

UX100

SAP Fiori – Foundation

PARTICIPANT HANDBOOK INSTRUCTOR-LED TRAINING

Course Version: 22

Course Duration: 5 Day(s)

Material Number: 50157747

SAP Copyrights, Trademarks and Disclaimers

© 2022 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 <https://www.sap.com/corporate/en/legal/copyright.html> 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.

These materials 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

3	Lesson: Exploring SAP Fiori Design
11	Lesson: Exploring SAP Fiori Launchpad
23	Lesson: Personalizing SAP Fiori
29	Lesson: Exploring SAP Fiori Data Handling

39 Unit 2: Technology

41	Lesson: Explaining User Interfaces
49	Lesson: Explaining Data Services
57	Lesson: Explaining Application Types

73 Unit 3: Architecture

75	Lesson: Examining ABAP Platform
83	Lesson: Examining SAP HANA
87	Lesson: Examining SAP S/4HANA
103	Lesson: Examining SAP Fiori Development

117 Unit 4: Configuration

119	Lesson: Managing SAP Fiori Content
125	Lesson: Creating SAP Fiori Spaces and Pages
131	Lesson: Creating SAP Fiori Groups
135	Lesson: Managing SAP Fiori Catalogs
141	Lesson: Creating Business Catalogs
147	Lesson: Creating Target Mappings
151	Lesson: Creating Technical Catalogs
159	Lesson: Creating Replicable Catalogs

171 Unit 5: Content Administration

173	Lesson: Exploring Rapid Activation for SAP Fiori
177	Lesson: Examining Basic Roles for SAP Fiori
181	Lesson: Configuring SAP Fiori Launchpad
185	Lesson: Troubleshooting SAP Fiori Launchpad

195 Unit 6: Adaptation

197	Lesson: Using the UI Theme Designer
203	Lesson: Using SAP Fiori Runtime Authoring
207	Lesson: Extending SAP Fiori Applications

225 Unit 7: Mobility

- | | |
|-----|-------------------------------------|
| 227 | Lesson: Examining SAP Fiori Mobile |
| 235 | Lesson: Examining SAP Fiori for iOS |

247 Unit 8: Integration

- | | |
|-----|---|
| 249 | Lesson: Exploring SAP Business Technology Platform Services |
| 257 | Lesson: Exploring SAP Launchpad Service |

267 Unit 9: Further Information

- | | |
|-----|-------------------------------------|
| 269 | Lesson: Getting Further Information |
|-----|-------------------------------------|

273 Unit 10: Appendix

- | | |
|-----|--|
| 275 | Lesson: Examining SAP Business Suite |
| 285 | Lesson: Creating Classic Target Mappings |
| 293 | Lesson: Configuring Analytical Applications |
| 301 | Lesson: Integrating SAP Fiori in SAP Enterprise Portal |
| 309 | Lesson: Integrating SAP Workflow in SAP Fiori |
| 315 | Lesson: Using SAP Screen Personas |

Course Overview

TARGET AUDIENCE

This course is intended for the following audiences:

- Application Consultant
- Development Consultant
- Technology Consultant
- Support Consultant
- Developer
- Enterprise Architect
- Solution Architect
- System Administrator
- System Architect

UNIT 1

End-User Perspective

Lesson 1

Exploring SAP Fiori Design

3

Lesson 2

Exploring SAP Fiori Launchpad

11

Lesson 3

Personalizing SAP Fiori

23

Lesson 4

Exploring SAP Fiori Data Handling

29

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 design language that brings great user experiences to enterprise applications
- Works seamlessly on desktops, tablets, and smartphones

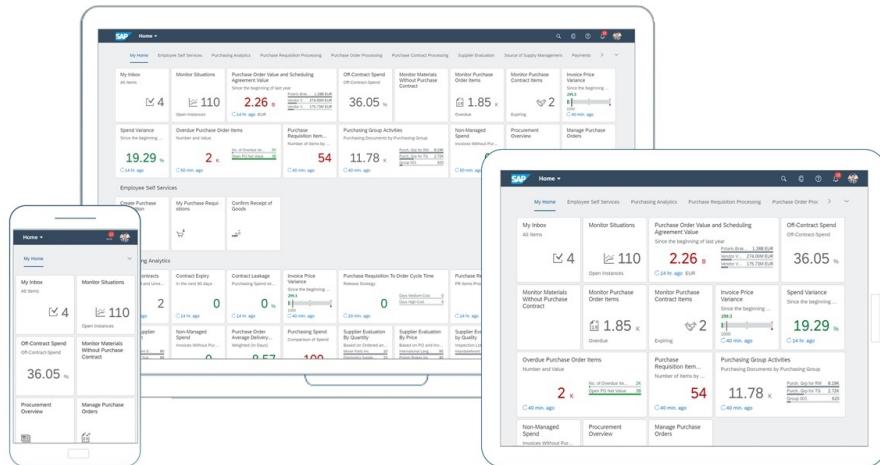


Figure 1: What is SAP Fiori

SAP Fiori is the design language that brings great user experiences to enterprise applications based on 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 2.0 was introduced with SAP S/4HANA 1610, taking the idea of SAP Fiori to the next level. Today SAP Fiori 3 is the current target design, which evolves the SAP Fiori design language for all SAP products to fully support the Intelligent Suite.

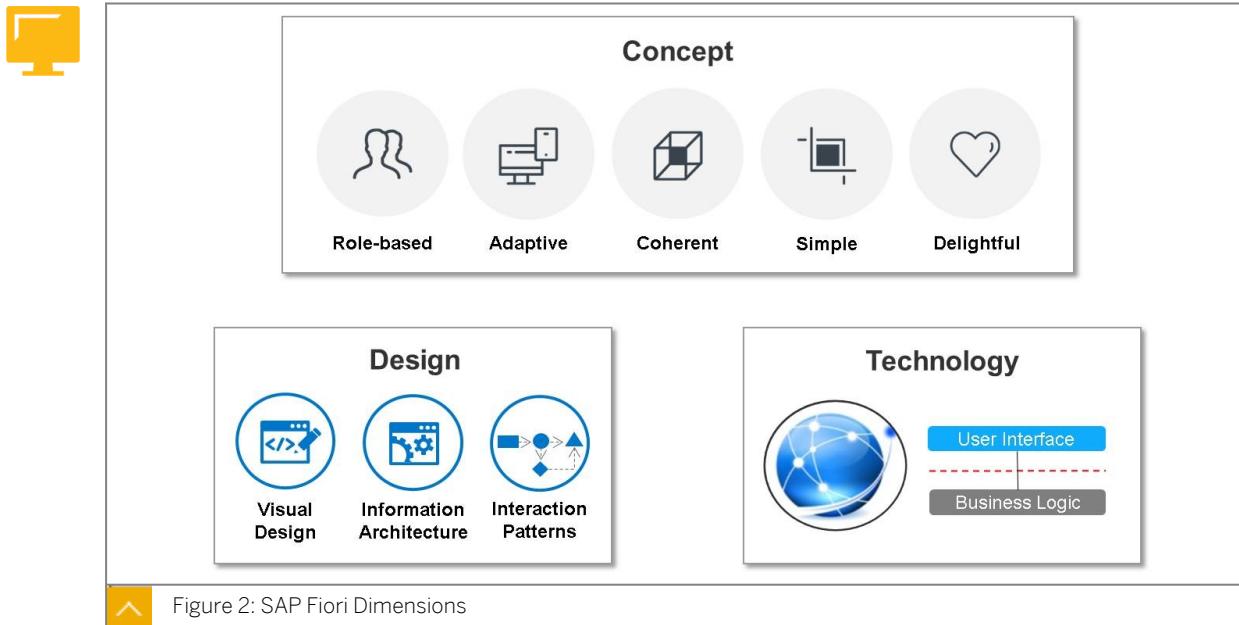


Figure 2: SAP Fiori Dimensions

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

The SAP Fiori Design Guidelines website features a dark-themed header with the SAP logo and "Fiori Design Guidelines". The main content area includes a welcome message, a large image of a mobile device and tablet displaying Fiori apps, and four smaller cards for "SAP Fiori for Android", "SAP Fiori for iOS", "SAP Fiori", and "Conversational UX". At the bottom, there is a link to experience.sap.com/fiori-design.

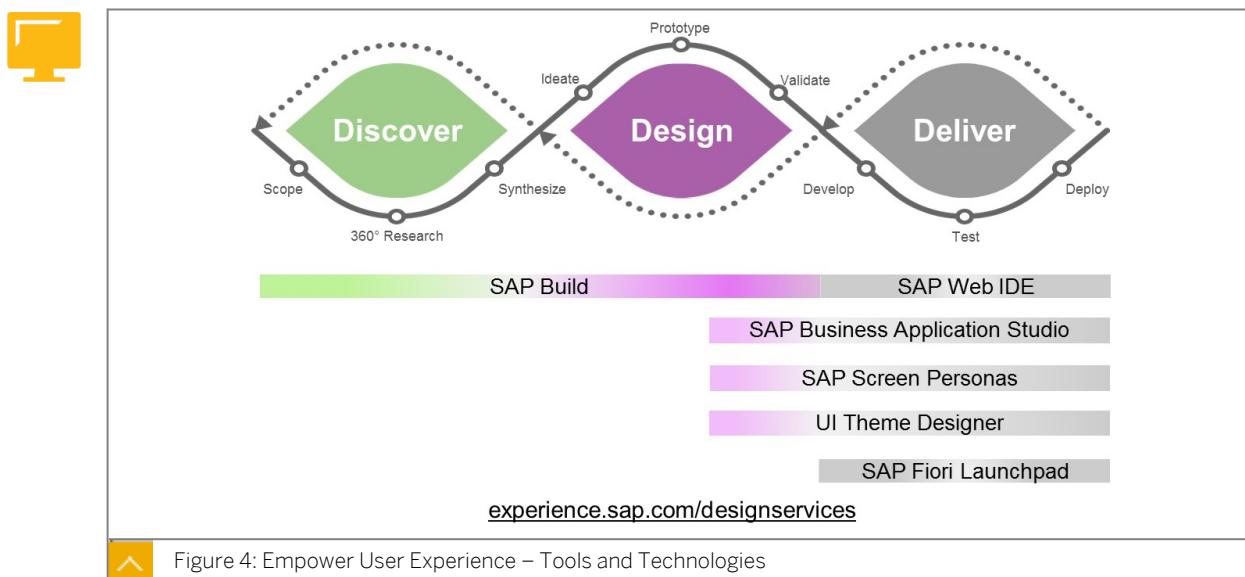
Figure 3: SAP Fiori Design Guidelines

All details of the SAP Fiori design are available as guidelines for general use. You can find all aspects of the SAP 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, the SAP icon font or font 72, 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 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. However, 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 was already possible in SAP Fiori 2.0 to use the SAP Conversational AI, and it became 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 SAP 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.



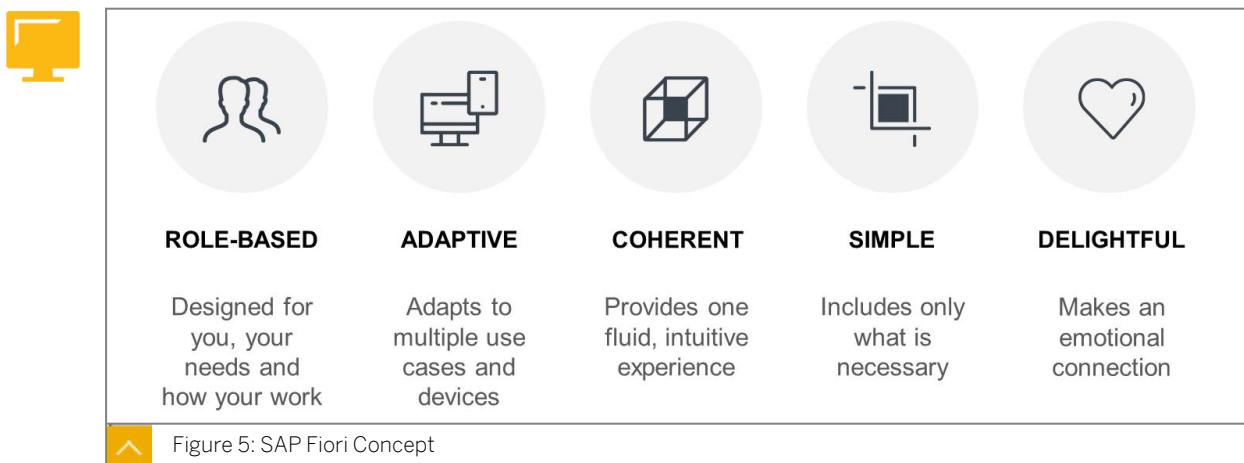
Note:

New enterprise licenses are no longer available for SAP Build. Existing customers may add additional users by contacting their SAP account executive or via the SAP Store. Everybody can use SAP Build freely in the trial version. SAP Build is supported until December 2024.

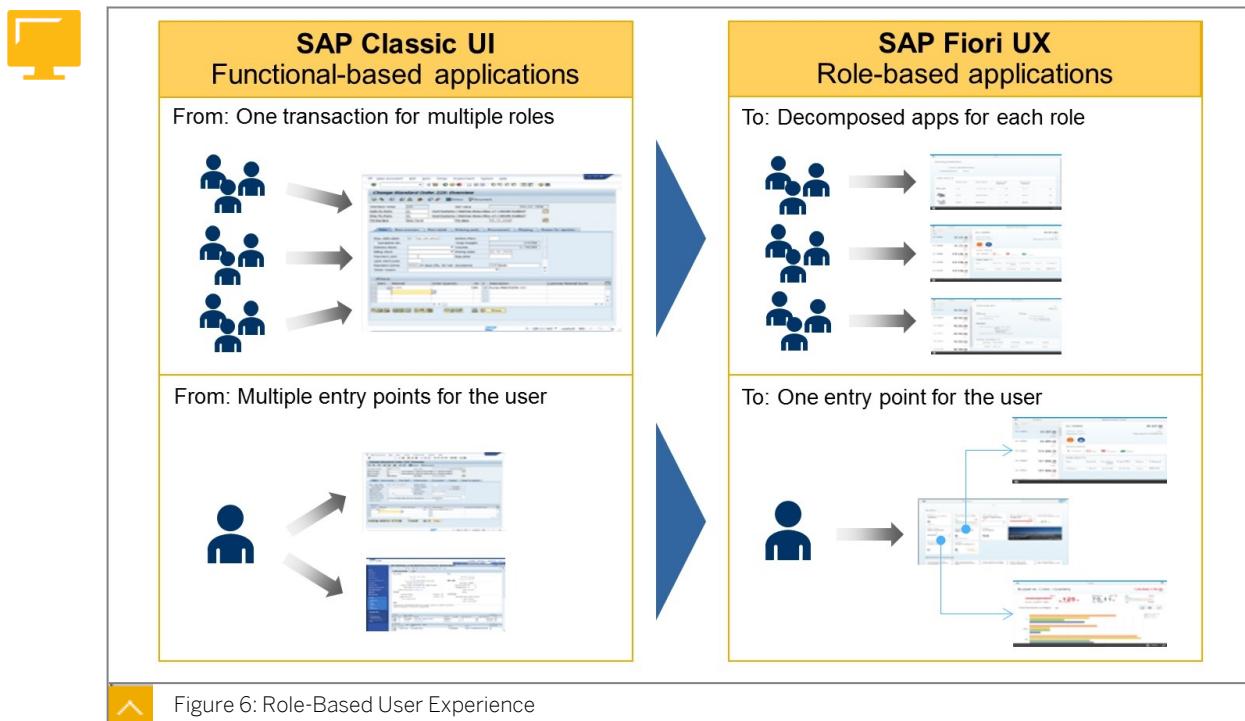
- SAP Web IDE is the Integrated Development Environment (IDE) for the web technology SAPUI5, the basis of all SAP Fiori web apps.
- SAP Business Application Studio is the next generation of SAP Web IDE provided in the SAP Business Technology Platform (BTP), Cloud Foundry environment.
- UI Theme Designer 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



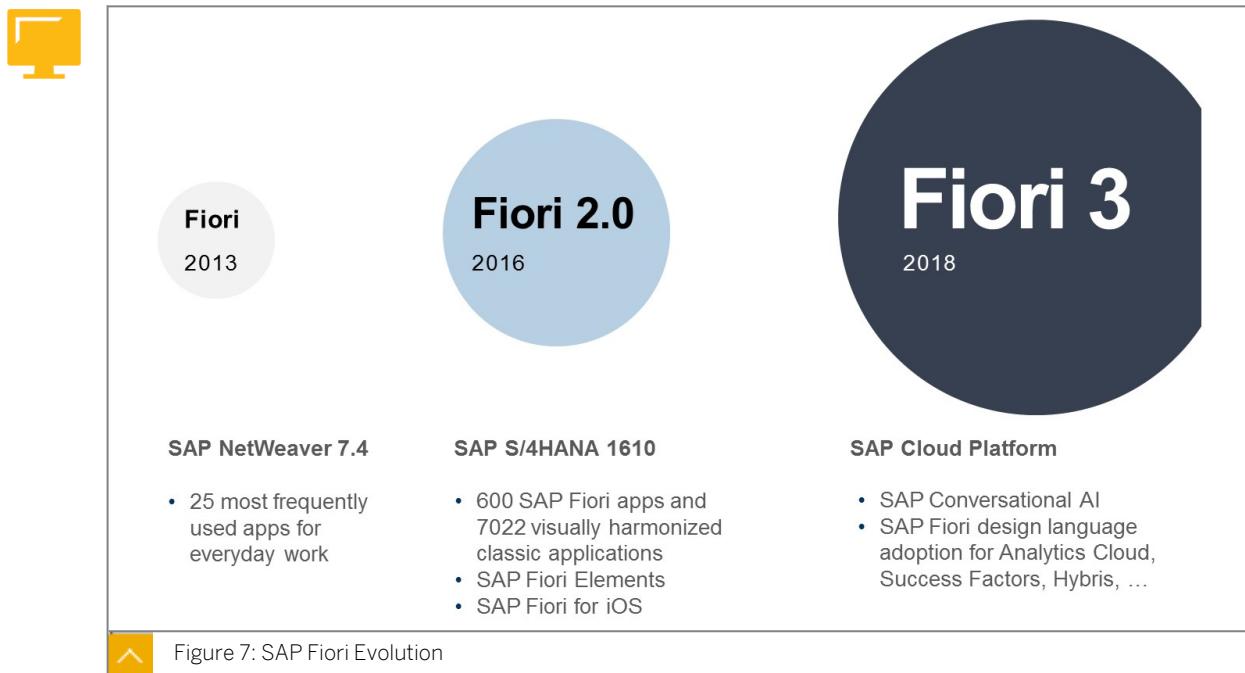
In the second dimension, SAP Fiori offers a unified user experience for a variety of clients. Users should have a consistent, coherent, simple, intuitive, and delightful user experience on all devices to be able to work better and more efficiently. The five design principles of SAP Fiori are at the core of every SAP 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 provides end users with 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.

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.

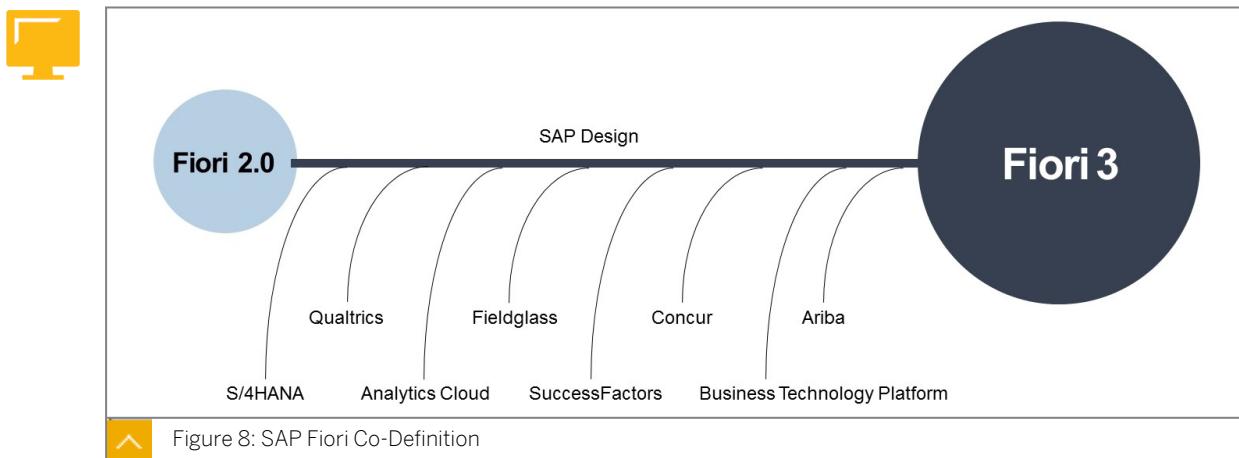
Technology



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 were fine-tuned in the following years.

In October 2018, the new design SAP Fiori 3 was introduced. Unlike previous versions, SAP did not deliver everything at once. SAP chose to deliver SAP Fiori 3 innovations in steps.



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 must be added to best support them all.

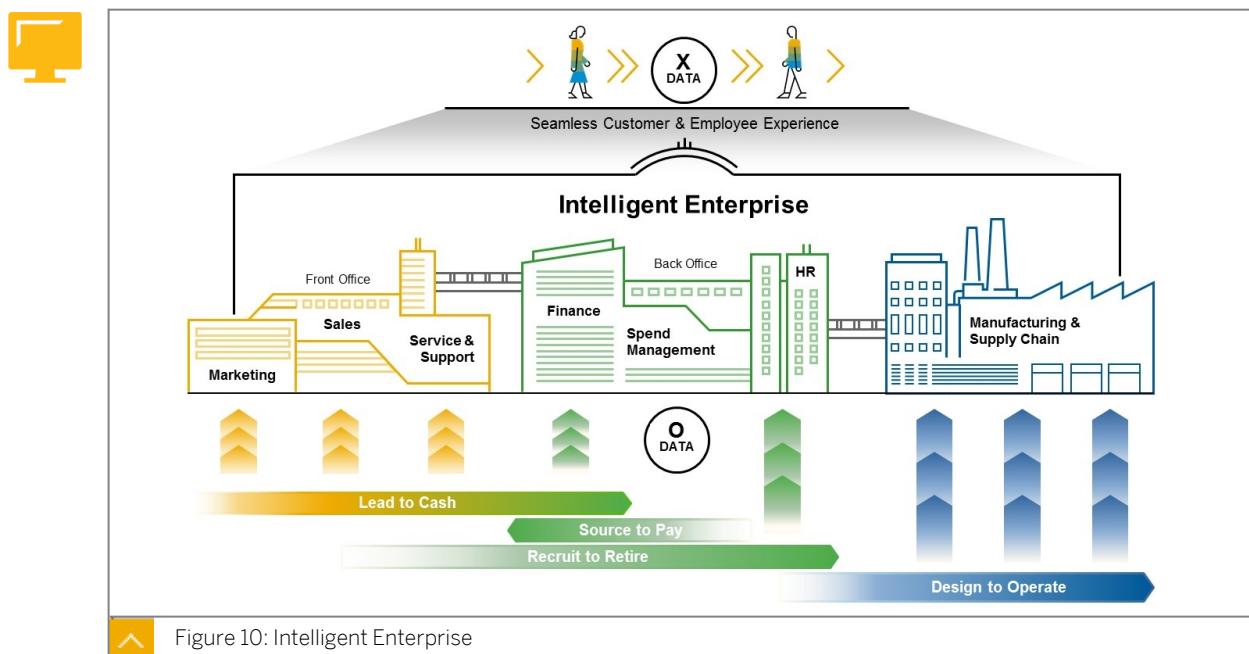
Figure 9: SAP Fiori 3

One of the most requested new aspects is to provide more information on home pages, which the design supports via cards. In addition, SAP reimaged how aspects of intelligence such

as natural language interaction and machine intelligence can become a key part of the experience. It deeply integrates them into the foundation of the design, providing the following features:

- A new theme for a fresh, modern look and feel.
- Intelligent home pages showing everything the user must 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).
- 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 enterprise 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 (X-data).

Businesses must be able to interpret and analyze large amounts of X-data, and understand its relationship to the operational data (O-data) of the business.

SAP delivers expertise across every industry in end-to-end processes that link together different systems. For example, "Recruit to Retire" links together financial planning, headcount planning, contractor management, and staffing into seamless end-to-end processes. This gives companies real-time understanding of the relationship between financial forecasts and workforce needs, costs and assumptions.



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

Evolution

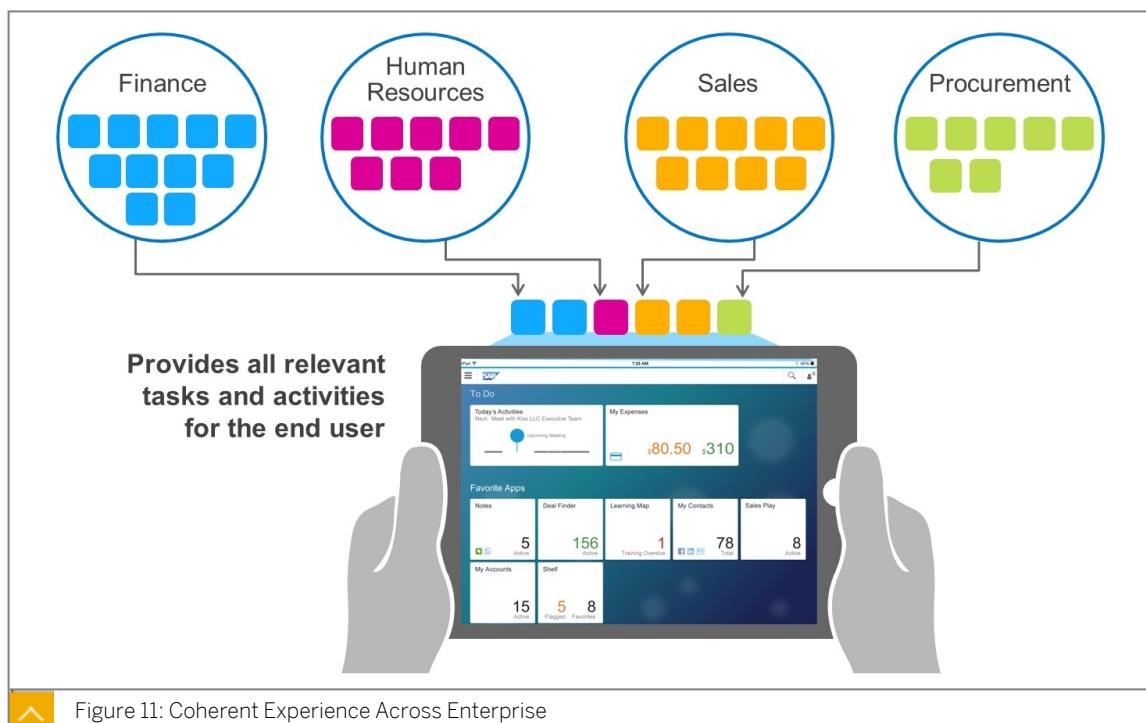
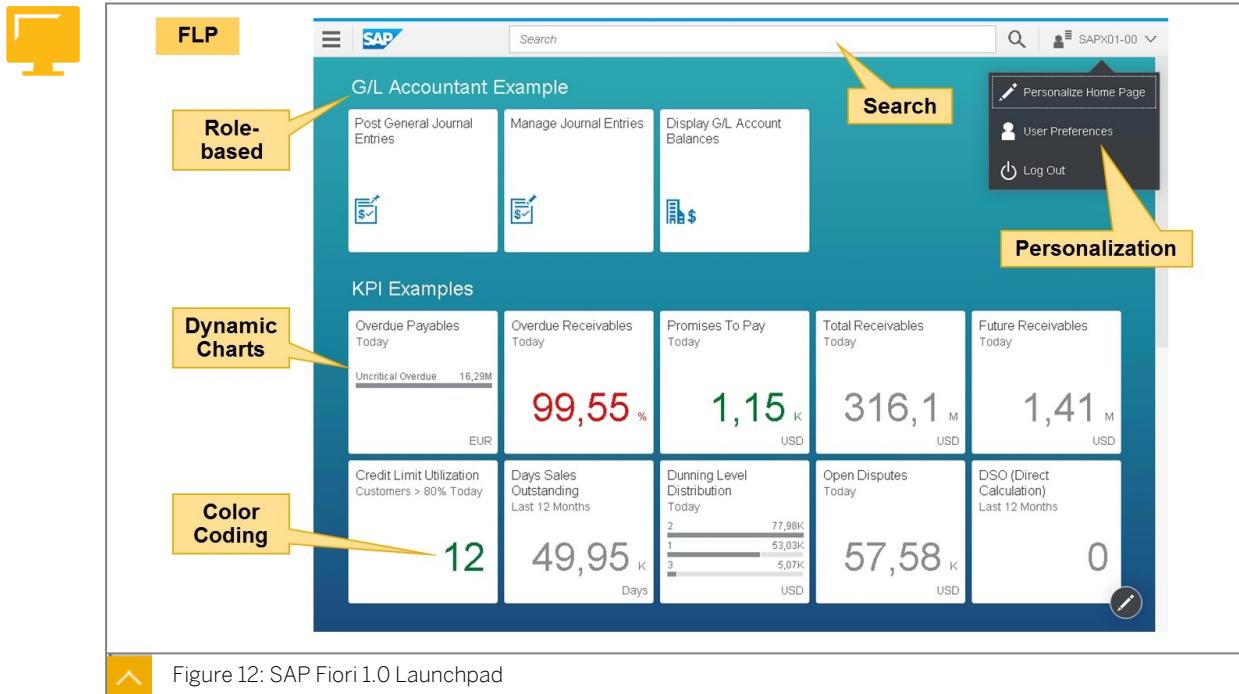
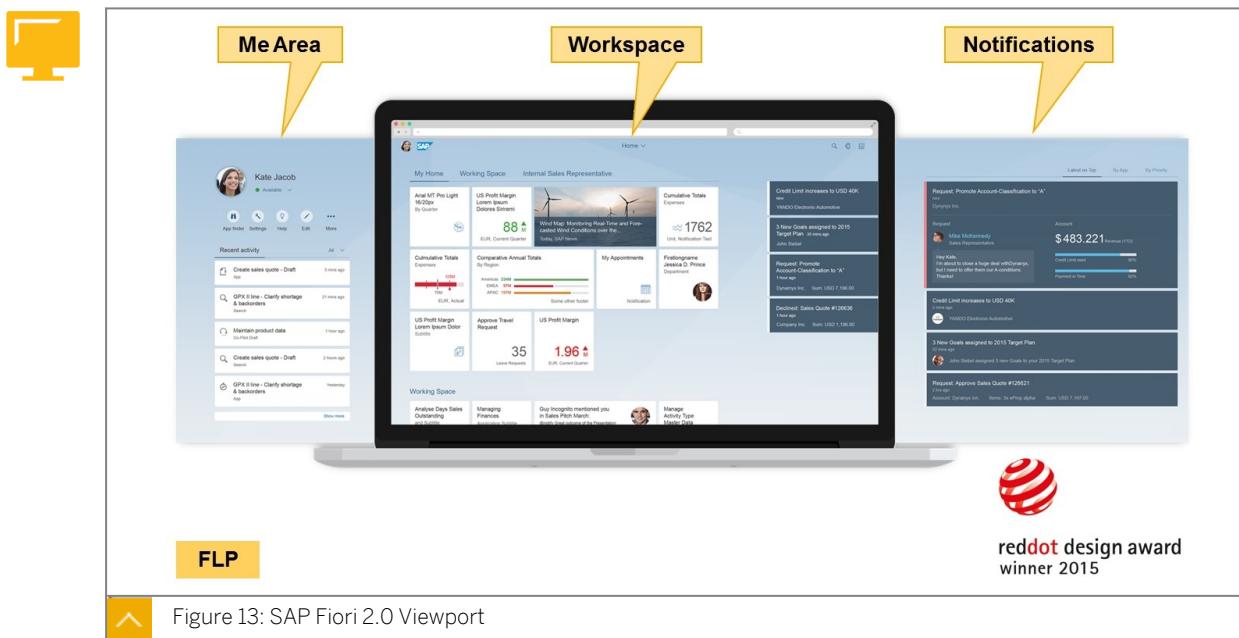


Figure 11: Coherent Experience Across Enterprise

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.



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 *launchpad* 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 such as the navigation menu at the top. The whole design of SAP Fiori 2.0 was rewarded with the “Red Dot Design Award” in 2015.

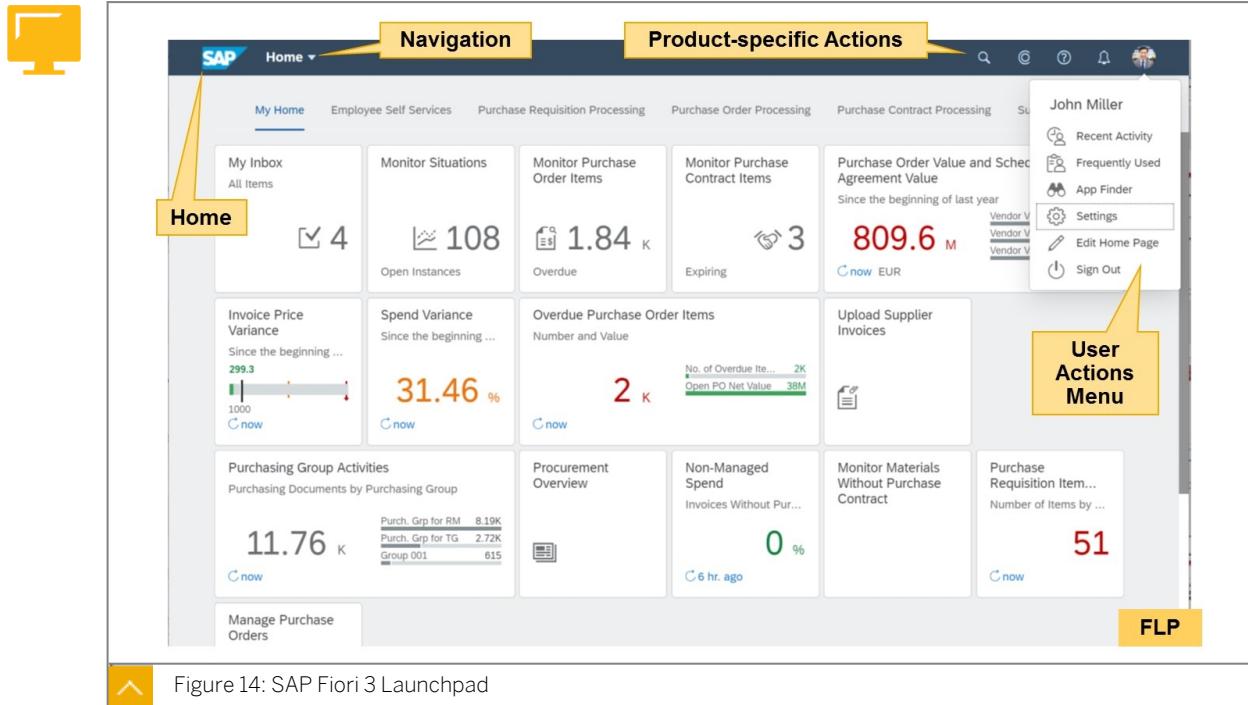


Figure 14: SAP Fiori 3 Launchpad

With SAP S/4HANA Cloud 1908 and SAP S/4HANA 1909, SAP Fiori 3 is available to customers. Compared to the previous versions, the colors and animations are withdrawn. The Me Area of SAP Fiori 2.0 is transformed in the *User Actions Menu* accessible via the user in the top right. More product-specific actions are available in the header bar and the home-button is visualized as the company logo.

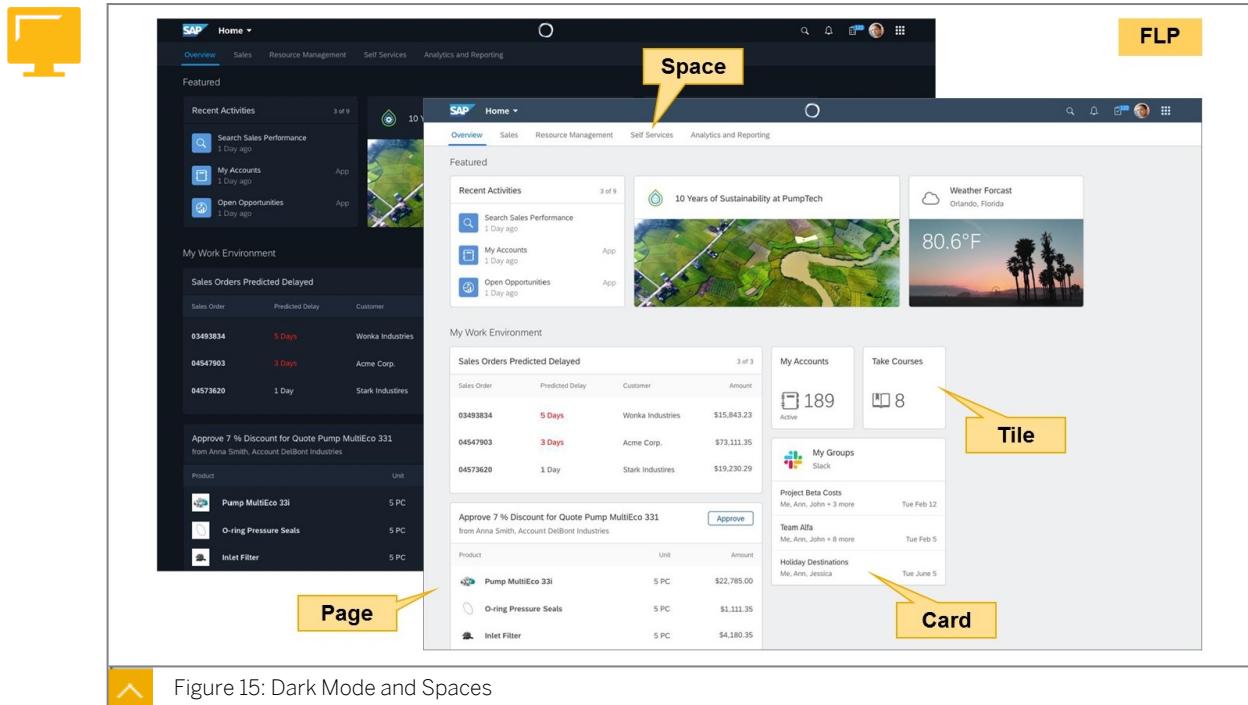


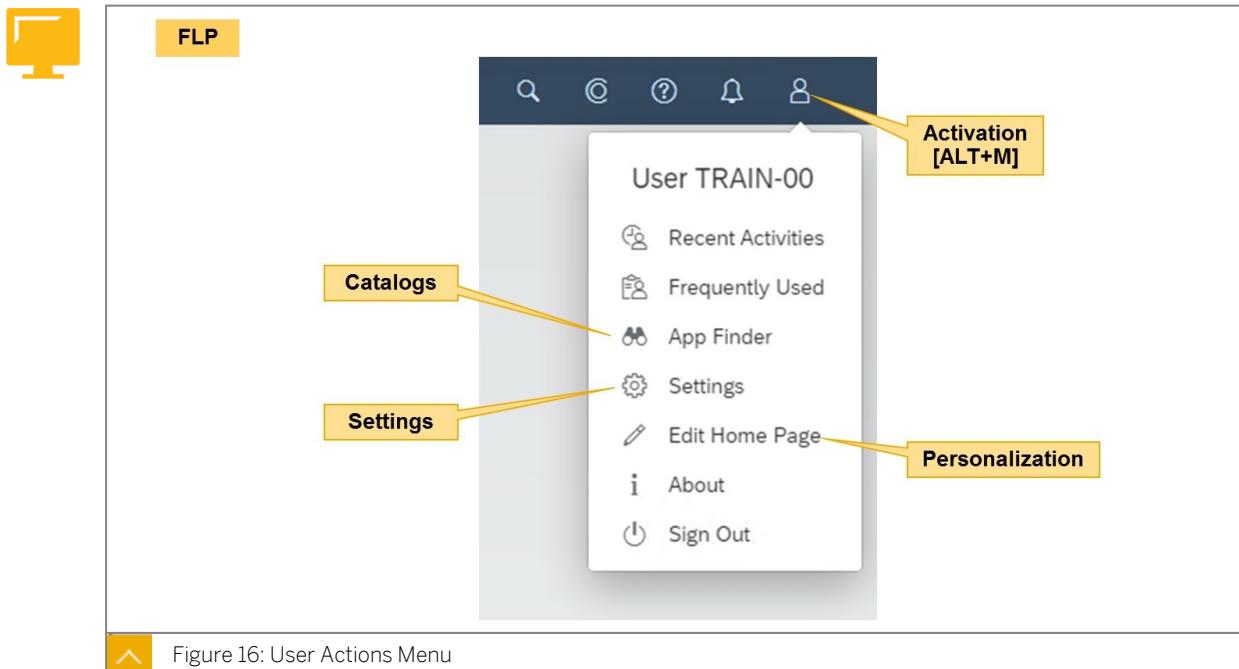
Figure 15: Dark Mode and Spaces

SAP Fiori 3 is delivered in a stepwise approach. More additions are available as design previews and some are already in production. Beside the SAP Quartz Light theme, a dark mode using the SAP Quartz Dark theme is available since SAP S/4HANA 1909 FPS02. The spaces concept offering customizable pages can be used since SAP S/4HANA 2020. New additions to SAP Fiori 3 for the future are always coming first to SAP S/4HANA Cloud.

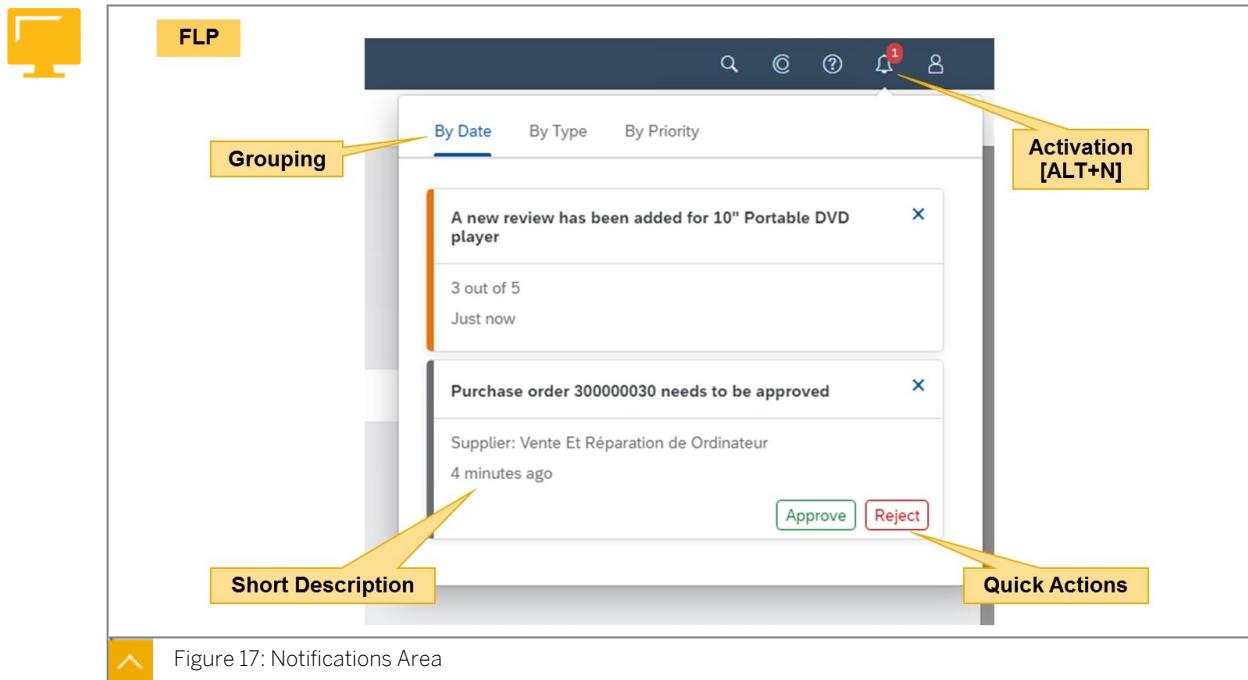


Hint:
For the latest information please visit
<https://community.sap.com/topics/fiori>.

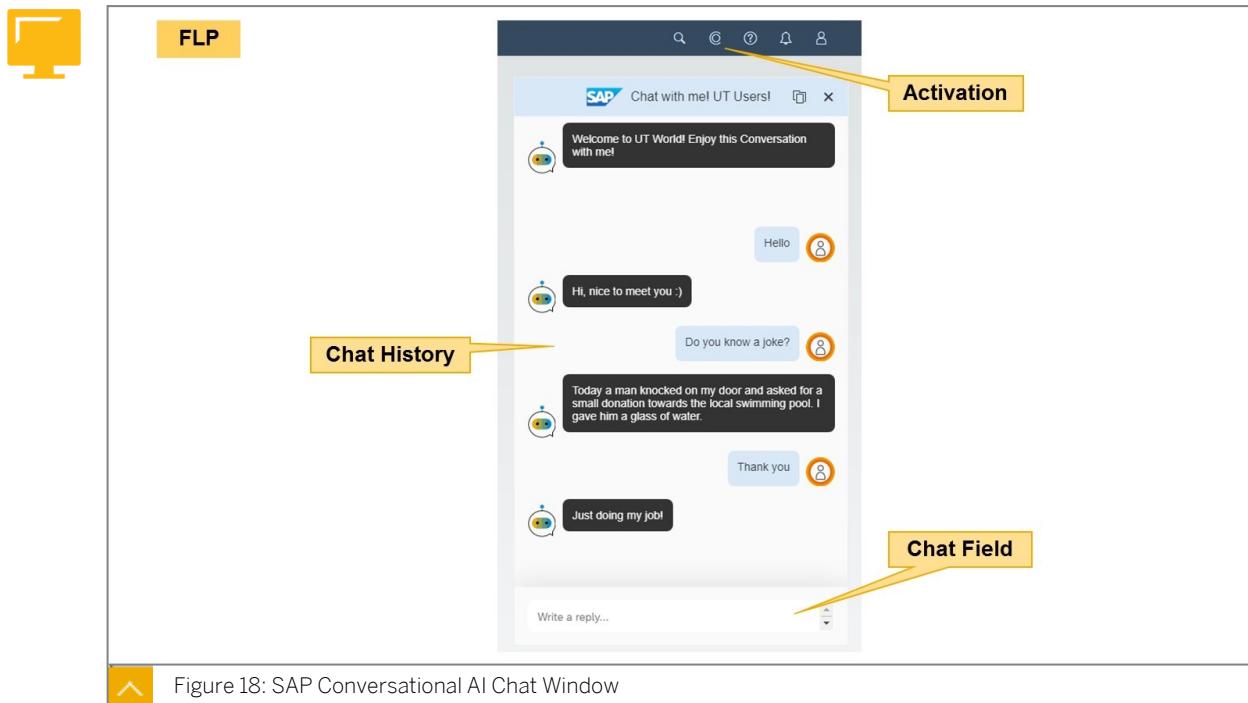
Features



The features in the *User Actions Menu* 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 area 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.



The integration of the SAP Conversational Artificial Intelligence (CAI) enables the use of digital assistants to interact with the SAP system. CAI is a cloud solution and provides everything around building, connecting, managing, and running so-called bots. CAI is part of the Conversational User Experience (CUX) for interacting with the intelligent enterprise.

Accessibility

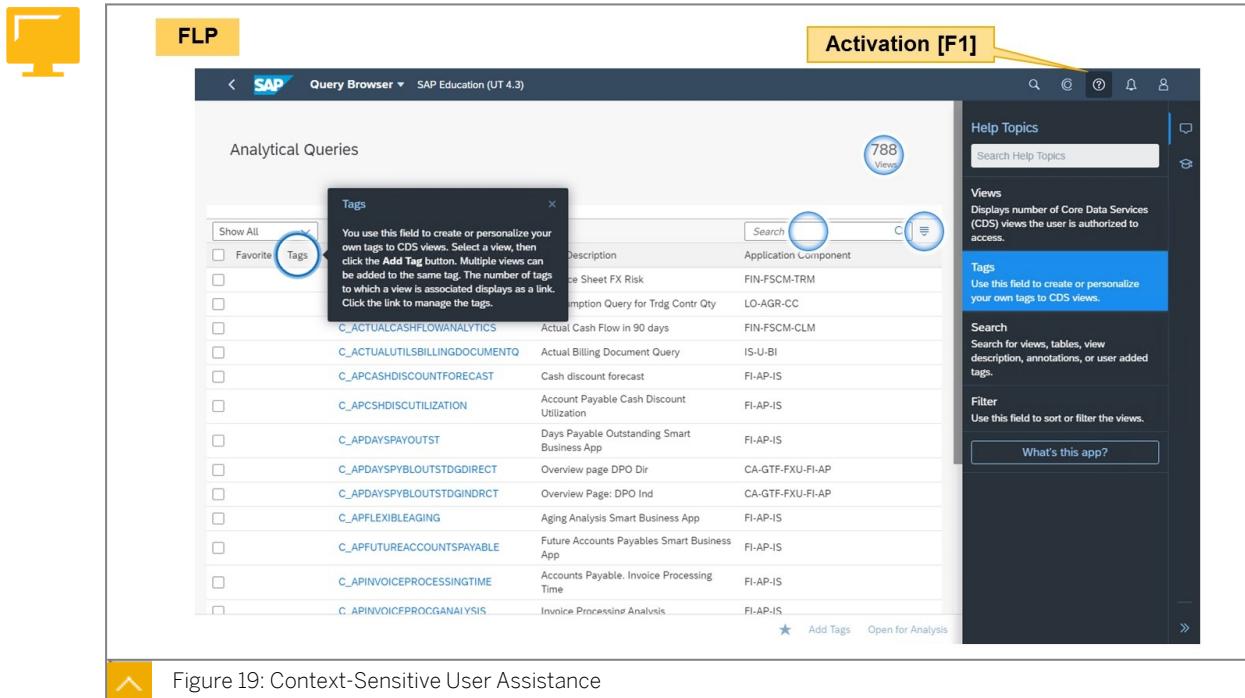


Figure 19: Context-Sensitive User Assistance

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 help topics shows up to the right of the app as well as bubbles pointing out the key elements inside the app. By selecting a topic or a 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.

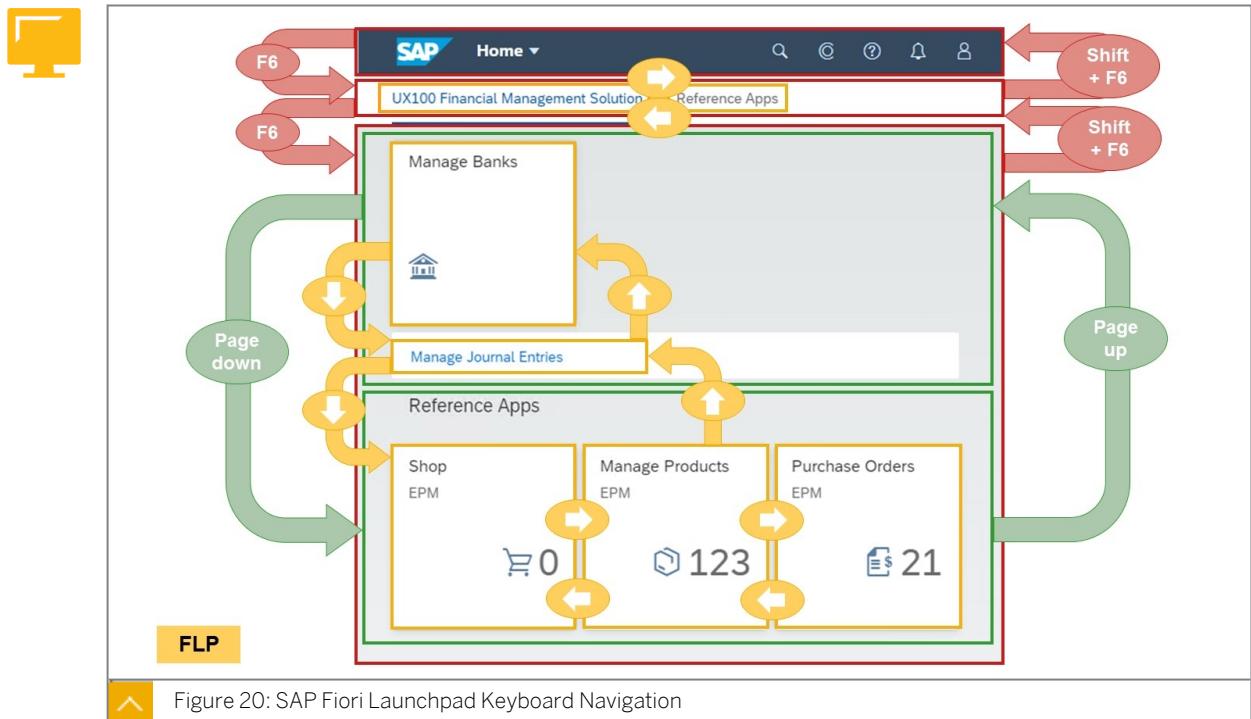


Figure 20: SAP Fiori Launchpad Keyboard Navigation

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. You can also use TAB to cycle through all elements.

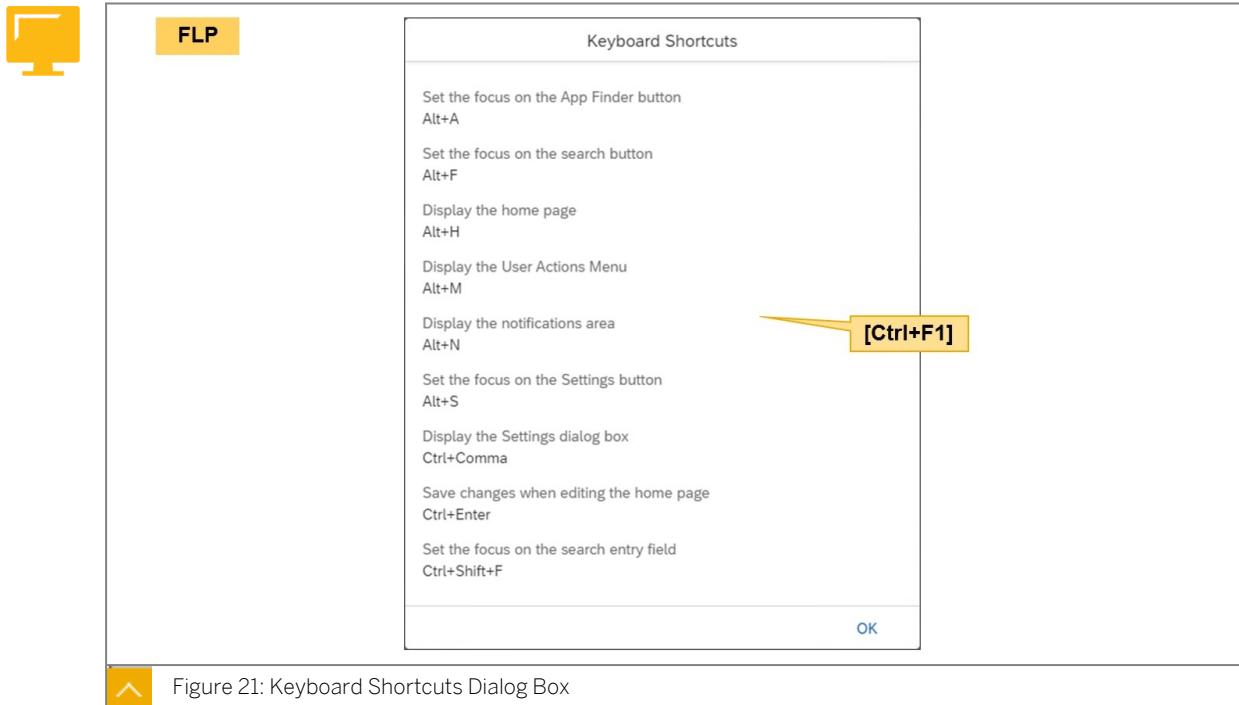
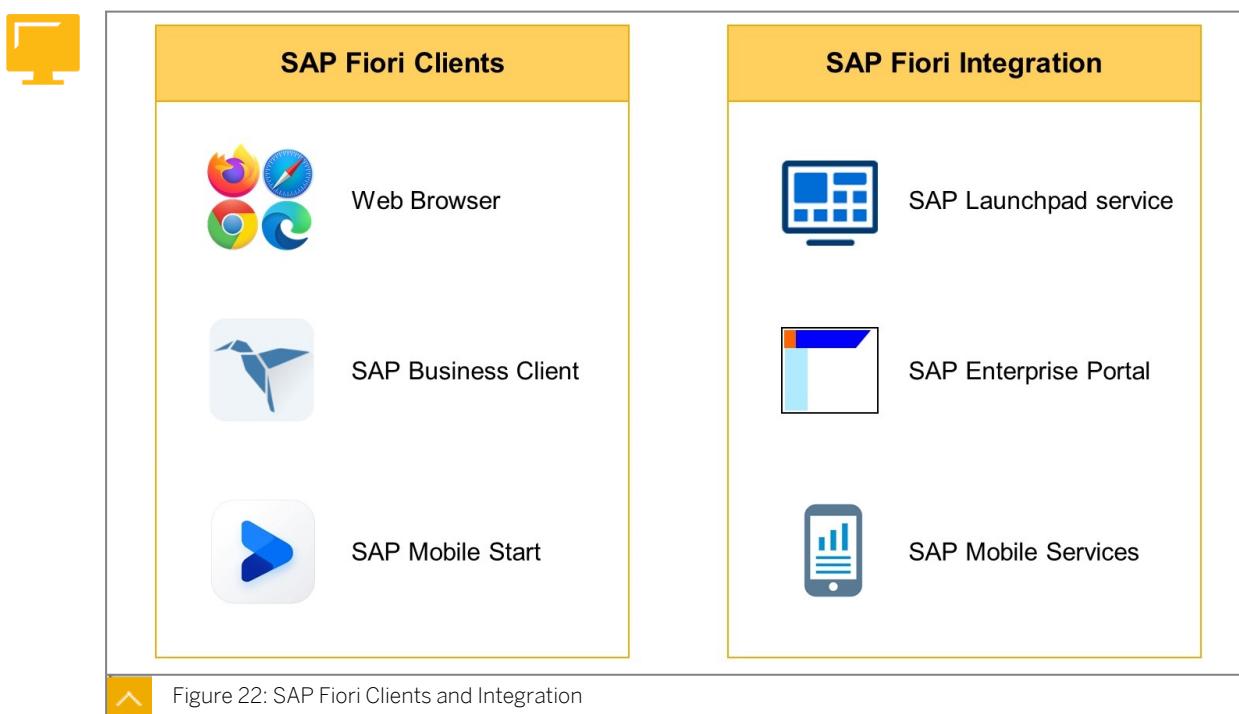


Figure 21: Keyboard Shortcuts Dialog Box

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 high contrast themes recommended for users who are visually impaired. The High-Contrast Black and High-Contrast White themes for SAPUI5 are predefined themes that are 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 SAP Fiori:

Web Browser

Any HTML5-ready browser can be used.

