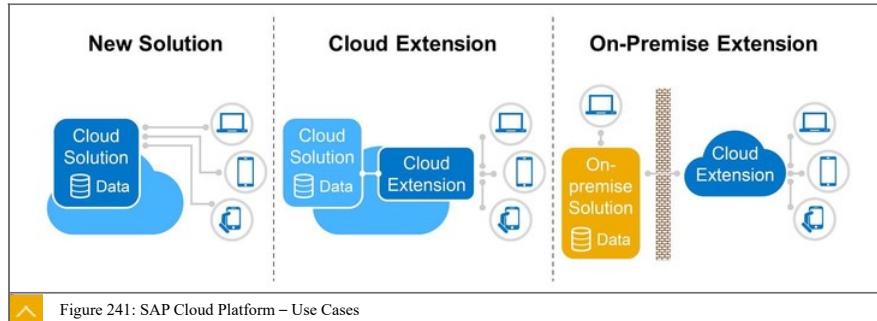


Figure 241: SAP Cloud Platform – Use Cases

Customers can increase their business speed and agility by extending SAP value to hybrid landscapes:

New Solution

SAP CP is an innovation platform to build new applications based on SAP's in-memory technology.

- Build all types of business applications
- Advanced real-time analytics
- Internet of things

Cloud Extension

SAP CP can be used to extend existing SAP cloud solutions and to integrate them with on-premise systems.

- Extend SAP cloud solutions
- Integrate cloud solutions with on-premise systems

On-Premise Extension

SAP CP can be used to extend existing SAP on-premise solutions and enable mobile scenarios, and as an outbound channel to reach consumers.

- Extend SAP on-premise solutions
- Enable mobile consumption
- Extend reach towards consumers



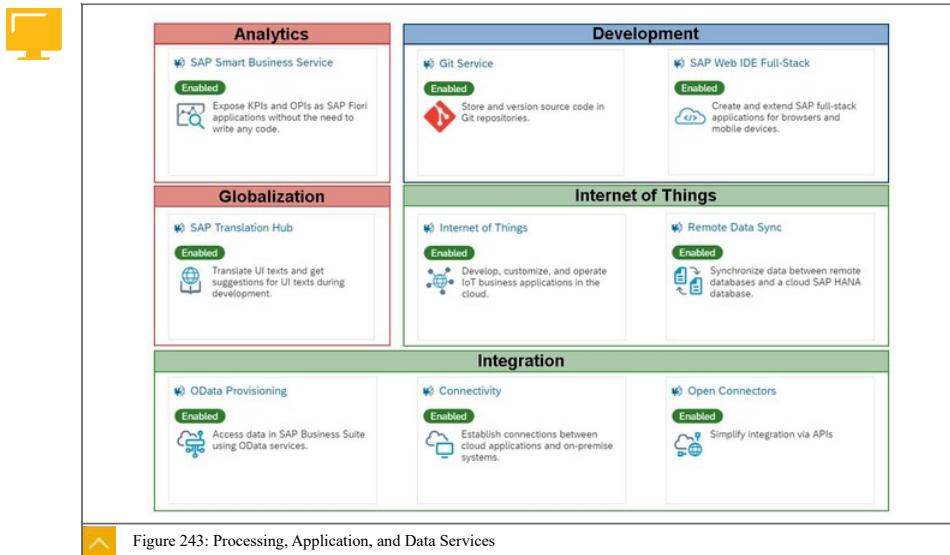
Analytics	Globalization	Internet of Things
<ul style="list-style-type: none"> • SAP Smart Business Service 	<ul style="list-style-type: none"> • SAP Translation Hub 	<ul style="list-style-type: none"> • Internet of Things • Remote Data Sync
Development	Mobile	Integration
<ul style="list-style-type: none"> • Git Service • SAP Web IDE 	<ul style="list-style-type: none"> • Mobile Services 	<ul style="list-style-type: none"> • Connectivity • OData Provisioning • Open Connectors
Collaboration	User Experience	Data and Storage
<ul style="list-style-type: none"> • Gamification • SAP Jam 	<ul style="list-style-type: none"> • Portal • SAP Build • UI Theme Designer 	<ul style="list-style-type: none"> • Document Service • SAP ASE • SAP HANA



Figure 242: SAP Cloud Platform – Services

SAP CP offers many services in different areas, especially for SAP Fiori . Analytics and globalization services are data processing services. Integration services include Internet of Things (IoT) and mobile services, which are also part of development services.

Application services consisting of collaboration and user experience are also available, as well as data services.



Processing services

- SAP Smart Business Service visualizes the analytical data in Fiori tiles and interactive dashboards.
- SAP Translation Hub offers developers support for translating UI texts during development.

Development services

- Git Service is an open-source source code repository for many programming languages.
- SAP Web IDE Full-Stack is the central development environment for apps used in SAP Cloud Platform.

Integration services

- Internet of Things (IoT) collects and manages data from sensors, actuators, and electronics.
- Remote Data Sync performs a two-way data synchronization between remote databases and a central consolidated SAP HANA database on SAP Cloud Platform.
- OData Provisioning is the front-end part of SAP Gateway.
- Connectivity establishes secure and reliable connectivity between cloud applications and on-premise systems.

- **Open Connectors** provides pre-built and feature-rich connectors to simplify the connectivity and seamless integration with a huge amount of non-SAP cloud applications.

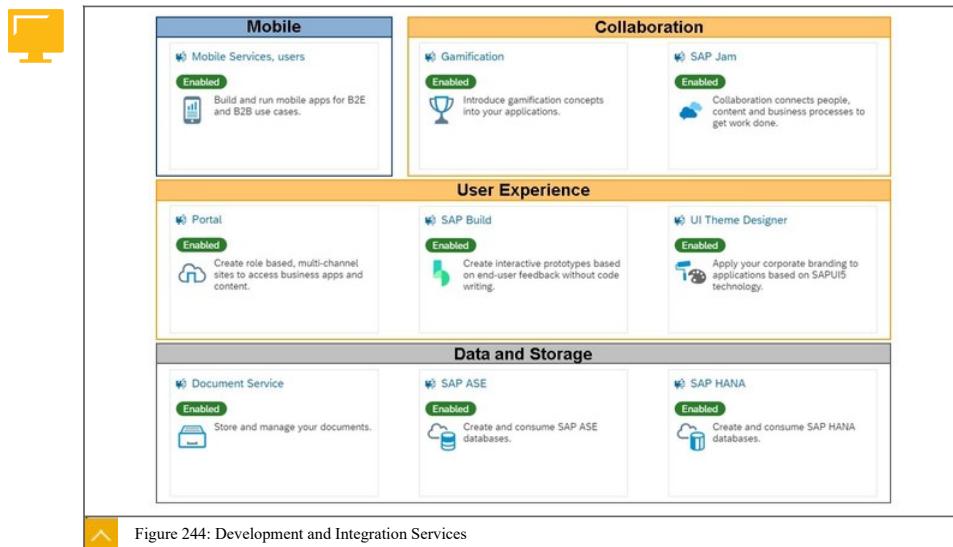


Figure 244: Development and Integration Services

Mobile Services

- Mobile Services provide mobile access to enterprise information as well as development and management of mobile apps.