SAP Business Client

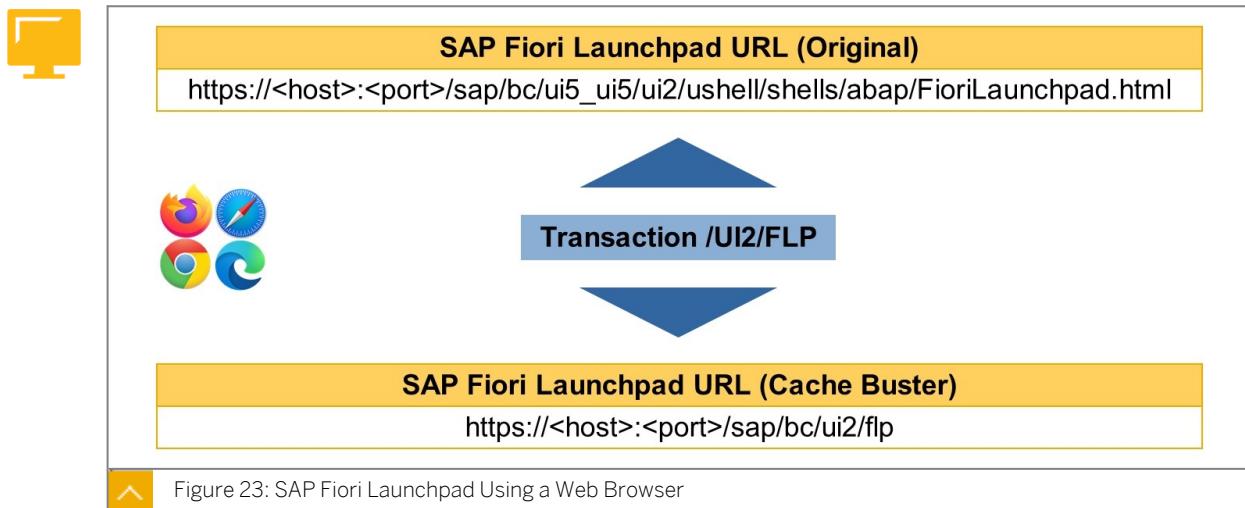
As of SAP Business Client 6.0, you can create SAP Fiori launchpad system connections.

SAP Mobile Start

You can access SAP Launchpad service web sites providing apps based on cloud and on-premise SAP solutions.

You can also integrate SAP Fiori in other system areas to add additional values like logon features or increased distribution range. Details and benefits of these integration options are discussed later or considered in more depth in other trainings:

- SAP Launchpad service
- SAP Enterprise Portal
- SAP Mobile Services



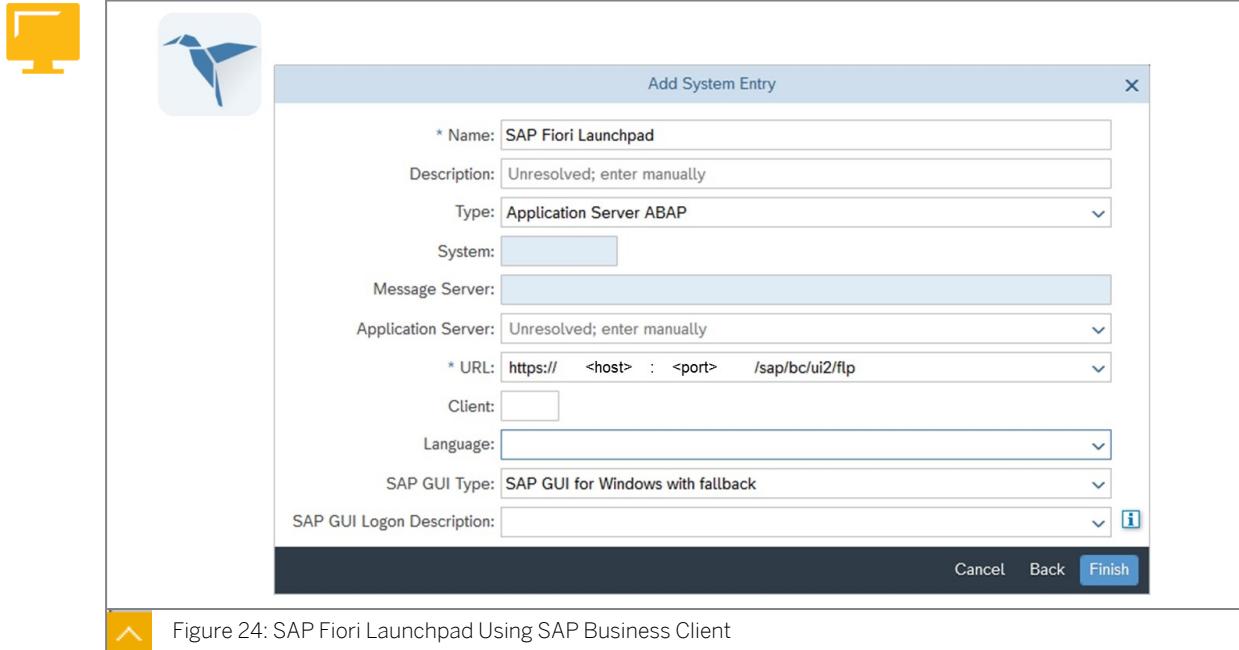
The original way to start an FLP is to enter the URL https://<host>:<port>/sap/bc/ui5_ui5/ui2/ushell/shells/abap/FioriLaunchpad.html in an HTML5-ready browser. This URL has been available since the beginning of SAP Fiori. Over time, SAP added additional options to start the launchpad.

Many customers access SAP software via a logon to an Application Server ABAP (AS ABAP). Therefore, the transaction `/UI2/FLP` is added. With this transaction, you can log on automatically with the credentials you used to log on to the AS ABAP. This approach is for users working in both worlds, ABAP transactions and SAP Fiori web apps, if no business client is available.

Note:

By default, transaction `/UI2/FLP` starts the FLP via the Internet Communication Manager (ICM) process of your application server instance. An entry in the database table `HTTPURLLOC` can be used to call a reverse proxy, such as SAP Web Dispatcher.

The current URL to start the FLP is <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.



SAP Business Client makes it possible to access SAP GUI and web applications in one client software. Therefore, in *SAP Business Client 6.0*, the ability to add system connections for FLP was introduced. The benefit is 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)*.

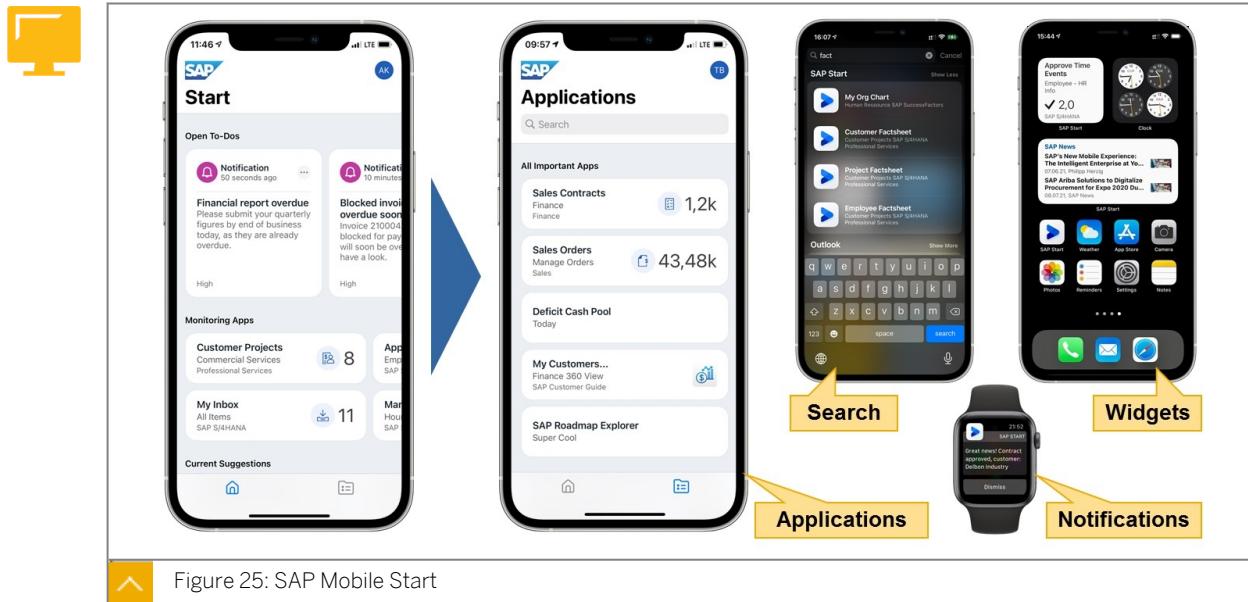


Figure 25: SAP Mobile Start

SAP Mobile Start was introduced in August 2021. It is a mobile application for Apple iOS (Google Android planned for Q1/2022) integrated with SAP S/4HANA and further SAP solutions, leveraging the SAP Business Technology Platform (BTP). The SAP Launchpad service, Notification service, and Mobile Services of BTP work together to provide web sites and apps for mobile usage. SAP Mobile Start integrates with mobile operating system features like notifications, spotlight search, and widgets running on smartphones, watches, and tablets.



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

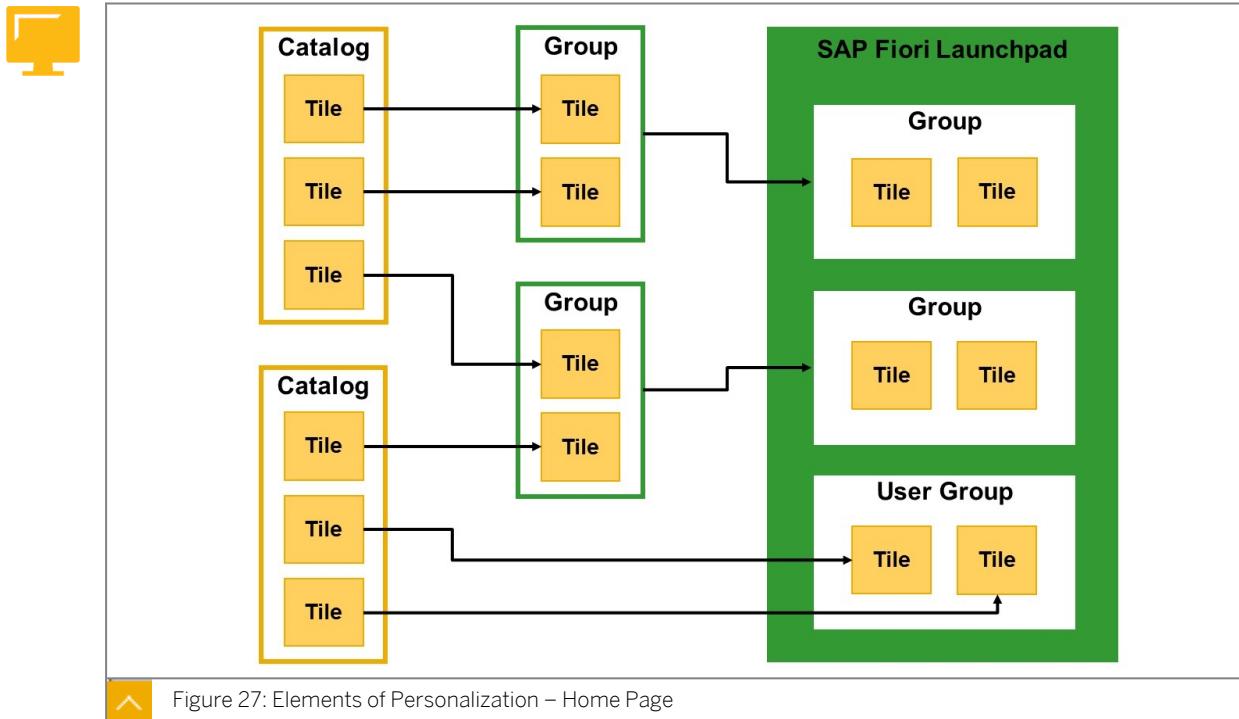


The screenshot shows the SAP Fiori Launchpad Settings interface. On the left is a sidebar with options: User Account (TRAIN-00), Appearance (SAP Quartz Light), Home Page, Spaces, User Activities, Language & Region (selected, showing EN | Time Format: 12h), Default Values, and Notifications. On the right is the 'Language & Region' section with dropdowns for Language and Region (set to English), Date Format (dd.MM.yyyy), and Time Format (12 h). A note says 'After you save your settings, the browser will refresh for the new language to take effect.' At the bottom are Save and Cancel buttons. In the top right corner, there's a yellow box labeled 'FLP'. In the bottom right corner, a yellow arrow points to the 'User TRAIN-00' section of the 'User Actions Menu' (UAM) which includes Recent Activities, Frequently Used, App Finder, Settings, Edit Home Page, About, and Sign Out.

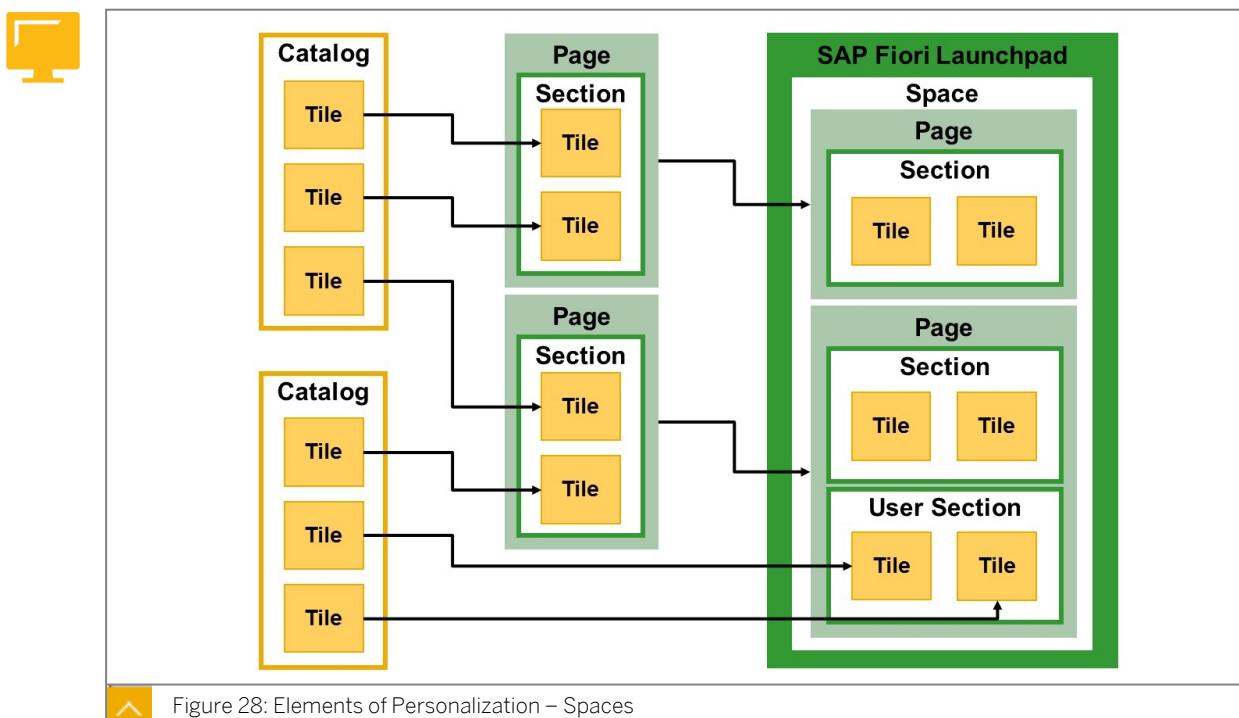
Figure 26: SAP Fiori Launchpad Settings

End users can personalize their own variant of the *SAP Fiori launchpad (FLP)*. The settings can be accessed via the *User Actions Menu*. You can 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
- Home page and language settings
- Activation of user profiling
- Maintenance of default values
- Appearance and behavior of notifications



In the home page concept, 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.



Introduced in SAP S/4HANA 2020, the spaces concept can be used as alternative to groups in the home page. A space is visualized as a ribbon or tab at the top of the FLP and defines a frame for one or more pages. A page consists of sections showing tiles in the same way as groups have done before. Spaces and pages are defined centrally in the system, but only

spaces are added to the launchpad including the pages assigned to the space. Sections are an integral part of pages and can also be created by the user in their launchpad.

💡

Hint:

You can activate spaces in the *SAP Fiori launchpad* settings under *Spaces*.

Figure 29: SAP Fiori Launchpad Personalization

To enter the action mode for personalization, a user must choose *Edit Home Page* in the *User Actions Menu* of the FLP. In this mode, tiles can be removed from existing groups or sections and new groups or sections can be created and rearranged. When adding a new tile to a group or section in the action mode or choosing *App Finder* in the *User Actions Menu*, the app finder is shown. Here, the user can choose tiles from all catalogs assigned to their user role.

💡

Hint:

You can transport home page and application personalization data to another system. The `/UIF/MIGRATE_FES_PERSO` report allows you to collect the personalization data and writes these to transport requests. For more information, please read SAP note [2789848](#).

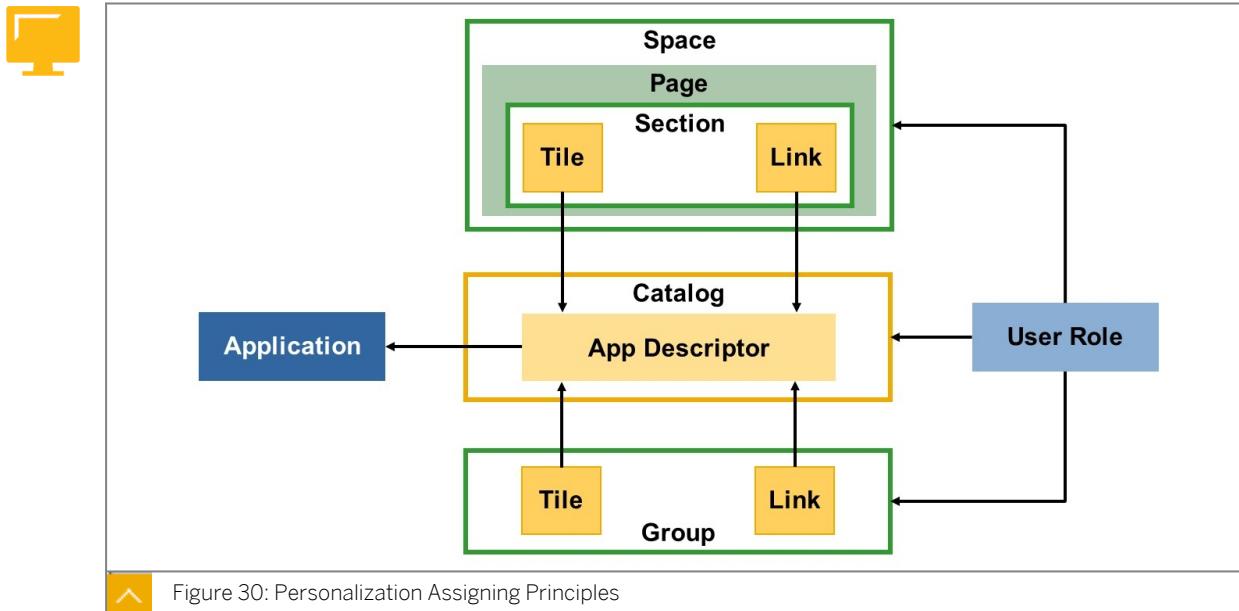


Figure 30: Personalization Assigning Principles

An app descriptor connects the FLP with the app implementation in the system. This includes starting the correct app with the correct parameters and the information shown on the tile. App descriptors are created in catalogs that collect all apps of one solution area. App descriptors and catalogs are delivered by SAP. However, they can also be created by customers.

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

App descriptors can also be referenced by pages and then visualized as tiles or links when assigning the page to a space. Spaces and pages do not add any additional settings and have no direct connection to the apps. Both catalogs and spaces can be assigned to user roles, but not pages. This works similar to catalogs and groups.

SAP also delivers template catalogs, groups, spaces, 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

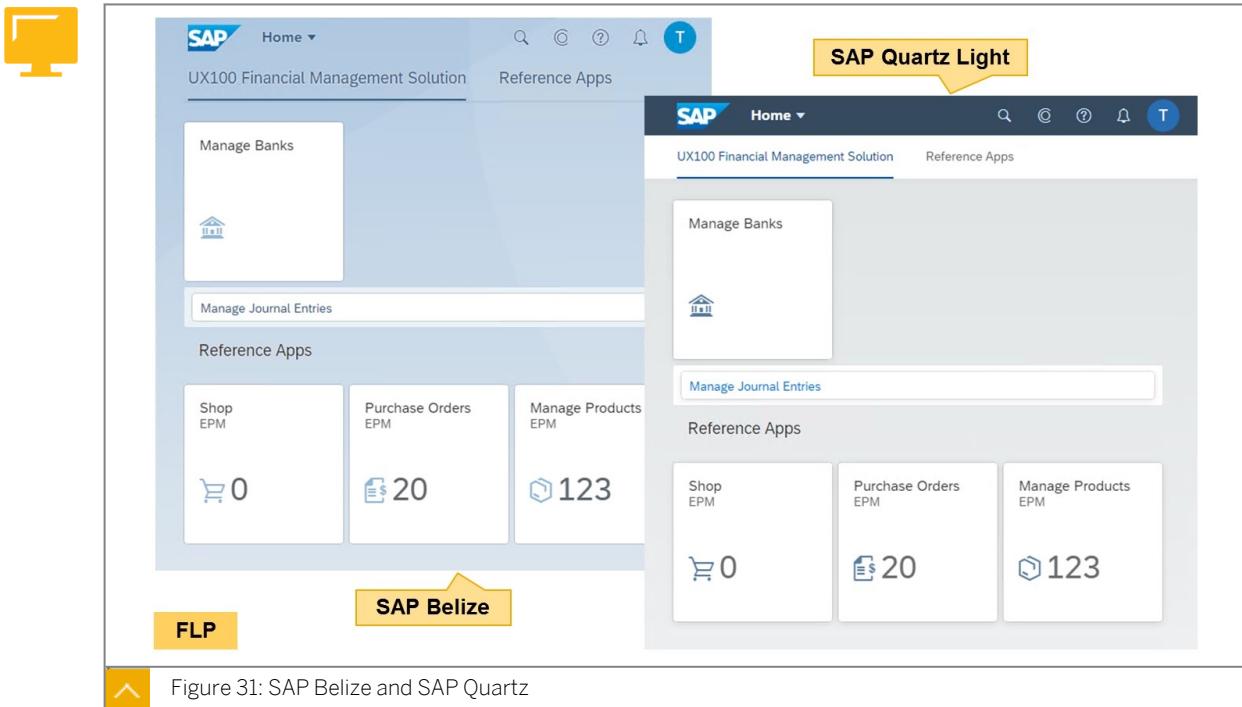


Figure 31: SAP Belize and SAP Quartz

The SAP Fiori themes SAP Belize (Deep) (introduced with SAP Fiori 2.0) and SAP Quartz Light/Dark (introduced with SAP Fiori 3) are not only a design for HTML-apps like SAPUI5, but also for applications running in SAP GUI. Beside changing colors and font, using SAP Belize or SAP Quartz in SAP GUI does also change the structure of the UI.

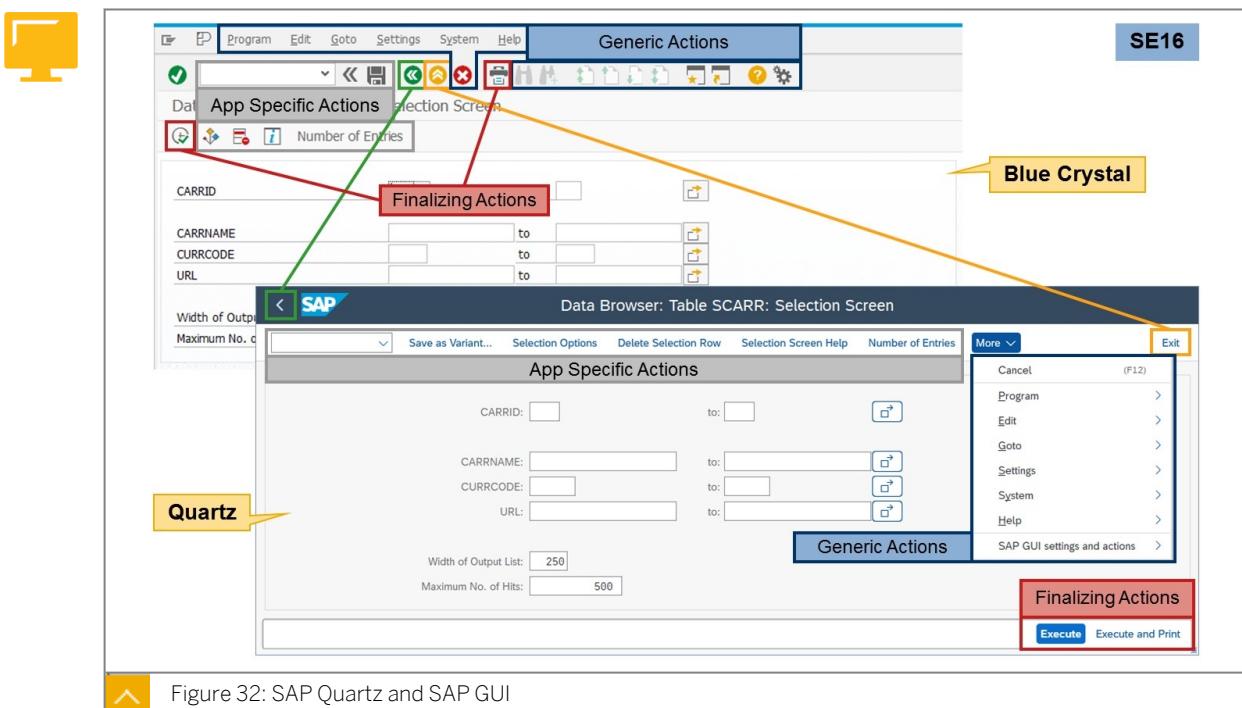


Figure 32: SAP Quartz and SAP GUI

Well-known functions such as **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 must be in the upper left corner. This all works out-of-the-box by using the SAP Belize or SAP Quartz theme with SAP GUI.

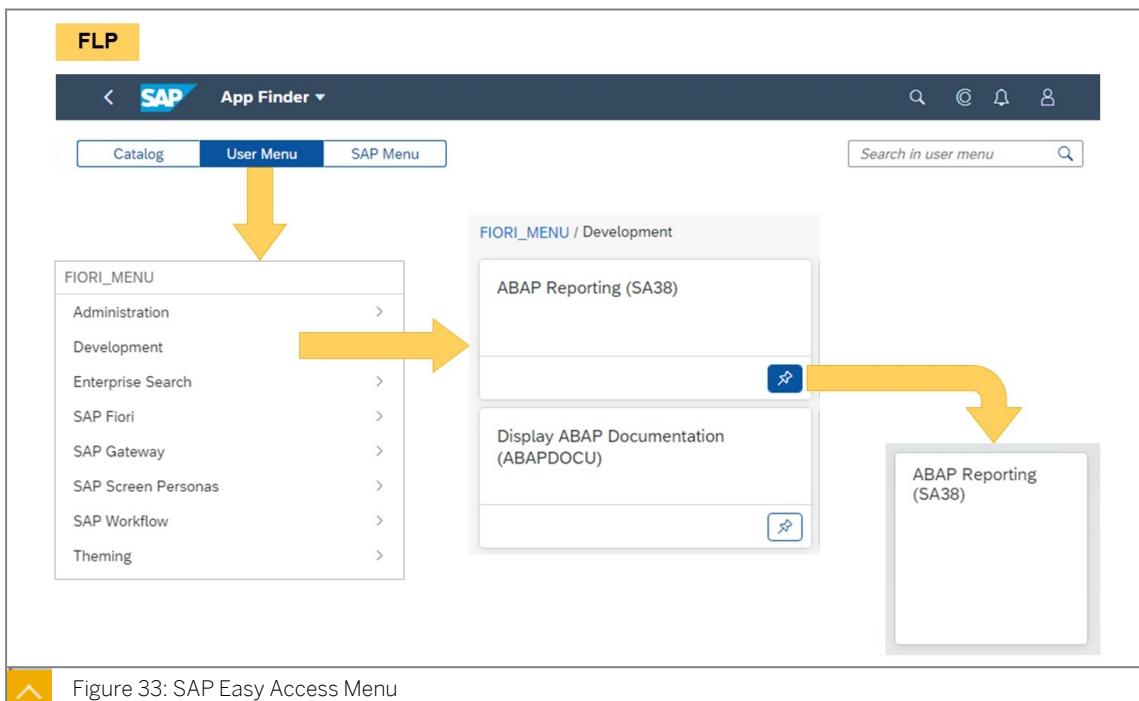
SAP Belize is available as of SAP GUI for HTML with SAP Kernel 7.49 and SAP GUI for Windows 7.50, but only if the user connects to an SAP S/4HANA 1610 or a newer SAP S/4HANA release. With SAP GUI for Windows 7.60, SAP Belize is available for all SAP products.

SAP Quartz is available as of SAP GUI for HTML with SAP S/4HANA 1909 and SAP GUI for Windows 7.70. A full documentation of all designs for SAP GUI and their prerequisites is available in SAP note [710719](#).



Hint:

In SAP GUI for Windows 7.70 it is also possible to replace *Microsoft Internet Explorer* with *Microsoft Edge* as the default HTML-control. For more information about this topic read SAP note [2913405](#)



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 must be done by an administrator. However, 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

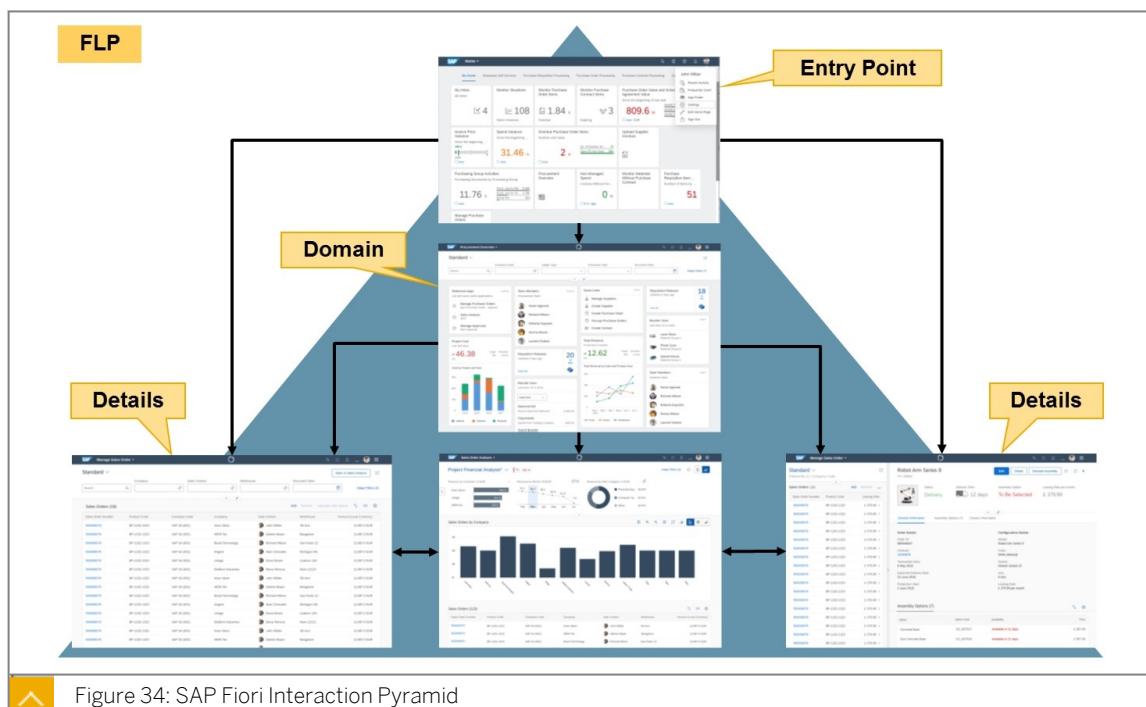


LESSON OBJECTIVES

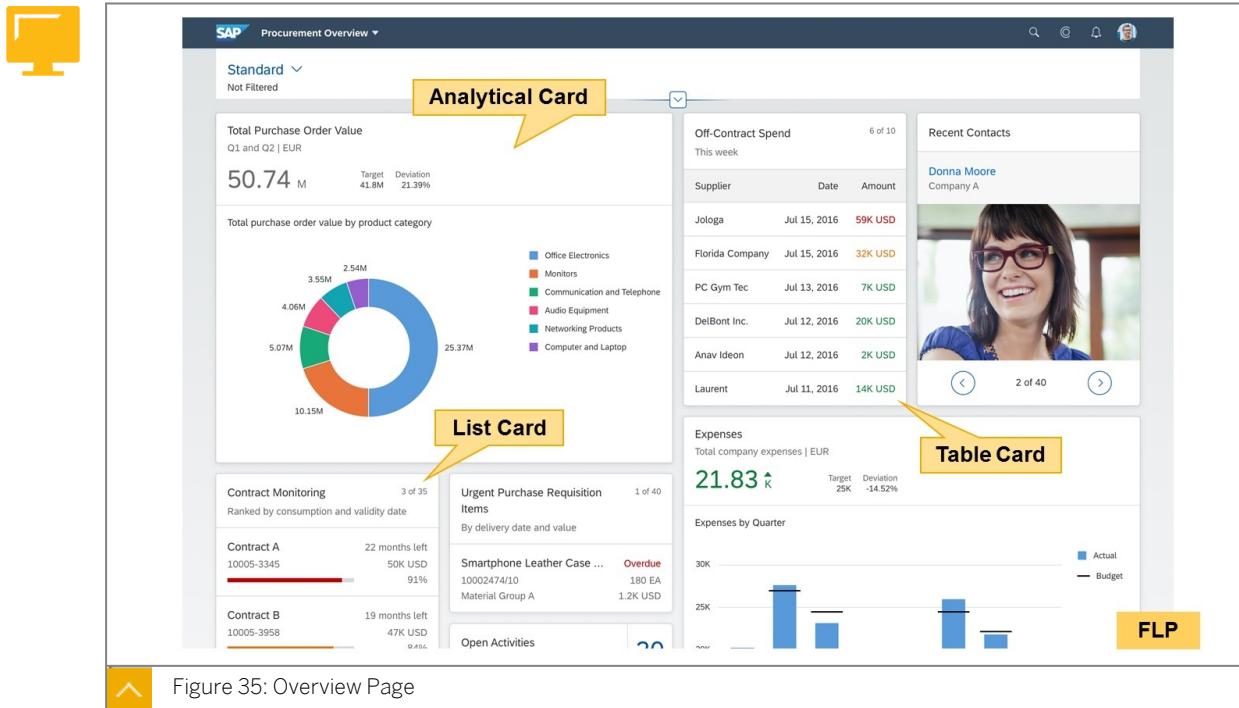
After completing this lesson, you will be able to:

- Explore SAP Fiori data handling

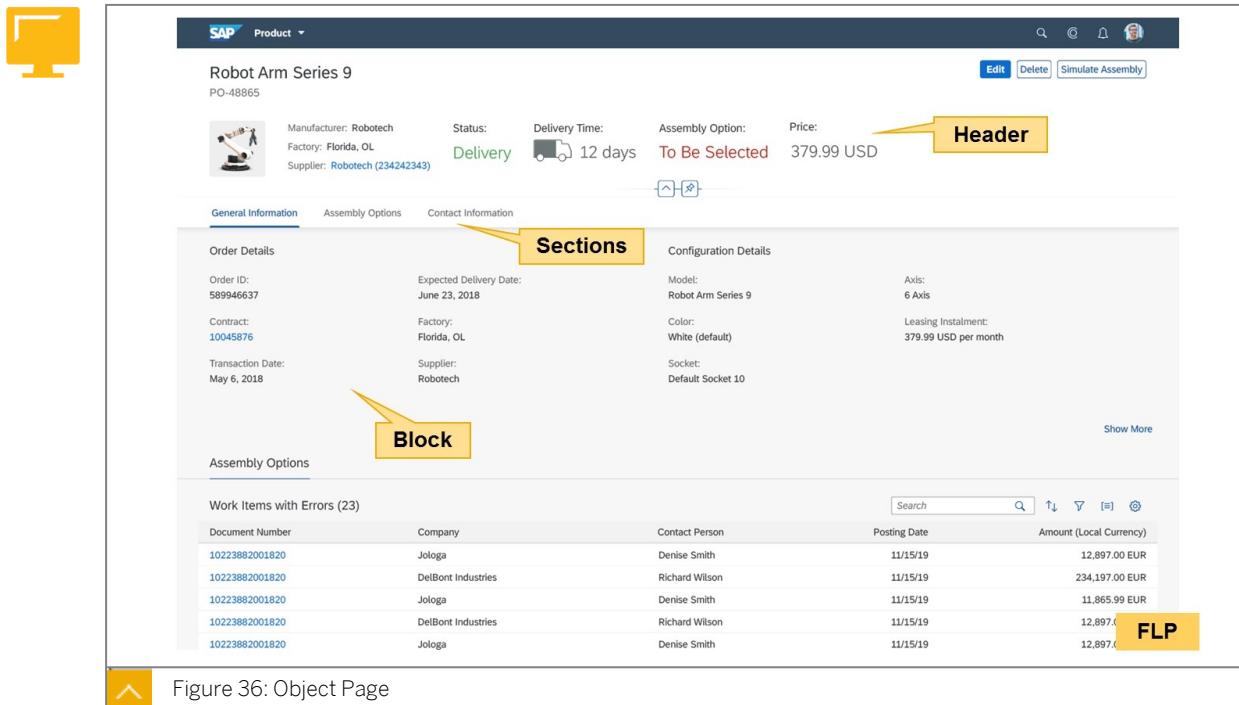
Interaction Floorplan



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.



The overview page (OVP) is a data-driven SAP Fiori app type and floorplan. It provides all the information a user needs in a single page, based on the specific domain or role of the user. It enables 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 enable the visualization of information in an attractive and efficient way.



The object page enables 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, similar to the

overview page, is based on SAP Fiori elements technology and uses an annotated view of app data.

SAP Fiori Search

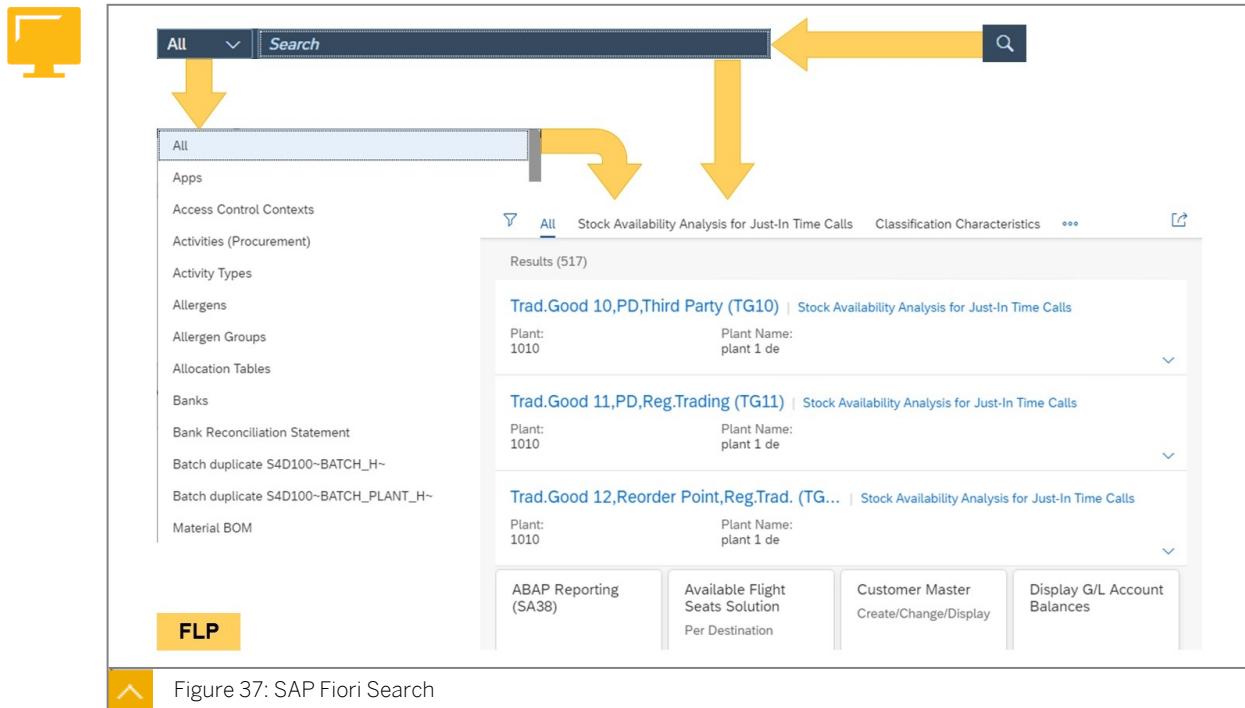


Figure 37: SAP Fiori Search

SAP Fiori offers a powerful search to find contents in the SAP Fiori launchpad (FLP). Using any database, you can search for text fragments of SAP 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.

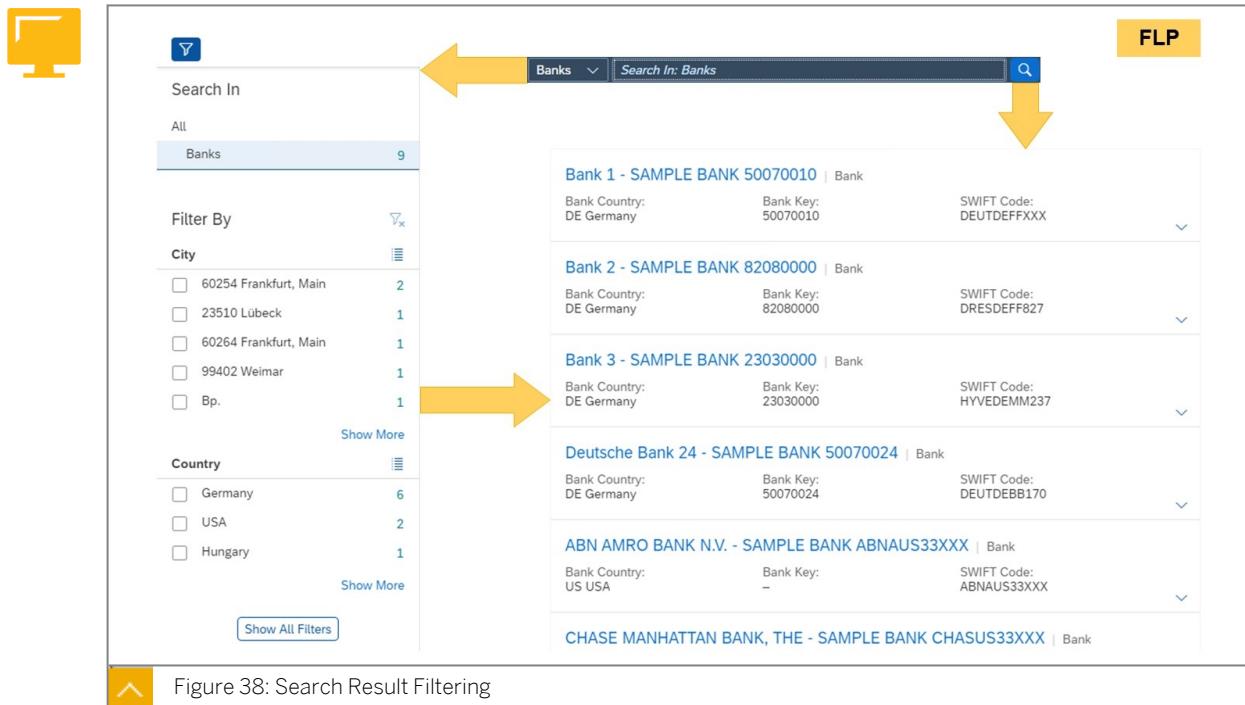
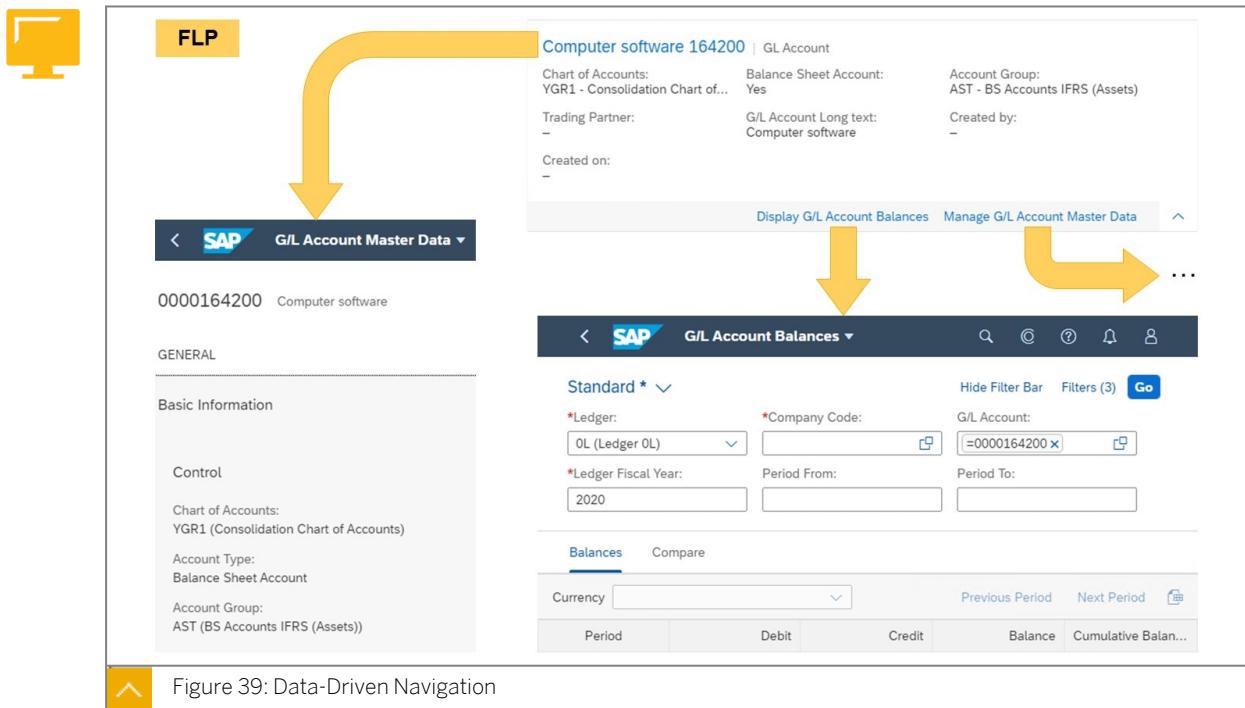


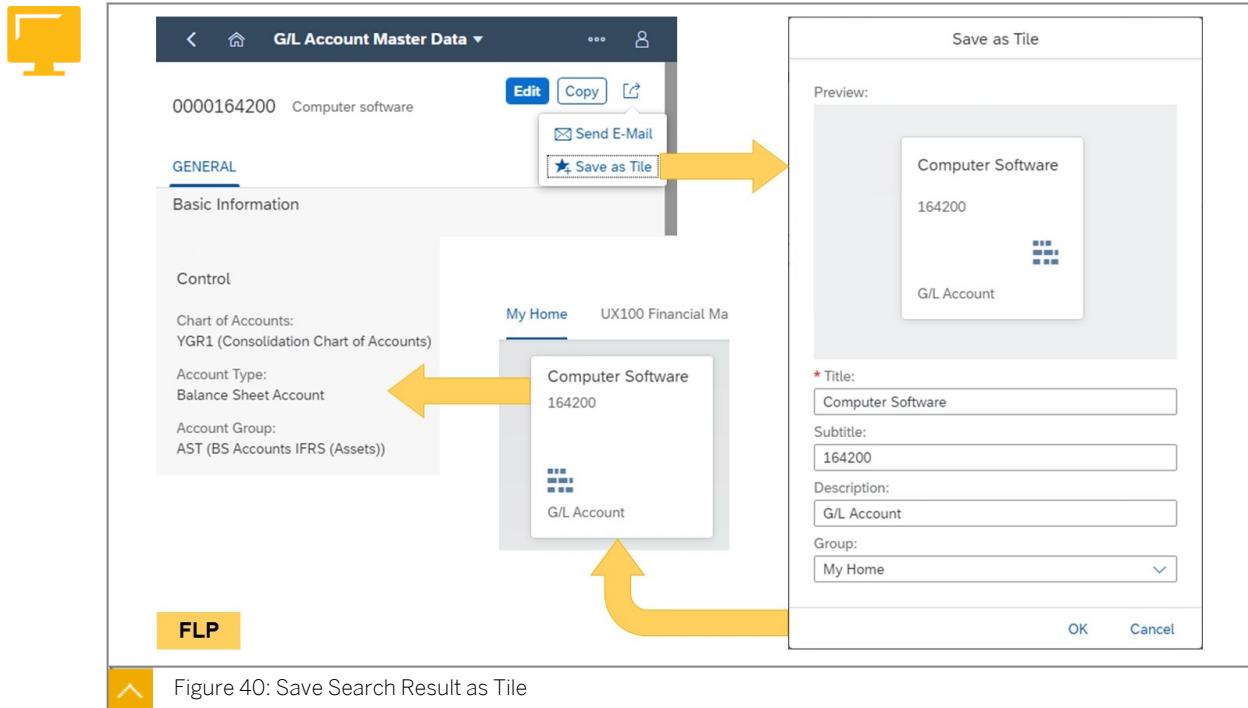
Figure 38: Search Result Filtering

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 need not be visible in the FLP. You can 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

Learning Assessment

1. Which dimensions define SAP Fiori?

Choose the correct answers.

- A Concept
- B Design
- C Performance
- D 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

3. Which clients can be used for SAP Fiori?

Choose the correct answers.

- A Web Browser
- B SAP Logon
- C SAP Easy Access
- D 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 Launchpad service
- E SAP Mobile Services

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

Choose the correct answers.

- A SAP Fiori group
- B SAP Fiori tile
- C SAP Fiori catalog
- D SAP Fiori space
- E SAP Fiori page

6. Which element has the application as target?

Choose the correct answer.

- A SAP Fiori user role
- B SAP Fiori catalog
- C SAP Fiori tile
- D SAP Fiori app descriptor

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

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 Easy Access
- D SAP Business Client

Correct. The following clients can be used in SAP Fiori: Web Browser 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 Launchpad service
- E SAP Mobile Services

Correct. The following platforms support the integration of SAP Fiori: SAP Enterprise Portal, SAP Launchpad service, and SAP Mobile Services.

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

Choose the correct answers.

- A SAP Fiori group
- B SAP Fiori tile
- C SAP Fiori catalog
- D SAP Fiori space
- E SAP Fiori page

Correct. The following personalization elements can be assigned to user roles: SAP Fiori group, space, and catalog.

6. Which element has the application as target?

Choose the correct answer.

- A SAP Fiori user role
- B SAP Fiori catalog
- C SAP Fiori tile
- D SAP Fiori app descriptor

Correct. The personalization element SAP Fiori app descriptor has the application as target.

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

41

Lesson 2

Explaining Data Services

49

Lesson 3

Explaining Application Types

57

UNIT OBJECTIVES

- Explain user interfaces
- Explain data services
- Explain Application Types
- Using SAP Fiori App Recommendations

Explaining User Interfaces

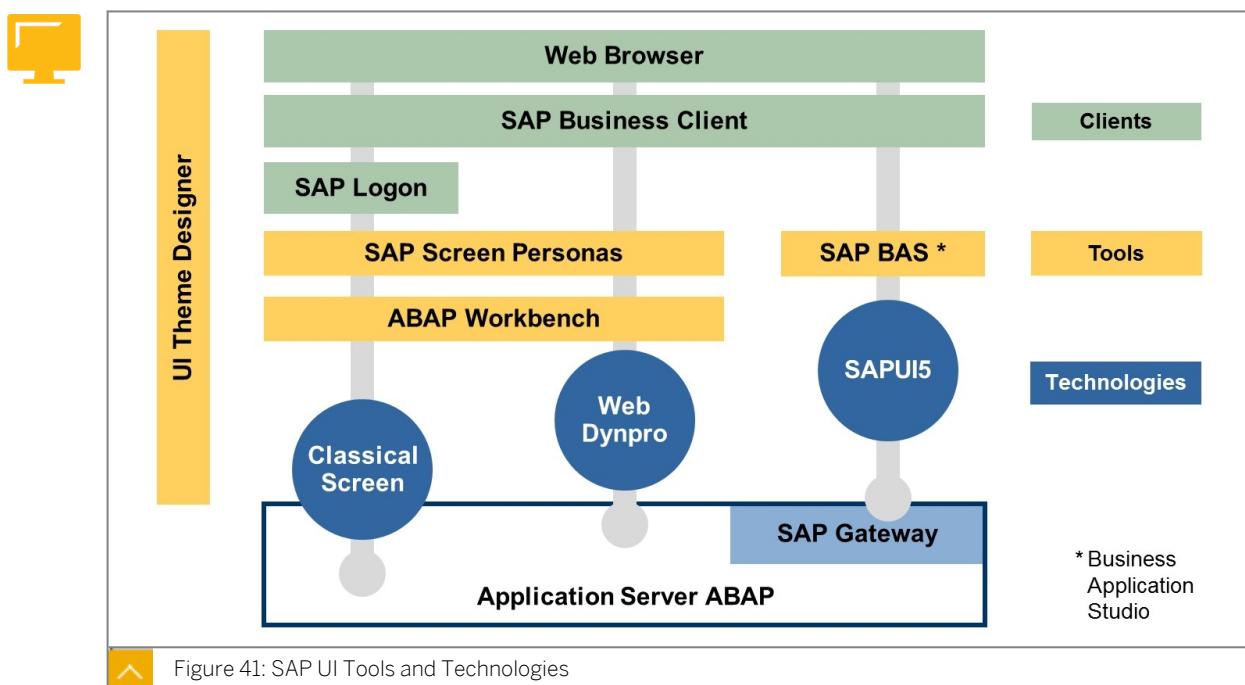


LESSON OBJECTIVES

After completing this lesson, you will be able to:

- Explain user interfaces

UI 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*, you can also 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 *Control Framework (CFW)*)
- *Web Dynpro* (including *Floorplan Manager (FPM)*)

SAP Business Application Studio

- *SAPUI5* (including SAP 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*. When it comes to the design of user interface, one tool handles all technologies: the *UI Theme Designer*.