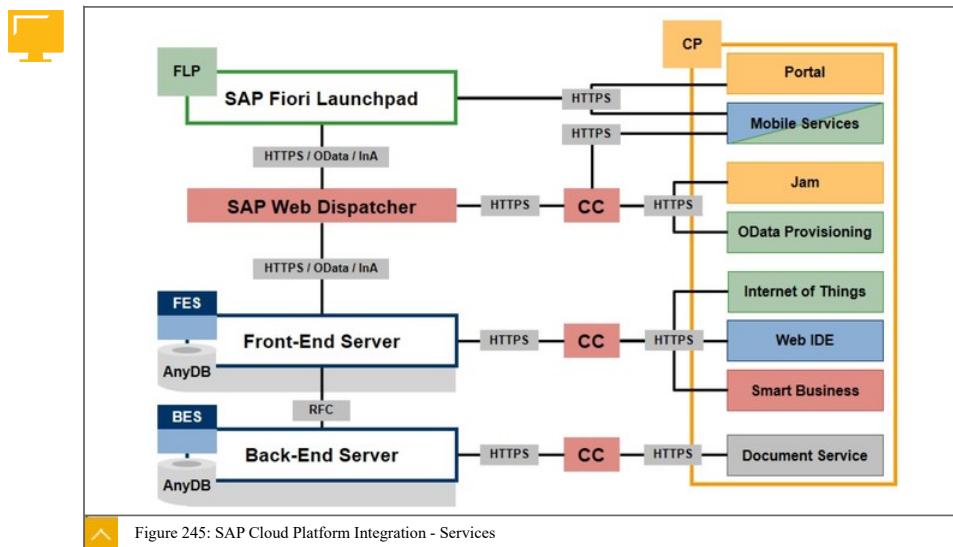
Application services

- Gamification is a gamification platform for including concepts and rules based on game design in business applications.
- SAP Jam is a secure collaboration platform, which can be deeply integrated in daily work with SAP Fiori apps.
- Portal offers features known from SAP Enterprise Portal to provide web sites such as the SAP Fiori launchpad.
- SAP Build is a tool to create and share user-friendly, interactive prototypes that are based on end-user feedback studies without writing any code.
- UI Theme Designer is a tool to build and apply corporate branding to applications created with SAP UI technologies.

Data services

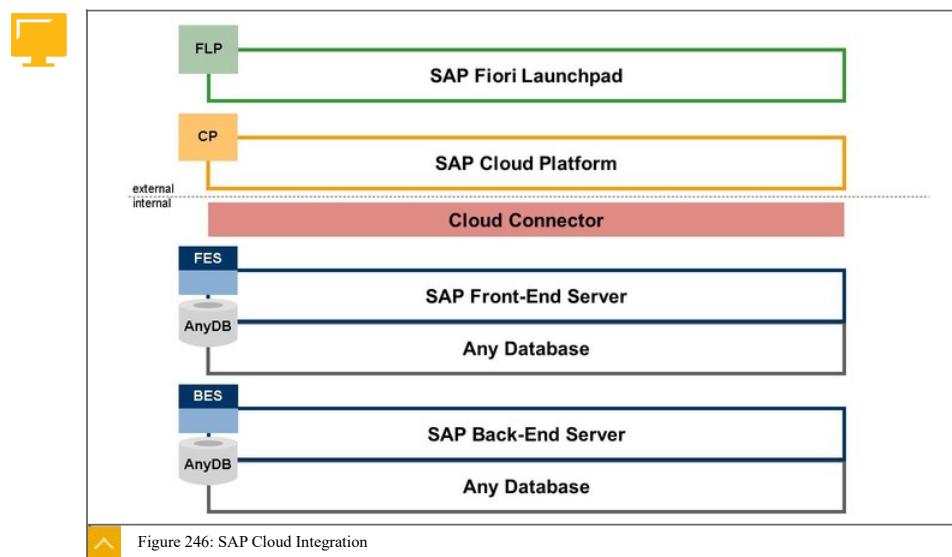
- Document Service offers secure filing for digital documents in the cloud.
- SAP ASE provides an SAP ASE database consumable by apps running in SAP Cloud Platform.

- SAP HANA provides an SAP HANA database consumable by apps running in SAP Cloud Platform .

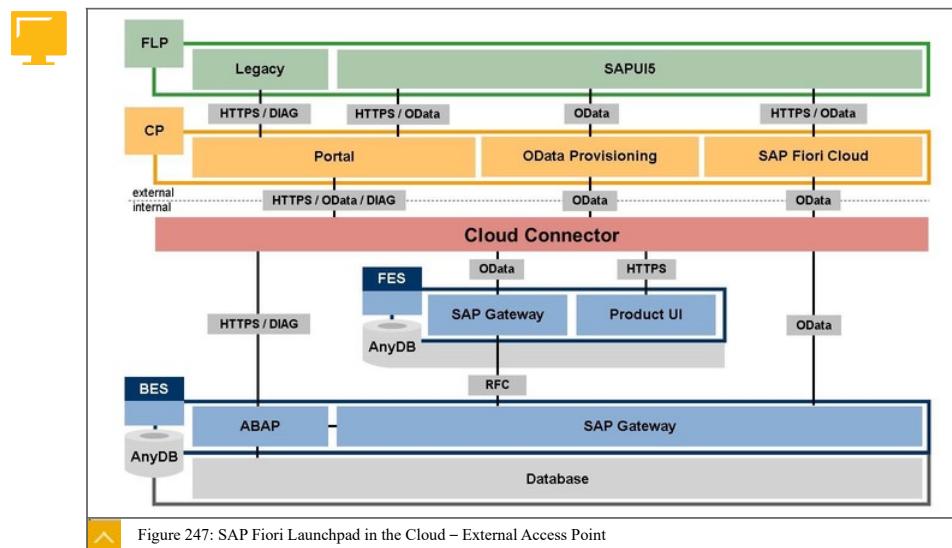


When integrating SAP Fiori for an on-premise landscape with SAP CP services, it depends on the nature of the service, and which system connects to which service. User Experience and Fiori Mobile enhance the FLP directly. Collaboration and OData Provisioning enhance the FLP indirectly with additional features called via routing rules of the SAP Web Dispatcher . The FES can request data from Internet of Things as well as Analytics and benefits from tools in Development for SAP Fiori . It is also possible to enhance data management of the BES with services in the area of Data and Storage . All connections rely on the SAP Cloud Connector , which is installed in the on-premise landscape as often as needed in terms of security and connectivity.

SAP Fiori Launchpad in the Cloud

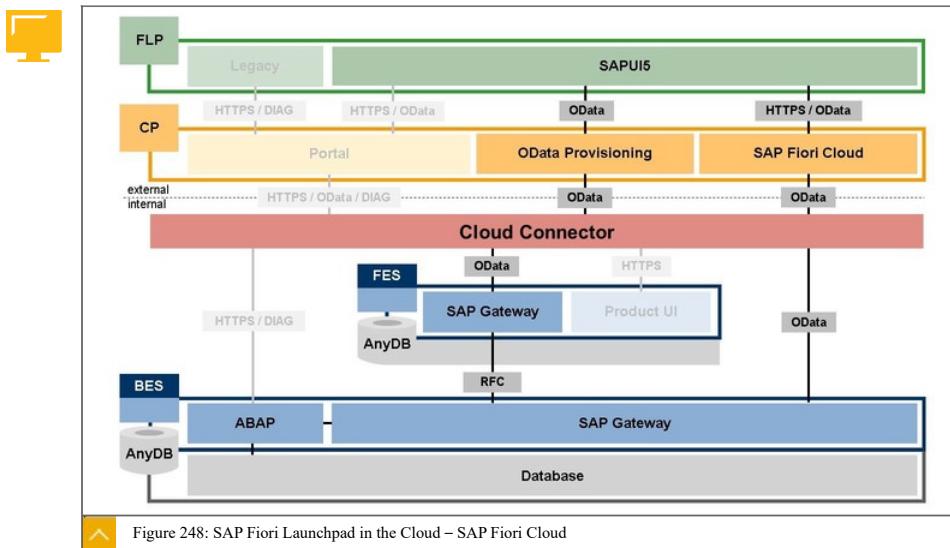


SAP Cloud Platform (CP) runs in the Internet. Therefore, clients need to connect to the Internet to consume SAP Fiori. When using apps on mobile devices, the client itself is already in the Internet.