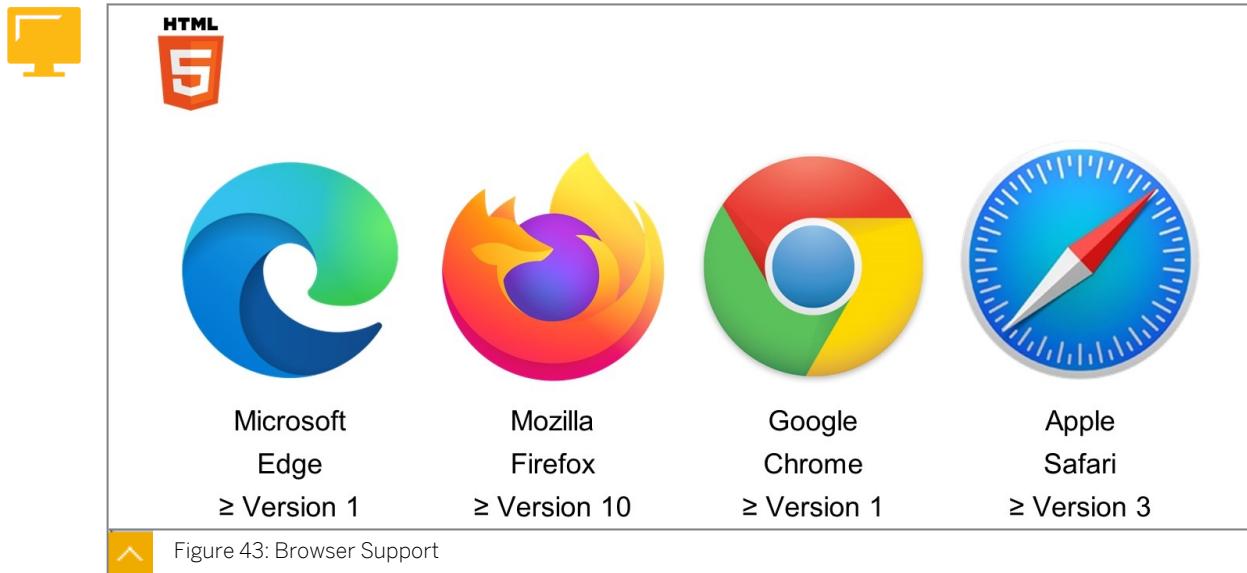


The majority of SAP Fiori apps are web apps built using SAPUI5 as UI technology. SAPUI5 is based on HTML5 and can be consumed on every device using a browser. The recommended development environment for SAPUI5 is the *SAP Business Application Studio*.

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>). You can also 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. CAI enables 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

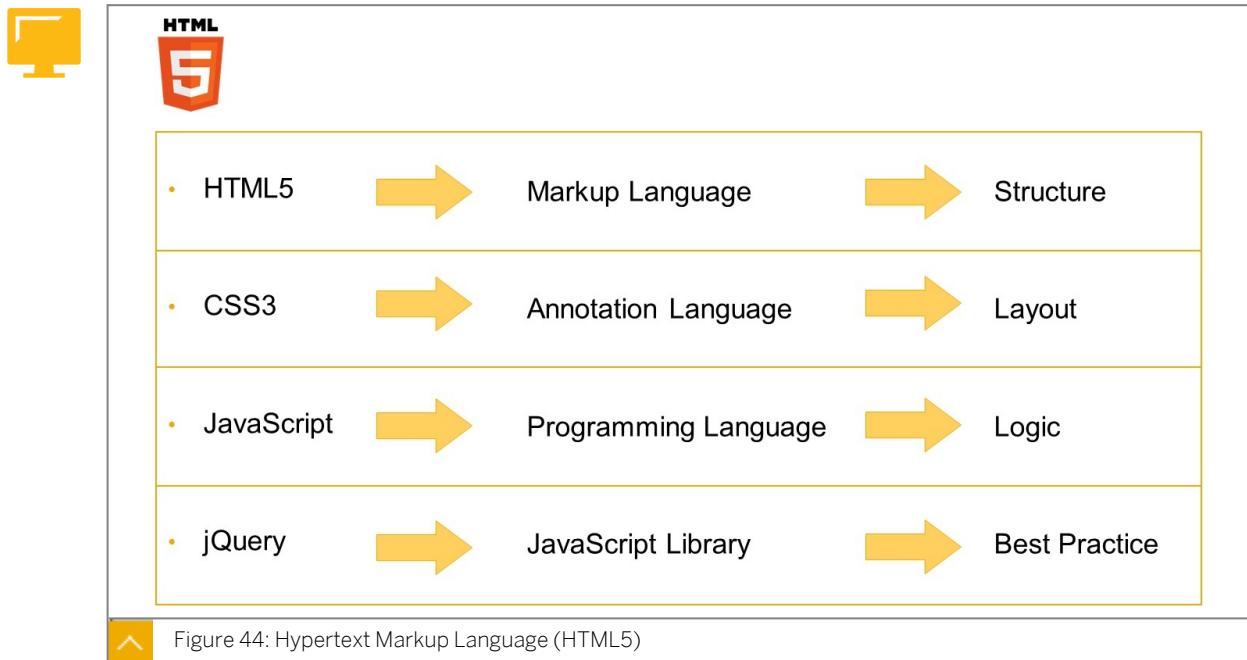


With SAPUI5 as a basis for web apps, browsers need a certain minimum release to process all code elements used in HTML5 and JavaScript. Google Chrome and Microsoft Edge both support HTML5 from the beginning. Mozilla Firefox and Apple Safari were updated to support it.



Note:

Although Microsoft Internet Explorer has officially supported HTML5 since version 8, it is recommended to switch to Microsoft Edge. Microsoft and SAP announced the end of support for Internet Explorer. For more information about this topic, please read SAP note [1672817](#).



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 code 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.



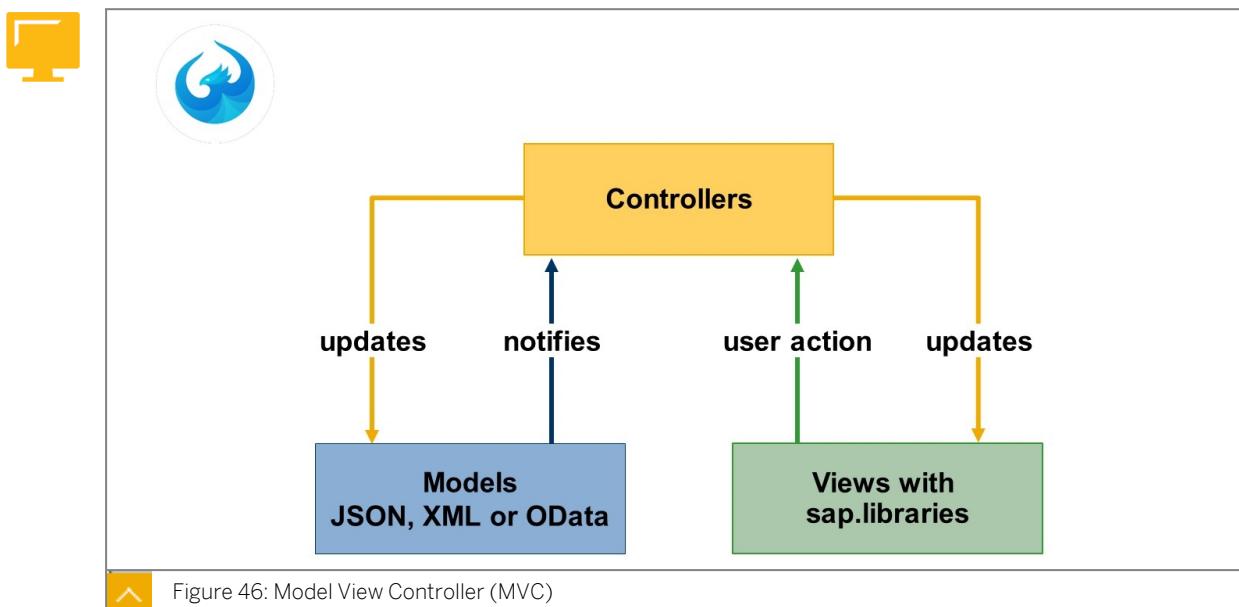
[!\[\]\(d72719a91e21a20aec23a11379fae991_img.jpg\)](https://ui5.sap.com)

- 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 45: 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, 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 user interface (UI) controls automatically adapt themselves to the capabilities of each device.



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 SAP 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.

SAPUI5 Versions Maintenance Status

This updated SAPUI5 maintenance strategy for SAP Business Technology Platform is available in [this document](#).

Version	Maintenance Status	End of Maintenance	End of Cloud Provisioning	SAP_UI version*	Front-end Server*
1.93 (2108)	Maintenance	Q1/2022		Cloud only	Cloud only
1.90 (2105)	Maintenance	Q1/2022	Q1/2023	7.56	
1.87 (2102)	Maintenance	Q3/2021	Q3/2022	Cloud only	Cloud only
1.84 (2011)	Maintenance	Long-term Maintenance, Q4/2023	Q4/2024	7.55	SAP FIORI FES 2020 FOR S/4HANA (SAP Note 2919182)

<https://sapui5.hana.ondemand.com/versionoverview.html>

This Application provides a ready-to-consume version of SAPUI5. The SAPUI5 sap-ui-core.js bootstrap file is available at: /sap/public/bc/ui5/_ui5/re

The following libraries are available:

Library	Version:1.84 (1.84.14)
com.sap.apf.asp-lib	1.84.2
com.sap.ca.scfld.md	1.84.3
com.sap.ca.ul	1.84.3
com.sap.collaboration.collaboration-commons	1.84.1
com.sap.gantt.gantt.lib	1.84.8
com.sap.hana.fileprocessing.sap.fileviewer	1.84.1
com.sap.ip.bi.zen.rt.components.crosstab.phx	1.84.11
com.sap.ip.bi.zen.rt.rsh.php	1.84.11
com.sap.ip.bi.zen.rt.zencopy.phx	1.84.11
com.sap.ipext.landviz.sap.landviz	1.84.1
com.sap.ovp.ovp-lib	1.84.6
com.sap.rules.sap.rules.ui	1.84.3
com.sap.suite.suite-ui-commons	1.84.7
com.sap.suite.suite-ui-microchart	1.84.13

/sap/public/bc/ui5/_ui5/index.html

Technical Information Dialog

SAPUI5-on-ABAP:
1.84.14 (built at 27.05.2021 13:32:00)

OpenUI5 Version:
1.84.13 (built at 26.05.2021 09:22:00)

User Agent:
Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML...)

App URL:
<https://s4dhost.wdf.sap.corp:44310/sap/bc/ui2/fp#Shell-home>

Use Debug Sources:
 Select specific modules

Activate Test Recorder Activate Support Assistant Close

Ctrl+Alt+Shift/Option+P

Figure 47: SAPUI5 Version Overview

SAPUI5 provides updates on a regular basis through maintenance and innovation versions. An innovation version is only maintained until the next version of SAPUI5 is released. Maintenance versions have an extended maintenance period in which SAP still provides patches even though a higher version is already available. The digits have the following meaning:

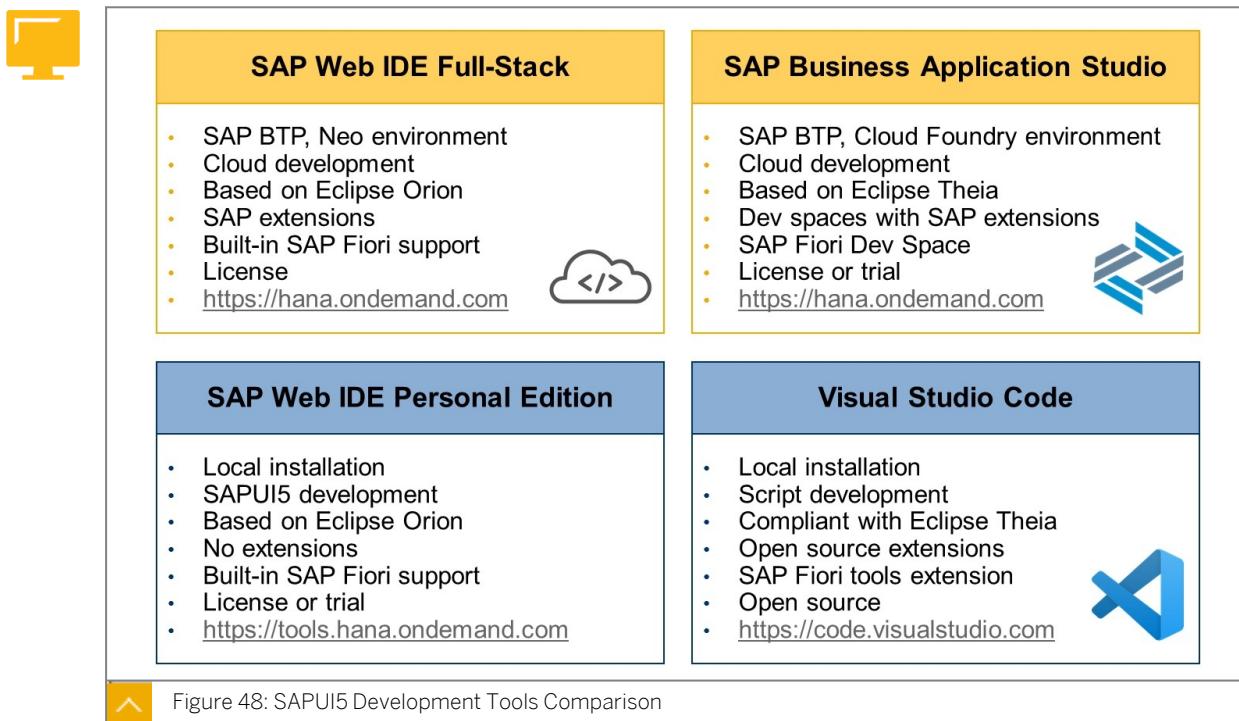
- The first digit specifies the release number (major version).
- The second digit specifies the version number (minor version).
- The third digit specifies the patch number.

A full list of all SAPUI5 versions including their end of maintenance can be found under <https://sapui5.hana.ondemand.com/versionoverview.html>.

Which SAPUI5 versions are available on an Application Server ABAP is provided via `https://<host>:<port>/sap/public/bc/ui5_ui5/index.html`.

The version a running app is using is visible in the technical information dialog opened via `CTRL+ALT+SHIFT+P` on Microsoft Windows or via `CTRL+ALT+OPTION+P` on Apple Mac.

SAPUI5 Development Tools



Since 2014, SAP provides the *SAP Web Integrated Development Environment (IDE)* based on Orion, an open source project by Eclipse. It is available in three flavors:

SAP Web IDE Full Stack

Development environment on SAP Business Technology Platform (BTP), Neo environment supporting for example SAP Fiori, SAP S/4HANA Cloud, SAP HANA, SAP Mobile Services including programming languages such as SAPUI5, Java, and Node.js.

SAP Web IDE for SAP HANA

Native development environment for SAP HANA provided by SAP HANA Extended Services Advanced including programming languages such as Java and Node.js.

SAP Web IDE Personal Edition

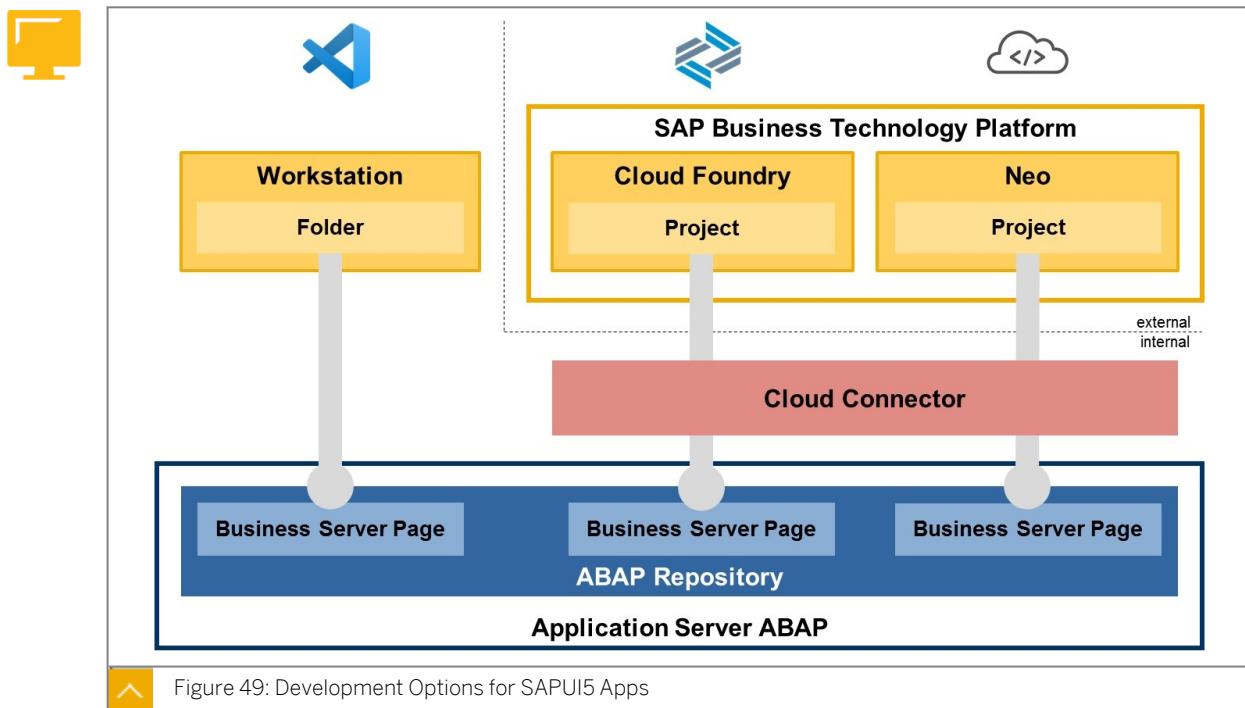
Local development environment for SAP Fiori and SAPUI5, which can be downloaded on <https://tools.hana.ondemand.com/>.

SAP Web IDE for SAP HANA is the only IDE not supporting SAPUI5. This leaves one IDE for developing SAPUI5 provided in the SAP Business Technology Platform (BTP) and one provided for on-premise. In developing SAP Fiori apps, the features mainly differ by release. One important difference is the missing support for extensions in the *SAP Web IDE Personal Edition*.

Since 2020, SAP provides the *SAP Business Application Studio (BAS)* based on Theia, an open source project by Eclipse. It is the next generation of *SAP Web IDE* provided in the

SAP BTP, Cloud Foundry environment. It provides pre-configured environments, so-called *Dev Spaces*, with pre-installed runtimes and tools tailored for key scenarios. These are based on extensions such as the SAP Fiori tools extension.

Another tool where the SAP Fiori tools extension can be included is Visual Studio (VS) Code. It is developed by Microsoft as an open source IDE following the same rules as Theia. Although not based on Theia, it is fully compatible. SAP offers the SAP Fiori tools beside others free of charge in the extensions marketplace of VS Code. VS Code can be downloaded under <https://code.visualstudio.com/>.



Developing an SAP Fiori web app means developing SAPUI5 using the built-in support in SAP Web IDE or the SAP Fiori tools extension in BAS or VS Code. A project in the SAP Web IDE can connect and, therefore, represents an SAP 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 Cloud Connector is a standalone software that is available free of charge to connect the services in the SAP BTP 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 BTP. It is not needed for SAP Web IDE Personal Edition or VS Code. The Cloud Connector is 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

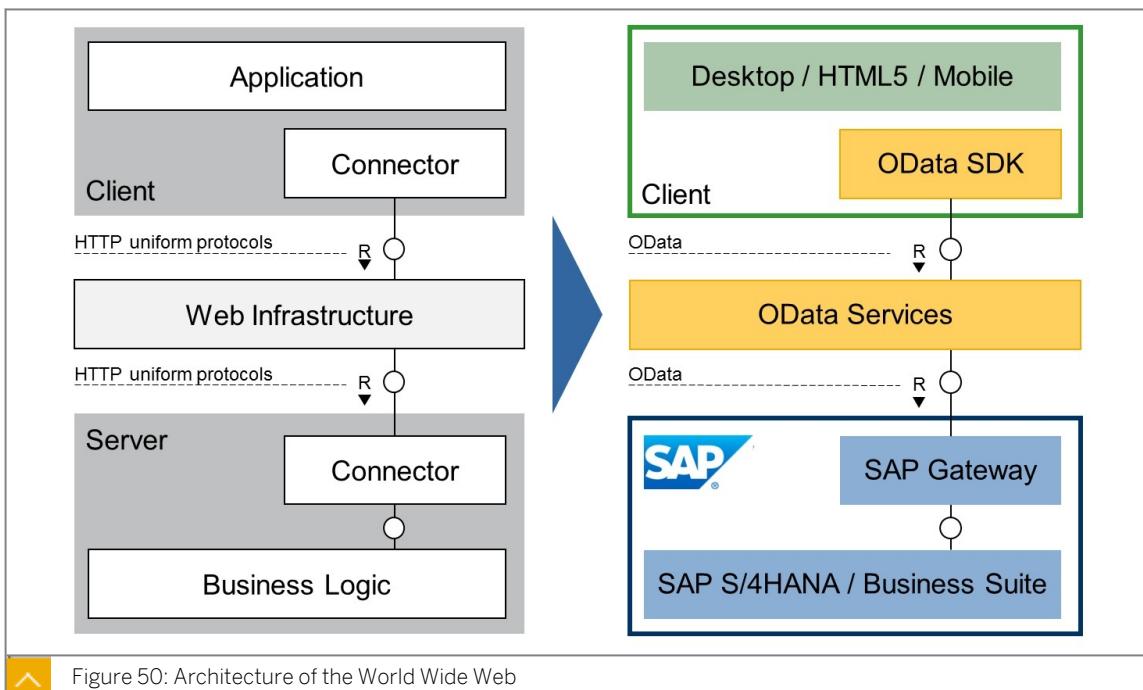


Figure 50: Architecture of the World Wide Web

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



- SAP Annotations enhance OData by adding additional information like dictionary labels.
- OData defines the standard to transfer data using already established technologies. It is also called "ODBC for the web".
- JSON (Java Script Object Notation) defines the data format for the transferred data.
- XML in combination with Atom Publishing and Atom Syndication is the alternative to JSON with a wider support range.
- HTTP(S) is the network protocol for communication.

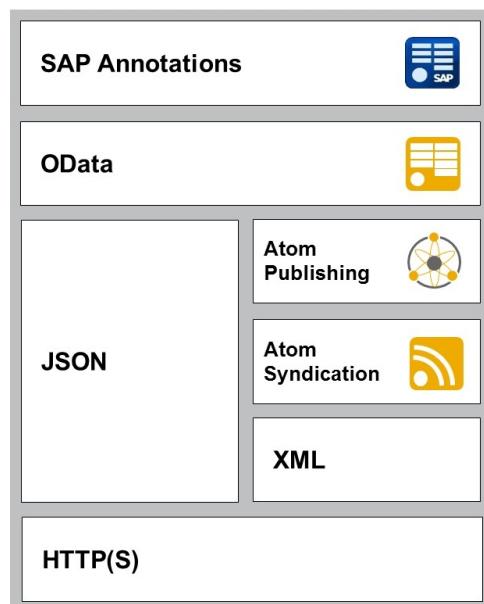


Figure 51: Open Data Protocol

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 enables SAP to supplement the data types used by OData with extra information from the ABAP Data Dictionary.

OData is available in version 2 (V2), version 3 (V3), and version 4 (V4). The versions are build on each other extending the previous version by adding new features. The majority of OData services are based on V2. SAP Gateway supports OData V2 since AS ABAP 7.00 and OData V4 since AS ABAP 7.50. OData V3 was skipped in SAP Gateway and is therefore not supported.



Atom	JSON
<pre><entry xmlns="http://www.w3.org/2005/Atom" xmlns:m="http://schemas.microsoft.com/ado/2007/08/dataservices/metadata" xmlns:d="http://schemas.microsoft.com/ado/2007/08/dataservices" metag="U"/><atom:updated>2015-09-10T15:22:03Z</atom:updated> <category term="/IWBEP/GWSAMPLE_BASIC.Product" scheme="http://schemas.microsoft.com/ado/2007/08/dataservices/scheme"/> <link href="/ProductSet('HT-1000')/edit" rel="edit" title="Product"/> <link href="/ProductSet('HT-1000')/ToSalesOrderLineItems" rel="http://schemas.microsoft.com/ado/2007/08/dataservices/related/ToSalesOrderLineItems" type="application/atom+xml;type=feed" title="ToSalesOrderLineItems"/> <link href="/ProductSet('HT-1000')/ToSupplier" rel="http://schemas.microsoft.com/ado/2007/08/dataservices/related/ToSupplier" type="application/atom+xml;type=entry" title="ToSupplier"/> <content type="application/xml"> <m:properties> <d:ProductID>HT-1000</d:ProductID> <d>TypeCode>PR</d>TypeCode> <d:Category>Notebooks</d:Category> <d:Name>Notebook Basic 15</d:Name> <d:NameLanguage>EN</d:NameLanguage> <d>Description> Notebook Basic 15 with 2,80 GHz quad core, 15" LCD, 4 GB DDR3 RAM, 500 GB Hard Disc, Windows 8 Pro </d>Description> <d>DescriptionLanguage>EN</d>DescriptionLanguage> <d:SupplierID>0100000000</d:SupplierID> <d:SupplierName>SAP</d:SupplierName> <d:TaxTariffCode>1</d:TaxTariffCode> <d:MeasureUnit>EA</d:MeasureUnit> <d:WeightMeasure>4.200</d:WeightMeasure> <d:WeightUnit>KG</d:WeightUnit> <d:CurrencyCode>EUR</d:CurrencyCode> <d:Price>956.00</d:Price> <d:Width>0.30</d:Width> <d:Depth>0.18</d:Depth> <d:Height>0.03</d:Height> <d:DimUnit>MM</d:DimUnit> <d:Created>2015-09-10T15:03:48.0000000</d:Created> <d:Changed>2015-09-10T15:03:48.0000000</d:Changed> </m:properties> </content> </entry></pre>	<pre>{ "d": { "metadata": (...), "ProductID": "HT-1000", "TypeCode": "PR", "Category": "Notebooks", "Name": "Notebook Basic 15", "NameLanguage": "EN", "Description": "Notebook Basic 15 with 2,80 GHz quad core, 15\" LCD, 4 GB DDR3 RAM, 500 GB Hard Disc, Windows 8 Pro", "DescriptionLanguage": "EN", "SupplierID": "0100000000", "SupplierName": "SAP", "TaxTariffCode": 1, "MeasureUnit": "EA", "WeightMeasure": "4.200", "WeightUnit": "KG", "CurrencyCode": "EUR", "Price": "956.00", "Width": "0.30", "Depth": "0.18", "Height": "0.03", "DimUnit": "MM", "CreatedAt": "/Date(1441897428000)/", "ChangedAt": "/Date(1441897428000)/", "ToSalesOrderLineItems": (...), "ToSupplier": (...) } }</pre>