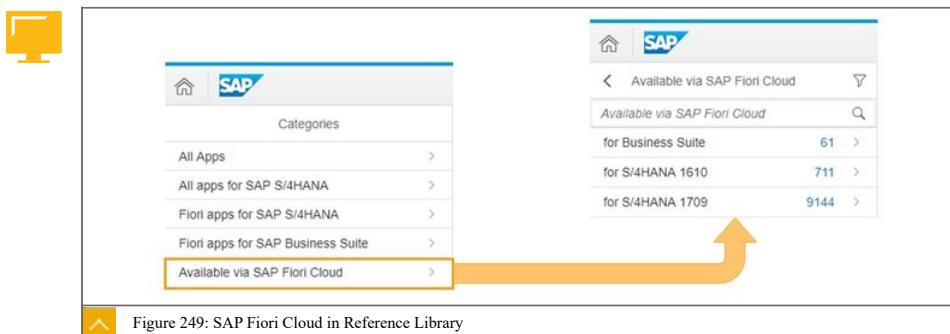
As for on-premise, data for apps is transferred between the client and the back-end using OData. When SAP CP is used as an external access point (EAP), OData requests are first processed by SAP CP OData Provisioning. This forwards the request via the Cloud Connector to SAP Gateway in the customer landscape. Here it is possible for the customer to directly connect the BES via the SAP Cloud Connector with SAP CP or use an FES in between. From a

technical perspective, both options are possible. However, from a security perspective, it is rarely permissible to connect the BES directly to the Internet.

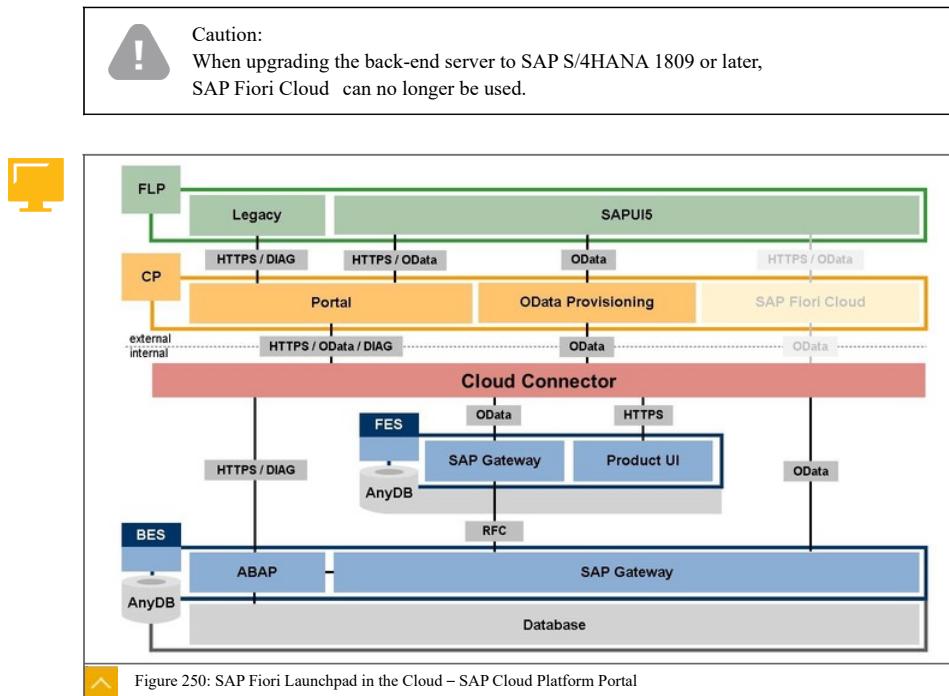


SAP Fiori Cloud enables customers to establish SAP Fiori in their enterprise by just connecting this cloud offering with their system landscape. All UI components are part of the SAP CP. There is no need to install any UI components in the customer landscape.

When integrating SAP Fiori Cloud, the system landscape does not have an SAP Web Dispatcher. Besides the SAP Cloud Platform, which is provided by an SAP data center, the Cloud Connector is a new component. It has to be installed in the system landscape of the customer, connecting the SAP CP in the Internet with the local customer network.

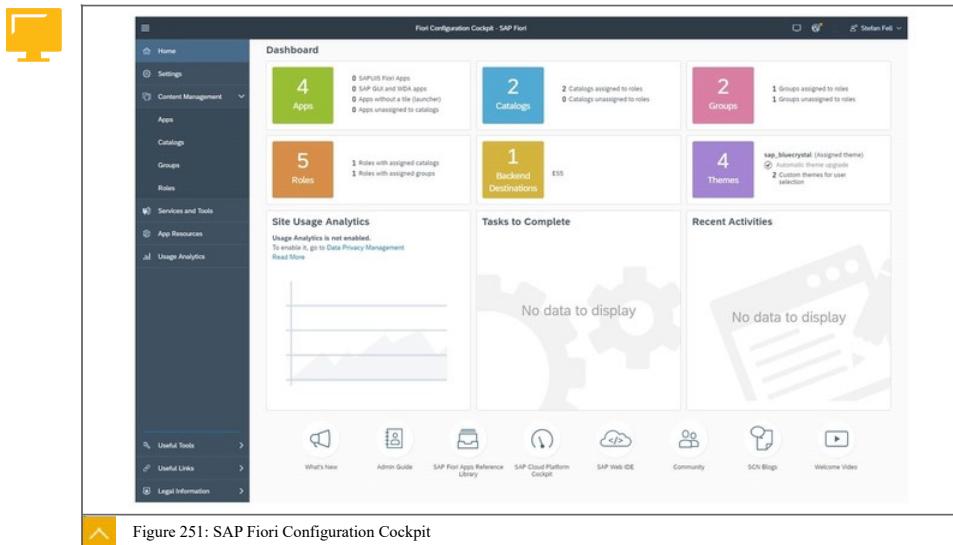


SAP Fiori Cloud is available for SAP Business Suite, SAP S/4HANA 1610, and SAP S/4HANA 1709. That means the UI components provided in SAP Fiori Cloud are available for these back-end server releases. There is no plan to extend this offering to other releases.

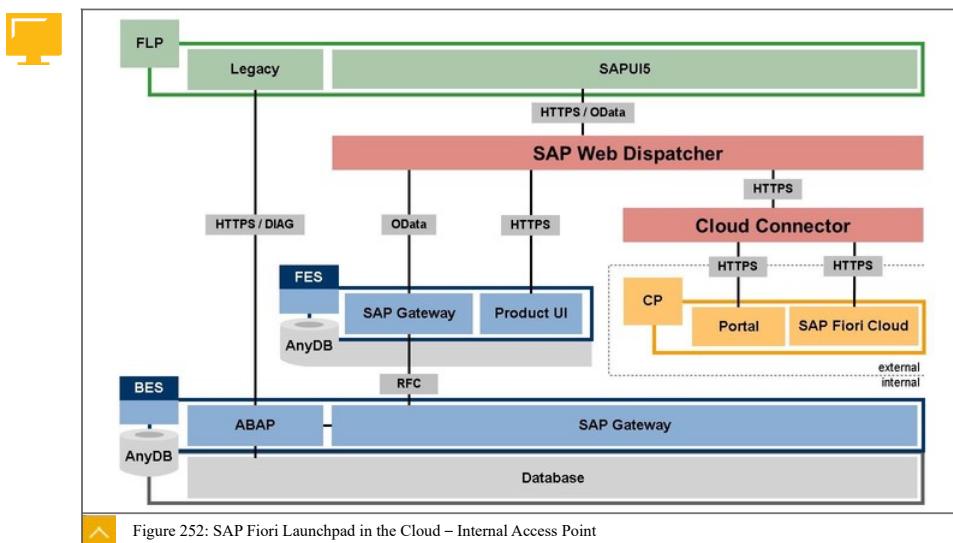


SAP Cloud Platform Portal offers features comparable to the SAP Enterprise Portal. In short this means providing sites for end users combining web UIs from various sources build with various technologies. One of these technologies is SAP Fiori.