Figure 52: 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 53: 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 through HTTP verbs.



```

▼<app:service xmlns:app="http://www.w3.org/2007/app" xmlns:atom="http://www.w3.org/2005/Atom"
  xmlns:m="http://schemas.microsoft.com/ado/2007/08/dataservices/metadata"
  xmlns:sap="http://www.sap.com/Protocols/SAPData" xml:lang="de"
  xml:base="http://
    /sap/opu/odata/IWBEP/GWSAMPLE_BASIC/"/>
  <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 54: 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.

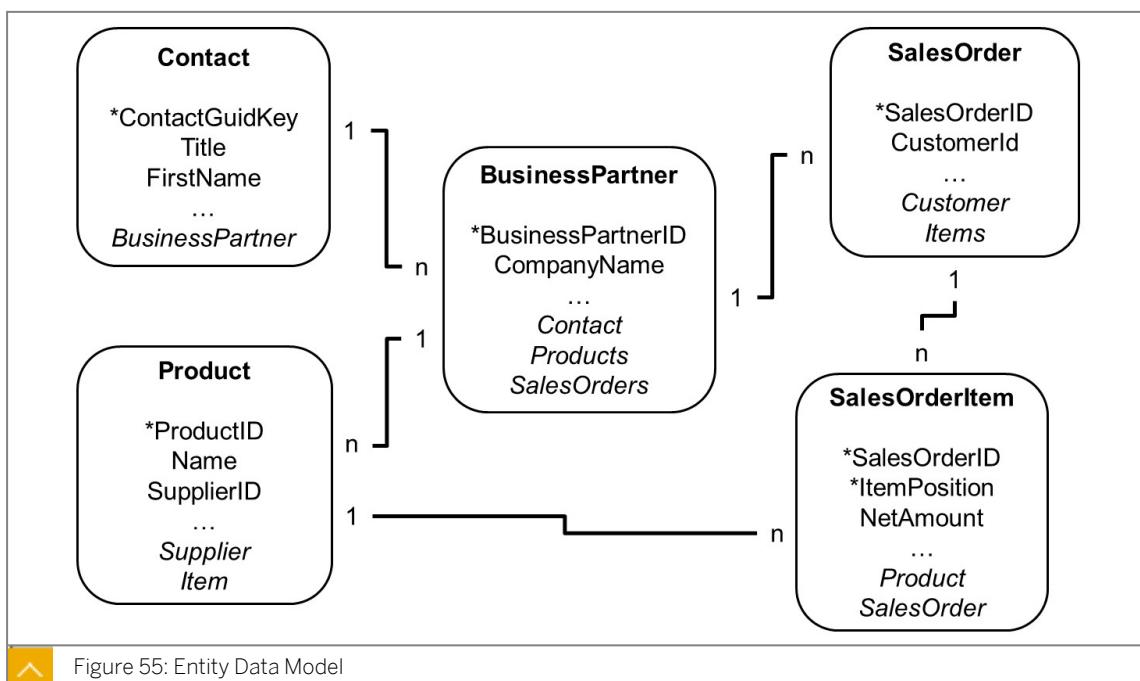


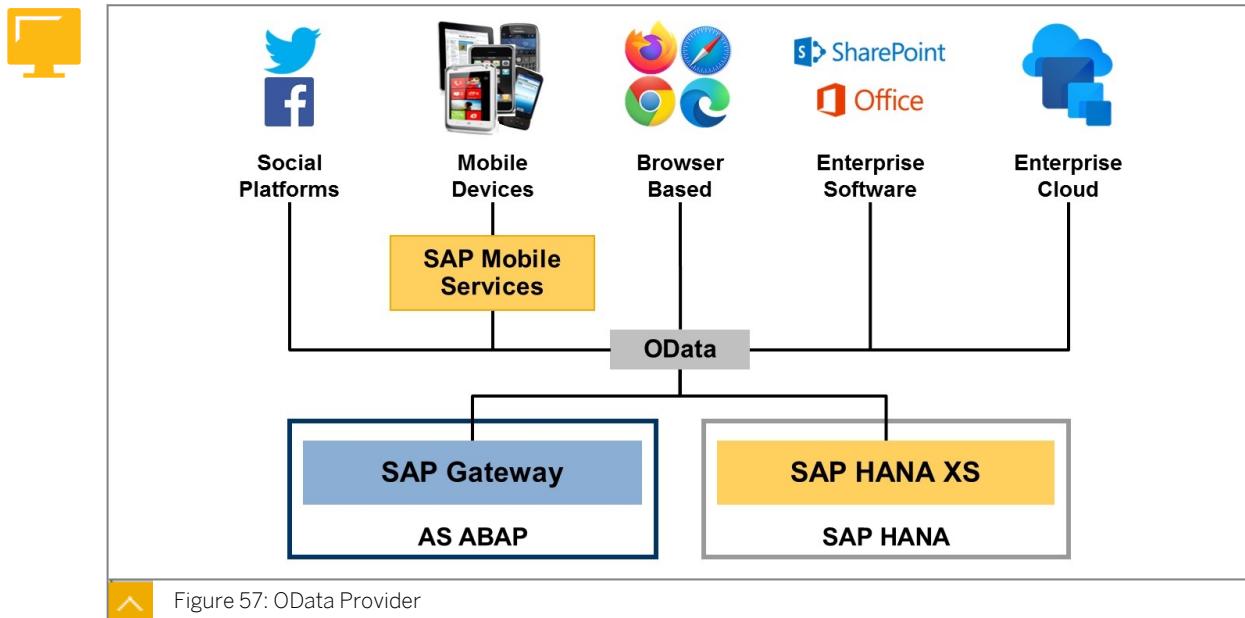
Figure 55: Entity Data Model

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



SAP Gateway (formally known as SAP NetWeaver Gateway) provides a single entry point to access business data of ABAP-based systems such as 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 Mobile Services adds additional value to the applications.

The screenshot shows the SAP Gateway Service Maintenance interface. At the top, it says "Activate and Maintain Services" and has a tab labeled "/IWFND/MAINT_SERVICE". Below this is a toolbar with various icons for service management. The main area is divided into three sections:

- Service Catalog:** A table listing registered OData services. The columns include Type, Technical Service Name, Service Description, External Service Name, Namespace, OAuth, Soft State Status, and Processing Mode. Some entries have checkboxes or dropdown menus for configuration.
- ICF Nodes:** A table showing ICF nodes with columns for Status, ICF Node, Session Time-out, Soft State, and Description. One entry is highlighted in blue.
- System Aliases:** A table for managing connections to other systems, with columns for SAP System Alias, Description, Default System, and Metadata.

Figure 58: SAP Gateway Service Maintenance (OData V2)

The transaction *SAP Gateway Service Maintenance* (/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 other systems)



Hint:

For services not connecting to other systems, the *Processing Mode* is set to **co-deployed only** and no system alias is assigned.

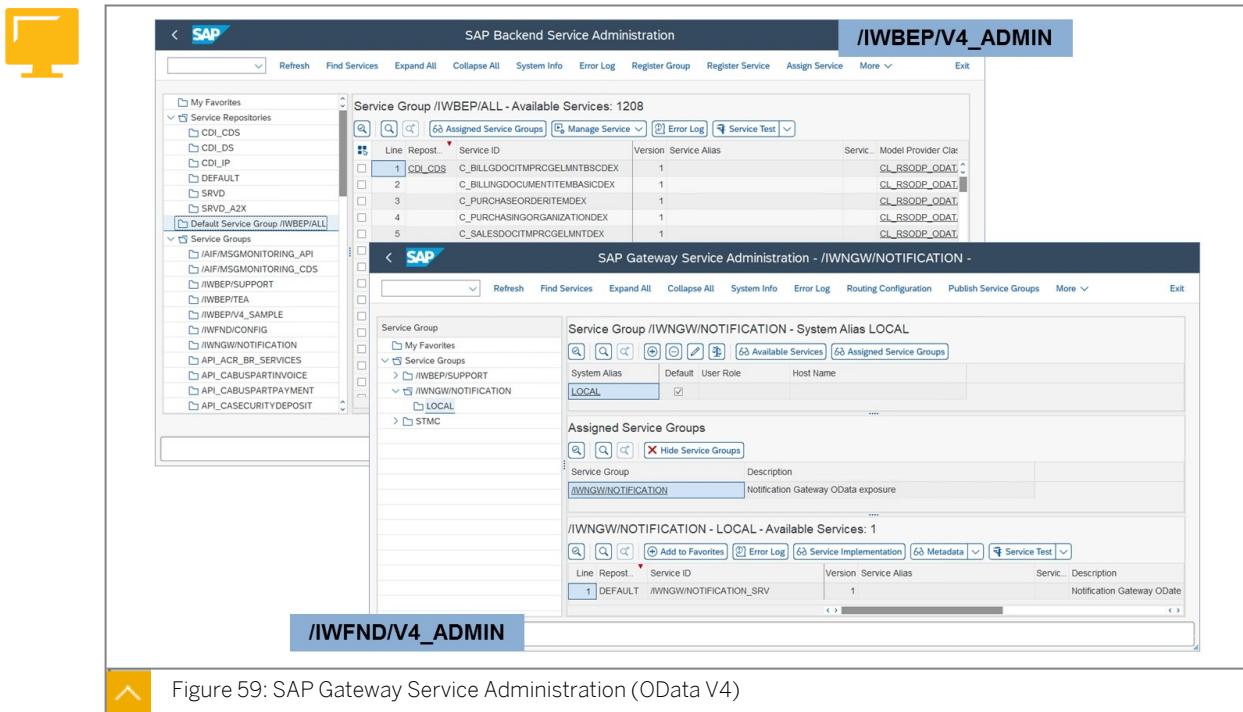


Figure 59: SAP Gateway Service Administration (OData V4)

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

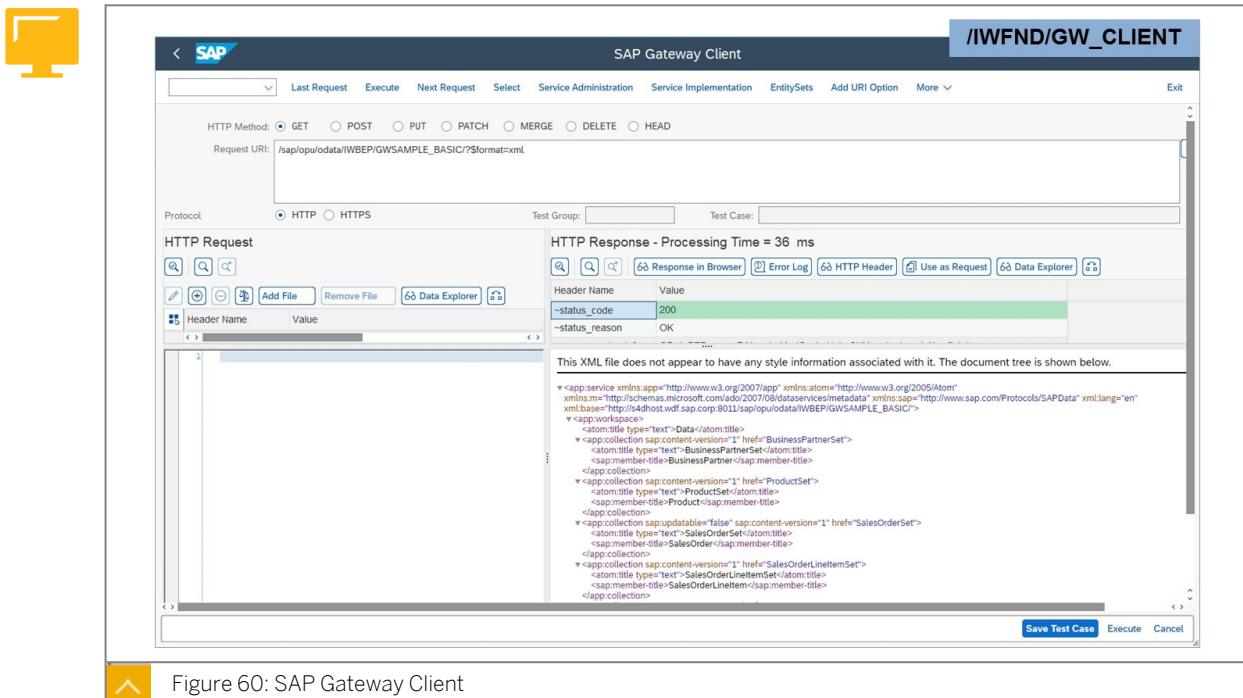
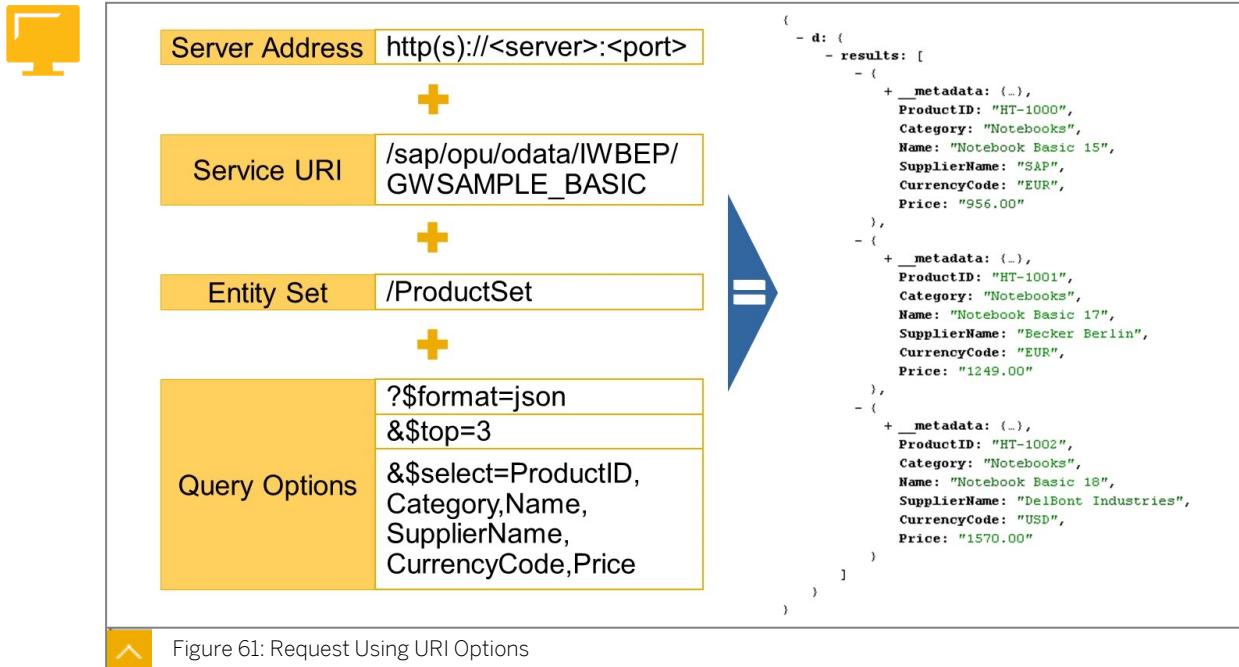


Figure 60: SAP Gateway Client

Using the transaction *SAP Gateway Client* (`/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>.



Note:

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

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



LESSON SUMMARY

You should now be able to:

- Explain data services

Unit 2

Lesson 3

Explaining Application Types

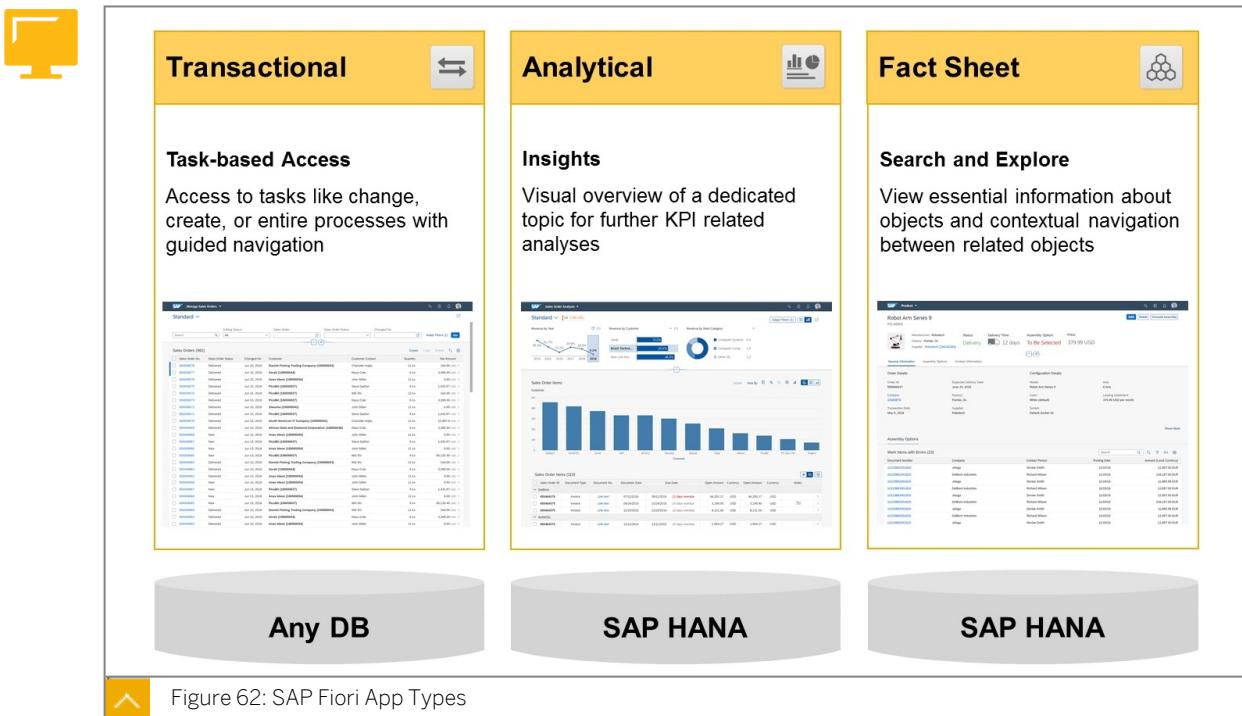


LESSON OBJECTIVES

After completing this lesson, you will be able to:

- Explain Application Types
- Using SAP Fiori App Recommendations

App Types Overview



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 SAP S/4HANA and SAP Business Suite on any database

Analytical Apps

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

Fact Sheet Apps

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

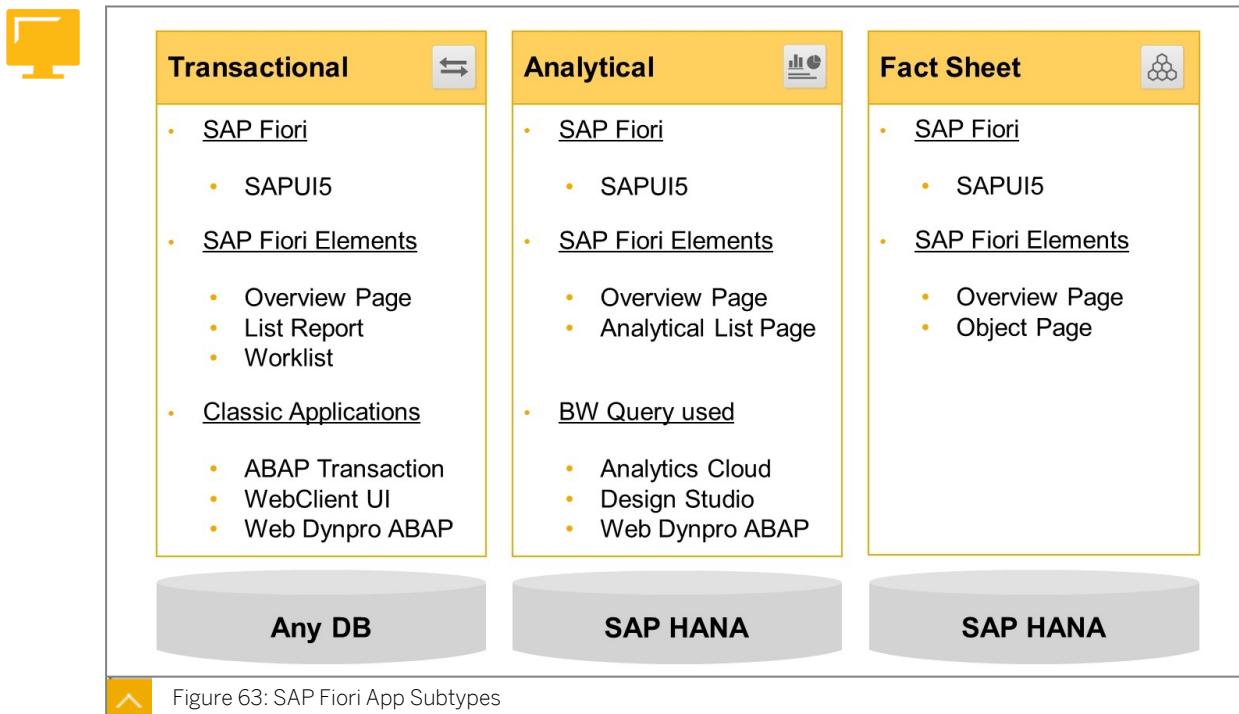


Figure 63: SAP Fiori App Subtypes

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

Enables 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

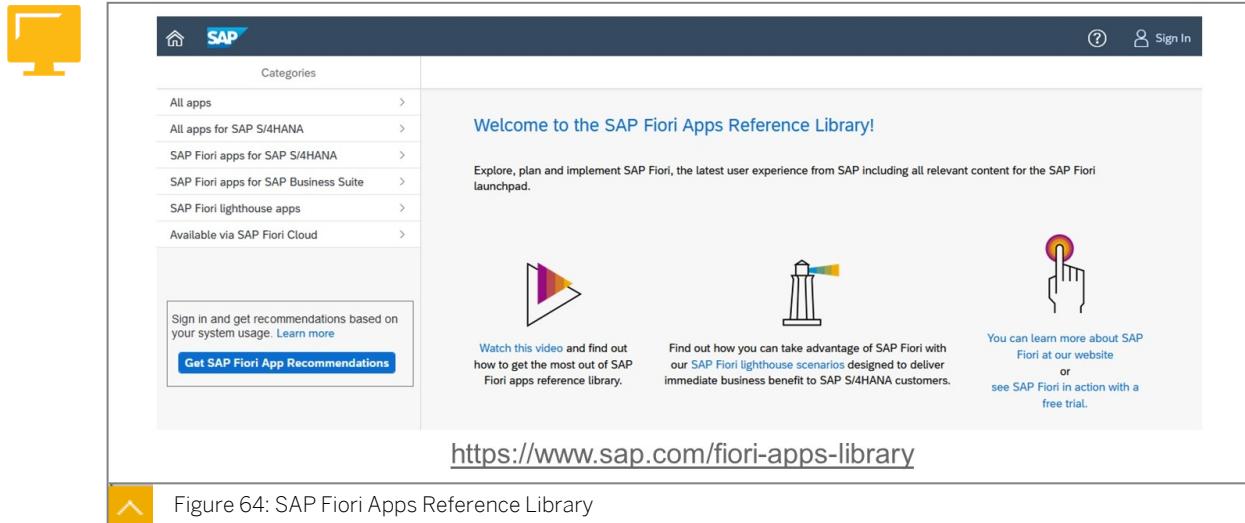
Identifies relevant areas within data sets or significant single instances using data visualization and business intelligence.

Worklist

Displays a collection of items that a user must process.

In addition, classic applications are part of transactional 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 classic applications 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 through SAP Analytics Cloud, the tool *Design Studio*, or by developing *Web Dynpro ABAP* applications.

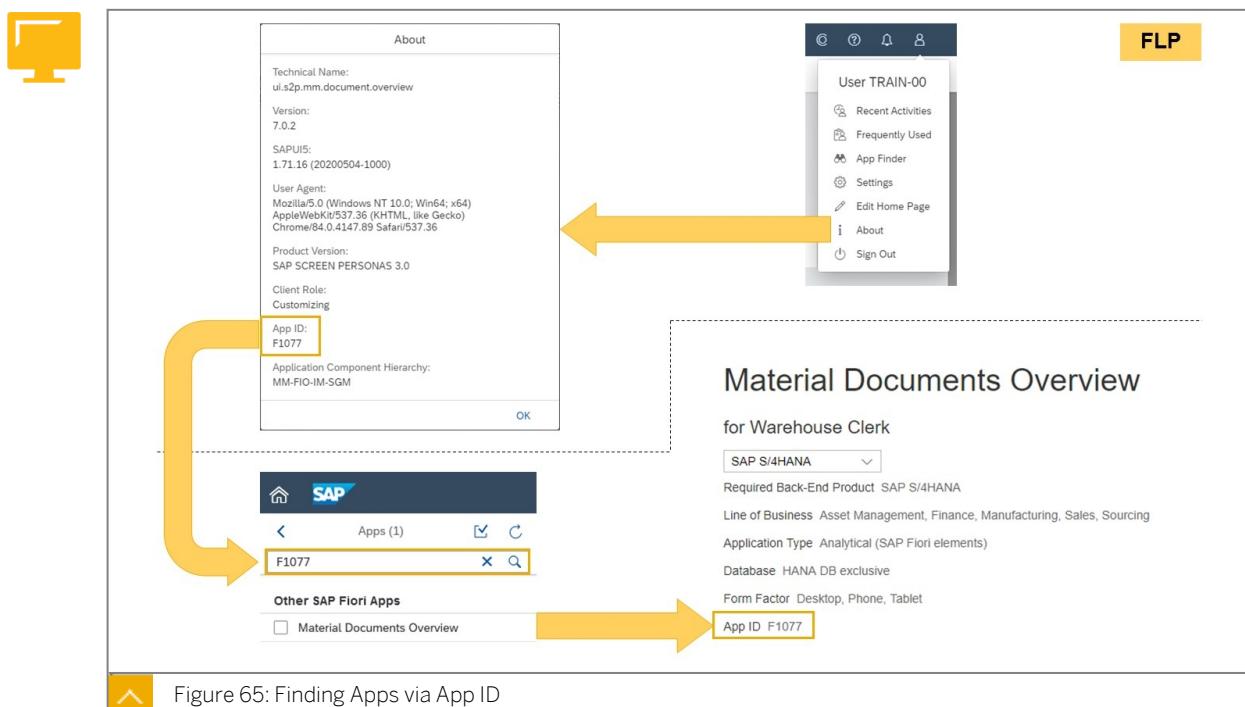


The screenshot shows the SAP Fiori Apps Reference Library homepage. It features a sidebar with categories like 'All apps', 'All apps for SAP S/4HANA', and 'Available via SAP Fiori Cloud'. A central panel displays a welcome message: 'Welcome to the SAP Fiori Apps Reference Library! Explore, plan and implement SAP Fiori, the latest user experience from SAP including all relevant content for the SAP Fiori launchpad.' Below this are three icons: a play button, a lighthouse, and a hand pointing, each with a brief description. At the bottom is a URL: <https://www.sap.com/fiori-apps-library>.

Figure 64: 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>.



The screenshot illustrates the process of finding an app by its App ID. It shows three main views: 1) An 'About' dialog for an app with 'App ID: F1077' highlighted. 2) The SAP Fiori launchpad showing 'Apps (1)' with 'F1077' selected. 3) The SAP Fiori Apps Reference Library documentation page for 'Material Documents Overview' where 'App ID F1077' is also highlighted. Arrows indicate the flow from the app's About dialog to the launchpad, and from the launchpad to the documentation page.

Figure 65: Finding Apps via App ID

Each SAP Fiori app has a unique *App ID*. When running the app in the *SAP Fiori launchpad*, choose *About* in the *User Actions Menu*. 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 apps reference library* and it is visible in the header of the documentation.



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



The screenshot shows the SAP Fiori Procurement Overview Page for Purchaser. The page header includes the title 'Procurement Overview Page' and 'for Purchaser'. On the left, there are three icons: a double arrow, a bar chart, and a cluster of circles. Below the header, the page displays the following information:

- Required Back-End Product: SAP S/4HANA
- Line of Business: Sourcing and Procurement, Supply Chain
- Application Type: Transactional, Analytical (SAP Fiori elements. Overview Page) [highlighted with a red box]
- Database: HANA DB exclusive [highlighted with a red box]
- Form Factor: Desktop, Phone, Tablet
- App ID: F1990

Below this information, there are two tabs: 'PRODUCT FEATURES' and 'IMPLEMENTATION INFORMATION'. The 'IMPLEMENTATION INFORMATION' tab is currently selected, containing the following text:

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 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 66: Overview Page Based on SAP Fiori Elements

The overview page is based on *SAP Fiori elements* technology and uses annotated views of app data. Thus, 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 67: 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 freetext search
- Search for sales orders by entering a value in one or more of the filter fields

Figure 68: 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 SAP Fiori element is the list report.



Create Material, Create Material &

SAP S/4HANA

Required Back-End Product SAP S/4HANA

Application Type **SAP GUI**

Database HANA DB exclusive

Form Factor Desktop

App ID **MM01**

PRODUCT FEATURES **IMPLEMENTATION INFORMATION**

This app is an SAP GUI for HTML transaction. These transactions are available in the SAP Fiori theme to support a seamless user experience across the SAP Fiori launchpad.

SAP Fiori 2.0 is the leading design for all SAP applications, providing a harmonized user experience across on-premise and cloud solutions. The visual theme, named Belize, comes in light and dark flavors. The central and role-based entry point for all apps, including SAP GUI for HTML transactions, is the SAP Fiori launchpad.

The image below shows the most important design features in the Belize theme.

SAP Fiori theme for classic applications: details

One visual design theme 'Belize' for all SAP Fiori applications: same colors, control design & form factor



Text toolbar instead of menu

Merged SAP Fiori header

Figure 69: SAP GUI Transaction

Since the release of SAP S/4HANA 1610, SAP GUI transactions can be found in the *SAP Fiori apps 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.



Process Maintenance Plan

for Maintenance Planner

SAP S/4HANA

Required Back-End Product SAP S/4HANA

Line of Business Asset Management, Manufacturing, Service

Application Type **Web Dynpro**

Database **HANA DB exclusive**

Form Factor Desktop

App ID W0009

PRODUCT FEATURES **IMPLEMENTATION INFORMATION**

Three apps are provided for processing a maintenance plan: Create Maintenance Plan, Change Maintenance Plan, and Display Maintenance Plan. As a maintenance planner, you can use these apps to plan recurrent maintenance work by creating time-based and performance-based single cycle plans, strategy plans, and multiple-counter plans. In time-based maintenance planning, maintenance is performed in specific cycles, for example, every two months or every six months. With performance-based maintenance plans, you can plan regular maintenance based on counter readings maintained for measuring points of pieces of equipment and at functional locations.

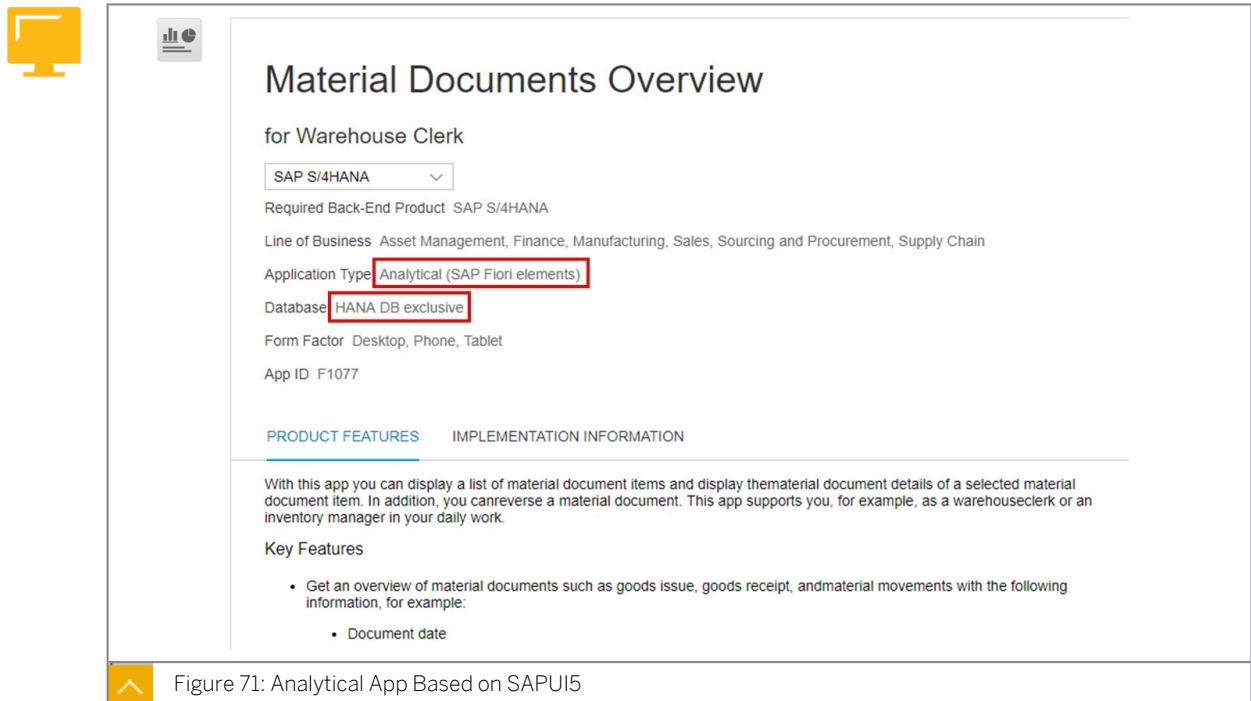
Key Features

If you have the business role Maintenance Planner, you can perform the following tasks:

Figure 70: Web Dynpro ABAP Application

Similar to 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 apps 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



The screenshot shows the SAP Fiori Launchpad interface. On the left, there's a yellow icon of a computer monitor. In the center, there's a grey icon of a document with a bar chart. The main title is "Material Documents Overview" in bold black font. Below it, the text "for Warehouse Clerk" is displayed. A dropdown menu shows "SAP S/4HANA". Underneath, it says "Required Back-End Product SAP S/4HANA". The "Line of Business" is listed as "Asset Management, Finance, Manufacturing, Sales, Sourcing and Procurement, Supply Chain". The "Application Type" is highlighted with a red box and labeled "Analytical (SAP Fiori elements)". The "Database" is also highlighted with a red box and labeled "HANA DB exclusive". "Form Factor" includes "Desktop, Phone, Tablet". "App ID" is F1077. At the bottom, there are two tabs: "PRODUCT FEATURES" (which is selected) and "IMPLEMENTATION INFORMATION". The "IMPLEMENTATION INFORMATION" tab contains the following text: "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." The "Key Features" section lists: "Get an overview of material documents such as goods issue, goods receipt, and material movements with the following information, for example:" and "Document date".

Figure 71: Analytical App Based on SAPUI5

Analytical 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 72: Design Studio App

The design studio SAP 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



Shopping Cart Item

for Employee - Procurement (SRM)

SAP Business Suite

Required Back-End Product SAP SRM, min. EHP 3

Line of Business Sourcing and Procurement

Application Type Fact sheet (SAP Fiori (SAPUI5))

Database HANA DB exclusive

App ID F0359

[PRODUCT FEATURES](#) [IMPLEMENTATION INFORMATION](#)

This factsheet displays contextual information about the shopping cart item.

Read more in [App Documentation](#)



Figure 73: Fact Sheet Based on SAPUI5

Fact sheet 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*.

Cost Center (S/4HANA)

for Controller

SAP S/4HANA

Required Back-End Product SAP S/4HANA

Line of Business Finance

Application Type Fact sheet (SAP Fiori elements)

Database HANA DB exclusive

Form Factor Desktop, Phone, Tablet

App ID F1721

PRODUCT FEATURES **IMPLEMENTATION INFORMATION**

This object page displays contextual information about the Cost Center business object. You can navigate to its related business objects and to related transactional apps, and you can access related transactions in ABAP back-end systems

This app is available for the role Controller.

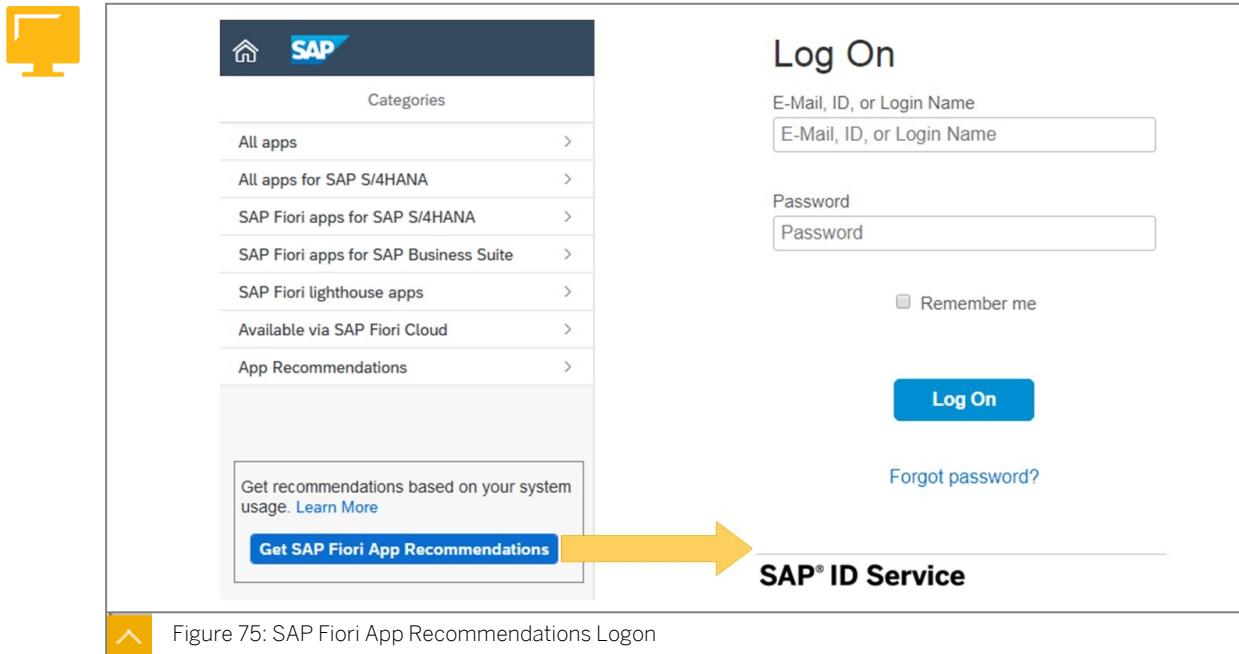
Key Features

- Display an overview of the cost center data, such as:
- Fixed assets

Figure 74: Object Page Based on SAP Fiori Elements

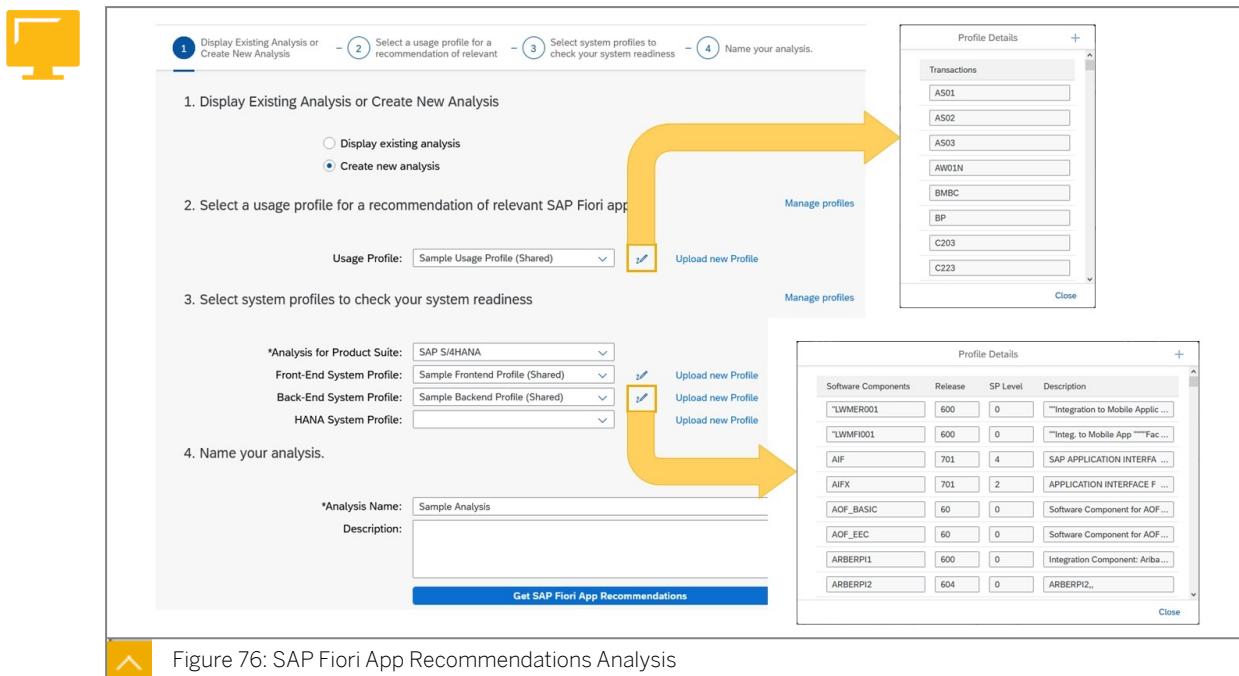
Object pages can change or create the data found via search directly in the app without opening other applications. The changes are performed in place. Fact sheet and object pages do not have tiles delivered by SAP. That is a task for customizing or configuration, or for the user when saving their search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



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 end-user feedback or by tracing the used transactions in the system, for example, through 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 functions of the SAP Solution Manager, for example, the *Landscape Management Database* (LMDB).

The screenshot shows the SAP Fiori App Recommendations Analysis List View. At the top, there's a navigation bar with a SAP logo, user name 'Stefan Fell', and a 'Manage profiles' link. Below the navigation is a search bar with 'Product Suite (SAP S/4HANA)' and 'Application Type (Fiori - Object page / Fact sheet)'. A yellow box labeled 'Filter' points to the search bar. The main area displays a list of apps under 'Apps (70) All apps'. One app, 'Supplier Accounting Document (S/4HANA)', is highlighted with a yellow box labeled 'Relevance'. Another app, 'Fixed Asset (S/4HANA)', is highlighted with a yellow box labeled 'Requirements'. A third app, 'Billing Document', is highlighted with a yellow box labeled 'Details'. A fourth app, 'Database', has a yellow box labeled 'Download' pointing to its download link. The list includes columns for 'App', 'Lighthouse', 'Relevance', 'System Readiness', 'Required Back-End', 'Description', and 'Screens hot'. At the bottom, there are 'Detail View' and 'List View' buttons.

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

Learning Assessment

1. What is the central frame application of SAP Fiori?

2. What is used as container for SAPUI5 apps in the ABAP repository?

3. Where are the services for SAPUI5 apps organized?

4. What does CRUD stand for?

5. What is JavaScript Object Notation (JSON)?

6. Which OData option grants access to the service definition?

7. What are the app types of SAP Fiori?

Choose the correct answers.

- A Transactional
- B Analytical
- C Smart
- D Principal
- E Fact Sheet

8. Which apps are meant by the term classic applications?

Learning Assessment - Answers

1. What is the central frame application of SAP Fiori?

SAP Fiori launchpad

2. What is used as container for SAPUI5 apps in the ABAP repository?

Business Server Pages (BSP)

3. Where are the services for SAPUI5 apps organized?

Internet Communication Framework (ICF)

4. What does CRUD stand for?

Create, Read, Update, Delete

5. What is JavaScript Object Notation (JSON)?

JSON is a format definition to structure data transferred using web technologies.

6. Which OData option grants access to the service definition?

\$metadata

7. What are the app types of SAP Fiori?

Choose the correct answers.

- A Transactional
- B Analytical
- C Smart
- D Principal
- E Fact Sheet

Correct. The SAP Fiori app types are: Transactional, Analytical, and Fact Sheet.

8. Which apps are meant by the term classic applications?

ABAP Transactions and Web Dynpro ABAP applications

Lesson 1

Examining ABAP Platform

75

Lesson 2

Examining SAP HANA

83

Lesson 3

Examining SAP S/4HANA

87

Lesson 4

Examining SAP Fiori Development

103

UNIT OBJECTIVES

- Examine SAP Fiori for ABAP Platform
- Examine Landscape for Transactional Apps
- Examine SAP HANA
- Examine SAP Fiori for SAP S/4HANA
- Examine SAP S/4HANA Embedded Analytics
- Examine SAP HANA Enterprise Search
- Examine Development Paradigm Shift
- Examine SAP Fiori Elements

Unit 3

Lesson 1

Examining ABAP Platform



LESSON OBJECTIVES

After completing this lesson, you will be able to:

- Examine SAP Fiori for ABAP Platform
- Examine Landscape for Transactional Apps

Landscape of SAP Fiori for ABAP Platform

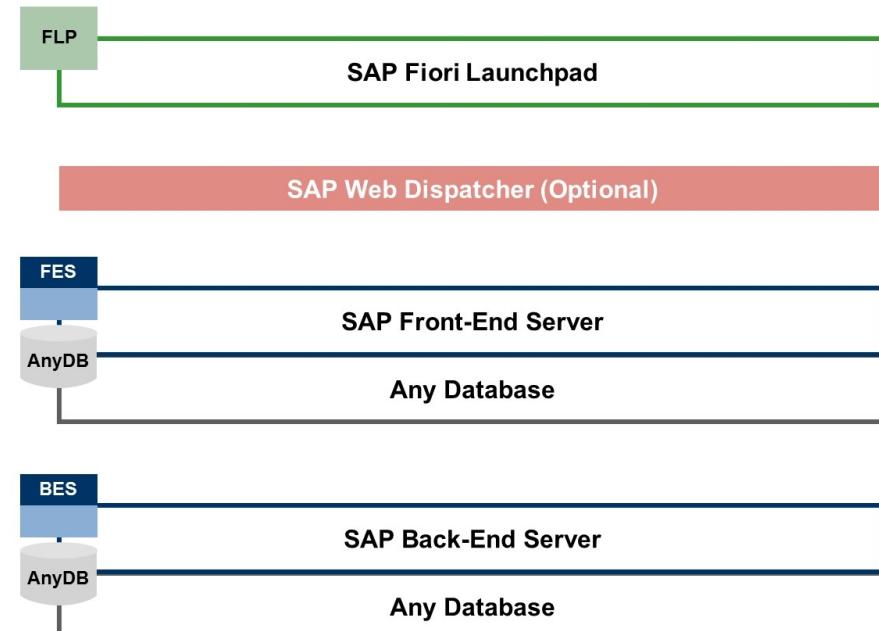


Figure 78: SAP Fiori System Landscape – Any Database

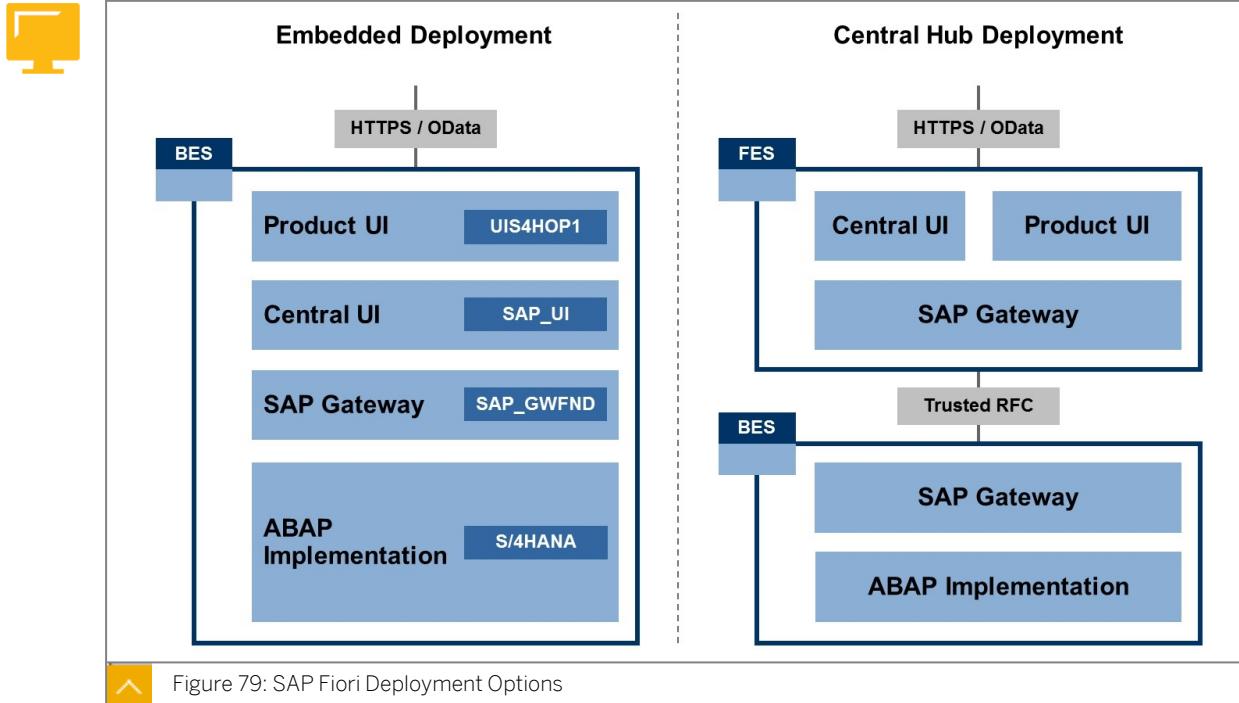
The system landscape for SAP Fiori consists mainly of a Front-End Server (FES) and an Back-End Server (BES). These are system roles of an Application Server (AS) ABAP or ABAP Platform in this landscape. The back-end server is an SAP Business Suite or an SAP S/4HANA holding applications and data of any 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 SAP Business Technology Platform (BTP), ABAP environment.

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.



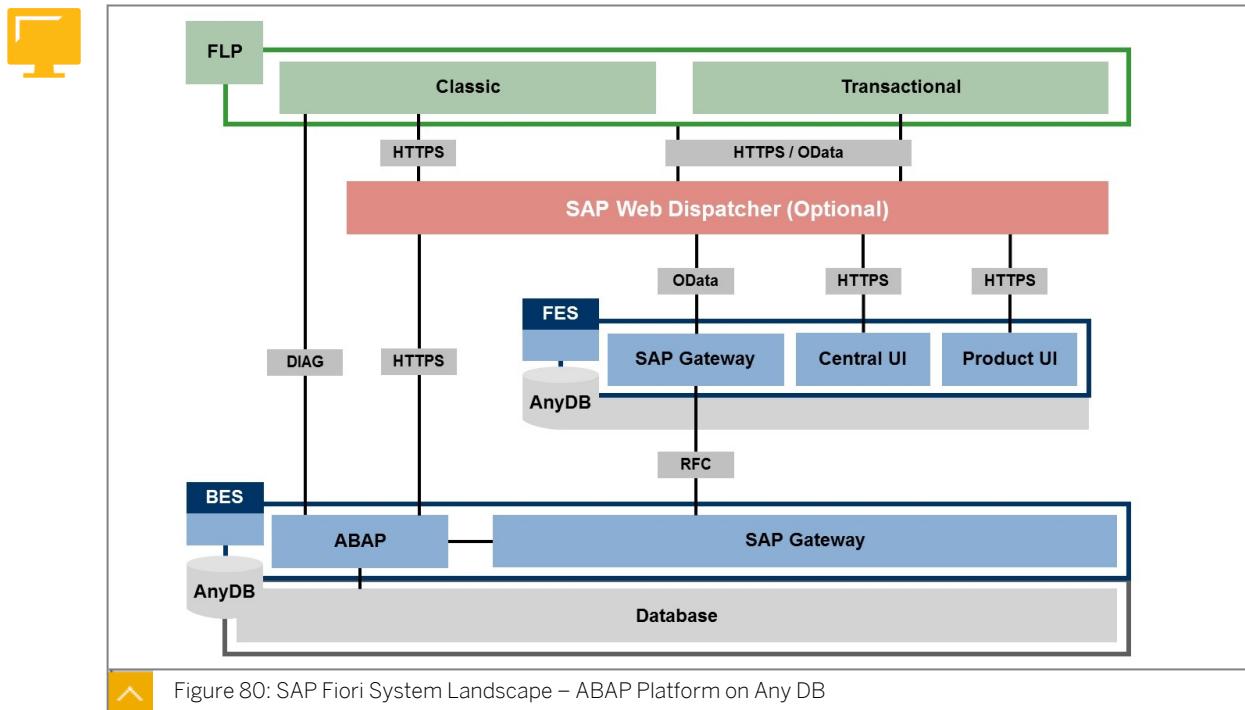
In some scenarios, it is not necessary to operate separated front-end and back-end servers. It can even be counterproductive to do so. Therefore, you can deploy all components of SAP Fiori in one system, in an embedded deployment.

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

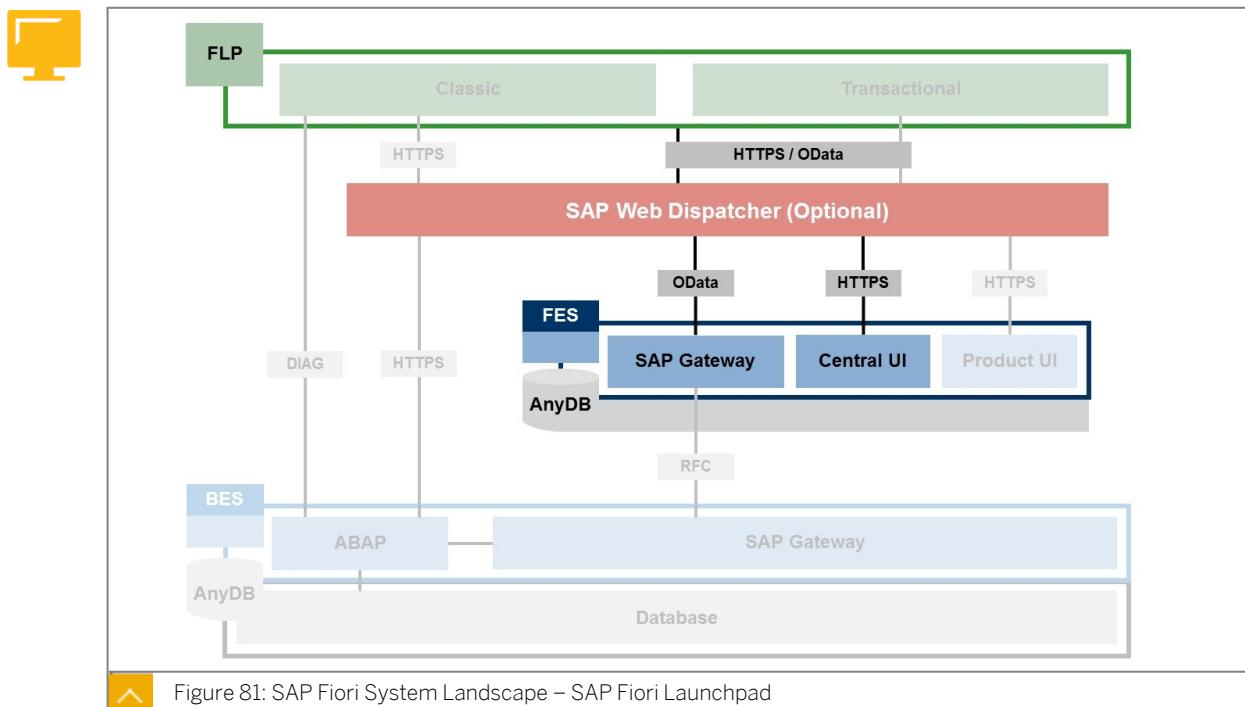
SAP Gateway, the only component in FES and BES, is the only part where you have to check carefully, which settings are in which system. As soon as there are multiple systems that want to provide SAP Fiori apps, the FES has many benefits in organizational and security terms.



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



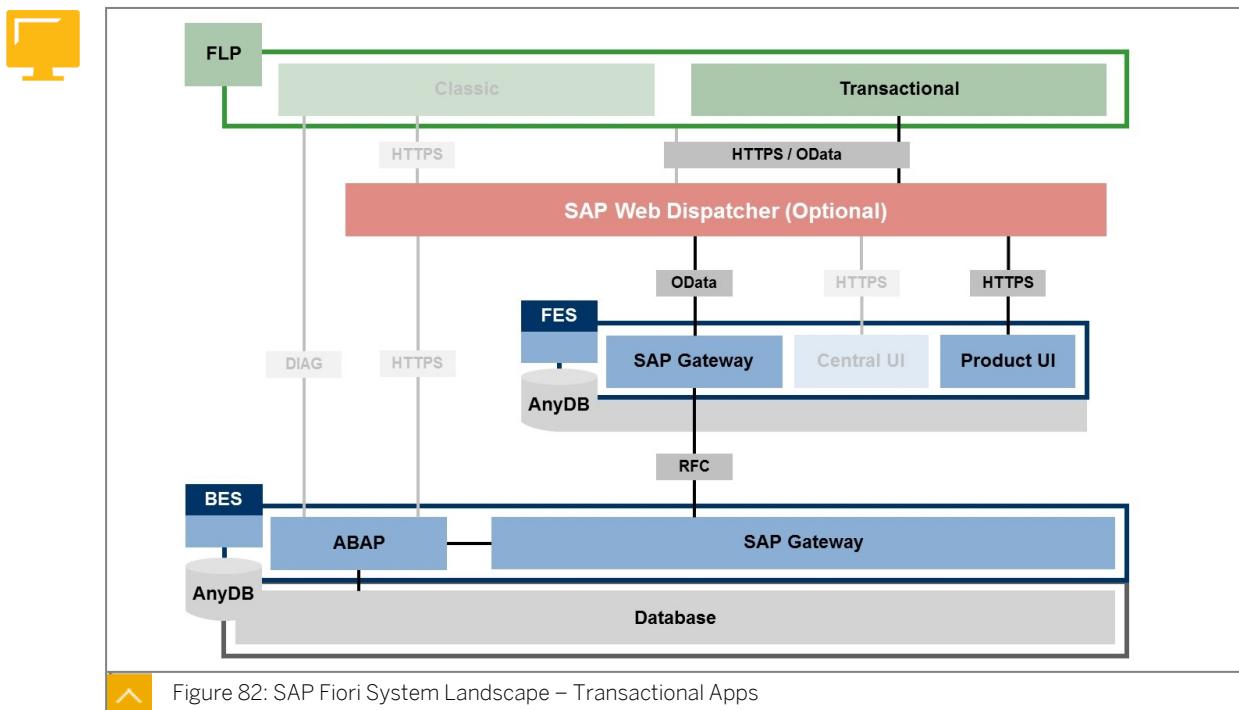
The SAP Fiori system landscape for a BES on any database supports transactional apps based on SAPUI5 and legacy apps based on Web Dynpro and ABAP transactions. Mandatory components of the FES are the SAP Gateway for OData communication, the central UI for general SAP Fiori functions, and product UIs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



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.

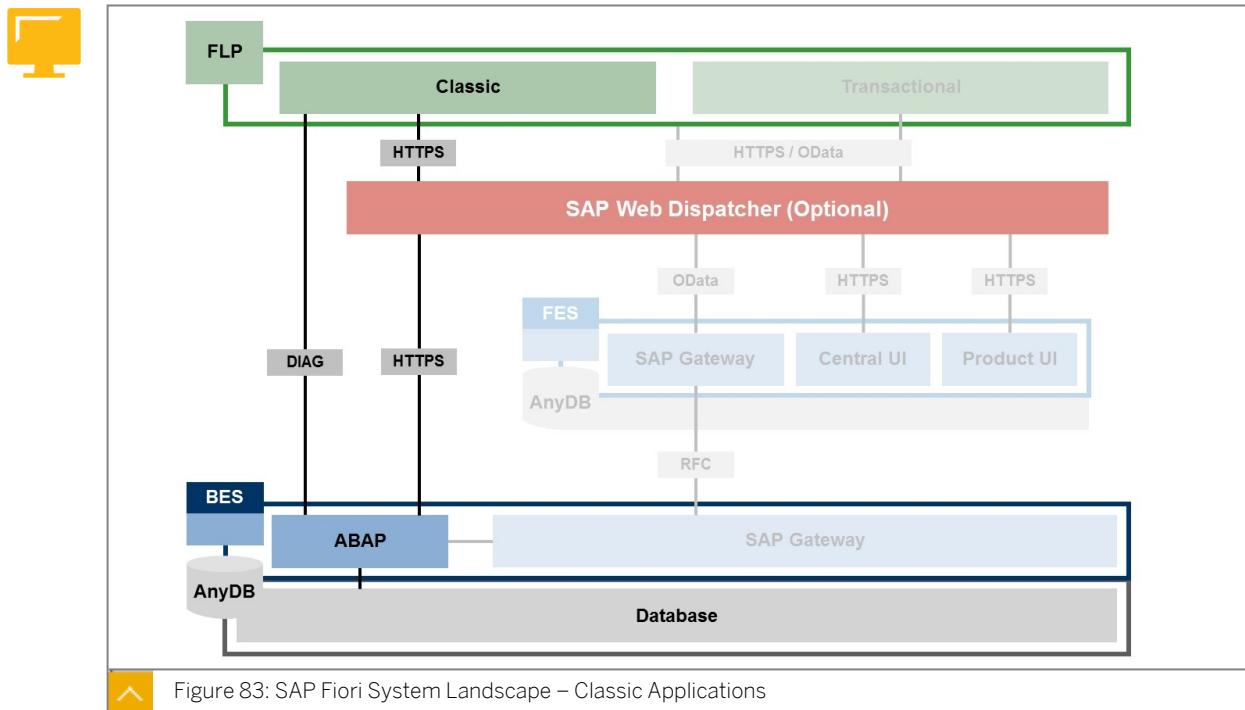
Therefore, the FLP must read and write data from and to the FES using OData, not only for personalization, but also in other areas.

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. Having only one place (system) for SAP Fiori settings is a huge benefit.



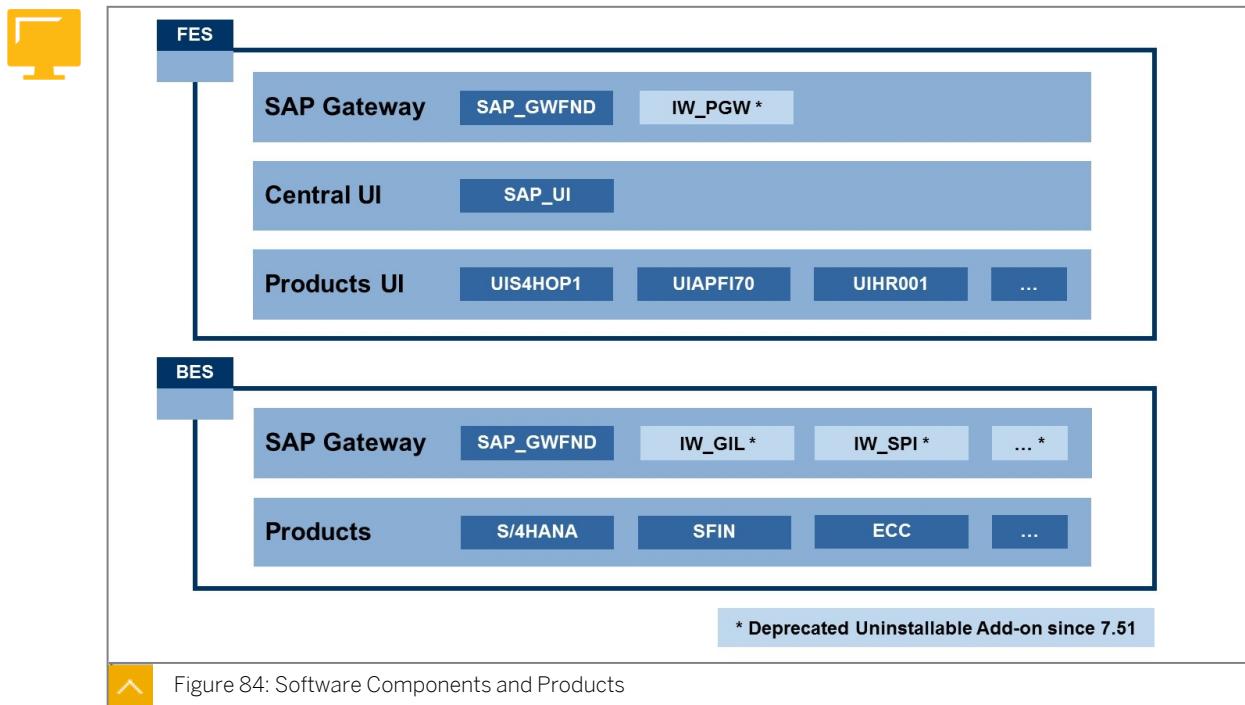
Disregarding the need for a running FLP, transactional apps consist of SAPUI5 apps and SAP Gateway services. SAPUI5 apps are saved in the ABAP repository as BSPs and delivered using product-specific UI add-ons. These are installed on the FES. The SAP Gateway services are delivered as part of updates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

The registration of the 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 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 code is used to access the data in the database.



Classic applications 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 code on the BES. The only task of the FES is to provide the connection information to the BES when a navigation request — such as 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 SAP 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 SAP 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 SAP 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 EHP7 FOR SAP ERP 6.0 SAP Fiori 1.0 for SAP ERP	
Support Package Stack	10 (07/2016)	
Software Component Version	UIEAAP01 100 - SP 0010	
Prerequisite for installation		UI FOR EHP7 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)
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 - SP 0012	
	SAP_APPL 618 - SP 0003	

Figure 85: SAP Fiori App Installation

In this example of the *SAP 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. All SAP Fiori applications are installed in such a way.

Transactional Applications based on SAP Gateway

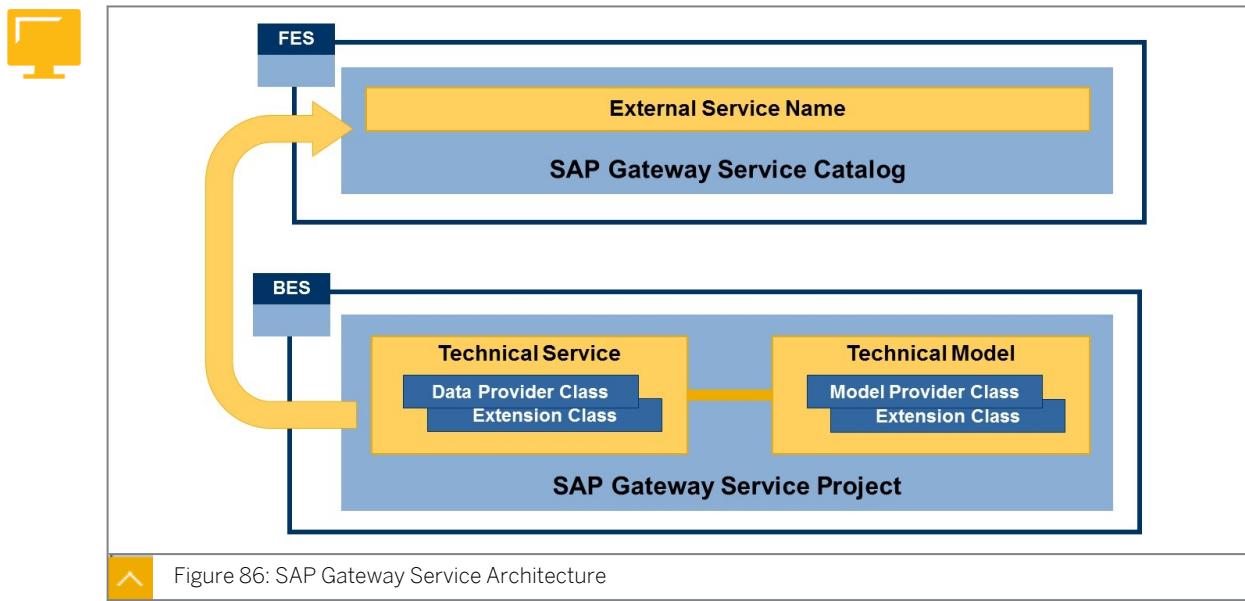


Figure 86: SAP Gateway Service Architecture

A Gateway service consists of ABAP classes and customizing elements in the FES, and a service catalog entry in the BES. The ABAP classes separate into two model provider classes (MPC) that provide the code for the service metadata and two data provider classes (DPC) that provide the code 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.

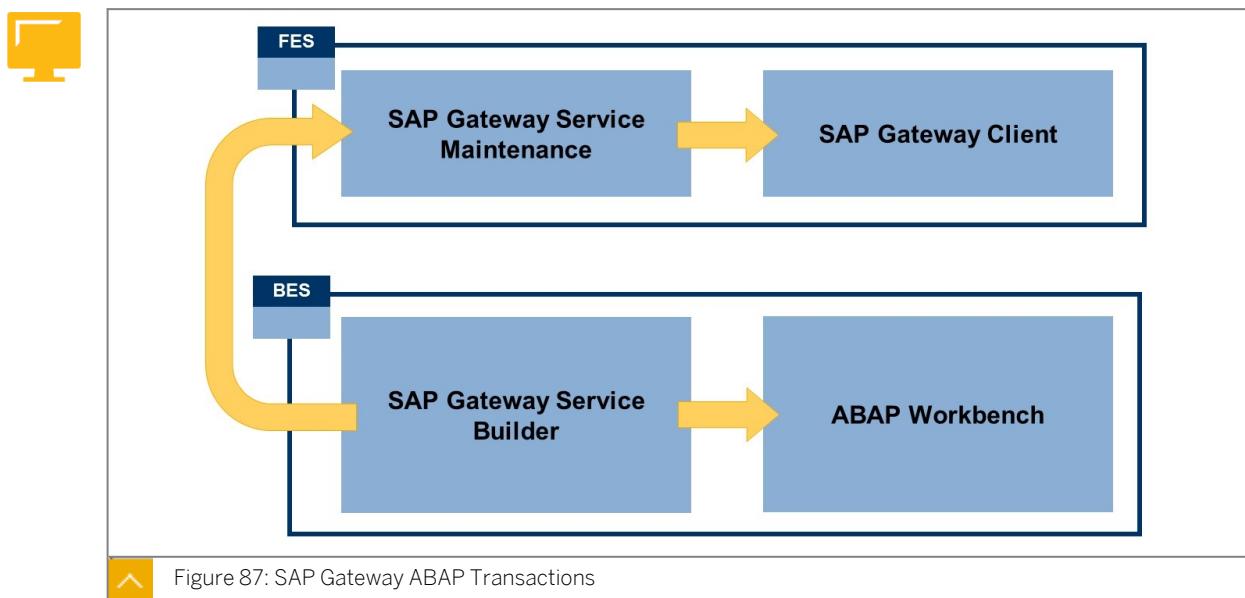


Figure 87: SAP Gateway ABAP Transactions

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

SAP Gateway Service Maintenance (/IWFND/MAINT_SERVICE)

Central SAP Gateway service management in the FES

SAP Gateway Client (/IWFND/GW_CLIENT)

SAP Gateway service test environment in the FES

SAP Gateway Service Builder (SEGW)

Central SAP Gateway development environment in the BES

ABAP Workbench (SE80)

Central ABAP development environment in the BES

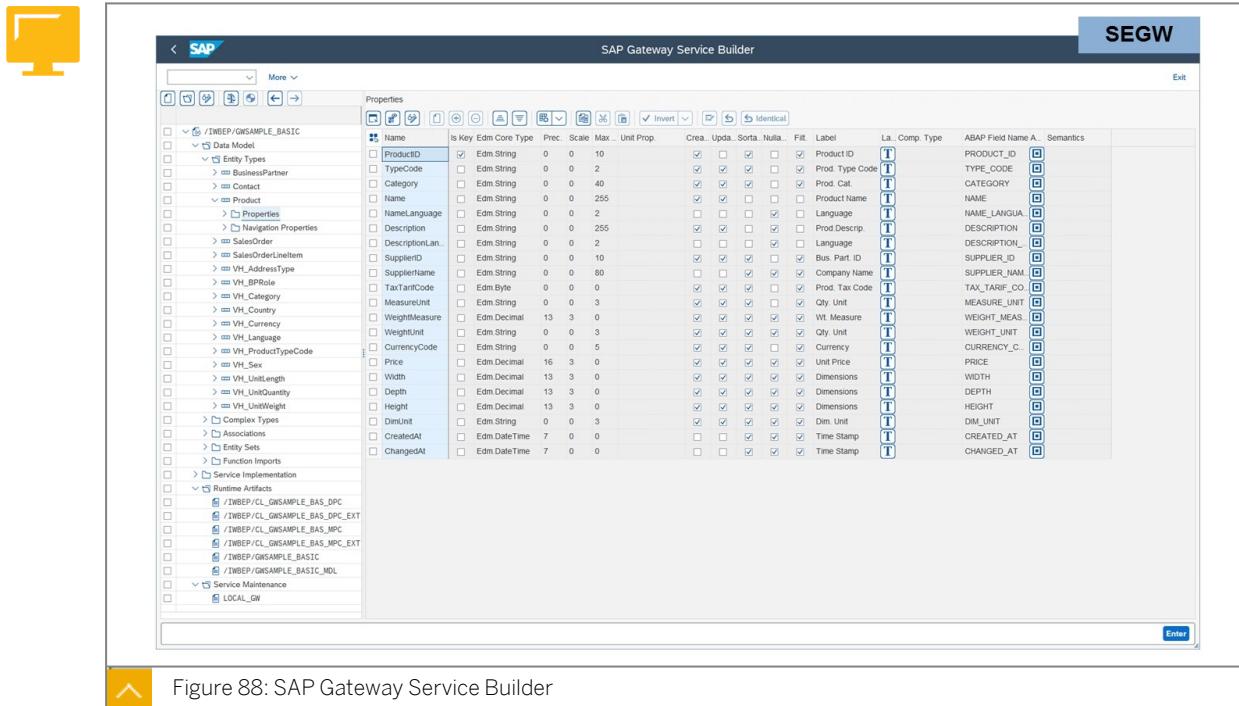


Figure 88: SAP Gateway Service Builder

The transaction **SAP Gateway Service Builder (SEGW)** 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
- Examine Landscape for Transactional Apps

Unit 3

Lesson 2

Examining SAP HANA



LESSON OBJECTIVES

After completing this lesson, you will be able to:

- Examine SAP HANA

SAP HANA

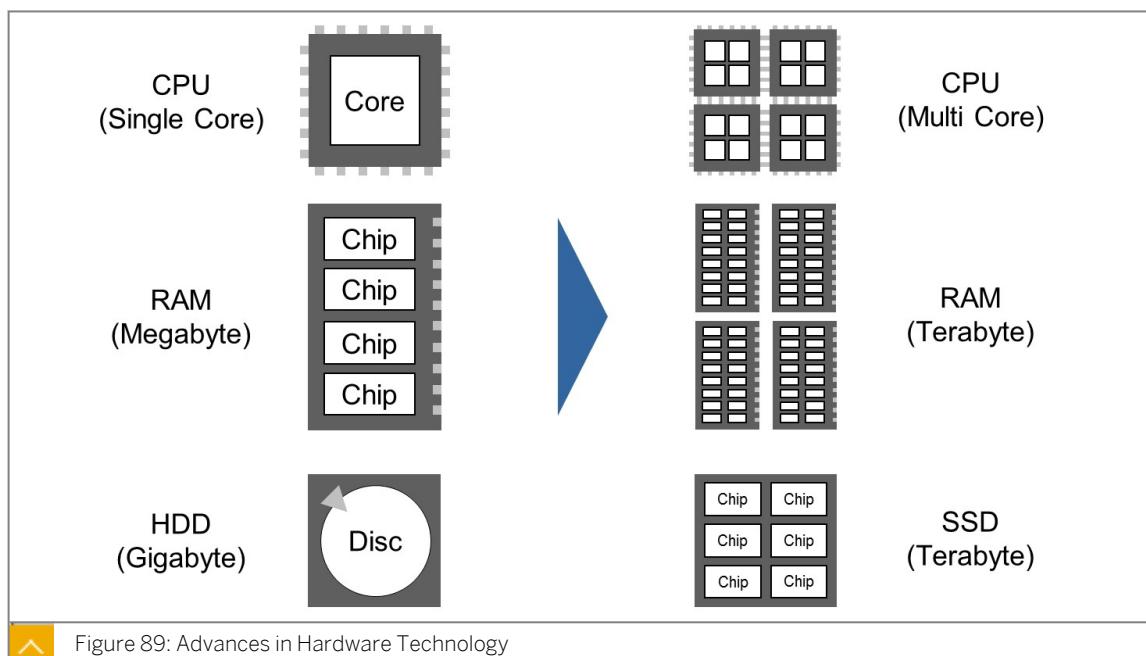
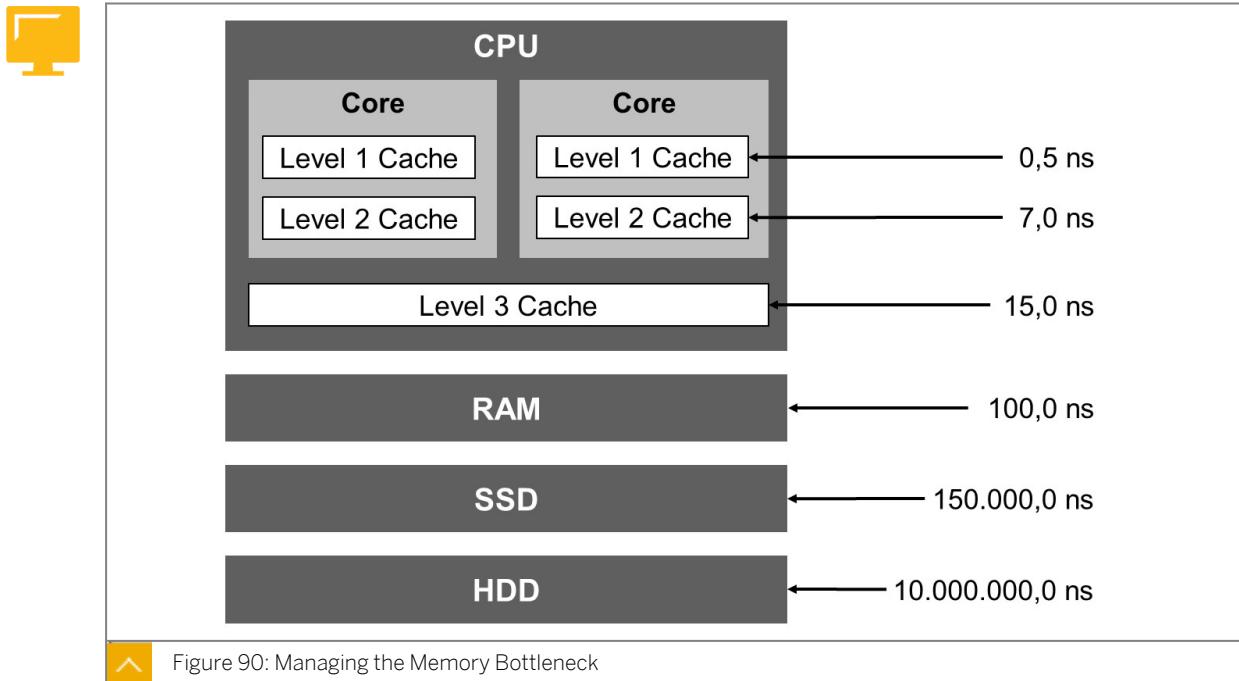


Figure 89: Advances in Hardware Technology

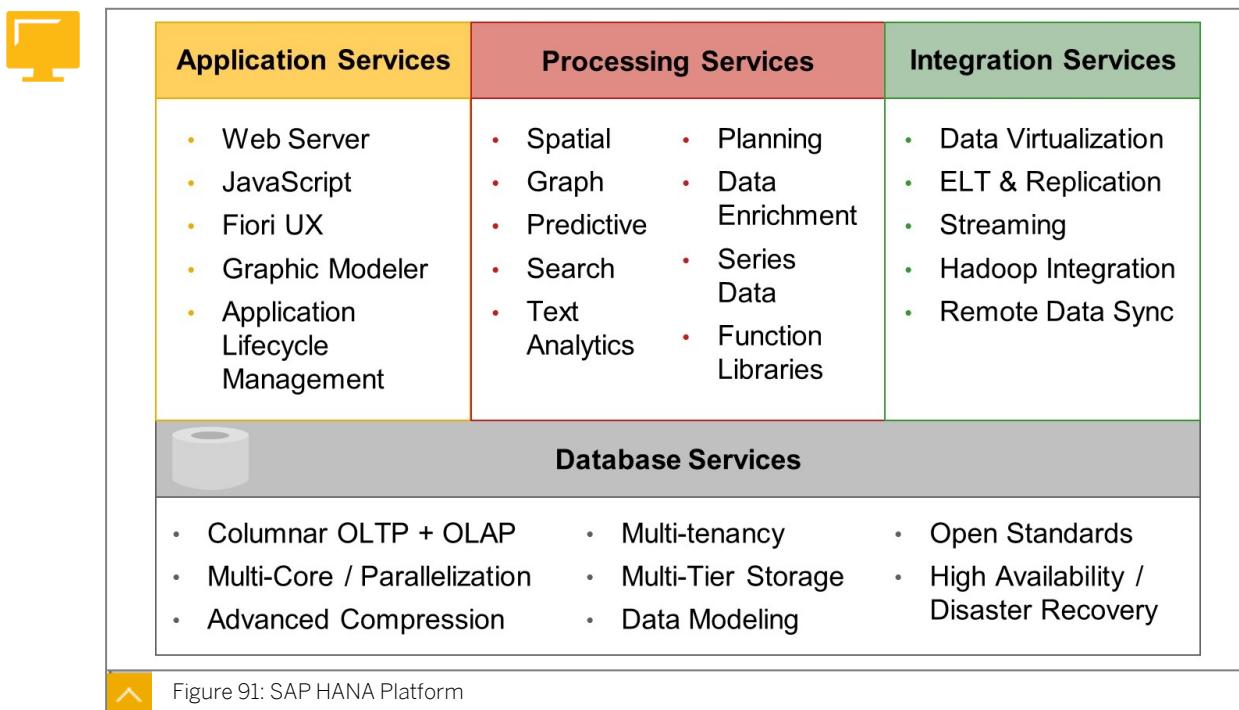
Historically, the amount of memory was limited because of its high costs. This caused a serious bottleneck in the data flow from disk to the Central Processing Unit (CPU). Today, memory is no longer a problem. SAP HANA runs on hardware with many terabytes of memory. With so much memory available, entire 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. Response times for even complex analytical tasks such as predictive analysis can be carried out in real time.



Modern processors can 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. 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

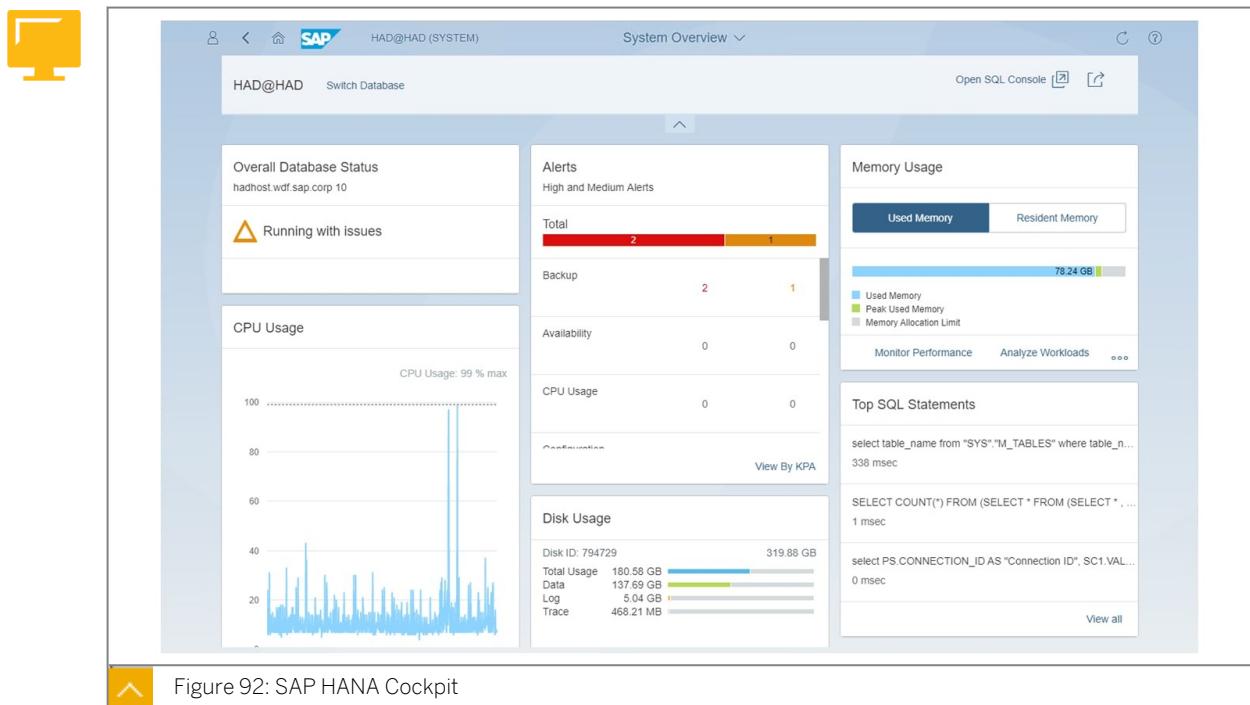
SAP HANA can handle many new types of data. This includes text, spatial, graph, and more. However, it is not enough to simply store these new data types to build applications that can process and integrate this data with traditional data types such as business transactions. SAP HANA provides native in-memory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



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 must enter database user credentials.

The *SAP HANA database explorer* is integrated into the *SAP HANA cockpit*. It enables 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).



Note:

More information about this topic can be found in HA200 (SAP HANA – Installation and Administration):

<https://training.sap.com/course/ha200>



LESSON SUMMARY

You should now be able to:

- Examine SAP HANA

Unit 3

Lesson 3

Examining SAP S/4HANA

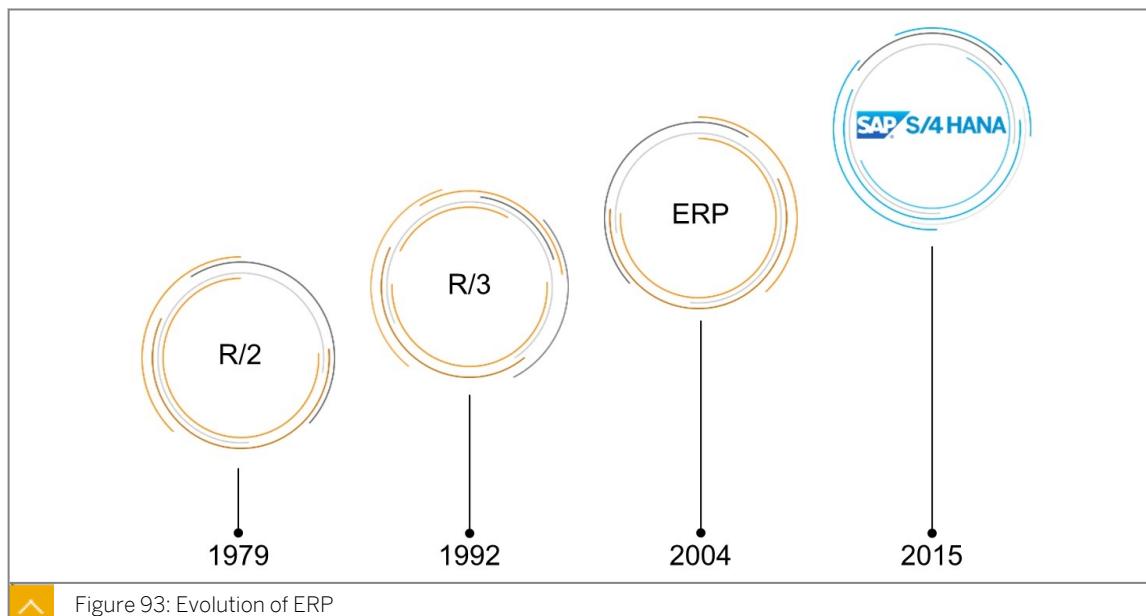


LESSON OBJECTIVES

After completing this lesson, you will be able to:

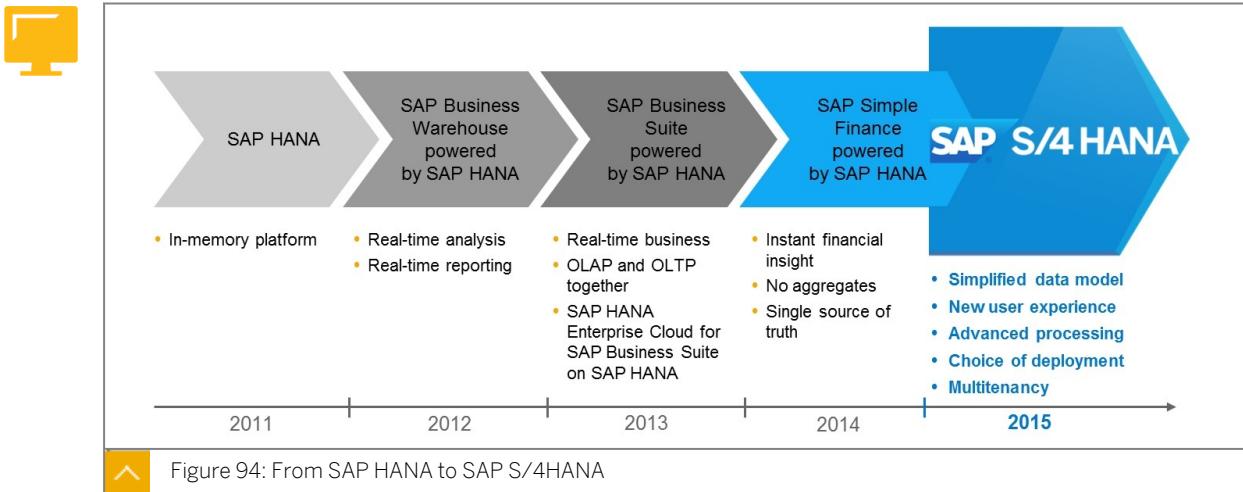
- Examine SAP Fiori for SAP S/4HANA
- Examine SAP S/4HANA Embedded Analytics
- Examine SAP HANA Enterprise Search

SAP S/4HANA

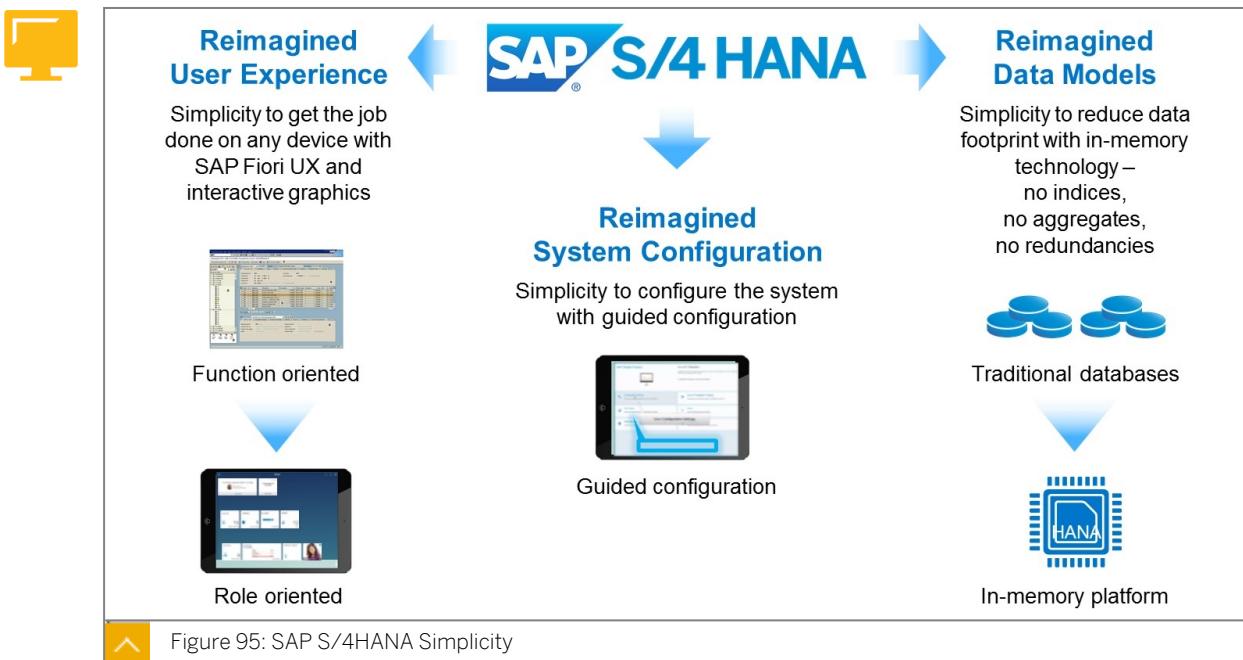


In SAP R/2, SAP started to deliver screens to specialists in order to enter information into the system. These users were experts in their fields, highly-skilled, and well-trained to use the system. Today, even the operation of an MRP Run is more like controlling a machine or using an interface to a 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 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 to run simply, 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 SAP S/4HANA that was available was SAP Simple Finance, the first optimized solution.

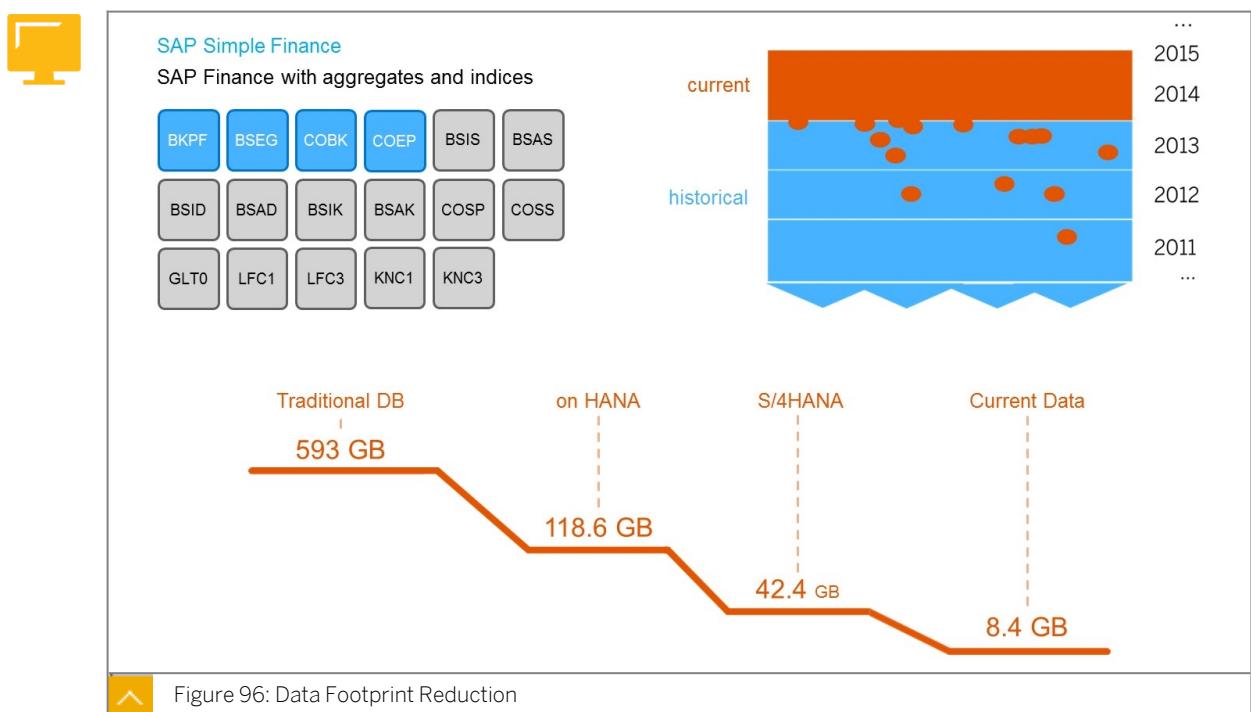


From an IT value perspective, SAP S/4HANA creates unique opportunities to dramatically simplify the landscape and reduce TCO, with SAP HANA as the great simplifier. Business users can leverage a simple and role-based user experience based on modern design principles, minimizing training efforts while 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:

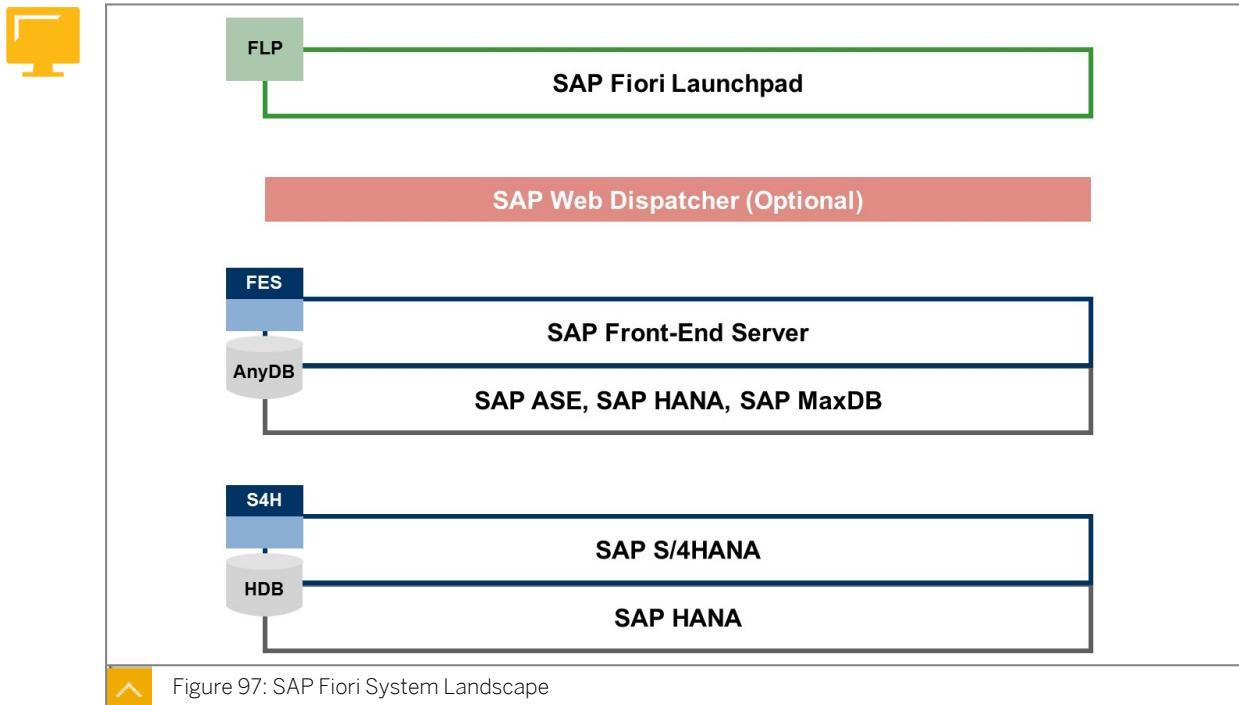
- Faster analytics and reporting
- Fewer process steps
- ERP, CRM, SRM, SCM, and PLM co-deployed
- All data: social, text, geo, graph, and processing
- No locking, parallelism
- Actual data (25%) and historical (75%)
- Smaller total data footprint
- SAP Fiori UX for any device



Tables in SAP HANA are automatically compressed so you can 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, the number of database tables reduces 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.

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 *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 now has optimized code 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](#)
- SAP Fiori front-end server 2020 for SAP S/4HANA: SAP Note [2919182](#)

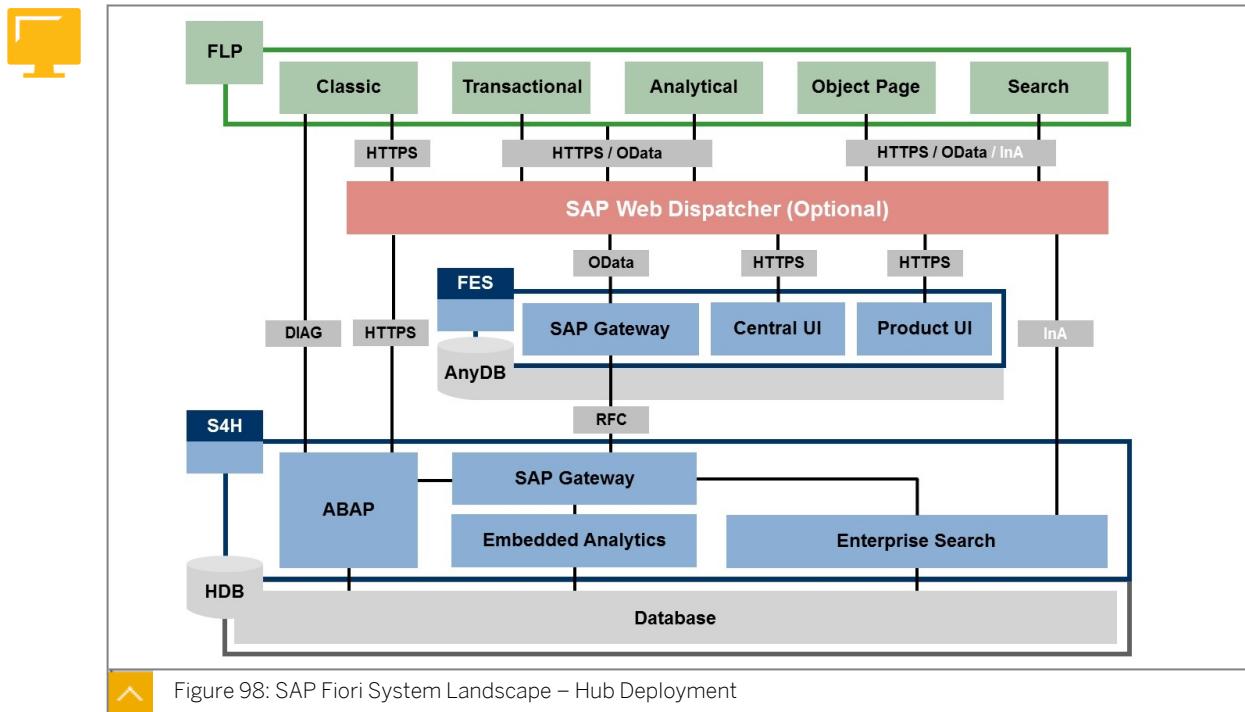


Figure 98: SAP Fiori System Landscape – Hub Deployment

All application types are available for SAP S/4HANA (S4H). The important difference to the SAP Business Suite 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 with the BES now an S4H. Only the classic applications and the SAP Fiori search connect directly to the S4H.

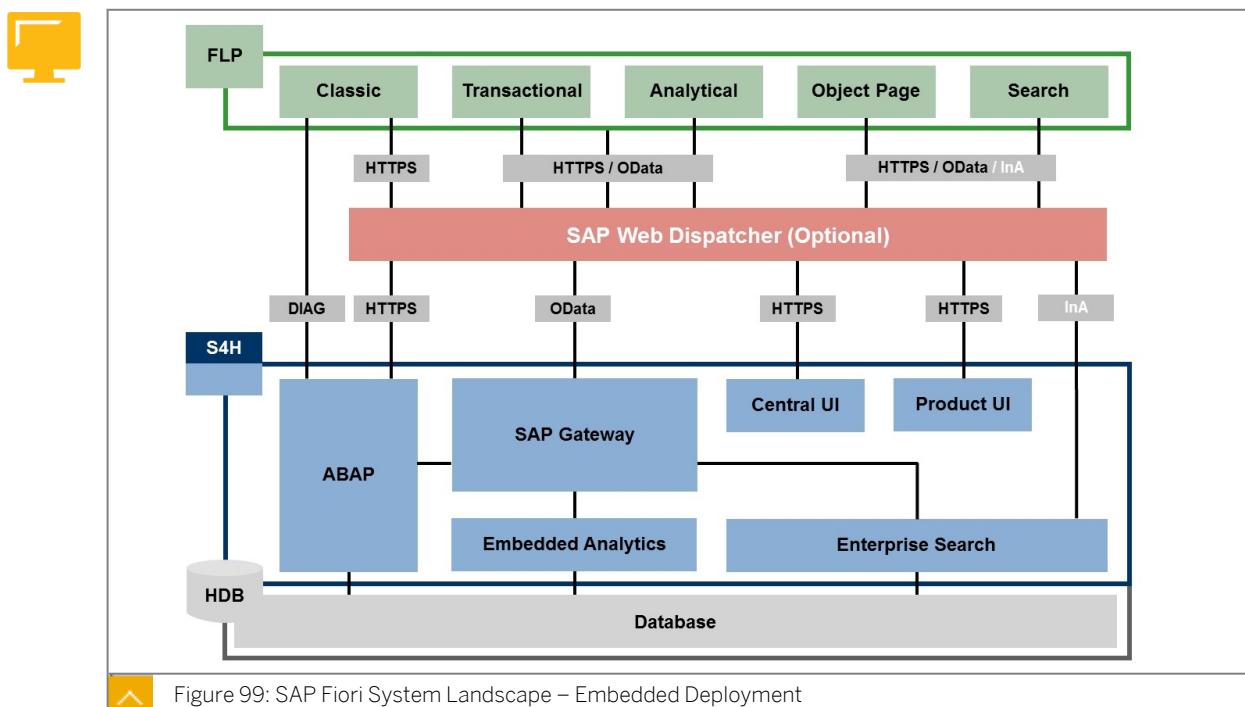


Figure 99: SAP Fiori System Landscape – Embedded Deployment

The recommended deployment option for SAP S/4HANA is very dependent on the customer scenario. In most cases, SAP recommends the embedded deployment for SAP S/4HANA. The main reason is that with SAP S/4HANA all data and functions reside in one system. A hub

system for routing requests to certain back-ends is no longer needed. In addition many configuration steps for SAP Fiori get easier when having everything in one system.



Note:

For more details please check the document "SAP Fiori Deployment Options and System Landscape Recommendations": <https://www.sap.com/documents/2018/02/f0148939-f27c-0010-82c7-eda71af511fa.html>

SAP S/4HANA Embedded Analytics

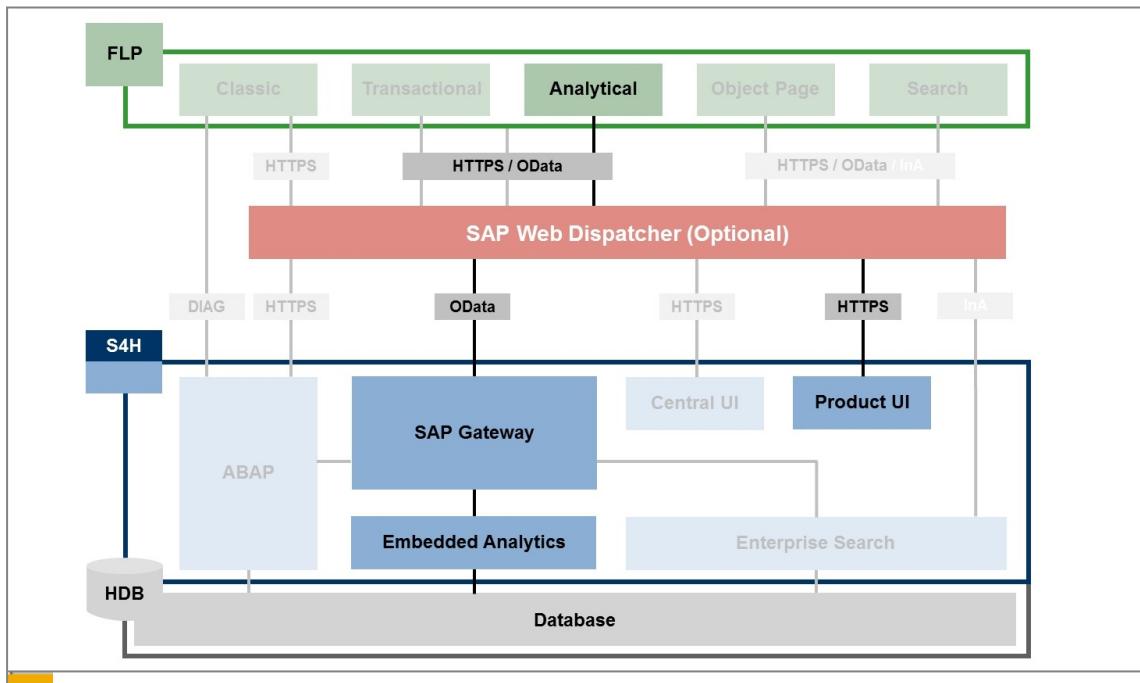


Figure 100: SAP Fiori System Landscape – Analytical Apps (S4H)

The UIs for analytical apps are part of the same product-specific UI add-ons that provide transactional apps. SAP Gateway manages again the OData services for the apps, but the source is now different. ABAP Core Data Services (CDS) Views are the basis for all analytical applications in S4H. ABAP CDS Views are able to integrate with SAP S/4HANA Embedded Analytics, which originated from SAP Business Warehouse (BW).



Note:

There is no need to access data views via the SAP HANA Extended Application Services (XS), for example, in the SAP Business Suite when using ABAP CDS Views.

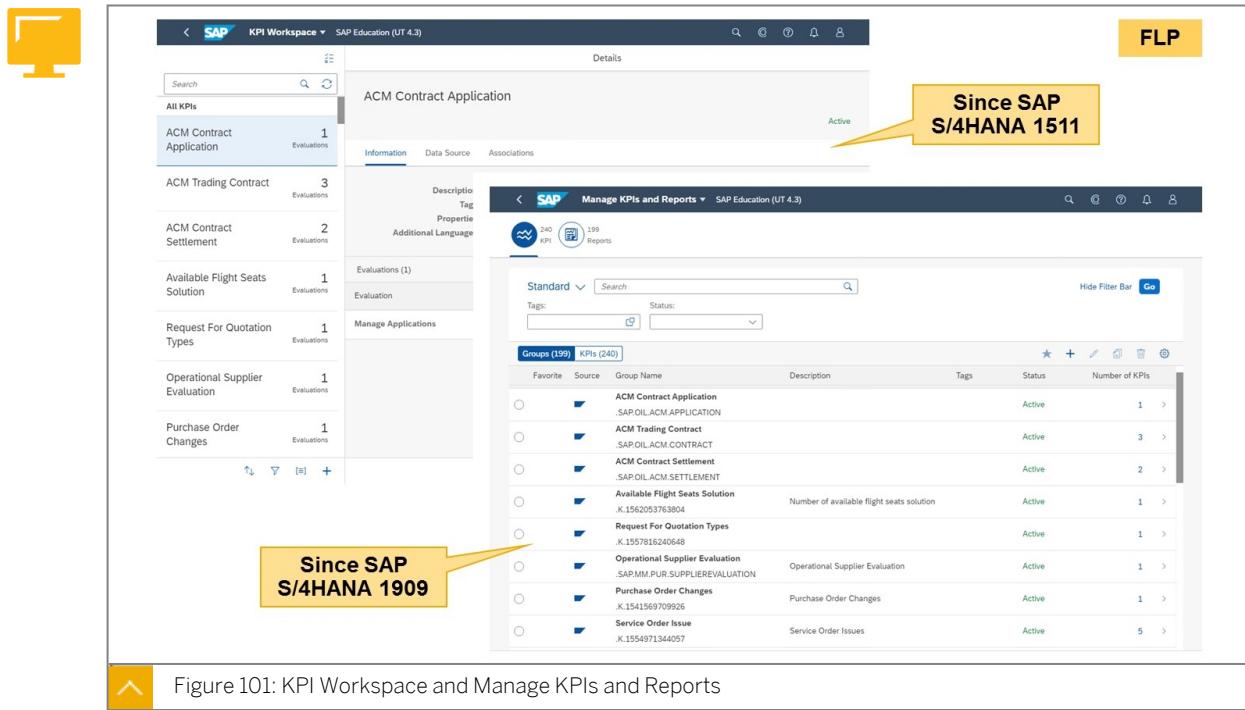


Figure 101: KPI Workspace and Manage KPIs and Reports

The area of business analytics specifically used in SAP Fiori is called SAP Smart Business. The core element there is the Key Performance Indicator (KPI). A KPI is build on top of ABAP CDS Views and define evaluations, tiles, and drill-downs. A drill-down is the visualization of a KPI offering predefined ways for the user to interact with the data – to drill-down.

Since SAP S/4HANA 1511, the *KPI Workspace* [F0818] is the central app for managing KPIs. Many other apps were created in each new release around this central app offering features such as defining KPI associations or reports. Since SAP S/4HANA 1909, *Manage KPIs and Reports* [F2814] is in place as the next generation of a central app in SAP Smart Business.

Note:

A predecessor of the *KPI Workspace* having the same name was already available in SAP Business Suite. The main difference is that not ABAP CDS Views but SAP HANA XS OData services are used as data source. This results in different and additional steps in managing KPIs such as authorizations.



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 SAP S/4HANA 1809
Support Package Stack	02 (05/2019) FP
Software Component Version	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_APOVRD_CDS	001	

Figure 102: SAP Fiori Analytical App Based on CDS View

In this example from the *SAP Fiori apps reference library*, you see the SAP S/4HANA 1809 as a back-end component. In addition, there is an OData service containing the shortage CDS in its name.



FLP

Analytical Queries

Show All ▾

View Name	View Description	Application Component
C_ABSLTNETBALSHTEXPSRQ	Balance Sheet FX Risk	FIN-FSCM-TRM
C_ACMTROGCONTROQUANTITIESQUERY	Consumpt	
C_ACTUALCASHFLOWANALYTICS	Actual Ca	
C_ACTUALUTILISBLINGDOCUMENTO	Actual Bill	
C_APCHASHDISCOUNTFORECAST	Cash disc	
C_APCHASHDISUTILIZATION	Account P	
C_APDAYSPIAVOUTSTDDIRECT	Days Payi	
C_APDAYSPIBLOUTSTDQINDRCT	Overview	
C_APFLEXIBLEAGING	Overview	
C_APFUTUREACCOUNTSPAYABLE	Aging Ana	
C_APINVOICEPROCESSINGTIME	Future Ac	
C_APINVOICEPROGANALYSIS	Accounts	
C_APINVOICEPR	Invoice Pr	
C_APJRNLENTIMAGINGGRID	Aging grid	
C_APJRNLENTIMAGINGGRID	Payables	

788 Views

Since SAP S/4HANA 1511

Views (44.274) Standard ▾