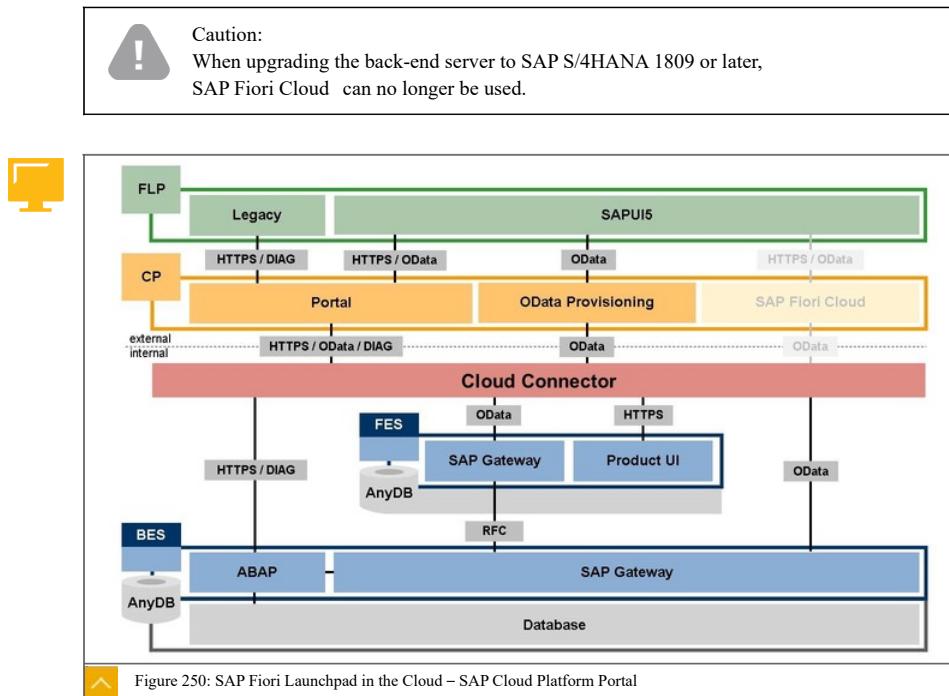
Like SAP Fiori Cloud, the Portal focuses on the UI parts providing an SAP Fiori launchpad with catalogs and groups to the user. In contrast to SAP Fiori Cloud, it is possible to define app definitions pointing to UI implementations in the on-premise system landscape. These can be SAPUI5 apps in the FES, transactions and Web Dynpro applications in the BES, or any other web source.



The SAP Fiori Configuration Cockpit is the central tool for administration, content management, and monitoring of portal sites. You define roles, catalogs, and groups comparable to the on-premise world as well as apps, which combine the target mapping and tile in one element. You can manage the app resources, destinations, and themes as well as monitor the usage of the site utilising the analytical power of the cloud.



Often, there is a security rule in companies that company data is not allowed to be exchanged with systems outside of the company network. In this scenario, the SAP Cloud Platform can be used just for the UI parts as an internal access point (IAP). The SAP Fiori launchpad in the client connects to an SAP Web Dispatcher provided in the company network. This



SAP Cloud Platform Portal offers features comparable to the SAP Enterprise Portal. In short this means providing sites for end users combining web UIs from various sources build with various technologies. One of these technologies is SAP Fiori.

Like SAP Fiori Cloud, the Portal focuses on the UI parts providing an SAP Fiori launchpad with catalogs and groups to the user. In contrast to SAP Fiori Cloud, it is possible to define app definitions pointing to UI implementations in the on-premise system landscape. These can be SAPUI5 apps in the FES, transactions and Web Dynpro applications in the BES, or any other web source.