Name	Description	Status	Application Component	Data Category
/1BS/CDS_EXP_SALESORDER	SalesOrder	Not Released	BC-ESI-ESF-BSA	>
/1bs/sadl_cds_exp	Test View for CDS Exposure	Not Released	BC-ESI-ESF-BSA	>
/1BS/SADI_CDS_Param	Test View for CDS Exposure	Not Released	BC-ESI-ESF-BSA	>
/1BS/s4_o_data_namespace	AIF Namespace for oData Service	Not Released	CA-GTF-AIF	>
/BOBFiv_eprm_product_name	BOPF EPM View for product value help	Not Released	BC-ESI-BOF	>
/BOFUICV_BPAddress	BOPF: /BOFU/BuPa Address	Not Released	CA-EPT-BRC	>

44274 Views

Since SAP S/4HANA 1610

Figure 103: Query Browser and View Browser

The entry points for users to create their own evaluations are the *Query Browser [F1068]* available since SAP S/4HANA 1511 and the *View Browser [2170]* available since SAP S/4HANA 1610. Both show a list of ABAP CDS Views, which are available in the system. The difference is that the *Query Browser* limits the list on pure analytical views whereas the *View Browser* is used by more advanced users offering a list of all ABAP CDS Views. Not only the consumption views providing data for applications but all views of the whole virtual data model (VDM) are visible. This enables a more detailed view on the data source but also demands more knowledge of the topic.

No matter which browser is used, after selecting an ABAP CDS View as data source, the *Custom Analytical Queries [F1572]* app can be started to actually define a query handling dimension, tables, and charts. The result can be saved as a tile to access the query directly in the *SAP Fiori launchpad*.



Note:

More information about this topic can be found in S4H400 (SAP S/4HANA Embedded Analytics Foundation):

<https://training.sap.com/course/s4h400>

SAP HANA Enterprise Search

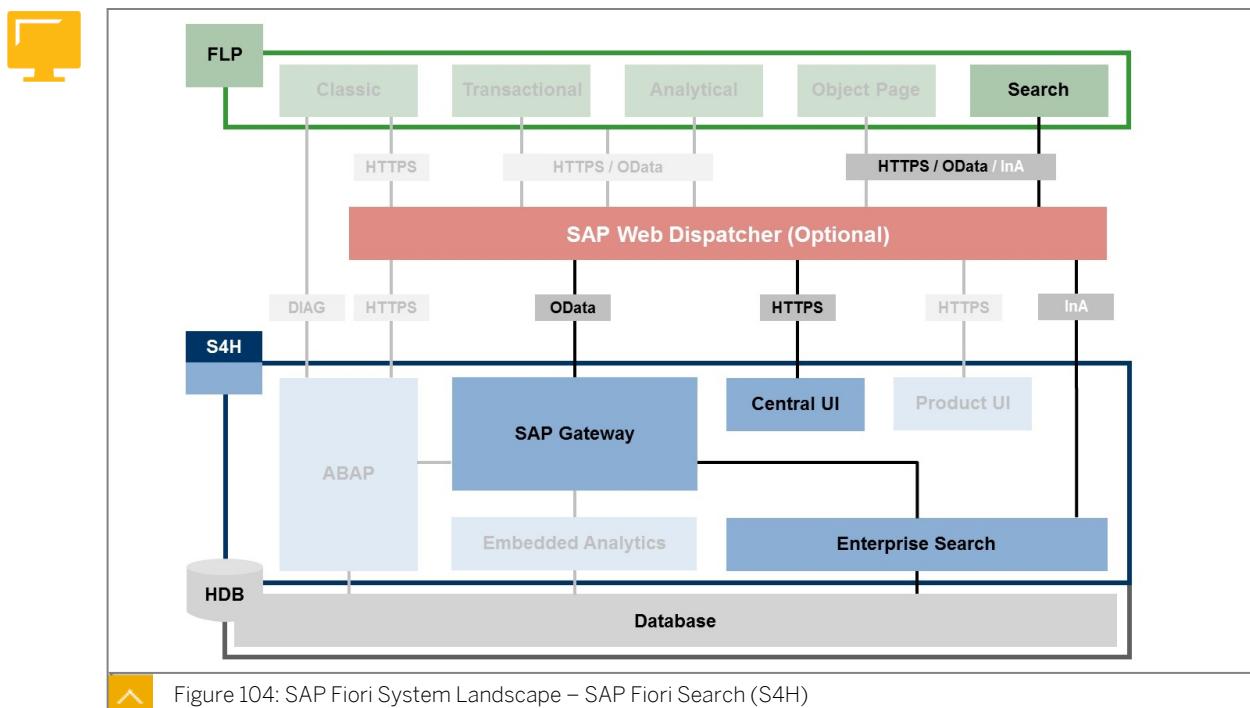


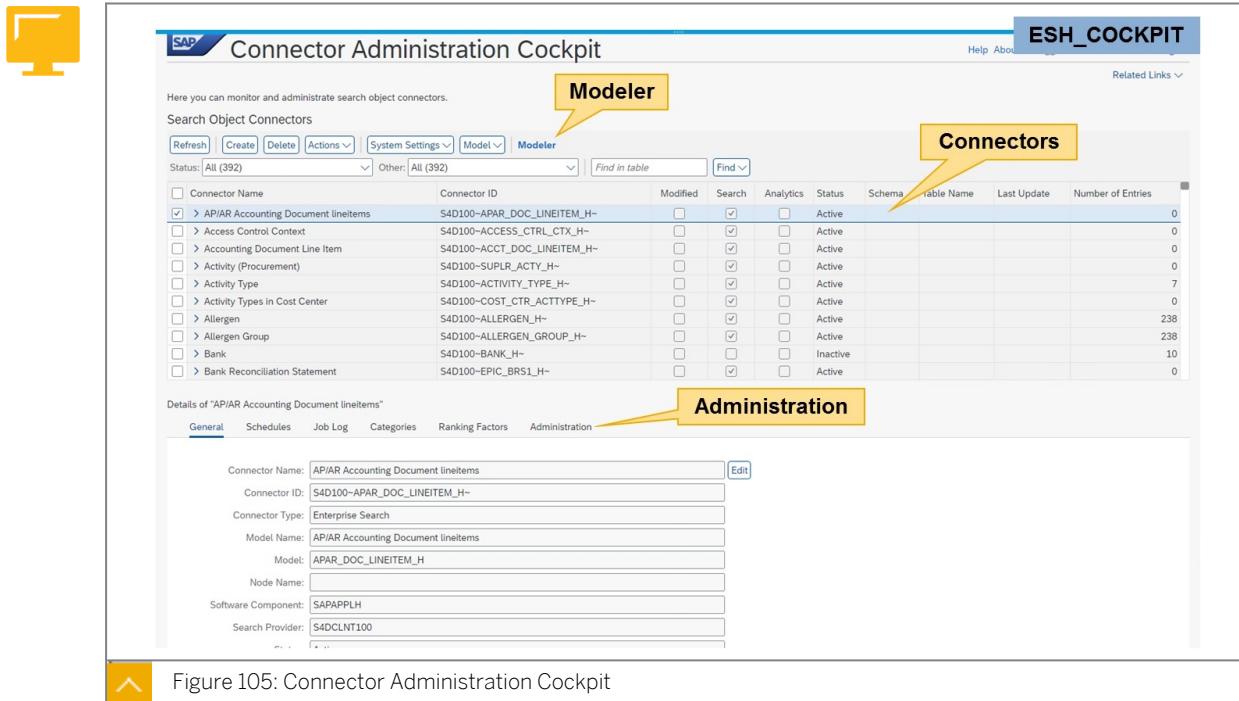
Figure 104: SAP Fiori System Landscape – SAP Fiori Search (S4H)

When users want to search for data in the *SAP Fiori launchpad*, the SAP Fiori search accesses the SAP HANA enterprise search, a successor of the SAP NetWeaver enterprise search. The protocol used for a direct network communication is the SAP proprietary Information Access (InA) protocol, which has some parallels to OData but is specialized for search requests and responses. Since SAP S/4HANA 1809, you can also use the SAP Gateway service ESH_SEARCH_SRV instead. So both, InA and OData are suitable ways accessing the SAP HANA enterprise search depending on the network landscape.



Hint:

Just by registering ESH_SEARCH_SRV, all search requests of the SAP Fiori search are routed through this service.



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 SAP Search and Classification (TREX) used by SAP NetWeaver enterprise search. The administration and handling is mainly the same when using SAP HANA or TREX. The following important transactions and applications work for both search engines:

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)

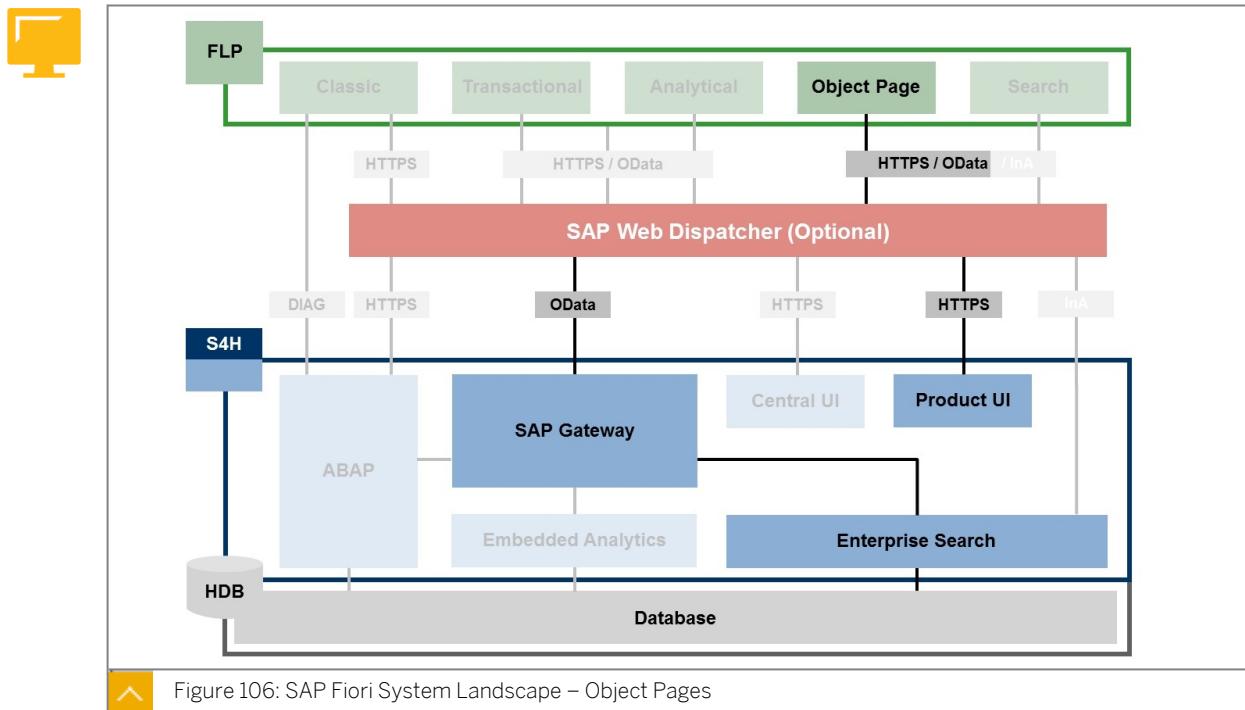
Area of the Implementation Guide (IMG) containing administration and configuration of the enterprise search

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.



Hint:

If you encounter duplicates of search connectors in the SAP Fiori launchpad search, please read SAP note [3007113](#) for possible solutions.



The UIs for object pages are part of the same product-specific UI add-ons that provide transactional and analytical apps. SAP Gateway manages again the OData services for the apps. ABAP Core Data Services (CDS) Views are the basis for all object pages in S4H. ABAP CDS Views utilize the search capabilities of the SAP HANA database by integrating with the SAP HANA enterprise search. That results in generated so-called CDS-based enterprise search (ES) connectors. Similar to the search connectors based on search models, they can be managed in the *Connector Administration Cockpit*. But the details about the source tables, views, and fields are not visible.



Hint:

When using ABAP CDS Views, there is no need to access search connectors directly via InA such as in the SAP Business Suite.

The screenshot shows two views of the SAP Fiori Manage Search Models app. The left view is a list titled 'Search Models (252)' with columns for Name, Application Area, Modified, and Status. Several entries are listed, including 'Employees', 'Journal Entry', 'Activity Types', 'Allergens', 'Allergen Groups', 'Allocation Tables', 'Payment Transaction References', 'Payments', 'Fixed Asset', 'Fixed Assets New Master Data', 'Employee-Workforce', and 'Banks'. The 'Banks' row is highlighted with a red border and has a yellow arrow pointing from it to the right view. The right view is a detailed edit screen for the 'Banks' search model. It shows the model's name is 'Banks', it belongs to the 'Information System' application area, and its status is 'Active'. The 'Enabled' checkbox is checked ('YES'). The 'General Information' tab is selected, showing details like 'Base CDS View: BNKA' and 'Data Control for private fields: OFF'. The 'Views' tab is also visible, listing various views and their configurations.

Figure 107: Manage Search Models

These details are available in the *Manage Search Models [F3036]* app since SAP S/4HANA 1909. There everything concerning a CDS-based ES-connector can be viewed such as fields, filters, relations, or even the source code of the ABAP CDS View. Since SAP S/4HANA 2020 it is also possible to de-/activate search connectors in the app, which is still possible in the *Connector Administration Cockpit*.



Note:

For more information about CDS-based ES-connectors, please read SAP note [2399860](#).

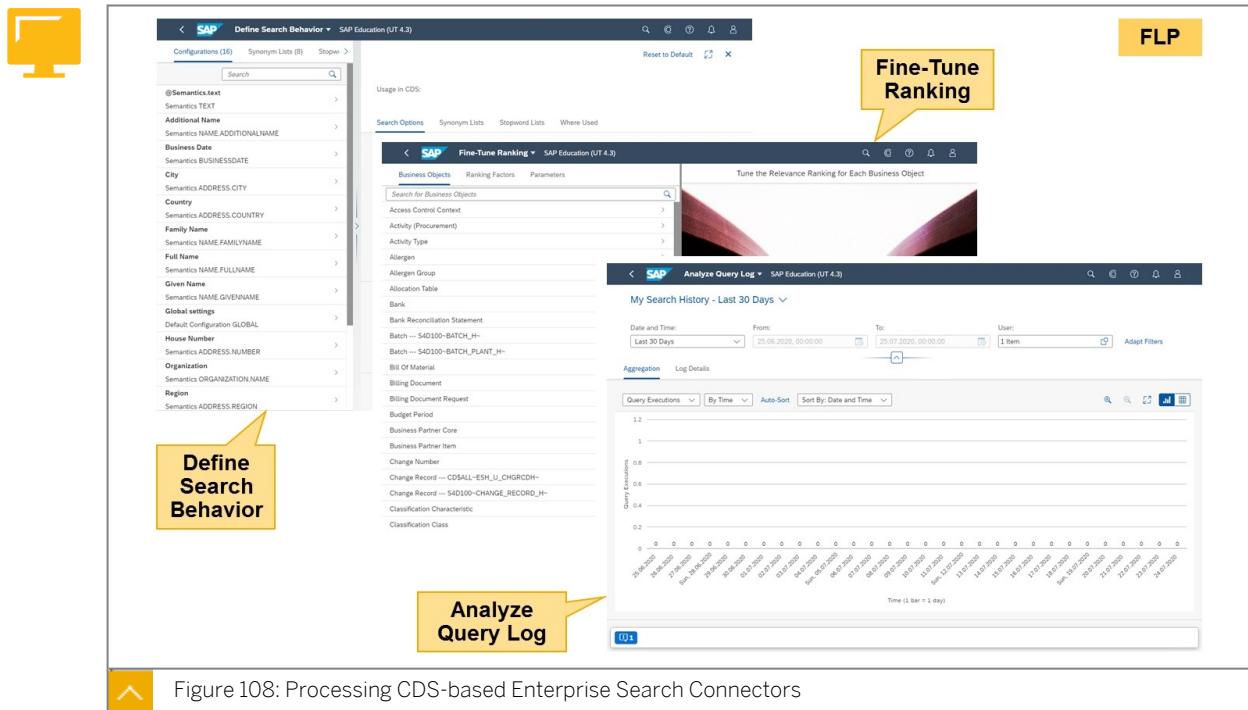


Figure 108: Processing CDS-based Enterprise Search Connectors

Since SAP S/4HANA 1809, additional SAP Fiori apps are available for managing the processing of CDS-based ES-connectors:

Analyze Query Log [F2571]

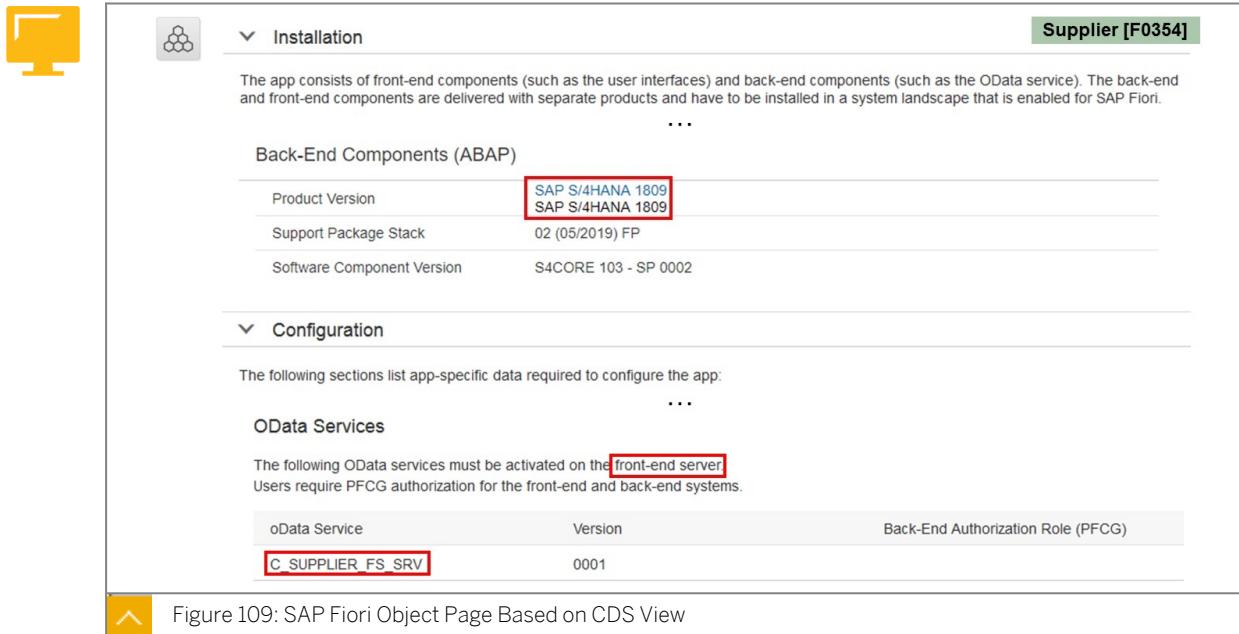
Evaluate log data containing user activities collected during searches, graphically in bar charts, or in tables.

Define Search Behavior (Synonyms) [F2700]

Display search configurations of business objects, create synonym and stop-word lists, and display where-used lists for configuration settings.

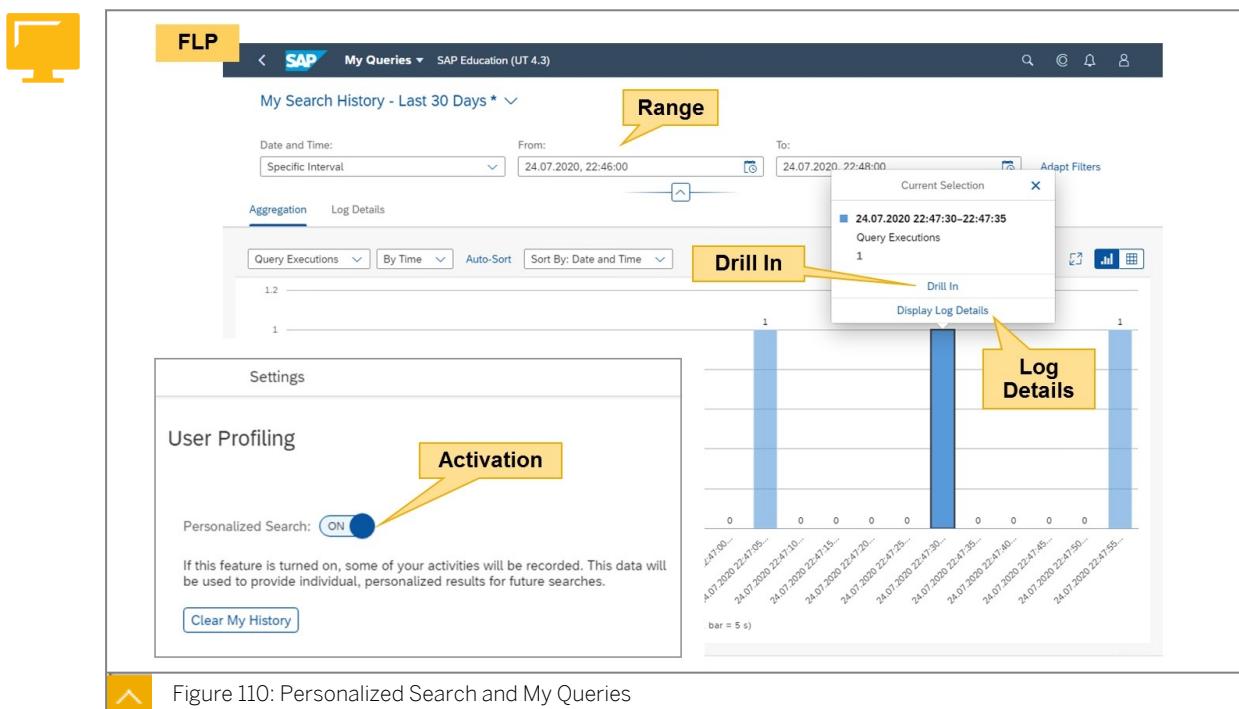
Fine-Tune (Search) Ranking [F2777]

Create and edit ranking factors and boosts, and test their effects immediately in a simulation.



The screenshot shows a SAP Fiori object page for a component named "Supplier [F0354]". The page is divided into sections: "Installation" and "Configuration". Under "Installation", it lists "Back-End Components (ABAP)" with a table showing "Product Version" as "SAP S/4HANA 1809" and "Support Package Stack" as "02 (05/2019) FP". Under "Configuration", it lists "OData Services" with a table showing "oData Service" as "C_SUPPLIER_FS_SRV" and "Version" as "0001". A callout box highlights the "front-end server" in the text: "The following OData services must be activated on the front-end server. Users require PFCG authorization for the front-end and back-end systems." The entire screenshot is framed by a yellow border.

In this example from the *SAP Fiori apps reference library*, you see the SAP S/4HANA 1809 as a back-end component. In addition, there is an OData service but without CDS in its name. Not every OData service using ABAP CDS Views is showing this in its name.



The screenshot shows the SAP Fiori launchpad interface. On the left, there is a "User Profiling" section with a "Personalized Search" toggle switch set to "ON". A callout box labeled "Activation" points to this switch. On the right, there is a "My Queries" app showing a bar chart of search activity. A callout box labeled "Range" points to the date range selector. Another callout box labeled "Drill In" points to a context menu over a bar in the chart. A third callout box labeled "Log Details" points to a button in the same context menu. The entire screenshot is framed by a yellow border.

Users of the *SAP Fiori launchpad* may activate a personalized search in the FLP settings under *User Profiling*. After activation, all search requests of the user in the SAP Fiori search are logged and made available via the *My Queries* [F3719] app. The user can view and evaluate her or his search activities and search terms either graphically in a bar chart or as a table. By default, the personalized search is deactivated and the history can be cleared at any time. These defaults for collecting user-specific data can be managed via the *Configure Personalized Search* [F2800] app.



Note:

More information about this topic can be found in UX200 (SAP Fiori – System Administration):

<https://training.sap.com/course/ux200>



LESSON SUMMARY

You should now be able to:

- Examine SAP Fiori for SAP S/4HANA
- Examine SAP S/4HANA Embedded Analytics
- Examine SAP HANA Enterprise Search

Examining SAP Fiori Development

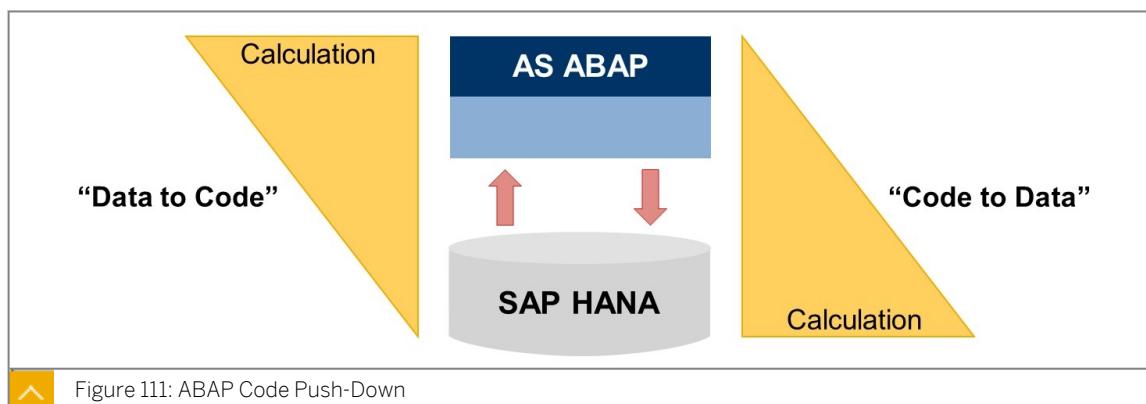


LESSON OBJECTIVES

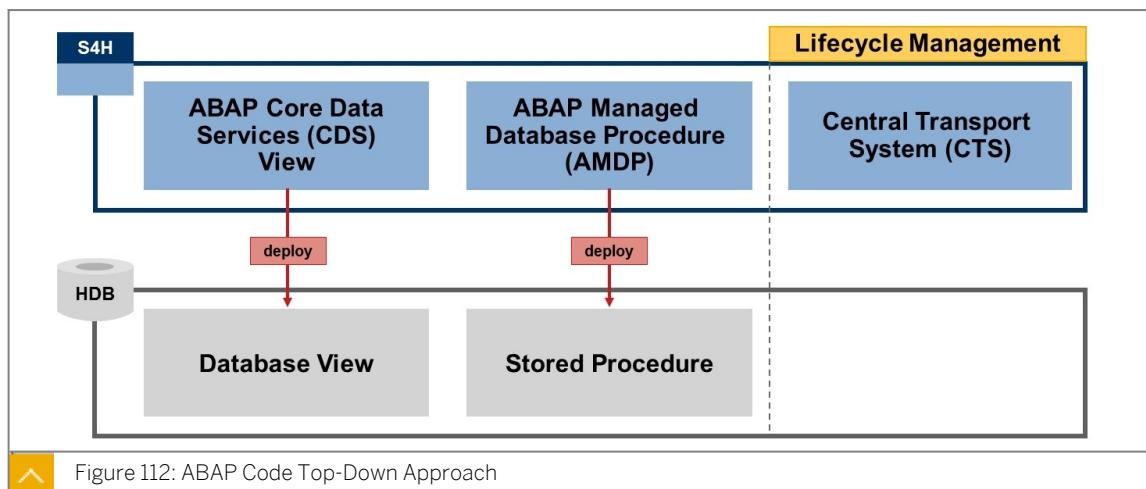
After completing this lesson, you will be able to:

- Examine Development Paradigm Shift
- Examine SAP Fiori Elements

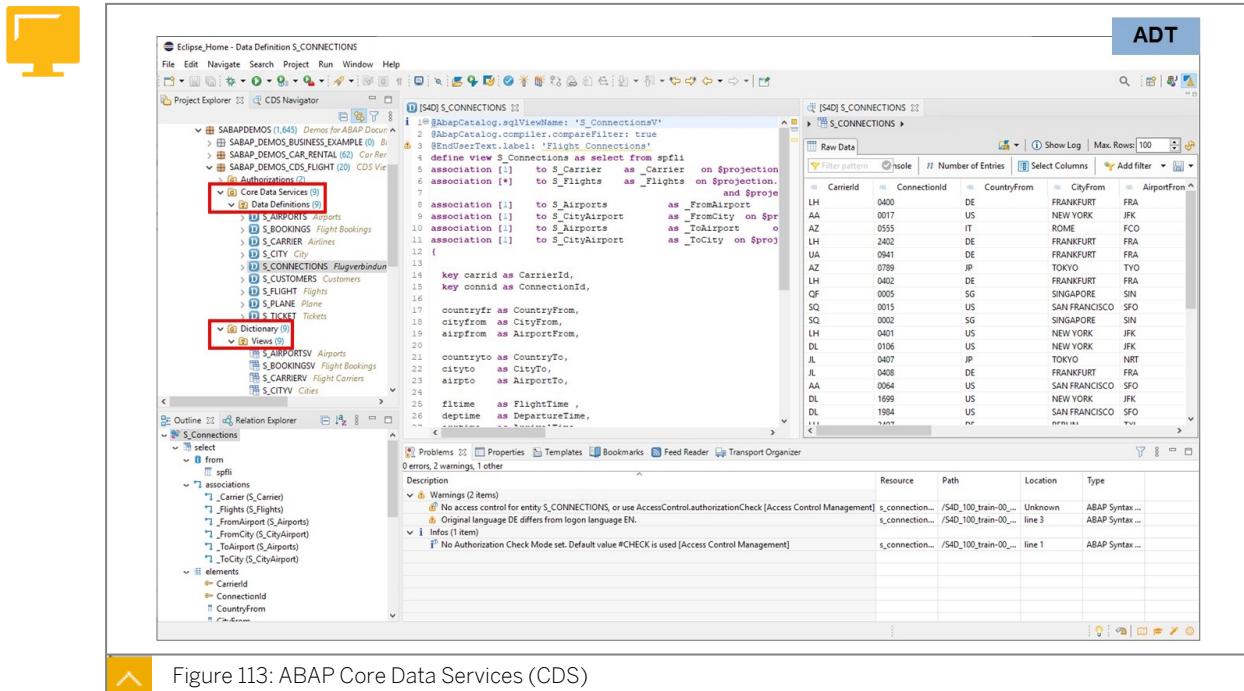
Development Paradigm Shift



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. 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, by pushing down calculations from AS ABAP in SAP HANA and only returning the results.



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. 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 SAP HANA views in the SAP 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.



On the one hand, ABAP CDS Views are classic database views compatible with any DB. On the other hand, they include side elements such as annotations, which can only be interpreted by an SAP 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.

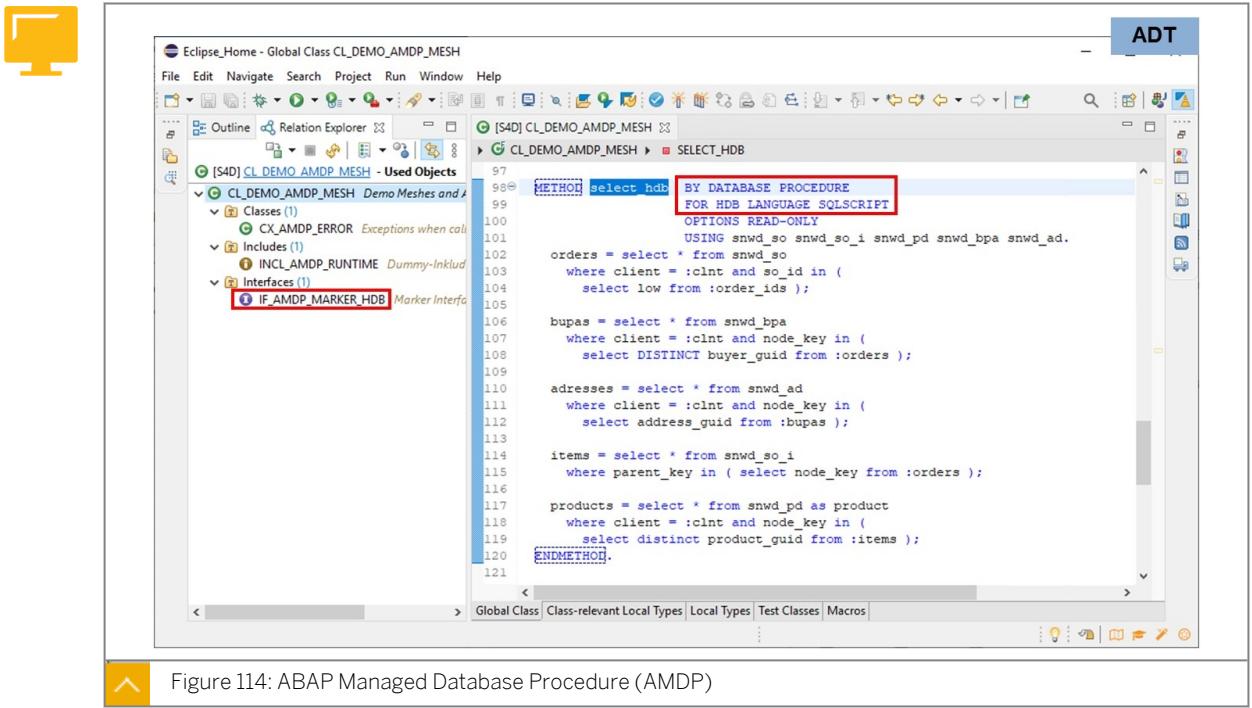
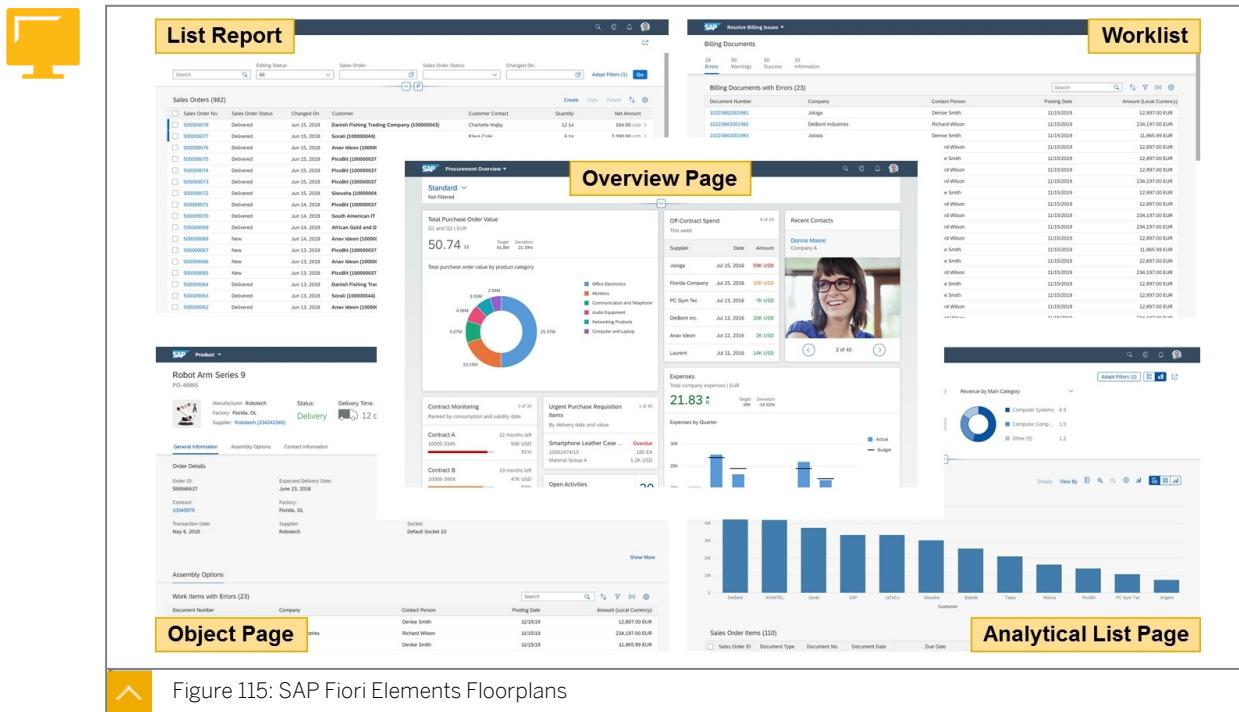


Figure 114: ABAP Managed Database Procedure (AMDP)

Like CDS Views, AMDPs are developed using ADT and can only be viewed using transaction SE80 in SAP GUI. An AMDP is a global class method 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. Use SQLScript 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 method name. Classes containing AMDPs implement the marker interface `IF_AMDP_MARKER_HDB`.

SAP Fiori Elements



SAP Fiori elements is a metadata-driven development of SAPUI5 applications and distinguishes the following floorplans:

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

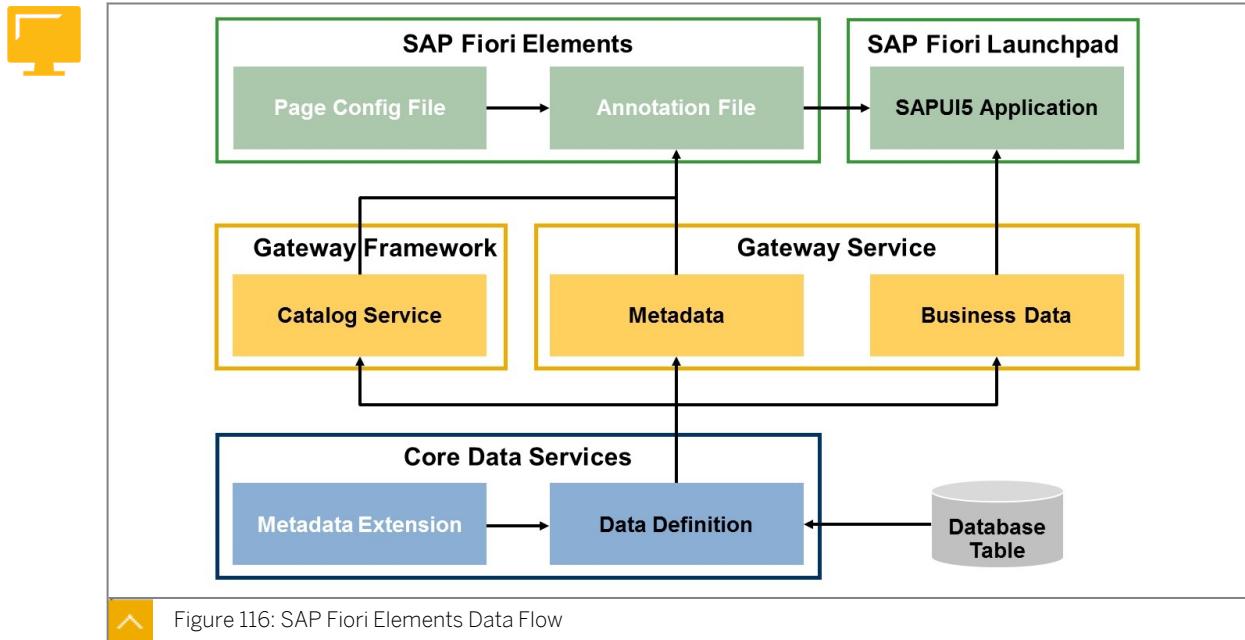
Visualizes large amount of data in cards with different formats for different types of content.

Analytical List Page

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

Worklist

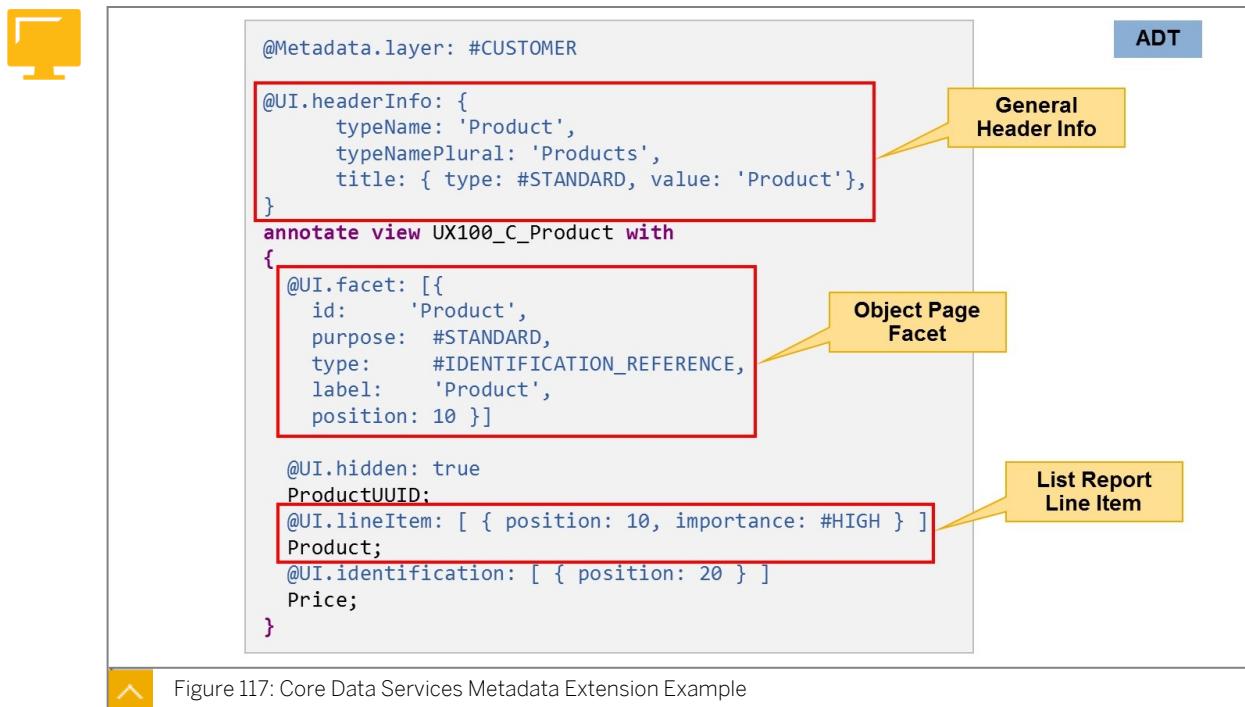
Displays a collection of items that a user must process.



The main source of the metadata for SAP Fiori elements are metadata extensions in CDS. Although you can add UI-annotations directly in data definitions, it is recommended to separate the definition of the UI from the data model definition.

The metadata of the Gateway service providing the business data provides the parts of the UI definition suitable to the OData standard. On top of that the /IWFND/CATALOGSERVICE of the Gateway framework provides all other parts, which are not allowed to be part of the metadata of an OData service.

In the SAPUI5 development environment, you can enrich the UI definition by adding annotations in the annotation.xml or by mapping page config files. These JSON-files describe certain parts of the UI in more detail.



This example shows the definition of a list report combined with an object page. At the top, some general header information for both SAP Fiori elements are set. Then an object page facet is created defining a frame for the data. The `ProductUUID` is hidden from the user. `Product` is the semantic key defined as line item in the list report. Finally `Price` is set as an identifier in the object page.

ADT

Service Binding: UX100_UI_PRODUCT_LIST

General Information
This section describes general information about this service binding

Binding Type: ODATA V2 - UI

Service Versions
Define service versions associated with the service binding

Version	Service Definition
0001	UX100_UI_PRODUCT_LIST

Service Version Details
View information on selected service version

Local Service Endpoint: Active

Service Information
Service URL: /sap/opu/odata/sap/UX100_UI_PRODUCT_LIST

Entity Set and Association
Product

Preview...

Standard

Price:

Search Adapt Filters Go

Products (123) Standard

Product ID	Category	Price	Weight
HT-1000	Notebooks	956.00 EUR	4.200 KG
HT-1001	Notebooks	1,249.00 EUR	4.500 KG
HT-1002	Notebooks	1,570.00 USD	4.200 KG
HT-1003	Notebooks	1,650.00 EUR	4.200 KG

Figure 118: Preview in Service Binding

Without using any SAPUI5 development environments, a preview of the list report can be provided directly via the service binding in the *ABAP Development Tools (ADT)*. This preview directly runs on the ABAP system the ADT is connected to but does not perform any deployment.

BAS

File Edit Selection View Go Run Terminal Help

Guided Development x

Select Project Page Type

All Guides Search guides

List Report Page

Add a custom action to a page using extensions

Add a custom filter to the filter bar

Add a new column as a contact view

Add a new column to a table

Add a new filter field to the Filter Bar

Add a progress indicator column to a table

Add a rating indicator column to a table

Add a smart micro chart to a table

Add custom columns to the table using extensions

Add semantic highlights to line items in tables based on their criticality

Add status colors and icons for a column

Configure multiple views

Enable condensed table layout

Enable draft toggle buttons

Enable export of table data

Enable flexible column layout

Enable multiple selection in tables

Add a custom controller for the action

Guide information Step 1 Create a new controller for the action Step 2 Add an extension definition to the manifest

Create a new controller for the action

Create a new controller for the action

Page Type Parameters Select the Page Type to which you want to attach new custom action.

New Function Parameters Enter a Function Name that will be the name of your new custom function.

Function Name Input a function name

Insert Snippet Copy Reset