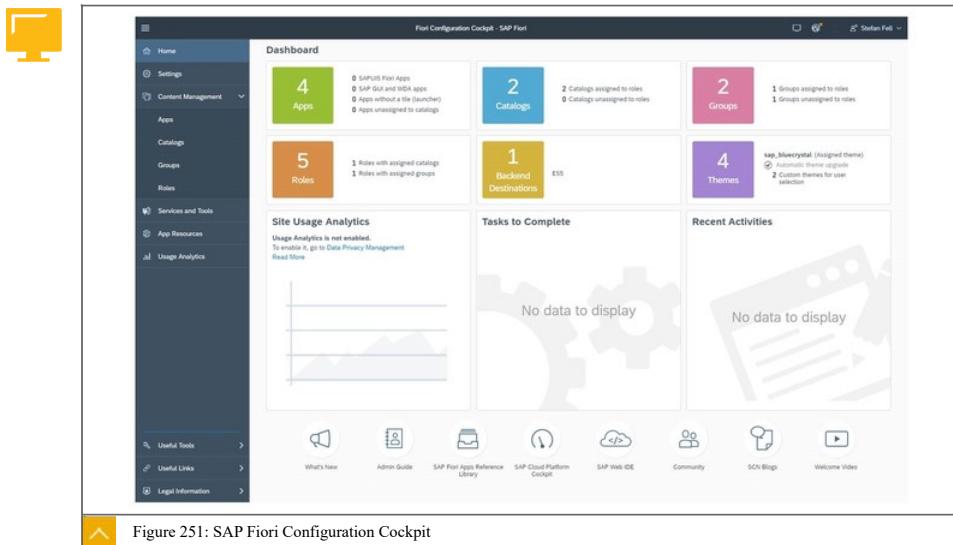
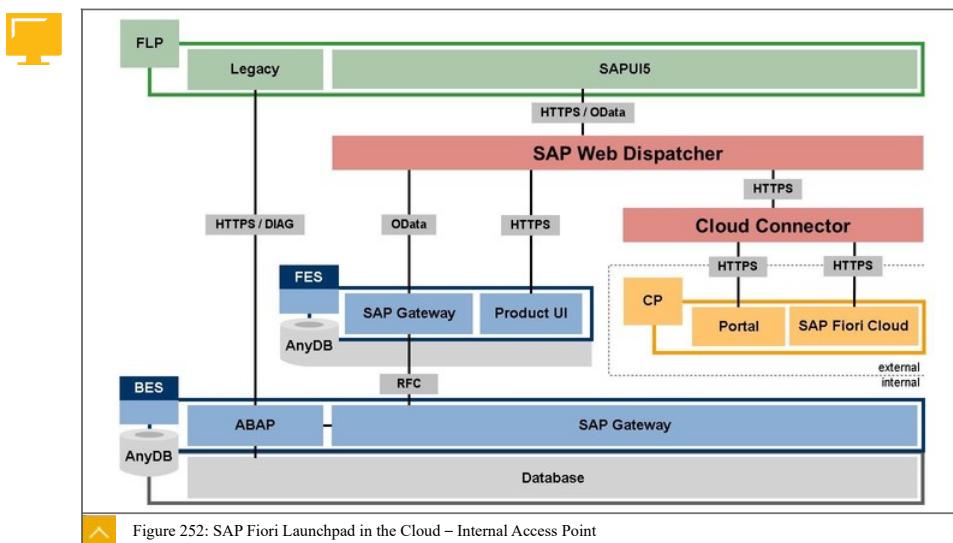


Figure 251: SAP Fiori Configuration Cockpit

The SAP Fiori Configuration Cockpit is the central tool for administration, content management, and monitoring of portal sites. You define roles, catalogs, and groups comparable to the on-premise world as well as apps, which combine the target mapping and tile in one element. You can manage the app resources, destinations, and themes as well as monitor the usage of the site utilising the analytical power of the cloud.



Often, there is a security rule in companies that company data is not allowed to be exchanged with systems outside of the company network. In this scenario, the SAP Cloud Platform can be used just for the UI parts as an internal access point (IAP). The SAP Fiori launchpad in the client connects to an SAP Web Dispatcher provided in the company network. This

Unit 7

Learning Assessment - Answers

1. What is the central app to display Workflow items in SAP Fiori?

My Inbox

2. In which system is the SAP Gateway workflow service implemented?

Front-End Server (FES)

3. What is used as a container for SAP Fiori in an Enterprise Portal?

Fiori iView

4. Which shell configurations are available to adapt the header of the app?

Choose the correct answers.

- A encapsulated
- B embedded
- C standalone
- D headerless
- E footless

Correct! The following shell configurations are available to adapt the header of the app: embedded, standalone, and headerless.

5. What is SAP Cloud Platform (CP)?

SAP's platform-as-a-service (PaaS) providing in-memory database and application services

6. What establishes a secure connection between on-premise systems and SAP CP?

Cloud Connector

7. Which service is responsible for OData services in the SAP CP?

SAP CP OData provisioning