```

1 sap.ui.controller("ext.controller.ListReportExt", {
2   onCustomAction : function(oEvent) {
3     alert('onCustomAction');
4   }
5 });
6

```

Back Next Exit Guide

Figure 119: SAP Fiori Tools – Guided Development

The SAP Fiori tools as part of the SAP Business Application Studio and available in *Visual Studio Code* are the recommended tool for developing the SAP Fiori elements artifacts on UI-level. The *Guided Development* of the SAP Fiori tools offers many guides to add additional features to the application. A guide offers a documentation of steps, screenshots of the expected result, code-snippets, and often a wizard for applying the feature directly in an application in the workspace.



LESSON SUMMARY

You should now be able to:

- Examine Development Paradigm Shift
- Examine SAP Fiori Elements

Learning Assessment

1. Which system holds the SAP Fiori launchpad in a hub deployment?

2. Which system component is split between front-end server and back-end server?

3. Which ABAP repository element contains the service logic?

4. How is the implementation of SAP Fiori apps delivered by SAP?

5. Which data processing capabilities does SAP HANA combine?

6. Which service areas are provided by the SAP HANA platform?

Choose the correct answers.

- A Application Services
- B Gateway Services
- C Integration Services
- D Connectivity Services
- E Processing Services
- F Database Services

7. What is SAP S/4HANA?

8. Which areas were re-imagined in SAP S/4HANA?

Choose the correct answers.

- A User experience
- B User management
- C Business processes
- D System configuration
- E Database models

9. Which system part provides the analytical data in SAP S/4HANA?

10. Which applications can be used as entry point to create evaluations?

Choose the correct answers.

- A Query Browser
- B My Queries
- C View Browser
- D Analyze Query Log

11. What is the data source definition for the SAP Fiori search?

12. What is the data source definition for object pages?

13. What is used in SAP S/4HANA to read data from the database?

14. What is used in SAP S/4HANA to run native code on the database?

15. Which are floorplans in SAP Fiori elements?

Choose the correct answers.

- A Object Page
- B Master Detail Page
- C List Report
- D Overview Page
- E Dynamic Page

16. In which layers is the metadata for SAP Fiori elements created?

Choose the correct answers.

- A User Interface
- B OData Service
- C Core Data Services
- D Database

Learning Assessment - Answers

1. Which system holds the SAP Fiori launchpad in a hub deployment?

Front-End Server (FES)

2. Which system component is split between front-end server and back-end server?

SAP Gateway

3. Which ABAP repository element contains the service logic?

Data Provider Class (DPC)

4. How is the implementation of SAP Fiori apps delivered by SAP?

Updates of the back-end server

5. Which data processing capabilities does SAP HANA combine?

Analytical and transactional

6. Which service areas are provided by the SAP HANA platform?

Choose the correct answers.

- A Application Services
- B Gateway Services
- C Integration Services
- D Connectivity Services
- E Processing Services
- F Database Services

Correct. The SAP HANA platform is offering application, integration, processing, and database services.

7. What is SAP S/4HANA?

The fourth generation of the SAP Business Suite based on SAP HANA renewing enterprise business

8. Which areas were re-imagined in SAP S/4HANA?

Choose the correct answers.

- A User experience
- B User management
- C Business processes
- D System configuration
- E Database models

Correct. User experience, business processes, system configuration, and database models were re-imagined in SAP S/4HANA.

9. Which system part provides the analytical data in SAP S/4HANA?

SAP S/4HANA Embedded Analytics

10. Which applications can be used as entry point to create evaluations?

Choose the correct answers.

- A Query Browser
- B My Queries
- C View Browser
- D Analyze Query Log

Correct. The entry points are Query and View Browser.

11. What is the data source definition for the SAP Fiori search?

Search connectors

12. What is the data source definition for object pages?

ABAP Core Data Services (CDS) View

13. What is used in SAP S/4HANA to read data from the database?

ABAP Core Data Services (CDS)

14. What is used in SAP S/4HANA to run native code on the database?

ABAP Managed Database Procedures (AMPD)

15. Which are floorplans in SAP Fiori elements?

Choose the correct answers.

- A Object Page
- B Master Detail Page
- C List Report
- D Overview Page
- E Dynamic Page

Correct. Object page, list report, and overview are floorplans in SAP Fiori elements.

16. In which layers is the metadata for SAP Fiori elements created?

Choose the correct answers.

- A User Interface
- B OData Service
- C Core Data Services
- D Database

Correct. In the user interface and in the Core Data Services is the metadata created.

Lesson 1

Managing SAP Fiori Content

119

Lesson 2

Creating SAP Fiori Spaces and Pages

125

Lesson 3

Creating SAP Fiori Groups

131

Lesson 4

Managing SAP Fiori Catalogs

135

Lesson 5

Creating Business Catalogs

141

Lesson 6

Creating Target Mappings

147

Lesson 7

Creating Technical Catalogs

151

Lesson 8

Creating Replicable Catalogs

159

UNIT OBJECTIVES

- Manage SAP Fiori content
- Create SAP Fiori spaces and pages
- Create SAP Fiori groups
- Manage SAP Fiori catalogs

- Create business catalogs
- Create target mappings
- Create technical catalogs
- Create replicable catalogs

Managing SAP Fiori Content

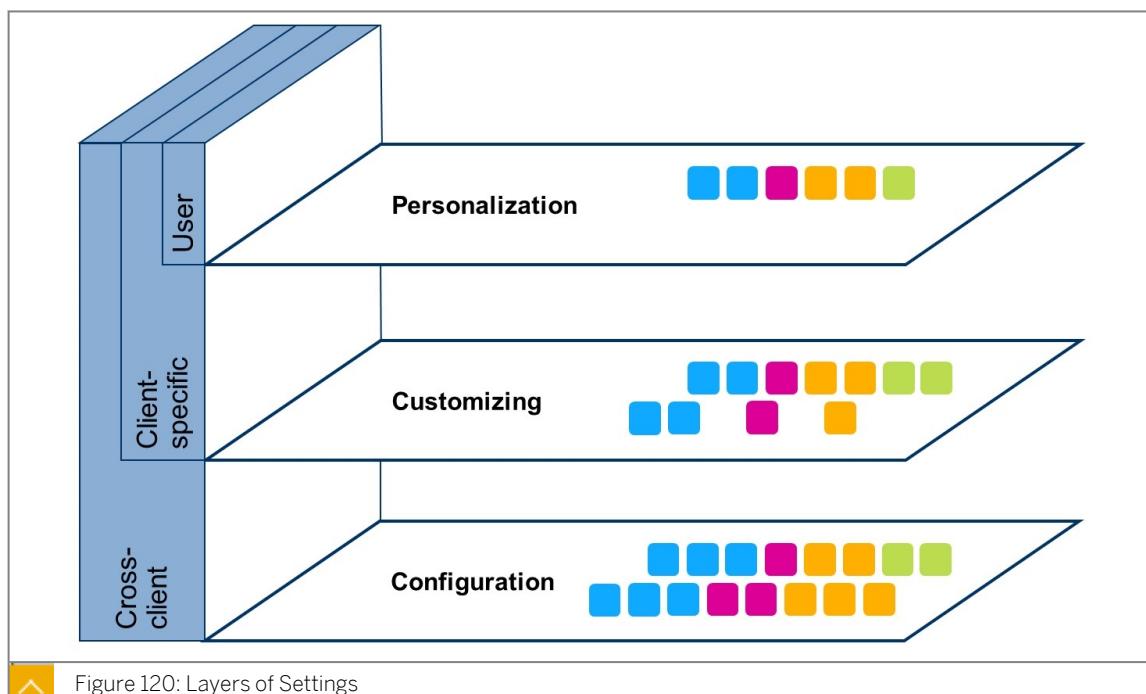


LESSON OBJECTIVES

After completing this lesson, you will be able to:

- Manage SAP Fiori content

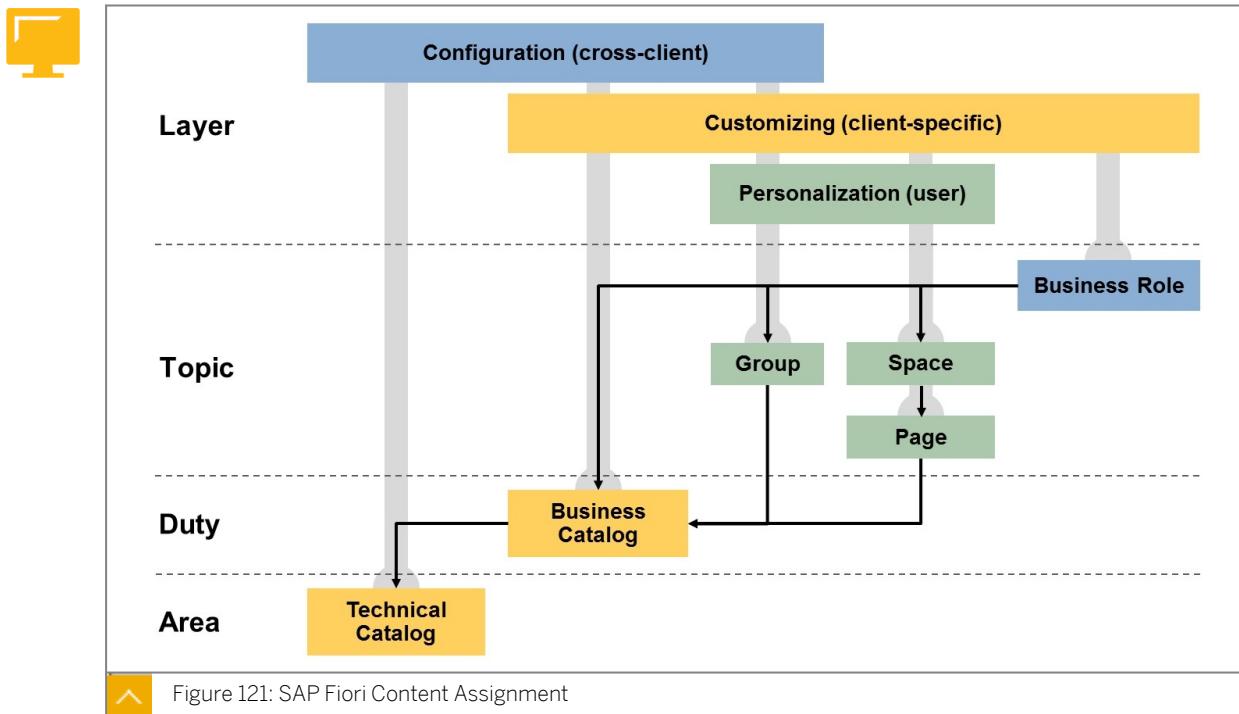
Content Assignment



The settings for the *SAP Fiori launchpad (FLP)* are organized in the three layers:

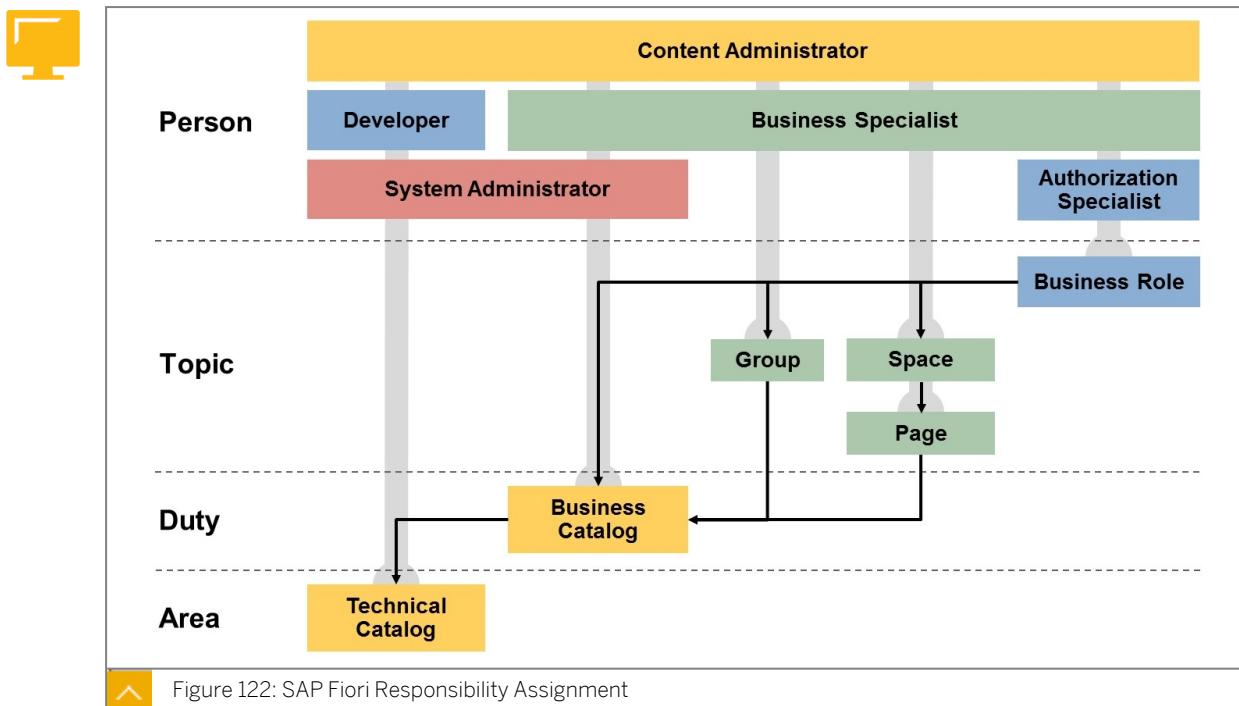
- The basic layer configuration consists of settings valid for all clients of an AS ABAP.
- The layer customizing builds on the configuration and fine tunes the settings per client.
- End users can then personalize these settings in their personalization layer.

Every layer can only change the elements made available by their sub-level or reduce elements for usage in the next layer. Users can only personalize apps that were originally provided in the configuration layer and made available in the customizing layer.



Catalogs are the basis of the SAP Fiori content in the FLP. Technical catalogs define apps per solution area, whereas business catalogs reference these apps per duty of the user. Groups, spaces, and pages reference tiles and links per business topic from business catalogs. Business catalogs, groups, and spaces can then be assigned via business roles to users.

A technical catalog can only be defined in configuration. Business catalogs can be defined in configuration and customizing. Groups are the only elements available in all three layers. Spaces and pages are defined in customizing and are changeable in personalization. Business roles are assigned to users in customizing .



The actual people creating, changing, or managing content for the *SAP Fiori launchpad* cannot be easily identified. Every customer defines their own roles or combines multiple tasks in one role. In general, the following roles can be distinguished:

Authorization Specialist

- Authorizes apps to users
- Generates authorization profiles based on business catalogs
- Creates roles and assigns business catalogs, groups, and spaces

Business Specialist

- Defines and handles business processes
- Creates roles and assigns business catalogs, groups, and spaces
- Creates business catalogs, groups, spaces, and pages

Content Administrator

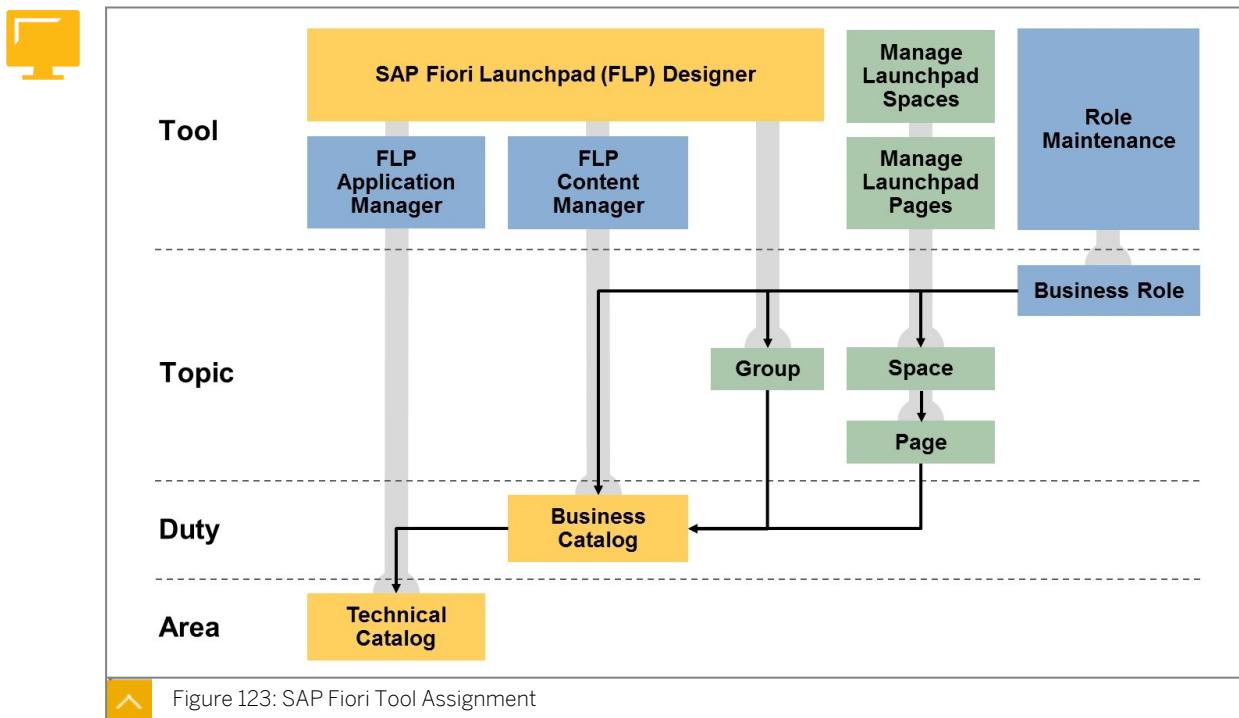
- Manages the whole SAP Fiori content
- Creates roles and assigns business catalogs, groups, and spaces
- Creates business catalogs, groups, spaces, and pages
- Creates technical catalogs for existing applications

Developer

- Develops and extends applications
- Provides data-driven navigation for SAP Fiori apps
- Creates technical catalogs for new or extended applications

System Administrator

- Administrates the SAP system on a basis level
- Provides the technical foundation for managing SAP Fiori content
- Creates technical and business catalogs for configuration tasks



The personalization is performed by each user in the FLP. For Configuration and customizing, each element of the SAP Fiori content has its own tool. The only exception is the *SAP Fiori launchpad designer*, which can handle groups, technical and business catalogs alike:

Role Maintenance (PFCG)

ABAP transaction for roles available for decades

SAP Fiori Launchpad Designer (FLPD)

Standalone SAPUI5 application for groups, technical and business catalogs available since SAP_UI 7.40

SAP Fiori Launchpad Content Manager (FLPCM)

ABAP transaction for business catalogs available since SAP_UI 7.53

SAP Fiori Launchpad Application Manager (FLPAM)

ABAP transaction for technical catalogs available since SAP_UI 7.55

Manage Launchpad Spaces

SAP Fiori launchpad app for spaces available since SAP_UI 7.55

Manage Launchpad Pages

SAP Fiori launchpad app for pages available since SAP_UI 7.55

Role Maintenance

The screenshot shows two main panels. On the left, the 'User Interface - Fiori Launchpad Admin' catalog is displayed, containing several tiles for managing Fiori Launchpad. On the right, the 'Role' maintenance screen shows the 'SAP_FLP_ADMIN' role assigned to the 'User Interface - Fiori Launchpad Administration' catalog. A yellow arrow points from the catalog in the FLP interface to the catalog in the PFCG interface.

Figure 124: SAP Fiori Administration Role – Catalogs

Since SAP UI 7.55, the SAP_FLP_ADMIN role includes all tools for maintaining SAP Fiori. The core are the two catalogs SAP_BASIS_BC_UI_FLA and SAP_BASIS_BC_UI_FLD. These enable the user to maintain SAP Fiori via the *SAP Fiori launchpad*.

The screenshot shows two main panels. On the left, the 'Fiori Launchpad' space is displayed, containing several tiles for managing Fiori Launchpad. On the right, the 'Role' maintenance screen shows the 'SAP_FLP_ADMIN' role assigned to the 'Fiori Launchpad' group. A yellow arrow points from the space in the FLP interface to the group in the PFCG interface.

Figure 125: SAP Fiori Administration Role – Space and Group

To make it even easier, a space and a group providing tiles for all tools are also part of the SAP_FLP_ADMIN role. In addition, plenty of additional ABAP transactions around SAP Fiori are provided in the SAP easy access menu.

**Note:**

The SAP_FLP_ADMIN single role replaces the SAP_UI2_ADMIN composite role.



SAP Change Roles PFCG

Role: **Z_00_BR_TRAINING** Obsolete
 Description: Training 00
 Target System: No destination

Description ● Menu ● Applications ● Authorizations ● User ● Personalization

+ Transaction From Menus

Hierarchy	Node Details	Catalog ID
Role Menu	Launchpad Catalog	SAP BASIS BC UI_FLA User Interface - Fiori Launchpad Admin
SAP Fiori Launchpad >	Launchpad Group	SAP BASIS BCG_UI_FLP Fiori Launchpad
SAP BW >	Launchpad Space	SAP BASIS SP UI_FLP Fiori Launchpad
Other >		

Figure 126: Role Maintenance

Catalogs, groups, and spaces can be added to user roles using the role maintenance transaction PFCG:

1. In the *Menu* tab, expand the *Transaction* button (or what was last added to the menu) and choose *SAP Fiori Launchpad*.
2. Choose what you want to add to the role.
3. In the following dialog box, enter the ID and continue.

**Hint:**

If a newly created element is not available in the dialog box, restart the PFCG.



LESSON SUMMARY

You should now be able to:

- Manage SAP Fiori content

Unit 4

Lesson 2

Creating SAP Fiori Spaces and Pages



LESSON OBJECTIVES

After completing this lesson, you will be able to:

- Create SAP Fiori spaces and pages

Manage Launchpad Spaces



The screenshot displays two separate instances of the SAP Fiori launchpad (FLP) application, each showing a list of 'Spaces'.

Top Instance (Customer-Created):

- Header: Manage Launchpad Spaces - SAP Education (UT 4.4)
- Filter: Customer-Created (highlighted with a red box)
- Table:
 - Spaces (1)
 - ID / Description: Z_00_SP_RESEARCH
 - Title: Research 00
 - Roles: 1
 - Created By / Created...: UX100 TRAIN-00 03.08.2021
 - Changed By / Chang...: UX100 TRAIN-00 03.08.2021

Bottom Instance (SAP-Delivered):

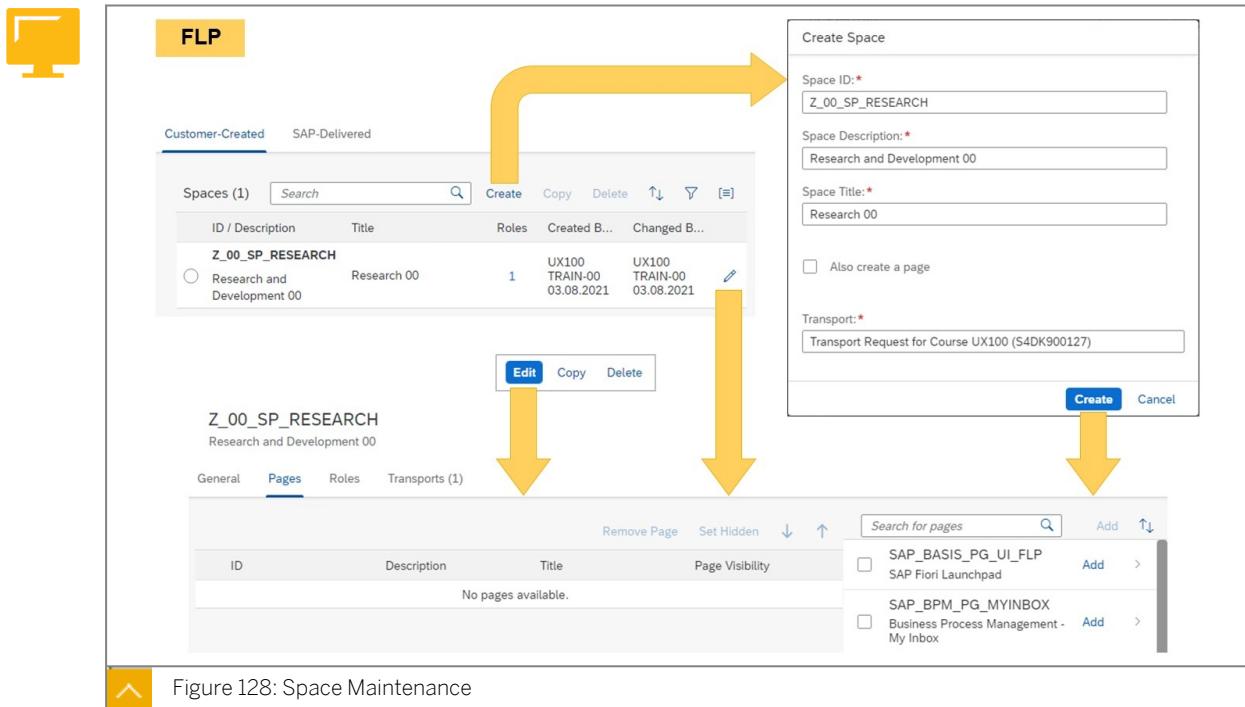
- Header: Manage Launchpad Spaces - SAP Education (UT 4.4)
- Filter: SAP-Delivered (highlighted with a red box)
- Table:
 - Spaces (211)
 - ID / Description: SAP_FIN_SP_JVA_ACCOUNTANT
 - Title: Joint Venture Accounting

Left Side: A large yellow arrow points upwards from the left side towards the first instance, indicating the creation of a customer space.

Right Side: A large yellow arrow points downwards from the right side towards the second instance, indicating the selection of a delivered space.

Figure 127: Manage Launchpad Spaces

Manage Launchpad Spaces is an SAP Fiori launchpad (FLP) app based on SAPUI5. It shows all spaces delivered by SAP and allows customers to create their own spaces.



When creating or changing a space, a customizing transport request is mandatory. It is not possible to do any local customizing not assigned to a transport request. Every space has an ID, description, title, and at least one page. When customers create their own space, they can assign pages by SAP or their own pages. Customer space IDs must start with Z or Y and should contain the abbreviation SP for space.



Note:
If you want to sort the order of spaces in the FLP, please read SAP note [3012443](#).

Manage Launchpad Pages

The screenshot shows the SAP Fiori Launchpad (FLP) interface for managing launchpad pages. It features two main tabs at the top: "Customer-Created" (highlighted with a red box and a yellow arrow pointing up) and "SAP-Delivered" (highlighted with a red box and a yellow arrow pointing down). Below each tab is a table listing pages. The "Customer-Created" table contains one entry: "Z_00_PG_ENGINEERING" (Engineering Overview 00). The "SAP-Delivered" table lists several standard SAP pages, such as "SAP_BASIS_PG_UI_FLP" (Fiori Launchpad), "SAP_BPM_PG_MYINBOX" (My Inbox), and "SAP_CDA_PG_MIG" (Data Migration).

Figure 129: Manage Launchpad Pages

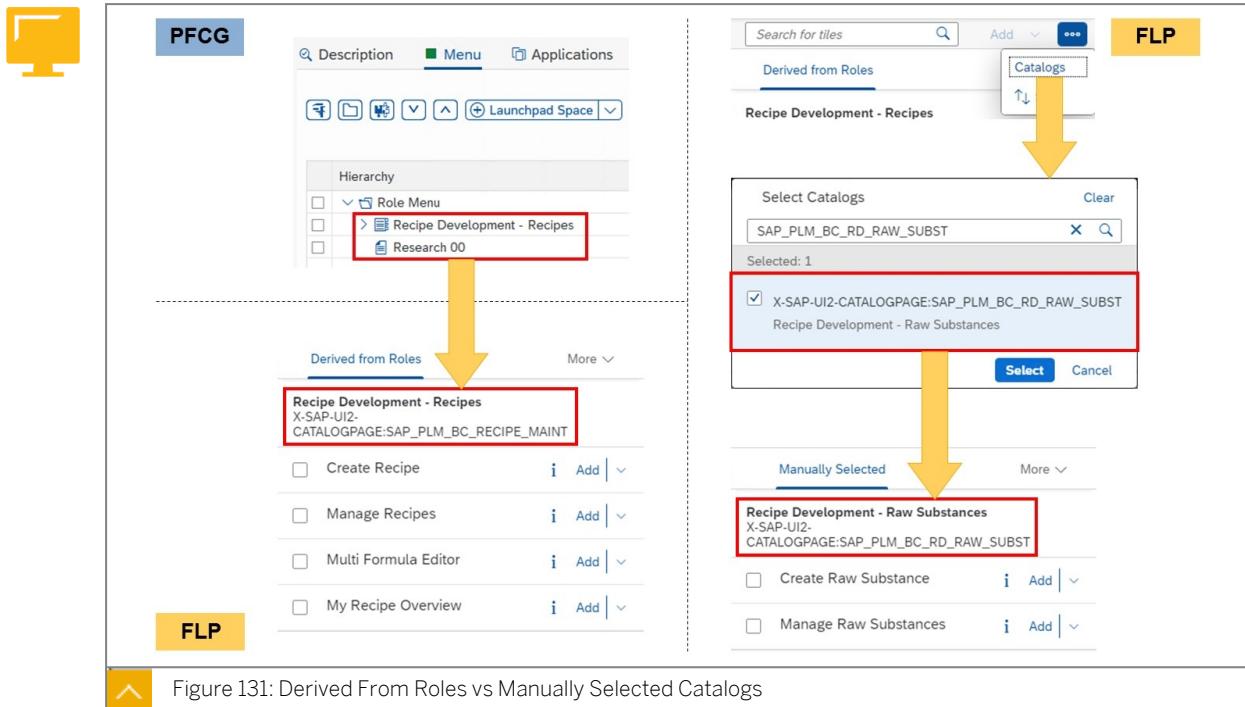
Manage Launchpad Pages is an SAP Fiori launchpad (FLP) app based on SAPUI5. It shows all pages delivered by SAP and allows customers to create their own pages.

The screenshot shows the SAP Fiori Page Maintenance application. On the left, there's a list of existing pages under the "Customer-Created" tab, with one page selected: "Z_00_PG_ENGINEERING" (Engineering Overview 00). On the right, a "Create Page" dialog is open, prompting for details like ID, description, title, and transport request. A large yellow arrow points from the "Create" button in the main list to this dialog. Below the main list, there's a "Page Content" section where a "Page Preview" is shown with edit, copy, and delete options. Another yellow arrow points from the "Edit" button in the preview to the "Create Page" dialog. At the bottom, there's a "Select Role Context (all)" button and a "Hide Catalogs" link.

Figure 130: Page Maintenance

When creating or changing a page, a customizing transport request is mandatory. It is not possible to do any local customizing not assigned to a transport request. Every page has an ID, description, title, and at least one section. Sections are an integral part of pages and cannot be reused in other pages. Sections group tiles by the level of details of a business task.

When customers create their own page, they can assign it to any of their own spaces. Customer page IDs must start with Z or Y and should contain the abbreviation PG for page.



Before adding apps to a page, the corresponding space should be assigned to a role. Business catalogs can be derived from this role to appear as sources for tiles in a page in *Manage Launchpad Pages*. This means, tiles can easily be assigned to the page and previewed.

It is also possible to select business catalogs manually in *Manage Launchpad Pages*. Hidden behind the ... button, all business catalogs can be selected as sources for tiles for the page. Tiles added in this way will show the warning *Out of role context* and will not be displayed in the preview. The prerequisite for the user to see the tile in the FLP is that the corresponding business catalog is also part of the user master record. In other words, the source catalog of the tile must be assigned to any role also assigned to the user.

The screenshot shows the SAP Fiori Launchpad interface. At the top, there's a navigation bar with 'SAP Home', 'SAP Education (UT 4.4)', and various dropdown menus like 'Fiori Launchpad', 'Cross Topic', 'Line of Business', and 'Reference Apps'. On the right side, there's a yellow button labeled 'FLP'. The main area is titled 'Daily Task' and contains three sections: 'Sourcing and Procurement', 'Stock Management', and 'Stock Monitoring'. Each section has several tiles representing different tasks. A large yellow arrow points from the top-left tile in the 'Stock Management' section towards the bottom-right tile in the 'Stock Monitoring' section, illustrating the flow from 'Insight' to 'Action'.

- **1** space per role
- **1-5** pages per space
- **2-5** sections per page
- **3-7** tiles per section
- **< 25** tiles per page

Figure 132: Best Practices for Spaces and Pages

Providing spaces and pages to users should support them in easily finding and accessing their most important content. So it is important to limit the number of tiles to a meaningful level. These are the [best practices for managing spaces and pages](#):

- A business role should only have one space, because both are targeting one business topic.
- A space should consist of one to five pages providing one business task on one page.
- A page should consist of two to five sections ordered from insight (top-left) to action (bottom right) tasks.
- A section should consist of three to seven tiles ordered in a logical way based on the topic.
- In total there should not be more than 25 tiles per page to keep it manageable.



LESSON SUMMARY

You should now be able to:

- Create SAP Fiori spaces and pages

Unit 4

Lesson 3

Creating SAP Fiori Groups



LESSON OBJECTIVES

After completing this lesson, you will be able to:

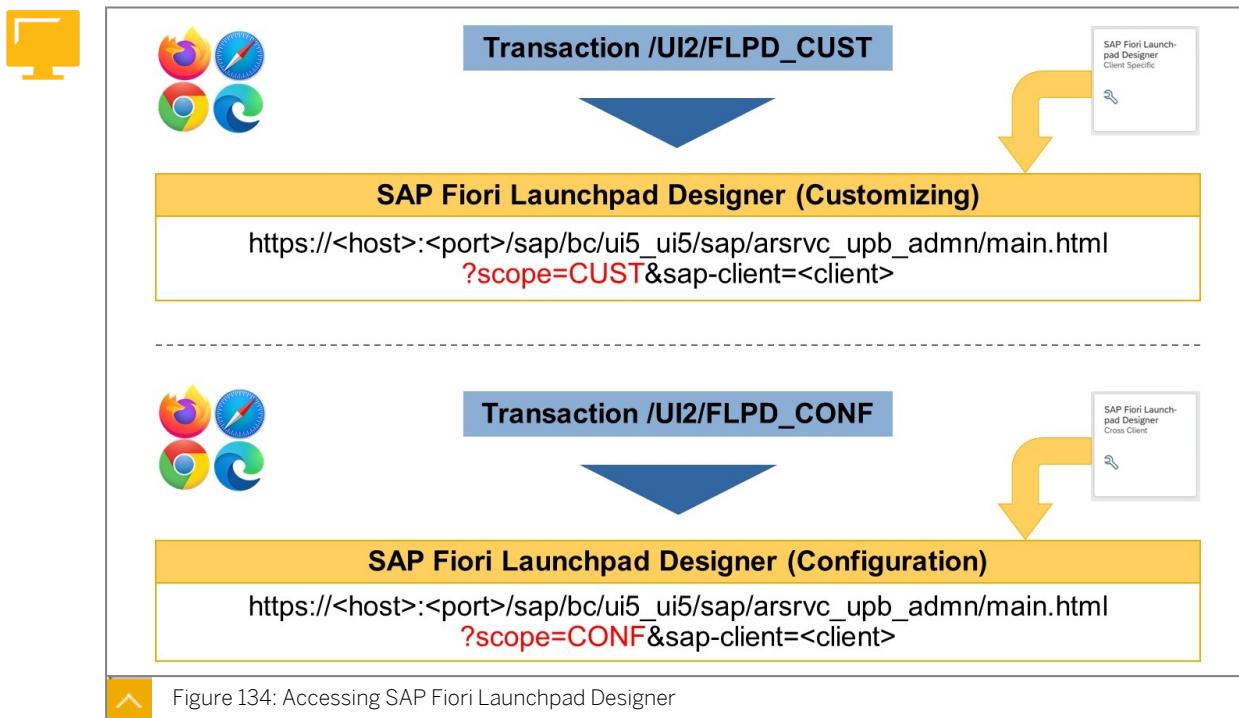
- Create SAP Fiori groups

SAP Fiori Launchpad Designer



Figure 133: SAP Fiori Launchpad Designer

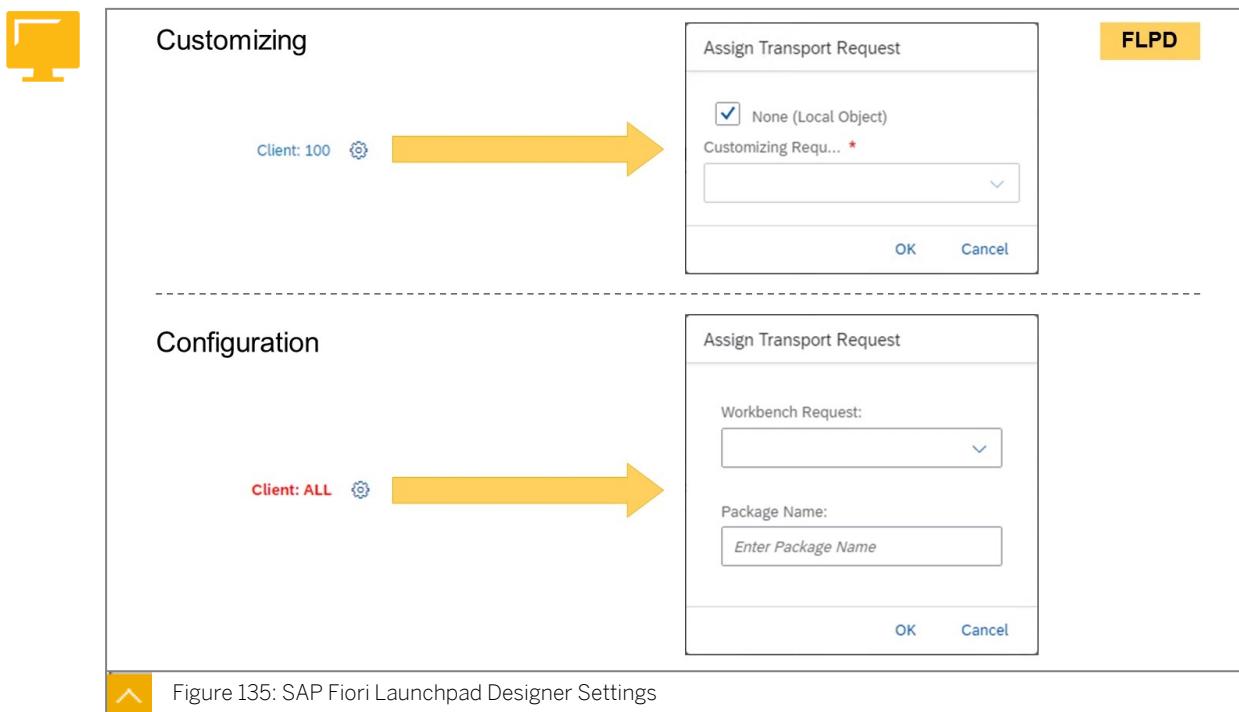
The tool *SAP Fiori Launchpad Designer* (*FLPD*) offers an environment for customizing and configuration. It is an SAPUI5 application following the SAP Fiori design guidelines, but is not an SAP Fiori launchpad (FLP) app documented in the *SAP Fiori apps reference library*. It is part of the central UI of every AS ABAP since 7.40, similar to the FLP.



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&sap-client=<client>

The FLPD offers the same options for customizing (client-specific) and configuration (cross-client). 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 must be saved in a development package and attached to a workbench request. Settings are accessible for customizing and configuration in the upper right corner of the FLPD.

Group Maintenance

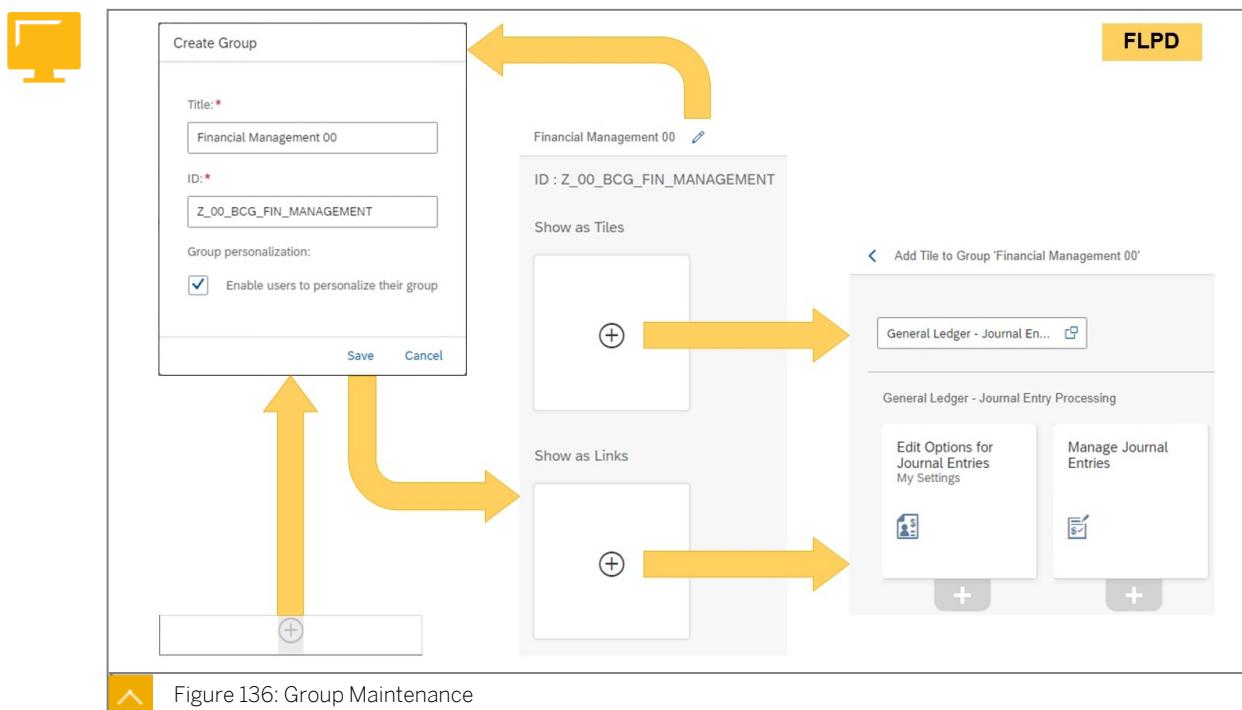


Figure 136: Group Maintenance

Groups can be created by choosing 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 later, by choosing the pencil icon next to the group heading. The group ID cannot be changed.

Groups delivered by SAP follow the naming schema SAP_<area>_BCG_<topic>. Groups created by customers should start with z_ and contain the abbreviation BCG (Business Catalog Group).

When assigning the group to a role, the prerequisite for the user to see the tiles of the group in the FLP is that the catalogs containing the tile definitions are also part of the user master record. In other words, the source catalog of the tiles must be assigned to the same role or to any other role also assigned to the user.



Caution:

The group ID must be written in capital letters.



LESSON SUMMARY

You should now be able to:

- Create SAP Fiori groups

Managing SAP Fiori Catalogs



LESSON OBJECTIVES

After completing this lesson, you will be able to:

- Manage SAP Fiori catalogs

Content Model for SAP S/4HANA

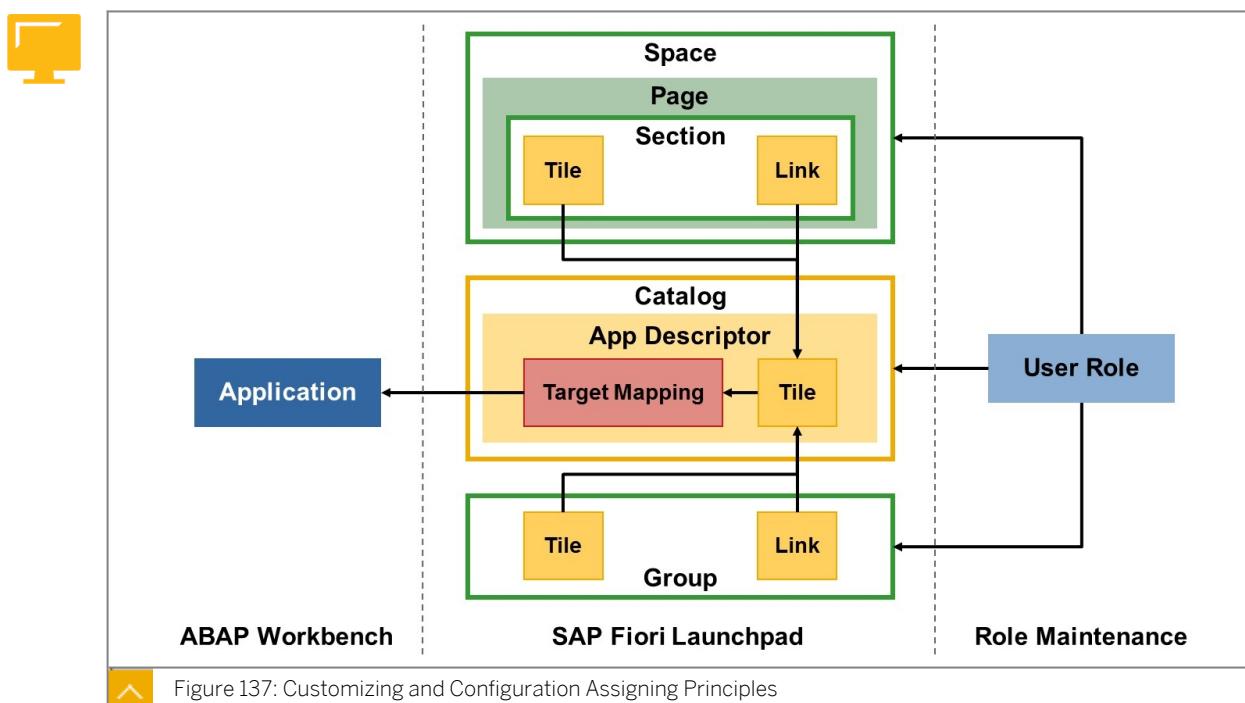
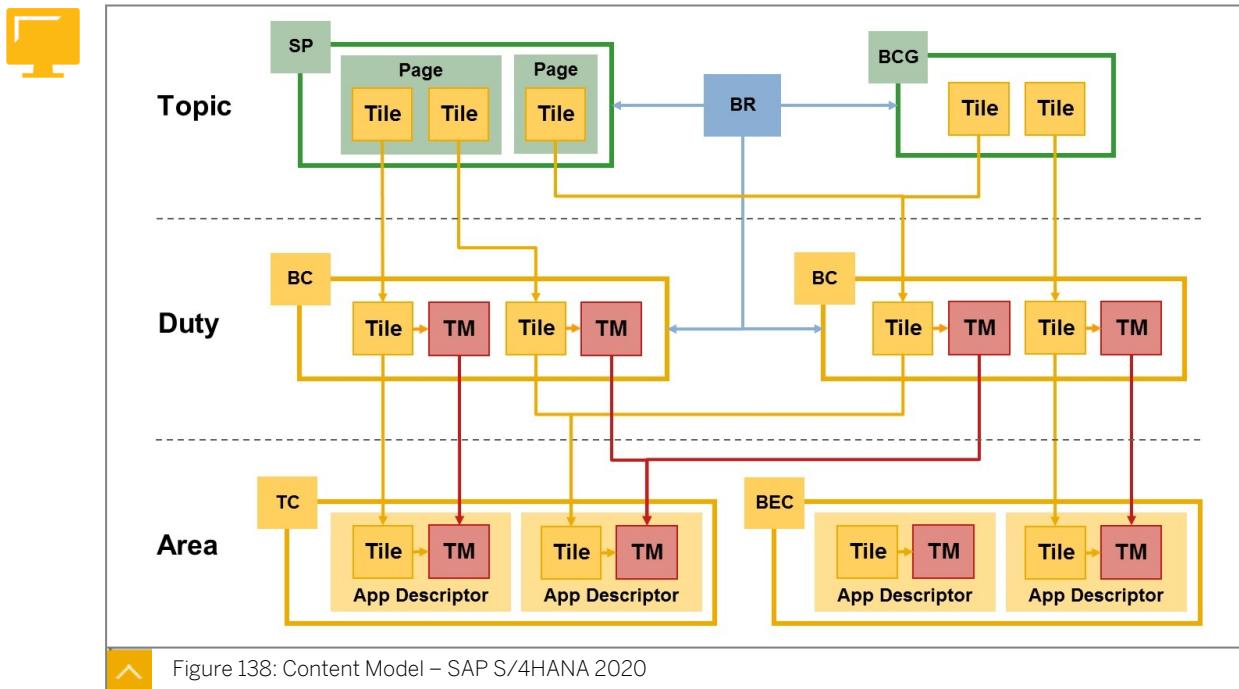


Figure 137: Customizing and Configuration Assigning Principles

In personalization, groups and catalogs just visualize tiles or links for calling an application. In customizing and configuration, catalogs define app descriptors consisting of tiles and target mappings. Target Mappings hold all the information needed to actually start an application.



The following table outlines the elements that can be distinguished in the content model since SAP S/4HANA 2020:

Element	Short	Naming Schema	Description
Business Role	BR	SAP_BR_<role>	Role for topic
Space	SP	SAP_<area>_SP_<topic>	Tiles for topic
Page	PG	SAP_<area>_PG_<task>	Tiles for daily task
Business Catalog Group	BCG	SAP_<area>_BCG_<subtopic>	Tiles for subtopic
Business Catalog	BC	SAP_<area>_BC_<duty>	Tiles for duty
Technical Catalog	TC	SAP_TC_<area>_<subarea>_COMMON	App descriptors for area
Back-End Catalog	BEC	SAP_TC_<area>_<subarea>_BE_APPS	App descriptors for area

Technical catalogs (TC) cut apps by solution area. TCs known as standard catalogs provide app descriptors consisting of tiles and target mappings for web applications like SAPUI5. TCs known as replicable or Back-End Catalogs (BEC) provide app descriptors for classic applications like ABAP transactions originating from back-end servers.

Business catalogs (BC) reference tiles and target mappings of TCs and BECs according to segregation of duty. Business catalog groups (BCG) contain apps from various business catalogs (BC), making a subtopic.

Business roles (BR) combine multiple BCs and BCGs in one topic. Spaces (SP) combine multiple pages (PG) in one topic, too. The space is then part of the BR of the same topic.



Note:

In the content model for SAP Business Suite, technical catalog roles (TCR) and business catalog roles (BCR) existed. The business roles (BR) replaced both.



↔

Technical Configuration

Technical Catalog	SAP_TC_FIN_ACC_COMMON
Technical Catalog Description	SAP
SAPUI5 Application	FIN_GLDOCPOST

...

Business Catalog(s)

<input type="checkbox"/> Catalog Name	Catalog Description
<input type="checkbox"/> SAP_SFIN_BC_GL_JE_PROC	General Ledger - Journal Entry Processing

Business Group(s)

Business Group	Group Description
<input type="checkbox"/> SAP_SFIN_BCG_DOC_ENTRY	Document Entry

Business Role(s)

<input type="checkbox"/> Role Name	Role Description
<input type="checkbox"/> SAP_BR_GL_ACCOUNTANT	General Ledger Accountant

Extend Apps Selection

↑ Figure 139: Content Model – SAP S/4HANA Example

In this example from the *SAP Fiori apps reference library*, you see content model elements under *Technical Configuration*, starting with TC and continuing with BC, BCG, and BR. The name of the BC describes the duty, the name of the BCG describes the topic, and the name of the BR is the end users role.

Catalog Management



FLPD

Catalogs

Catalog Collection
Drag to add

be_apps

x

SAP Sales and Distribution: ... 394
ID : X-SAP-U12-ADCAT:SAP_TC_CEC_SD_BE_APPS:S4SD

SAP SCM Demand Driven Rep... 2
ID : X-SAP-U12-ADCAT:SAP_TC_SCM_DDMRP_BE_APPS:S4SCM

SAP SCM Extended Wareho... 398
ID : X-SAP-U12-ADCAT:SAP_TC_SCM_EVMW_AD_BE_APPS:S4SCM

SAP SCM Extended Wareho... 295
ID : X-SAP-U12-ADCAT:SAP_TC_SCM_EVM_BE_APPS:S4SCM

SAP SCM Extended Wareho... 2
ID : X-SAP-U12-ADCAT:SAP_TC_SCM_EVM_S4_BE_APPS:S4SCM

+ Show similar target mappings

SAP Sales and Distribution: Classic Apps - Read-Only

ID : X-SAP-U12-ADCAT:SAP_TC_CEC_SD_BE_APPS:S4SD

Search 🔍

Configure Columns 🕒

Semantic Object	Action	Navigation Type	Information	Desktop	Table
SalesOrderWithoutCharge	display	Transaction		✓	✓
SalesInquiry	display	Transaction		✓	✓
CreditMemoRequest	display	Transaction		✓	✓
DebitMemoRequest	display	Transaction		✓	✓
CustomerReturn	display	Transaction		✓	✓
SalesDocument	display	Transaction		✓	✗
SalesPriceConditionTable	display	Transaction		✓	✓
BillingDocumentRequest	display	Transaction		✓	✓
InvoiceList	display	Transaction		✓	✓
BillingDocumentList	display	Transaction		✓	✓
PriceCondition	listPriceListSteppedDisplay	Transaction		✓	✗
PriceCondition	listIndivCustomerPriceList	Transaction		✓	✗
PriceCondition	listPriceGroups	Transaction		✓	✗
PriceCondition	listMaterialList	Transaction		✓	✗
Customer	listCompareMasterDataCust	Transaction		✓	✗
Promotion	createSalesDeal	Transaction		✓	✗

Create Target Mapping
Configure
Create Reference
Delete
Where Used
Original

↑ Figure 140: Read-Only Catalogs in SAP Fiori Launchpad Designer

Standard, replicable and back-end catalogs are visible as read-only catalogs in the SAP Fiori launchpad designer (FLPD) and SAP Fiori launchpad content manager (FLPCM). Both tools can only display the app descriptors defined inside these catalogs as tiles and target mappings. That is the task of the SAP Fiori launchpad application manager (FLPAM).

Replicable and 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 as remote catalog. 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 SAP Fiori installation. A remote catalog can only provide new tile configurations, not new application implementations. The application code must 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. 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. Another use case is the integration of SAP Jam tiles from the SAP Business Technology Platform or having multiple FESs pointing to one central FES with custom catalogs.



Note:

Defining remote catalogs manually in the FLPD is deprecated since the introduction of back-end catalogs.

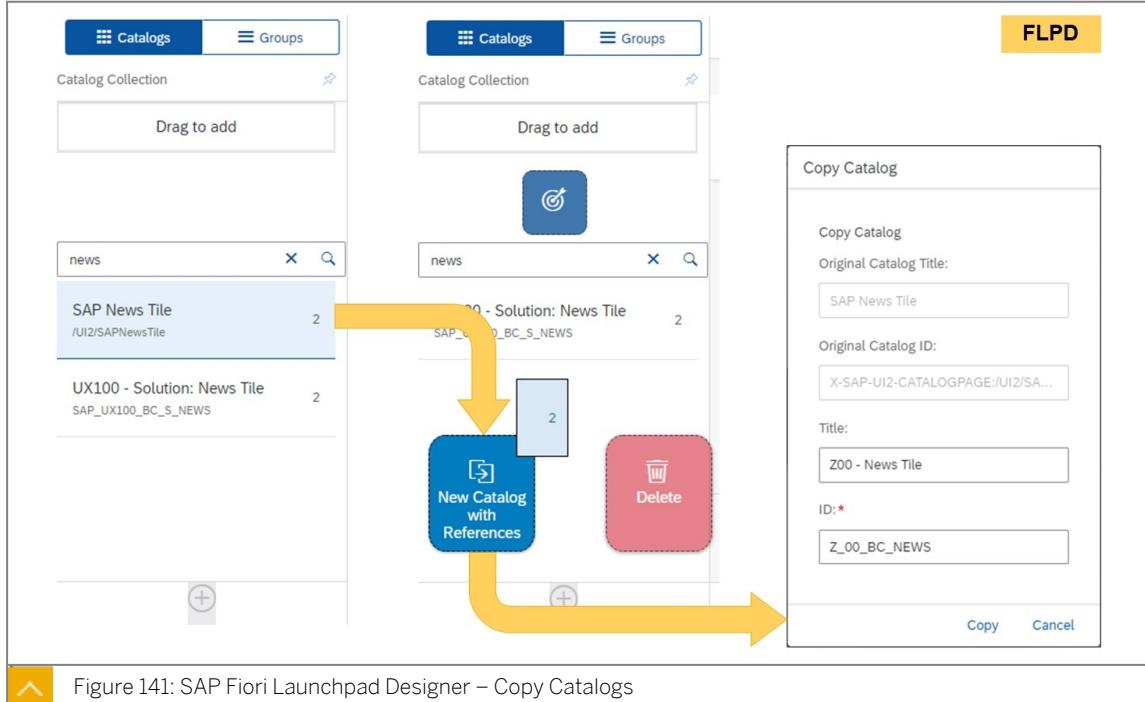


Figure 141: SAP Fiori Launchpad Designer – Copy Catalogs

Catalogs can be referenced in the FLPD 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

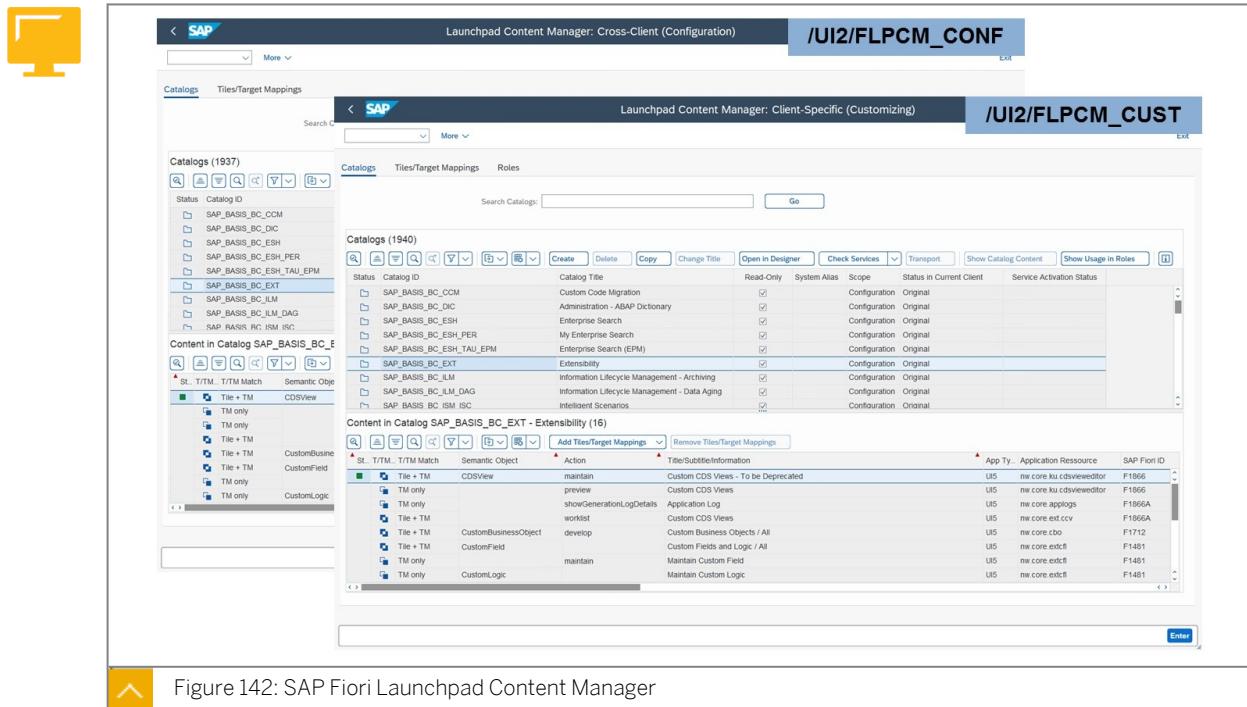
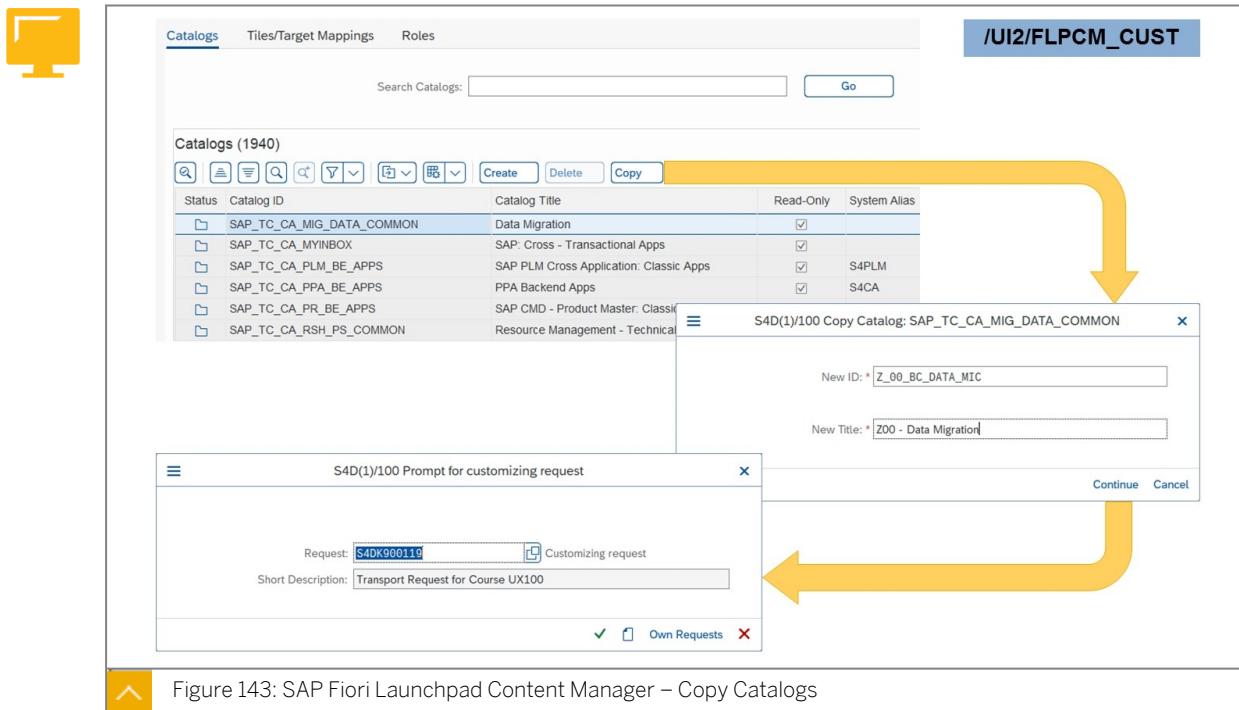


Figure 142: SAP Fiori Launchpad Content Manager

In SAP S/4HANA 1809 SPS03, the *SAP Fiori launchpad content manager (FLPCM)* 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. Similar to the *SAP Fiori launchpad designer (FLPD)*, you can use the FLPCM for configuration (transaction `/UI2/FLPCM_CONF`) or customizing (transaction `/UI2/FLPCM_CUST`).

Whereas the FLPD focuses on single catalogs or groups, the FLPCM 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 FLPD. However, it is not possible to create groups, tiles, or target mappings. Standard and replicable catalogs can only be created in the *SAP Fiori launchpad application manager (FLPAM)*.

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



The FLPCM, FLPD, and FLPAM can also be used in combination. For example, after a copying a catalog in the FLCM, the new catalog can be opened directly in the FLPD (business catalog) or FLPAM (standard catalog) in order to adapt the tiles and target mappings. Combined with the ability to also jump from the FLPCM to the *Role Maintenance* (PFCG), the FLPCM acts as the center for maintaining SAP Fiori catalogs.



LESSON SUMMARY

You should now be able to:

- Manage SAP Fiori catalogs

Unit 4

Lesson 5

Creating Business Catalogs



LESSON OBJECTIVES

After completing this lesson, you will be able to:

- Create business catalogs

Catalog Maintenance

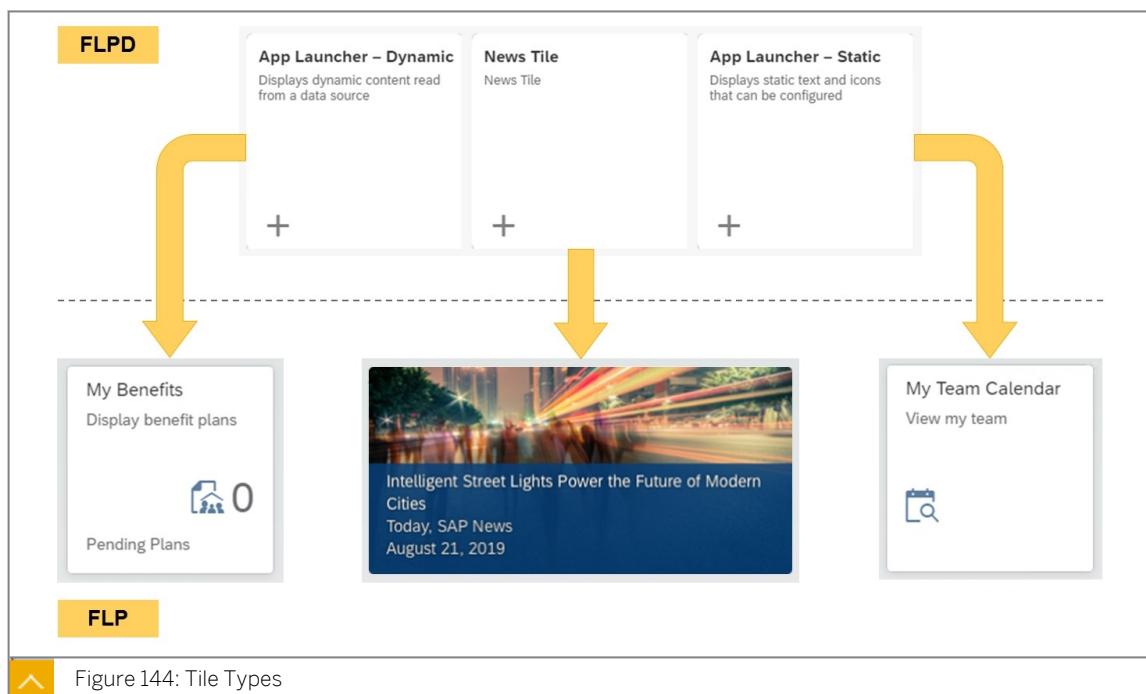


Figure 144: Tile Types

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 such as 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.



Note:

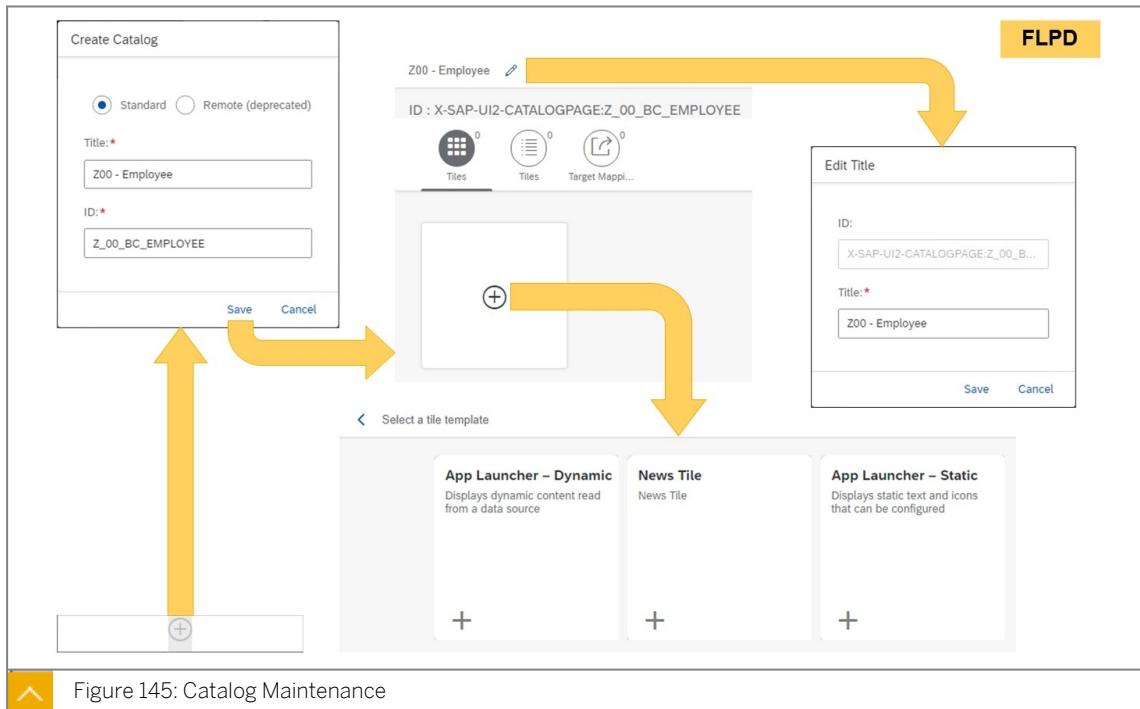
For more information about the news tile, please read SAP note [2990265](#).

Figure 145: Catalog Maintenance

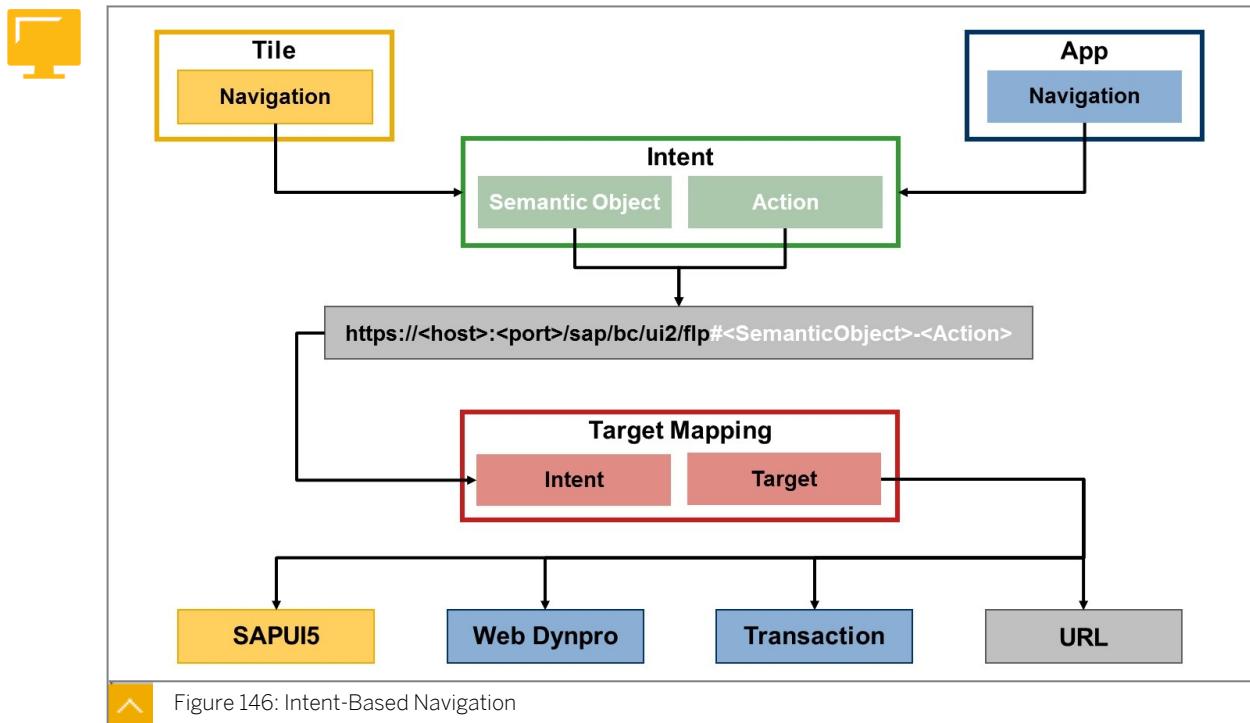
Catalogs can be created by choosing the plus symbol in the lower left corner of the catalog overview. This opens a dialog box for entering the catalog title and technical ID, and specifying if the catalog is pulled from another system as a remote catalog. The catalog title can be changed at a later point 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 abbreviation BC for business catalog.



Caution:

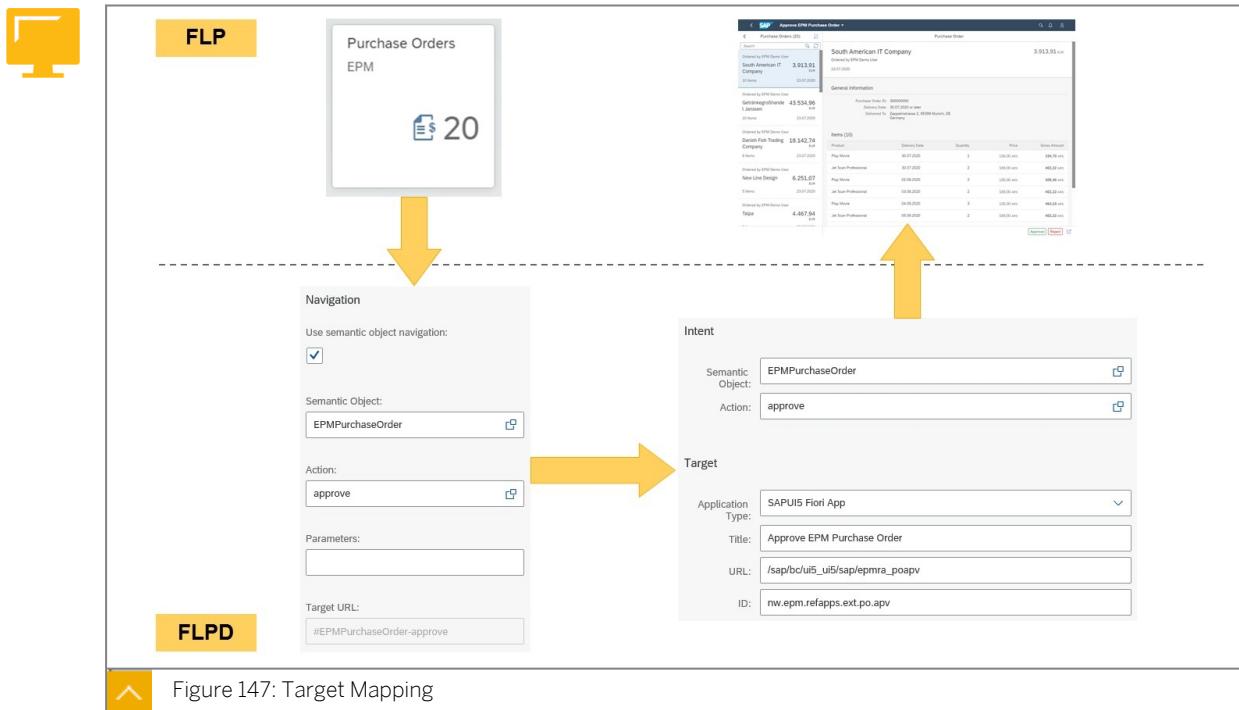
The catalog ID must be written in capital letters.

Intent-Based Navigation



The most important part of a tile is its navigation information, which contains an intent that connects to a target mapping. The term “intent” is short for “intention” and consists of a semantic object such as `SalesOrder` and an action such as `display`. If the user wants to see the details of a sales order, their intention is to display a sales order. SAP Fiori adds this intention behind the FLP URL, separated by a #, to call a target mapping having this intent. In this way, you can 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. The same intent can lead to different results for different users.



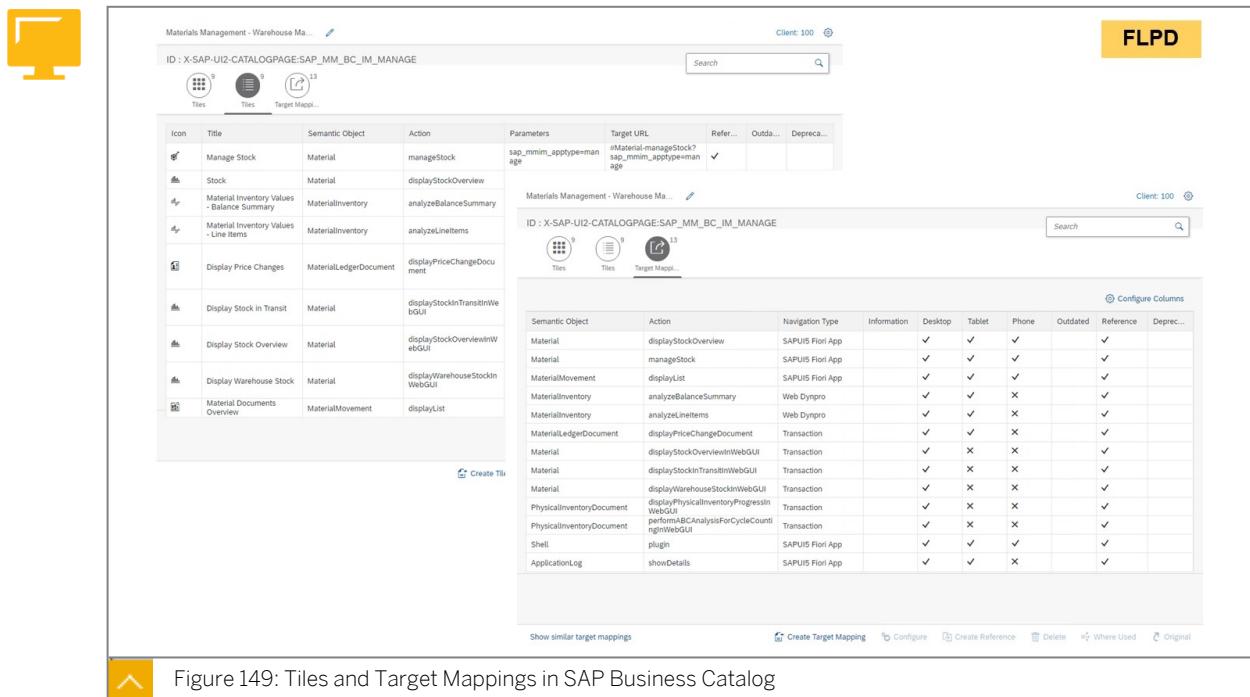
Target mappings contain the information about which app to start, with what parameters, and on what device types. They are identified by its intent, the combination of semantic object and action. If the user chooses a tile, the intent-based navigation defined in the tile is started. The suitable target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

The figure shows the SAP Fiori application interface for the transaction /UI2/SEMOBJ_SAP. It displays a list of semantic objects under the heading 'Semantic Objects - SAP Delivery'. The table includes columns for Semantic Object, Semantic Object Name, Appl. Component, and Semantic Object Description. The list includes entries such as ABAPSystem, ABOPCustomSort, ABOPRun, ABOPSegment, ABOPVariant, ACMMasterData, ACMOperationsClerk, ACMSettlementClerk, ACMTMIntegrationMonitor, ACMTMLoadOrderAssignment, ACMTrader, ADBKFC, ADSConfiguration, AIFMessage, AIFRecipient, AIFUserRecipientAssignment, AIFValueMapping, APARToleranceGroup, and APSLog. Each entry has a detailed description of its purpose and application component.

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



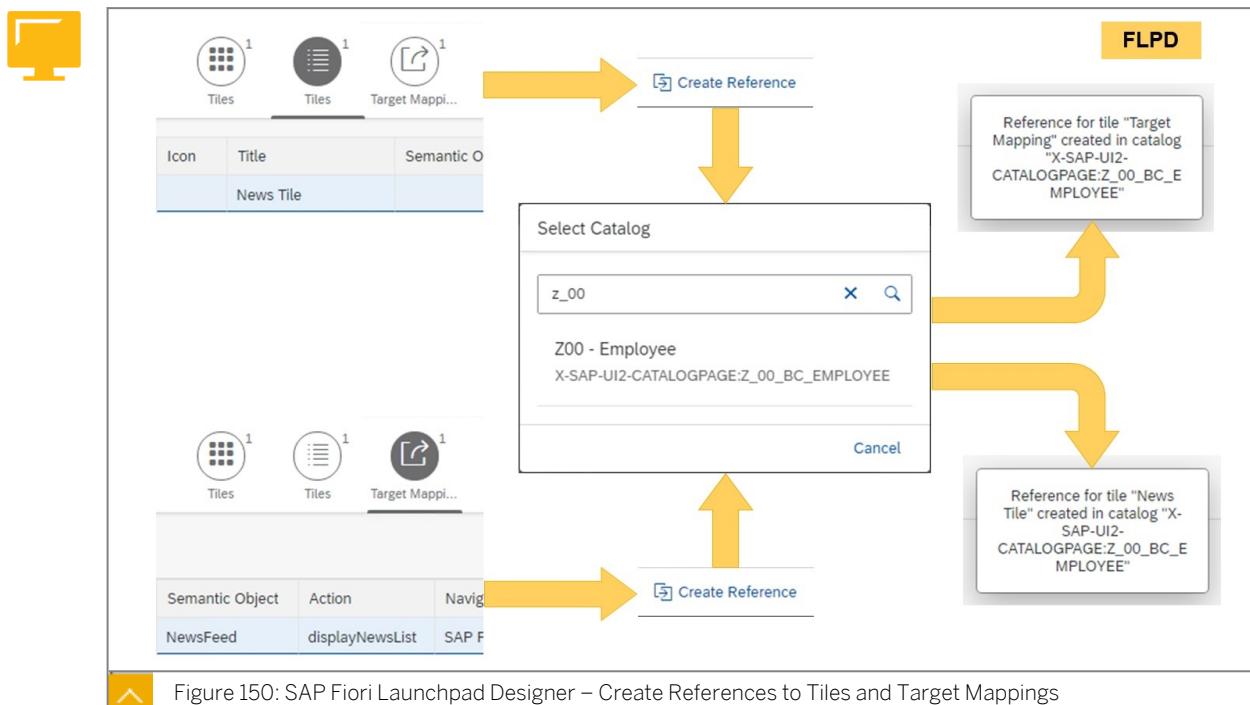
The screenshot shows the SAP Fiori Launchpad Designer interface. At the top, there are three tabs: 'Tiles' (selected), 'Tiles' (disabled), and 'Target Mapp...'. Below the tabs, there are two main sections:

- Tiles Section:** A table listing tiles with their icons, titles, semantic objects, actions, and parameters. One row is selected, showing 'Manage Stock' with action 'manageStock' and parameter 'sap_mm_im_apptype=man age'.
- Target Mappings Section:** A table listing target mappings with columns for Semantic Object, Action, Navigation Type, and various device support flags. One row is selected, showing 'Material' with action 'displayStockOverview' and navigation type 'SAPUI5 Fiori App'.

At the bottom, there are buttons for 'Create Tiles', 'Create Target Mapping', 'Configure', 'Create Reference', 'Delete', 'Where Used', and 'Original'.

Figure 149: Tiles and Target Mappings in SAP Business Catalog

SAP delivers numerous catalogs containing many tiles and target mappings. All definitions are tested and distinguished, 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.



The screenshot illustrates the creation of references between tiles and target mappings across different catalogs. It shows two main steps:

- Create Reference from Target Mapping:** A 'Create Reference' button is highlighted, leading to a 'Select Catalog' dialog where 'z_00' is selected. A callout box indicates the reference is for the 'Target Mapping' tile in catalog 'X-SAP-U12-CATALOGPAGE:Z_00_BC_EMPLOYEE'.
- Create Reference from Tile:** Another 'Create Reference' button is highlighted, leading to the same 'Select Catalog' dialog where 'z_00' is selected. A callout box indicates the reference is for the 'News Tile' tile in catalog 'X-SAP-U12-CATALOGPAGE:Z_00_BC_EMPLOYEE'.

Figure 150: SAP Fiori Launchpad Designer – Create References to Tiles and Target Mappings

To create a reference to 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.

The screenshot shows the SAP Fiori Launchpad Content Manager interface. It consists of three main sections:

- Top Section:** Catalogs (3 of 1940) table. Shows catalogs like Z_00_BC_EMPLOYEE, Z_00_BC_MIG_DATA, and Z_00_BC_NEWS. A yellow arrow points from the 'Add Selected Tiles/TMs to Other Catalog' button in the toolbar to the 'Content in Catalog Z_00_BC_EMPLOYEE - Z00 - Employee (0)' section.
- Middle Section:** Content in Catalog Z_00_BC_EMPLOYEE - Z00 - Employee (0) table. Shows a row for 'BusinessPartner' with action 'manage'. A yellow arrow points from the 'Action' column to the 'Content in Catalog Z_00_BC_EMPLOYEE - Z00 - Employee (1)' section.
- Bottom Section:** Content in Catalog Z_00_BC_EMPLOYEE - Z00 - Employee (1) table. Shows the same row for 'BusinessPartner' with action 'manage'. This section is identical to the middle one.

Toolbar buttons include: Search, Create, Delete, Copy, Change Title, Open in Designer, Add Tiles/Target Mappings, Remove Tiles/Target Mappings, Add Tiles/TMs to Selected Catalog, and Add Selected Tiles/TMs to Other Catalog.

Search bar: MANAGE BUSINESS PARTNER MASTER DATA

Page header: /UI2/FLPCM_CUST

To create a reference to 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 business catalogs

Unit 4

Lesson 6

Creating Target Mappings



LESSON OBJECTIVES

After completing this lesson, you will be able to:

- Create target mappings

SAPUI5 Mapping Elements

The screenshot shows the SAP Fiori apps reference library interface for the application "Post General Journal Entries [F0718]". The main title is "Configuration". Below it, a note states: "The following sections list app-specific data required to configure the app: Please note that this app has related apps which need to be configured". A section titled "SAPUI5 Application" lists ICF nodes for the application. A table provides the mapping between SAPUI5 Components and ICF Nodes:

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_gldocdisplay	fin.gl.documentdisplay
	FIN_GL_POST_LIB *	/sap/bc/ui5_ui5/sap/fin_gl_post_lib	sap.fin.gl.lib.posting
	FIN_LIB *	/sap/bc/ui5_ui5/sap/fin_lib	

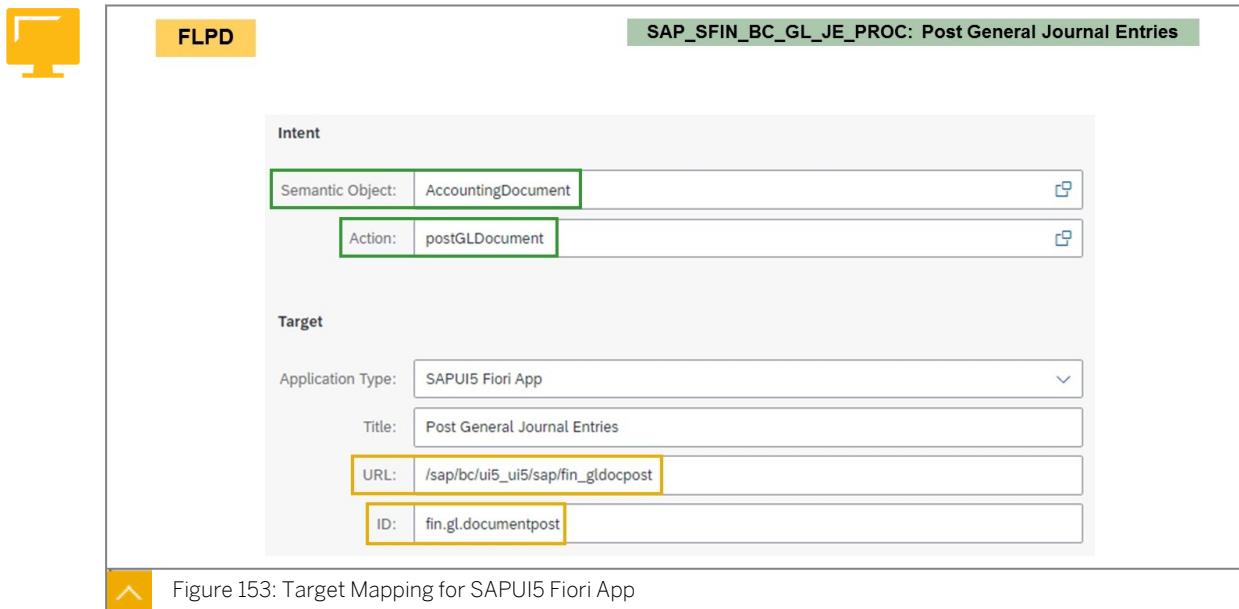
A note at the bottom of the table says: "* Added automatically due to dependencies". Below the table, there is a section titled "Technical Configuration" with three items listed:

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

At the bottom left of the screenshot is a yellow navigation icon with an upward arrow.

Figure 152: Technical Configuration for SAPUI5 App

In this example from the *SAP Fiori apps reference library*, you see the technical name of the SAPUI5 application, the ICF-path, and the SAPUI5 component. The semantic object and action form the intent.

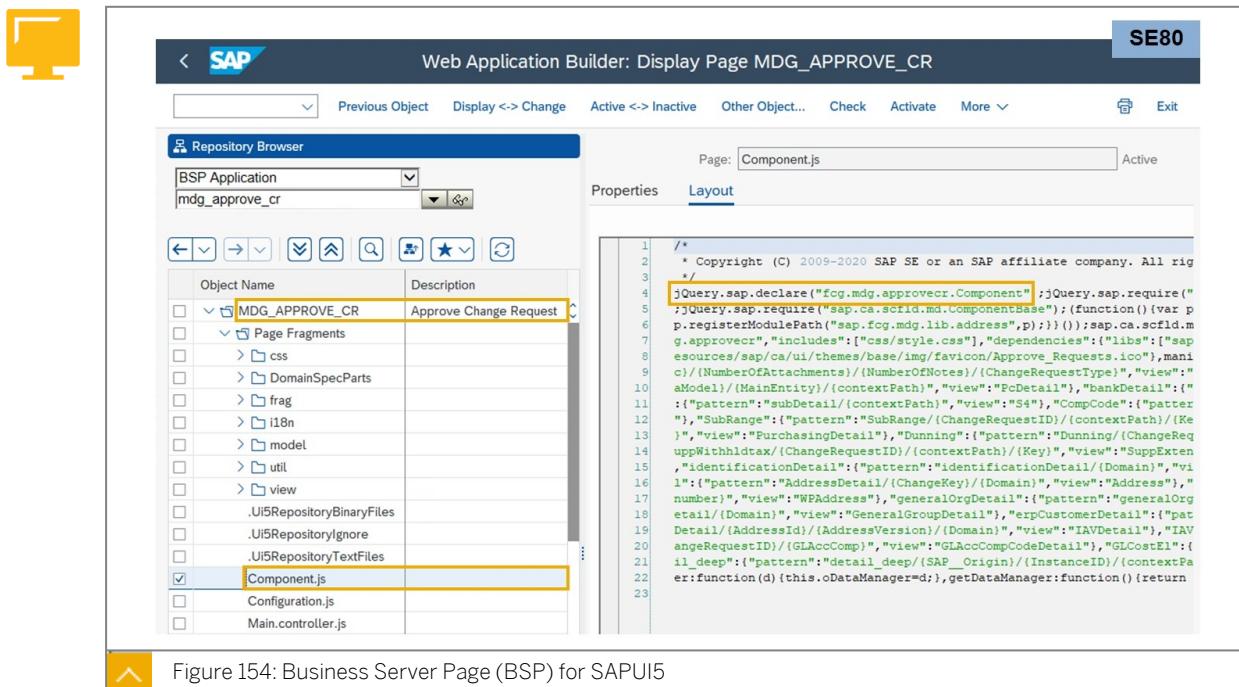


In the *SAP Fiori launchpad designer (FLPD)*, the intent can be found in the target mapping of the catalog. The application type *SAPUI5 Fiori App* is set as target. The URL holds the ICF-path and the ID holds the SAPUI5 component.



Note:

The transaction `LPD_CUST` was the first way to map an intent-based navigation to the implementation of an app before the FLPD fully took over. URL and ID were defined in an `LPD_CUST` instance of an `LPD_CUST` role and referenced in the FLPD via an application alias.



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`.

The screenshot shows the SAP Web Application Builder interface. The title bar reads "Web Application Builder: Display Page FIN_GLDOCPOST". The top right corner has a blue button labeled "SE80". Below the title bar is a toolbar with various icons and buttons like "Previous Object", "Display <-> Change", "Active <-> Inactive", "Other Object...", "Check", and "More". The main area is divided into two panes. The left pane is the "Repository Browser" showing the structure of the "fin_gldocpost" BSP application. It lists "Object Name" and "Description" for each item. The "Component.js" file is selected and highlighted with a yellow border. The right pane shows the "Properties" and "Layout" tabs. Under "Properties", there is a "Page:" field set to "Component.js" and a "Status" dropdown set to "Active". The "Layout" tab is active, displaying a code editor with the following content:

```

1
2
3
4  on(U,R,D,E){"use strict";return U.extend("fin.g1.documentpost.Component",
5    prototype.createContent.apply(this,arguments);r.addStyleClass(this.getCont
6    hasClass("sapUiSizeCompact"))(this._sc).contentDensityClass"";else if(!D.s
7
8

```

A yellow callout with the word "Extension" points to the code editor area.

Figure 155: Business Server Page (BSP) for SAPUI5 Extension

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`, instead, `return U.extend` holding the path to the `Component.js`.



LESSON SUMMARY

You should now be able to:

- Create target mappings

Unit 4

Lesson 7

Creating Technical Catalogs



LESSON OBJECTIVES

After completing this lesson, you will be able to:

- Create technical catalogs

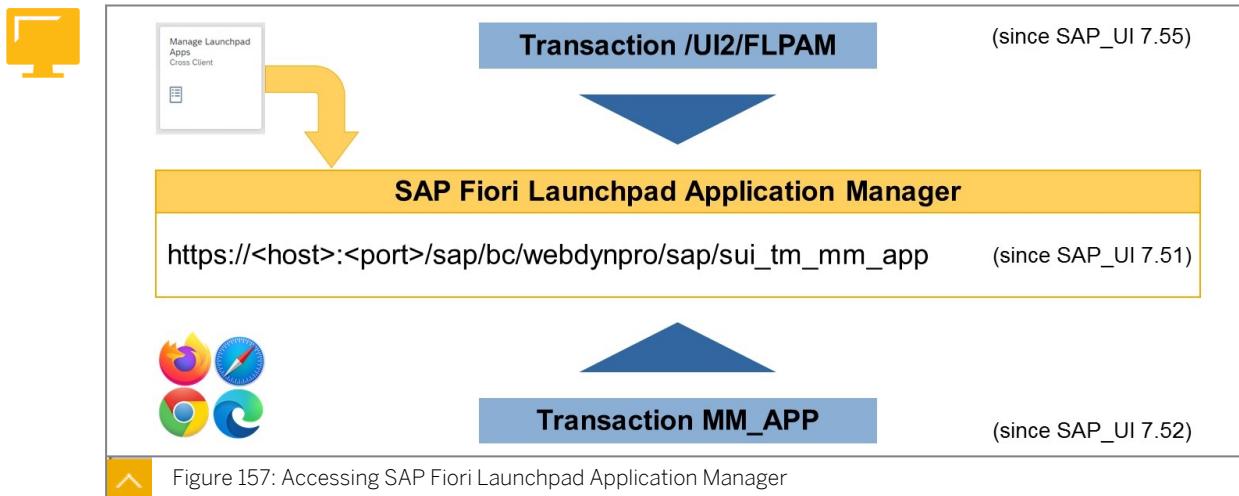
SAP Fiori Launchpad Application Manager



The screenshot displays the SAP Fiori Launchpad Application Manager interface across three main sections: **Search**, **Facet Filter**, and **Catalog Entry**. The **Search** section is the primary view, featuring a search bar for Catalog ID and Title, and a table listing 319 catalogs with columns for Catalog ID, Change Date, Catalog Type, and Creation Date. The **Facet Filter** section shows filter criteria for attributes like Catalog ID, App Type, and Semantic Object. The **Catalog Entry** section provides detailed entry fields for Catalog ID, Title, and Type.

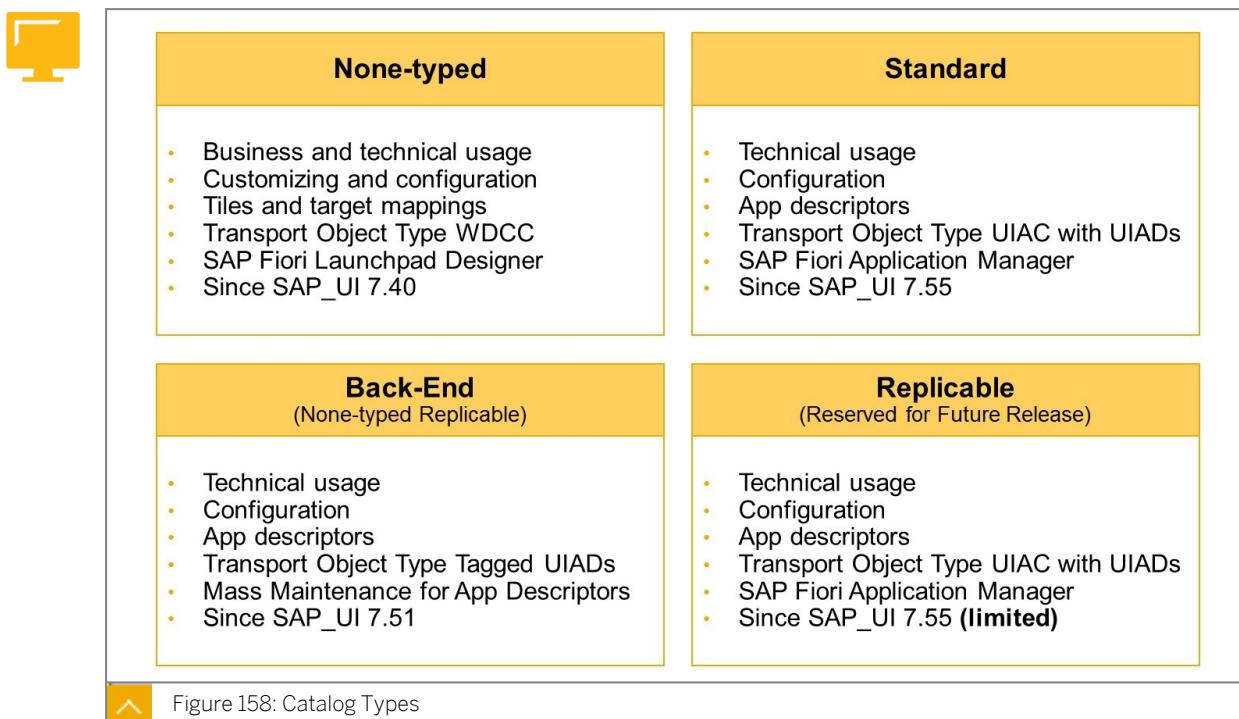
Figure 156: SAP Fiori Launchpad Application Manager

In SAP S/4HANA 2020, the *SAP Fiori launchpad application manager (FLPAM)* was introduced to create, change, and maintain technical catalogs and app descriptors. It offers the views *Search*, *Facet Filter*, and *Catalog Entry*, which can be switched at any item via the buttons in the header. The main view is *Search* offering all common tasks around technical catalogs.



In fact, the FLPAM was already introduced in SAP S/4HANA 1610 as *Mass Maintenance for App Descriptors* focusing purely on classic applications. In SAP_UI 7.51, the Web Dynpro application **SUI_TM_MM_APP** had to be called directly. In SAP_UI 7.52, transaction **MM_APP** was introduced to make starting the application easier. In SAP_UI 7.55, transaction **/UI2/FLPAM** was introduced and added to the **SAP_FLP_ADMIN** role as *Manage Launchpad Apps* tile.

Catalog and Application Types

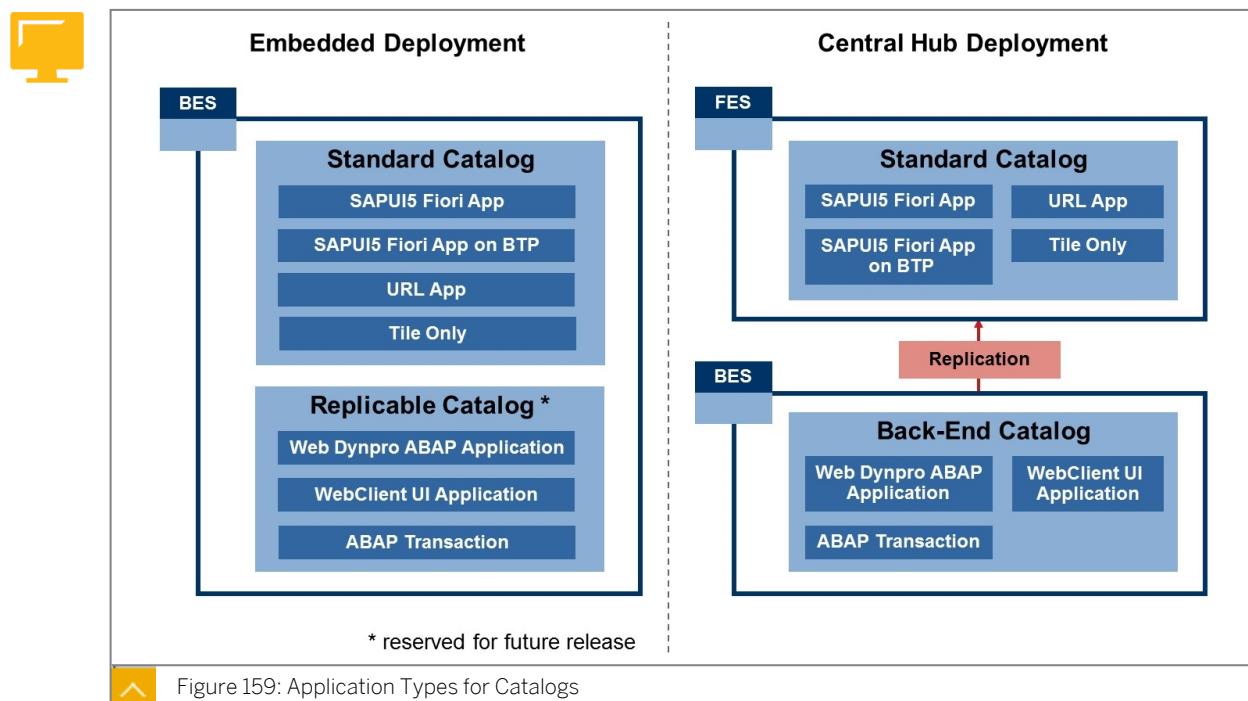


With the introduction of the *SAP Fiori launchpad application manager (FLPAM)*, catalog types were also introduced. Until then, business and technical catalogs were just semantic interpretations of SAP Fiori catalogs. These are now called none-typed catalogs, defining tiles and target mappings per client (customizing) or cross-client (configuration). The transport object type is Web Dynpro Component Configuration (WDCC) and the according tool is the *SAP Fiori launchpad designer (FLPD)*.

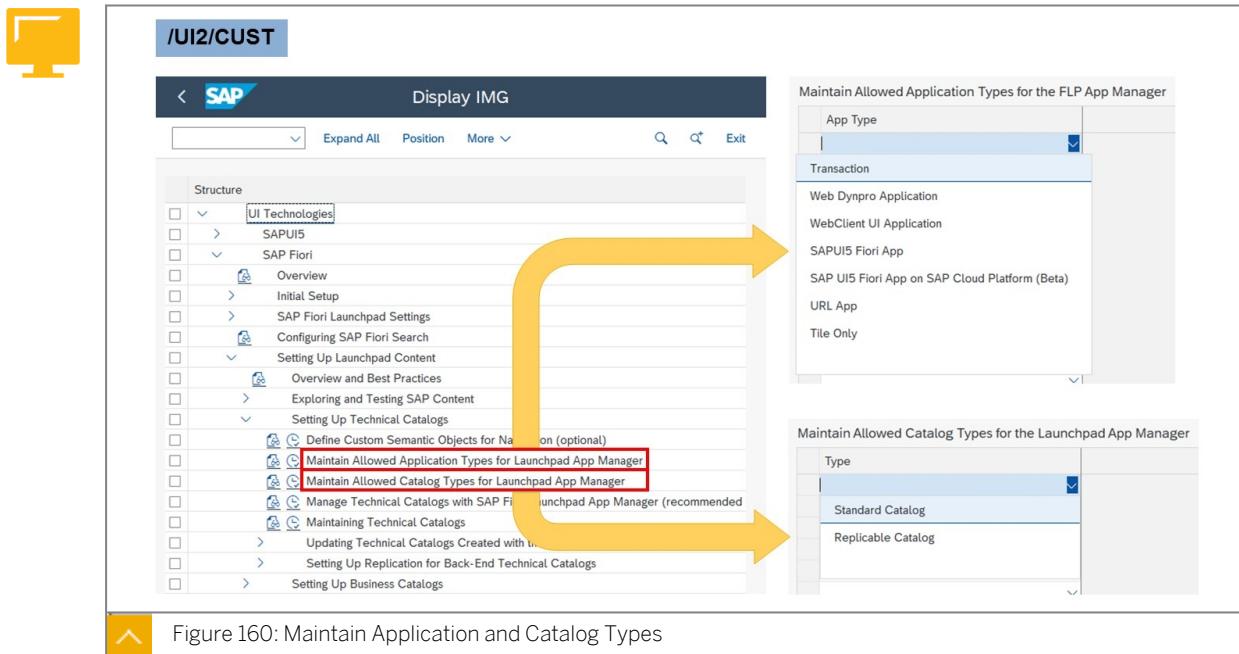
Before the introduction of catalog types, only the back-end catalog differed in its structure on a technical level, by defining app descriptors instead of separated tiles and target mappings. Back-End catalogs are always cross-client (configuration) and are available since SAP_UI 7.51. The transport object type is User Interface App Descriptors (UIAD) tagged with the catalog name, and the corresponding tool is the *Mass Maintenance for App Descriptors*.

With SAP_UI 7.55 SP01, most technical catalogs are standard catalogs. They are always cross-client (configuration) and define app descriptors instead of separated tiles and target mappings. The transport object type is User Interface App Catalog (UIAC) consisting of User Interface App Descriptors (UIAD) and the according tool is the FLPAM. UIAC is a development object visible in development tools (ADT, SE80) making it easier to find and organize catalogs for example in transport requests. Therefore it is recommended to use standard catalogs as soon as possible.

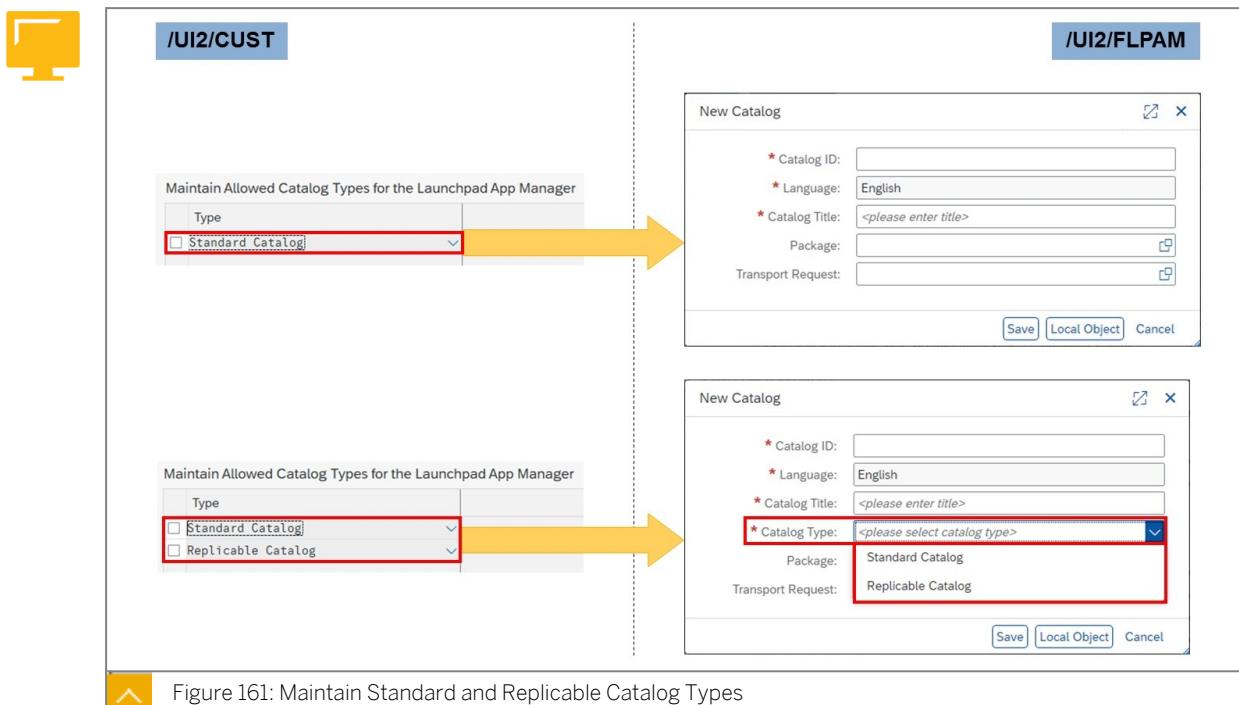
Although also introduced with SAP_UI 7.55 SP01, replicable catalogs are reserved for use in a future release. They are nearly identical to standard catalogs (UIAC with UIAD in FLPAM) but are replicable like the back-end catalogs. Customers should not use replicable catalogs in the initial release, because their feature set is not yet complete.



In parallel to catalog types, application types were introduced as cross-client setting. In an embedded deployment, all application types are allowed in technical catalogs. In a central hub deployment, standard catalogs consist of SAPUI5 Fiori apps (for BTP), URL apps, and tiles without target mappings in the front-end server (FES). Back-end catalogs in the back-end server (BES) consist of Web Dynpro ABAP applications, WebClient UI applications, and ABAP transactions. The transaction /UI2/APPDESC_GET in the FES is used to replicate back-end catalogs from BES as remote catalogs to the FES.



Catalog and application types are defined cross-client via the implementation guide. Transaction /UI2/CUST can be used to access the UI parts of the implementation guide directly. For an embedded deployment, it is recommended not to define any application type. This allows the use of any application type in technical catalogs.



For an embedded deployment, **Standard Catalog** is recommended as the only catalog type. This will mean, only standard catalogs can be defined in the FLPAM. When setting **Standard Catalog** and **Replicable Catalog**, FLPAM offers the catalog types as dropdown during creation.

Technical Catalog Maintenance

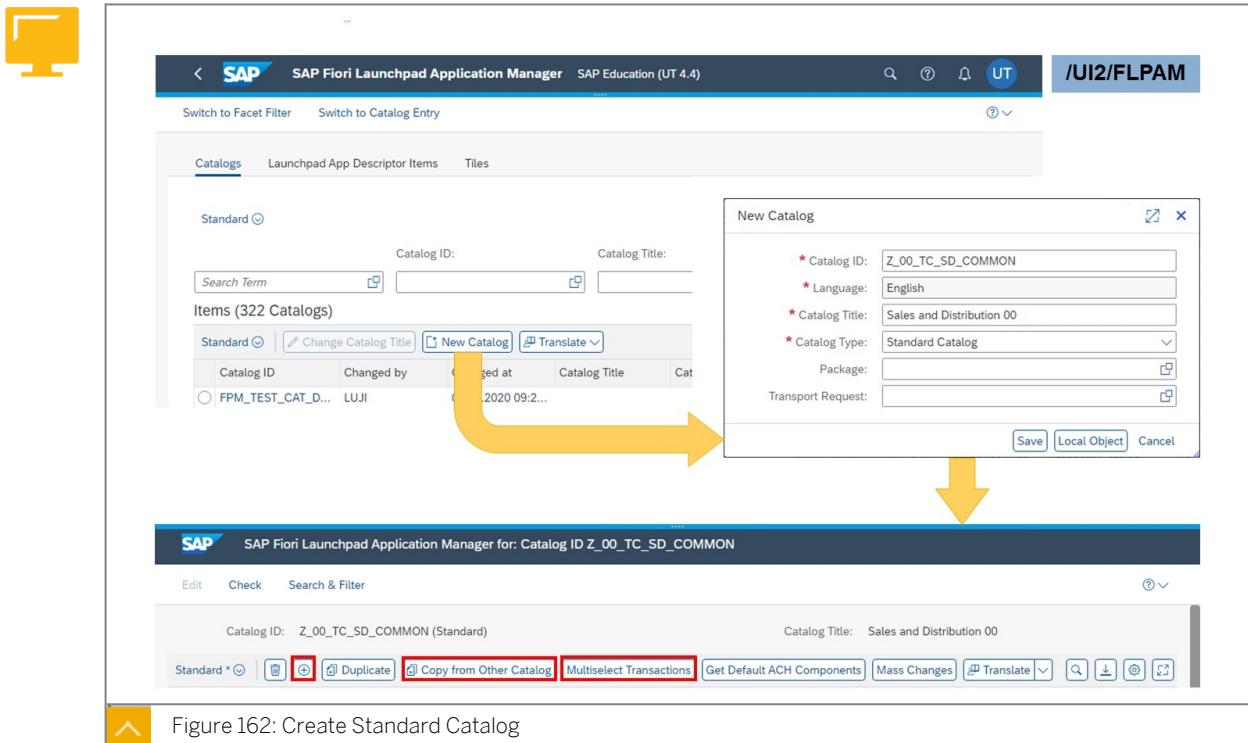


Figure 162: Create Standard Catalog

New standard catalogs can be created in the Search view of the *SAP Fiori launchpad application manager (FLPAM)*. They are saved in a package and assigned to a workbench request unless defined as local object. Standard catalogs created by customers must start with `Z` and should contain the abbreviation `TC` for technical catalog and end with `_COMMON`.

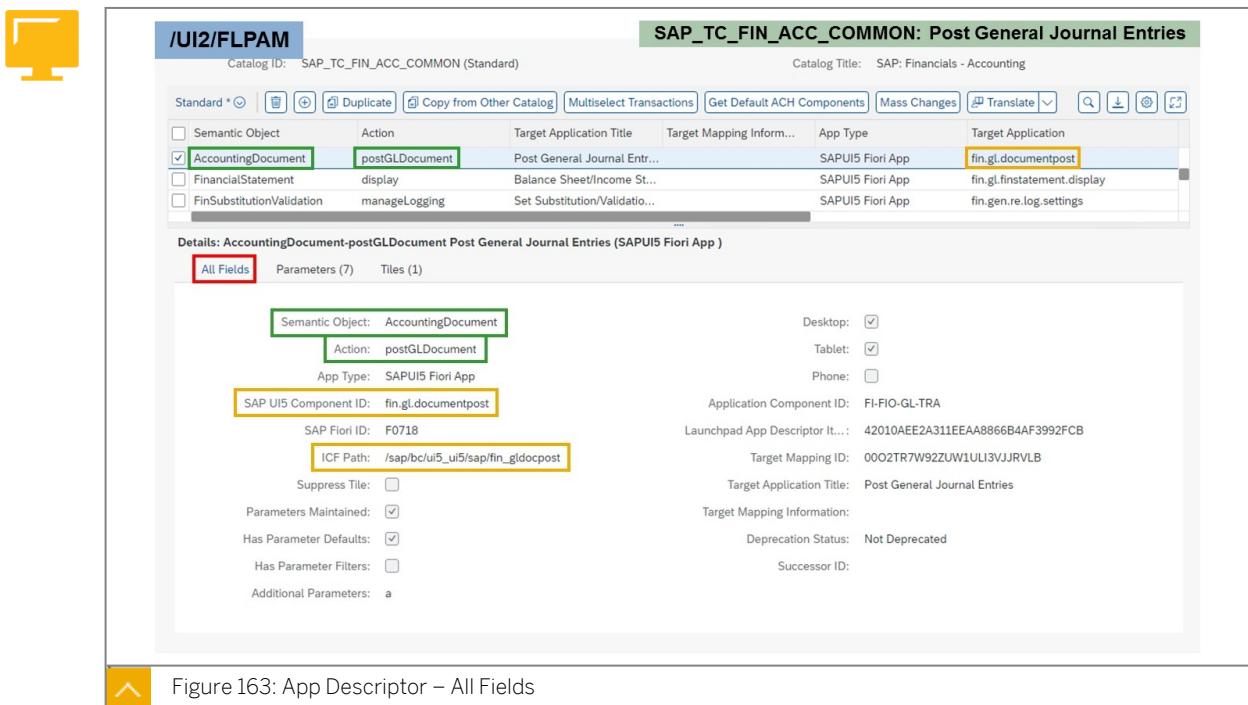
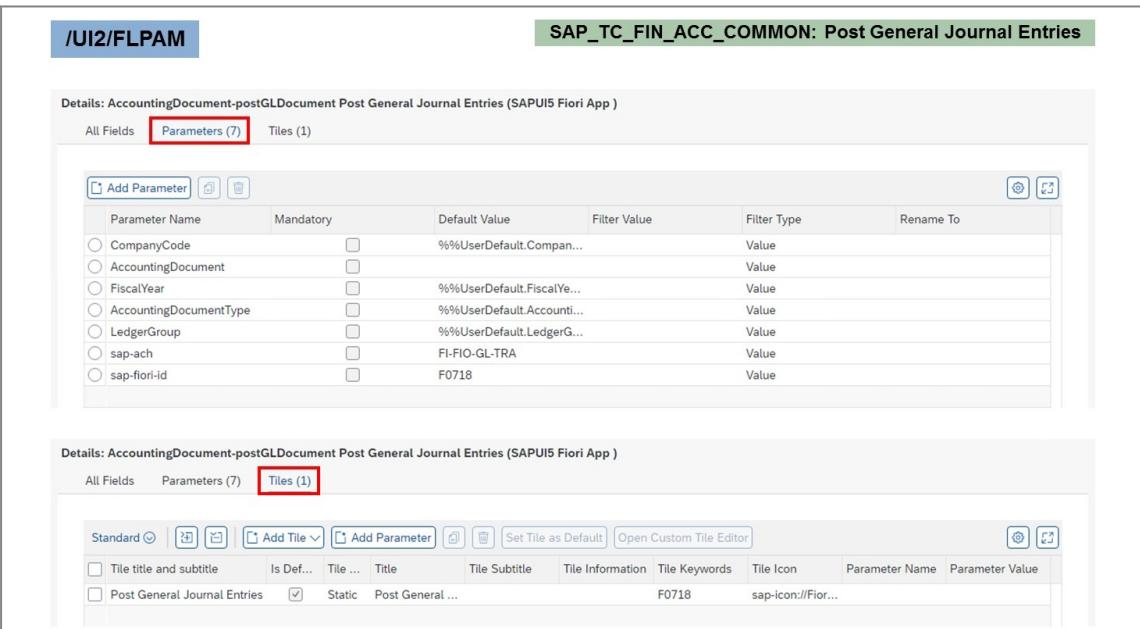


Figure 163: App Descriptor – All Fields

App descriptors can be created declarative, copied from other catalogs, or generated by selecting transactions. The mandatory fields are semantic object, action, app type, and one or

more additional fields, depending on the selected app type. For SAPUI5 Fiori apps, it is the SAPUI5 component ID. All fields can be viewed or changed inside the table or in the *All Fields* view.

 Hint:
The FLPAM opens catalogs in a new browser window. There is no "back" or "close" button. Just close the browser window.

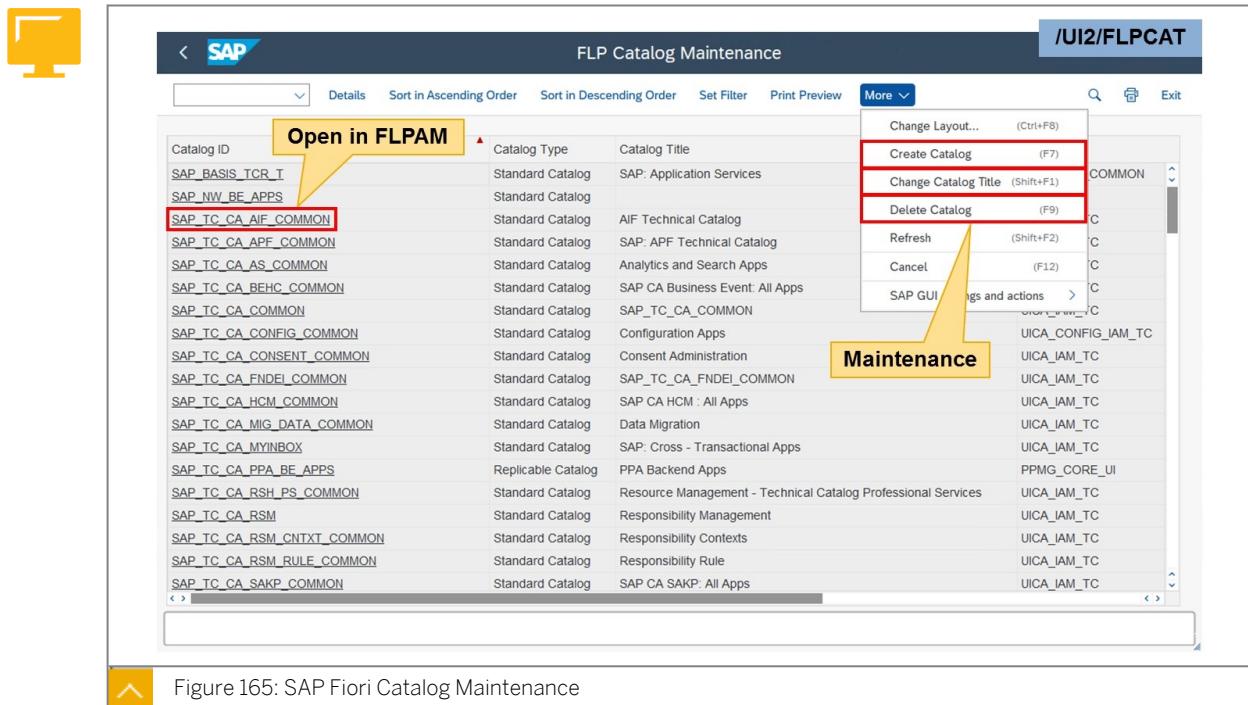


The screenshot shows the SAPUI5 Fiori App Descriptor interface. At the top, there's a header bar with the application name and a sub-header for the specific configuration. Below this, there are two main tabs: 'All Fields' and 'Parameters (7)', with 'Parameters' being the active tab and highlighted with a red box. Under the parameters tab, there's a table listing various parameters with their names, mandatory status, default values, filter values, and filter types. Another tab, 'Tiles (1)', is also present. Below the parameters, another section for tiles is shown, with a single tile configuration row. The tile has a title 'Post General Journal Entries', is static, and is set as the default tile.

Figure 164: App Descriptor – Parameters and Tiles

Parameters allow to start an application for a certain dataset or in a certain mode. Providing parameters as optional allows more flexibility in calling the application. Providing parameters as mandatory allows to define requirements for starting the application. The values of the parameters are provided in the application starting the intent-based navigation or via a tile.

An app descriptor can define multiple tiles for an application offering multiple visualizations or ways to start the application. The integrated tile types are *App Launcher – Static*, *App Launcher – Dynamic*, and *News Tile*. When adding tiles, one has to be set as default tile.



Another tool for maintaining technical catalogs is the *SAP Fiori Catalog Maintenance* (/UI2/FLPCAT). It offers a list of technical catalogs and opens the FLPAM for the content of a single catalog. In the menu, options for creating and deleting catalogs and changing the catalog title are offered.



LESSON SUMMARY

You should now be able to:

- Create technical catalogs

Unit 4

Lesson 8

Creating Replicable Catalogs



LESSON OBJECTIVES

After completing this lesson, you will be able to:

- Create replicable catalogs

Back-End Catalog Maintenance



The screenshot shows two overlapping dialog boxes. The top box is titled 'Maintain Allowed Catalog Types for the Launchpad App Manager' and has a 'Type' field with a red border. The bottom box is titled 'Maintain Allowed Application Types for the FLP App Manager' and shows a list of application types: 'WebClient UI Application', 'Transaction', and 'Web Dynpro Application', all with red borders. Arrows point from both boxes down to the SAP Fiori Launchpad Application Manager interface. The interface includes fields for 'Catalog ID' and 'Semantic Object', and a message bar at the bottom stating 'Only backend app types are allowed in this system'. A 'Continue' button is visible.

Figure 166: Mass Maintenance for App Descriptors

Back-end catalogs are shipped by SAP as part of updates of back-end systems (BES). They can be examined in the BES using the Web Dynpro ABAP application SUI_TM_MM_APP formally called *Mass Maintenance for App Descriptors*. With the availability of the SAP Fiori launchpad application manager (FLPAM), a catalog type must not be defined in the implementation guide to enable the usage of back-end catalogs in FLPAM. The application types must be **Transaction**, **Web Dynpro ABAP Application**, and **WebClient UI Application**. The FLPAM then starts in the *Catalog Entry* view and shows the warning that only back-end catalogs are allowed in the system.



Note:

In the version called *Mass Maintenance for App Descriptors*, the *Catalog Entry* view was the only one.

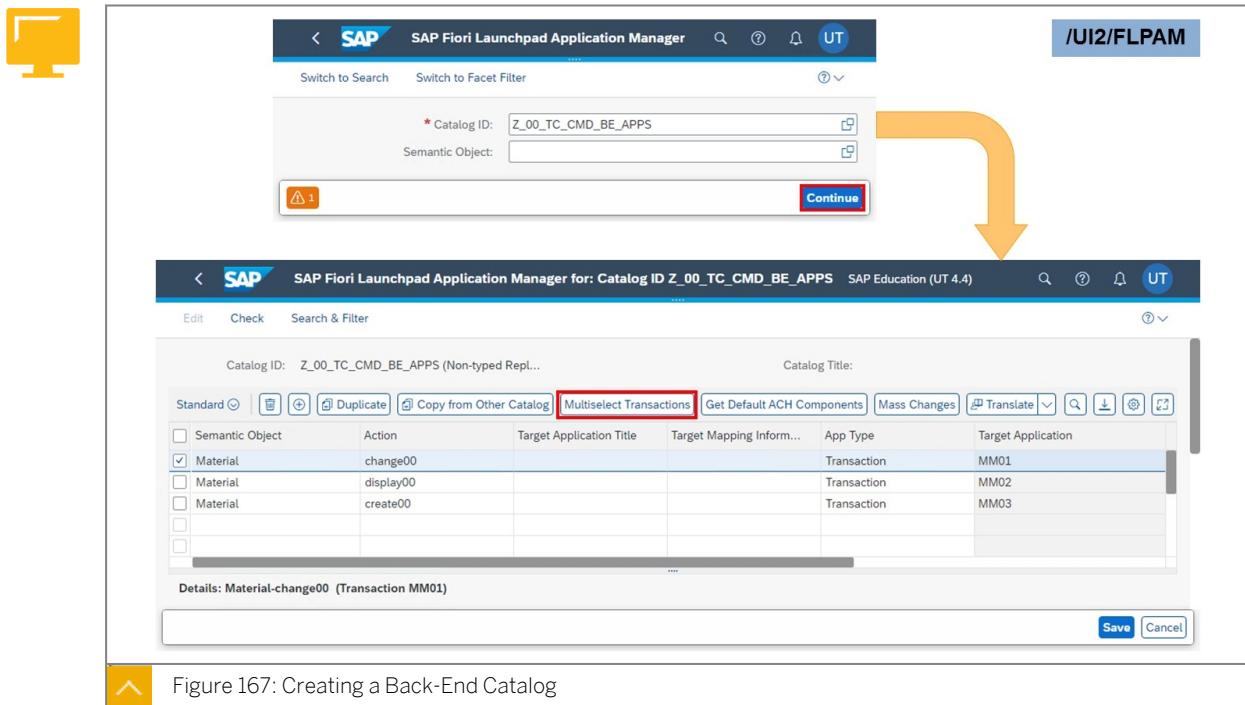


Figure 167: Creating a Back-End Catalog

When only back-end catalogs are allowed in the system, entering a nonexistent catalog ID in the *Catalog ID* field and choosing *Continue* will automatically create a back-end catalog with this ID. Back-end catalogs created by customers must start with `z_` and should contain the abbreviation `TC` for technical catalog and end with `_BE_APPS`.

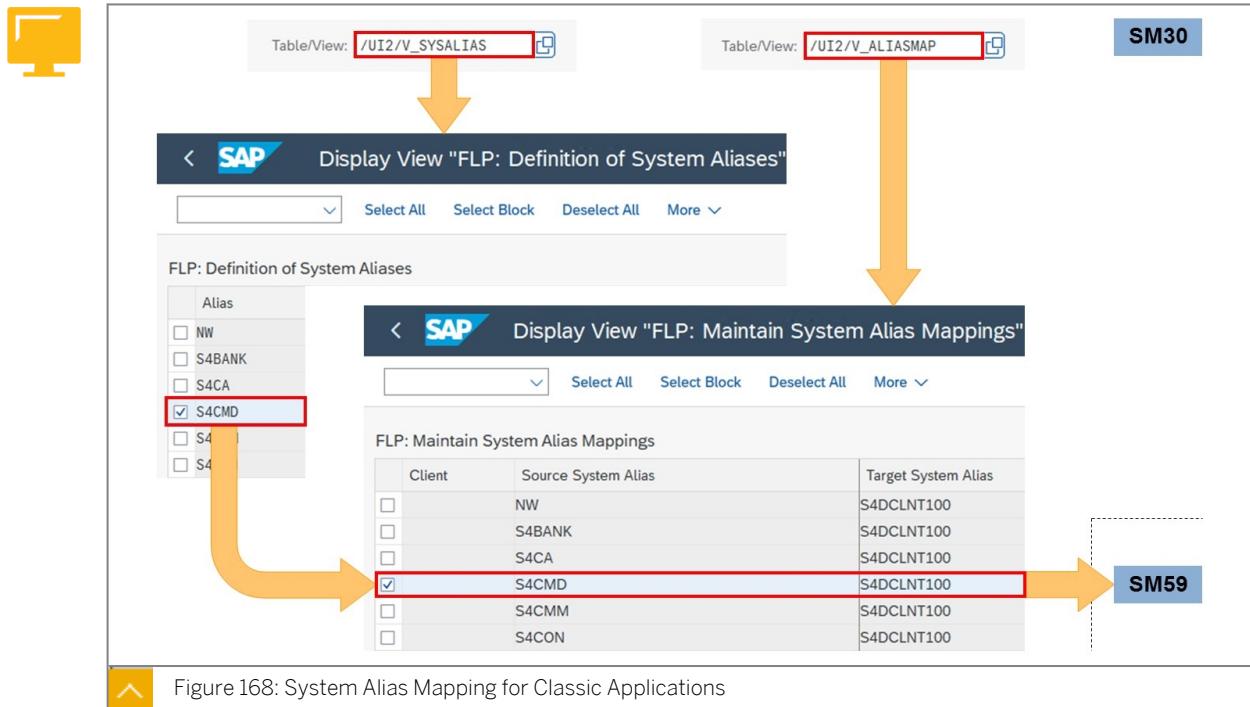


Note:

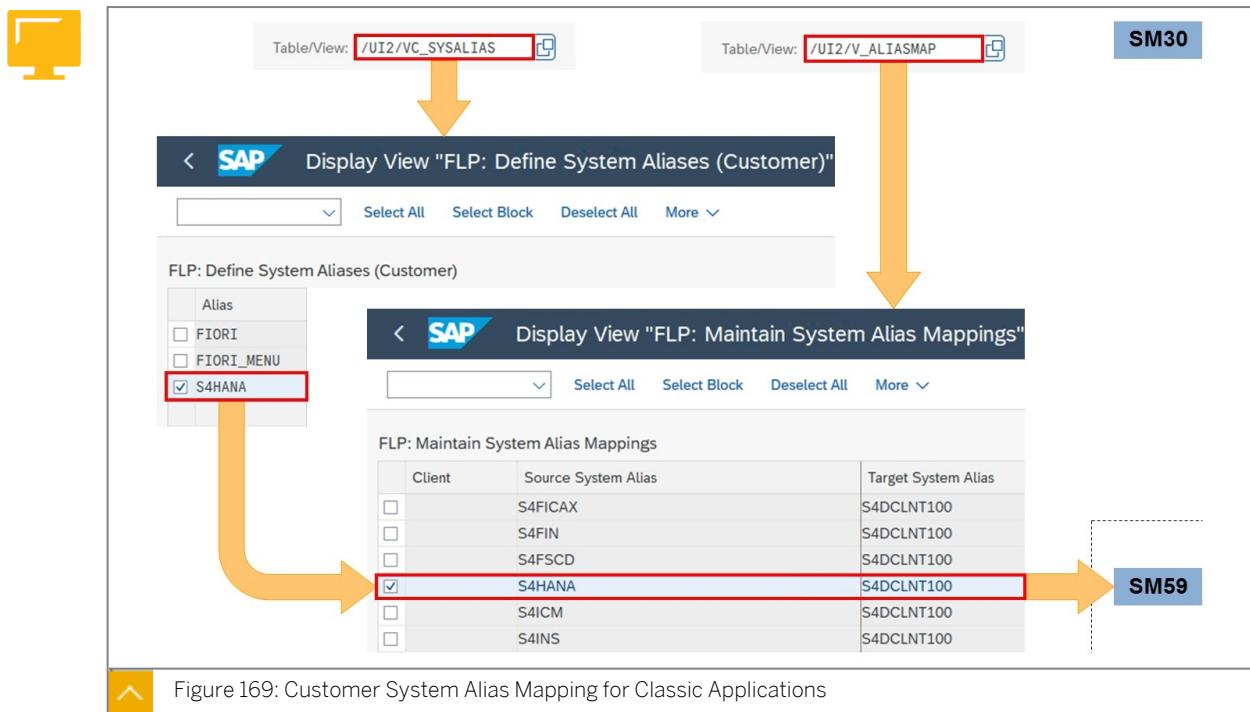
In the version called *Mass Maintenance for App Descriptors*, a dialog box appears asking if this back-end catalog should be created

Back-end catalogs are called *None-typed Replicable Catalogs* in the FLPAM, because no catalog type is defined in the system and they must be replicated to a front-server to be usable as source for business catalogs. A replication is a kind of extraction of the app descriptors as tiles and target mappings in a remote catalog. Independent of the deployment scenario, a system alias for classic applications must be assigned to the back-end catalog for the replication to work.

System Alias for Classic Applications



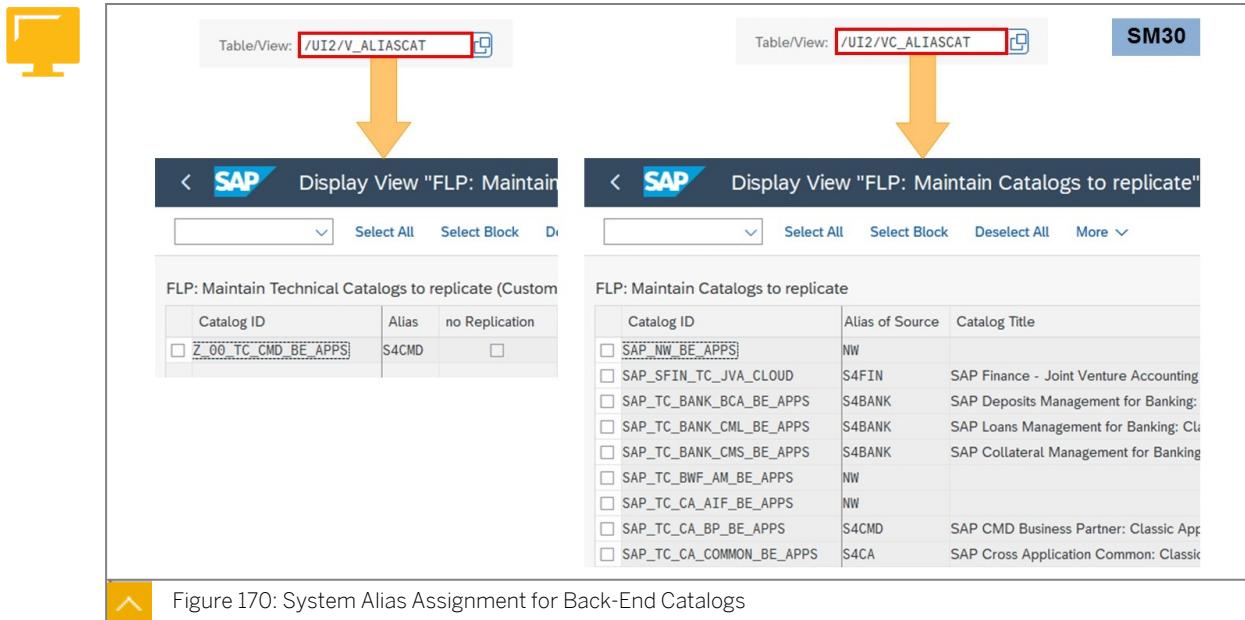
System aliases for classic applications are delivered by SAP and distinguished by solution area. They are viewable in the maintenance view `/UI2/V_SYSALIAS`. For each one an RFC (`<alias>_RFC`) and HTTPS (`<alias>_HTTPS`) destination should be created in the SM59. If several aliases point to the same system such as 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.



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 or an RFC (<alias>_RFC) and HTTPS (<alias>_HTTPS) destination should be created in the SM59.

Since SAP Fiori rapid activation in an embedded deployment, the destinations FIORI_CLASSIC_UI_RFC and FIORI_CLASSIC_UI_HTTPS are generated in SM59. An empty target system alias for a source system alias in /UI2/V_ALIASMAP is then sufficient to forward the request correctly.

Catalog Replication



Each back-end and replicable catalog needs a system alias for classic applications 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.

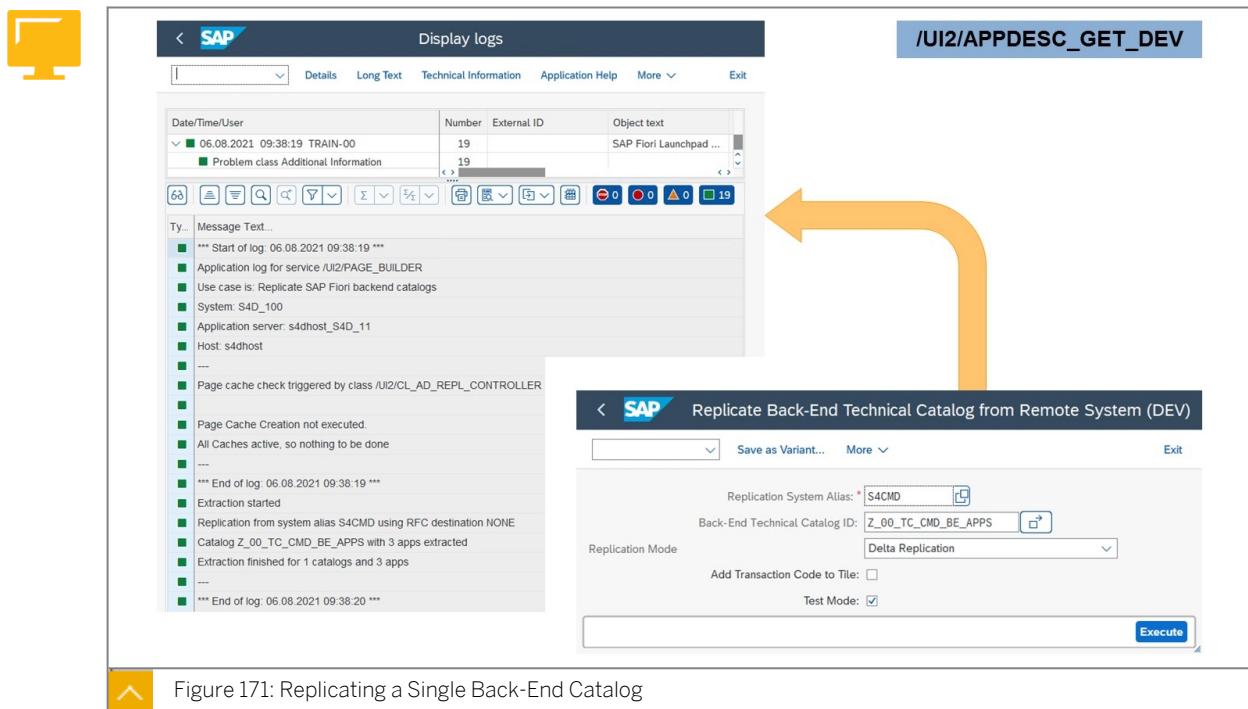


Figure 171: Replicating a Single Back-End Catalog

To replicate back-end and replicable catalogs to the front-end server (FES), SAP provides the following transactions:

/UI2/APPDESC_GET

Replicates all back-end and replicable catalogs referenced in selected business catalogs.

/UI2/APPDESC_GET_ALL

Replicates all back-end and replicable catalogs for all system aliases.

/UI2/APPDESC_GET_DEV

Replicates selected back-end and replicable catalogs for selected system alias.

The transactions are called on the FES. In all transactions, you can choose between a full or delta replication. A *Testmode* allows a test run before replicating anything.



Hint:

The transaction /UI2/APPDESC_GET_DEV was introduced in SAP S/4HANA 2020. If it is not available in your system, start the report /UI2/GET_APP_DESCR_REMOTE_DEV via transaction SA38.



LESSON SUMMARY

You should now be able to:

- Create replicable catalogs

Learning Assessment

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

Choose the correct answers.

- A Configuration
- B Personalization
- C Customizing
- D Internationalization

2. Which elements of SAP Fiori can be assigned to business roles?

Choose the correct answers.

- A Spaces
- B Pages
- C Business Catalog Groups
- D Technical Catalogs
- E Business Catalogs

3. How are the tools for managing SAP Fiori spaces and pages provided?

4. How many spaces should be assigned to a business role according to the best practices?

5. Which tool provides the definition of SAP Fiori groups?

6. What is the prerequisite for a user to see the tiles of a group?

7. Which element groups SAP Fiori apps based on solution area?

8. Which element groups SAP Fiori apps based on business duty?

9. 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

10. 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

11. What is needed to define a target mapping for an SAPUI5 application?

Choose the correct answers.

- A Component ID
- B Gateway service
- C ICF path
- D BSP name
- E System alias

12. Which is used to maintain technical catalogs?

13. What are the catalog types in SAP Fiori?

Choose the correct answers.

- A Replicable
- B Custom
- C Standard
- D Front-end
- E Back-end

14. What is meant by the term app descriptor?

15. Which catalog type must be set to enable back-end catalog maintenance in the SAP Fiori launchpad application manager?

16. What needs to be assigned to a back-end catalog for the replication to a front-end server?

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 elements of SAP Fiori can be assigned to business roles?

Choose the correct answers.

- A Spaces
- B Pages
- C Business Catalog Groups
- D Technical Catalogs
- E Business Catalogs

Correct. Business catalogs, business catalog groups, and spaces can be assigned to business roles.

3. How are the tools for managing SAP Fiori spaces and pages provided?

They are provided as apps in the SAP Fiori launchpad.

4. How many spaces should be assigned to a business role according to the best practices?

One

5. Which tool provides the definition of SAP Fiori groups?

SAP Fiori launchpad designer

6. What is the prerequisite for a user to see the tiles of a group?

The catalogs containing the tile definitions are part of the user master record.

7. Which element groups SAP Fiori apps based on solution area?

Technical Catalog (TC)

8. Which element groups SAP Fiori apps based on business duty?

Business Catalog (BC)

9. 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.

10. 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.

11. What is needed to define a target mapping for an SAPUI5 application?

Choose the correct answers.

- A Component ID
- B Gateway service
- C ICF path
- D BSP name
- E System alias

Correct. The component ID and ICF path of the SAPUI5 application is needed for a target mapping.

12. Which is used to maintain technical catalogs?

SAP Fiori launchpad application manager

13. What are the catalog types in SAP Fiori?

Choose the correct answers.

- A Replicable
- B Custom
- C Standard
- D Front-end
- E Back-end

Correct. Catalog types are back-end, standard, and replicable.

14. What is meant by the term app descriptor?

The combination of tiles and target mappings in standard, replicable, and back-end catalogs.

15. Which catalog type must be set to enable back-end catalog maintenance in the SAP Fiori launchpad application manager?

None

16. What needs to be assigned to a back-end catalog for the replication to a front-end server?

System alias for classic applications

Lesson 1

Exploring Rapid Activation for SAP Fiori

173

Lesson 2

Examining Basic Roles for SAP Fiori

177

Lesson 3

Configuring SAP Fiori Launchpad

181

Lesson 4

Troubleshooting SAP Fiori Launchpad

185

UNIT OBJECTIVES

- Explore Rapid Activation for SAP Fiori
- Examine Basic Roles for SAP Fiori
- Configure SAP Fiori launchpad
- Maintain SAP Fiori Launchpad Content

Unit 5

Lesson 1

Exploring Rapid Activation for SAP Fiori



LESSON OBJECTIVES

After completing this lesson, you will be able to:

- Explore Rapid Activation for SAP Fiori

Rapid Activation



Activation of individual apps leads to high upfront investment to experience SAP Fiori



Experience **Rapid Activation** for SAP Fiori along business roles

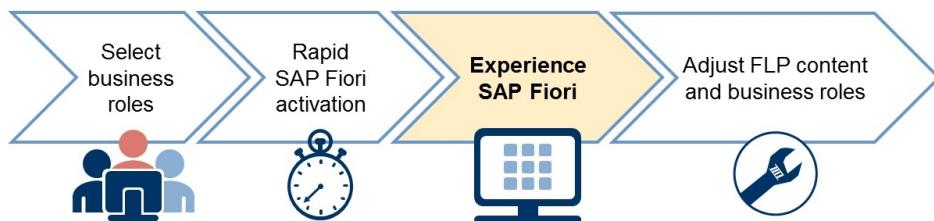


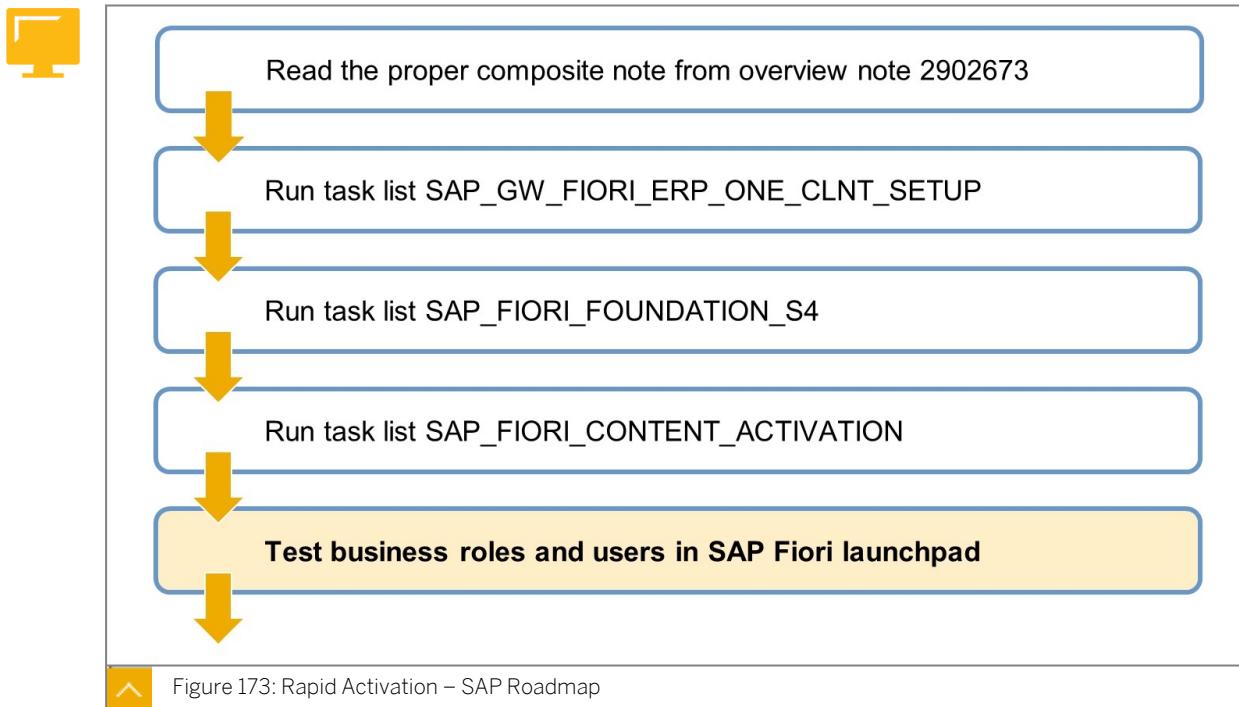
Figure 172: Rapid Activation – Goal

Since SAP S/4HANA 1709, rapid activation for SAP Fiori is available to speed up the activation time for SAP Fiori content. Instead of activating app by app, checking technical prerequisites step by step, task lists are in place to ease and speed up the configuration process role by role. Most importantly, the time needed until key-users can test apps in the *SAP Fiori launchpad* shrinks from weeks to days.



Caution:

Rapid activation for SAP Fiori in SAP S/4HANA only works within an embedded deployment.



The first part of rapid activation for SAP Fiori is about setting up the foundation for SAP Fiori followed by activating SAP Fiori apps shipped by SAP. The overview note [2902673](#) offers links to the composite notes for each release of SAP S/4HANA documenting every step of the activation. The following task lists are the main steps:

SAP_GW_FIORI_ERP_ONE_CLNT_SETUP

The technical setup for SAP Gateway, SAPUI5, and *SAP Fiori launchpad* in one client makes it possible to start the *SAP Fiori launchpad*.

SAP_FIORI_FOUNDATION_S4

The initial setup for *SAP Fiori launchpad* and SAP Fiori apps in SAP S/4HANA makes it possible to use all features of SAP Fiori.

SAP_FIORI_CONTENT_ACTIVATION

The content activation for business roles shipped by SAP makes it possible to use all SAP Fiori apps defined in the business role.

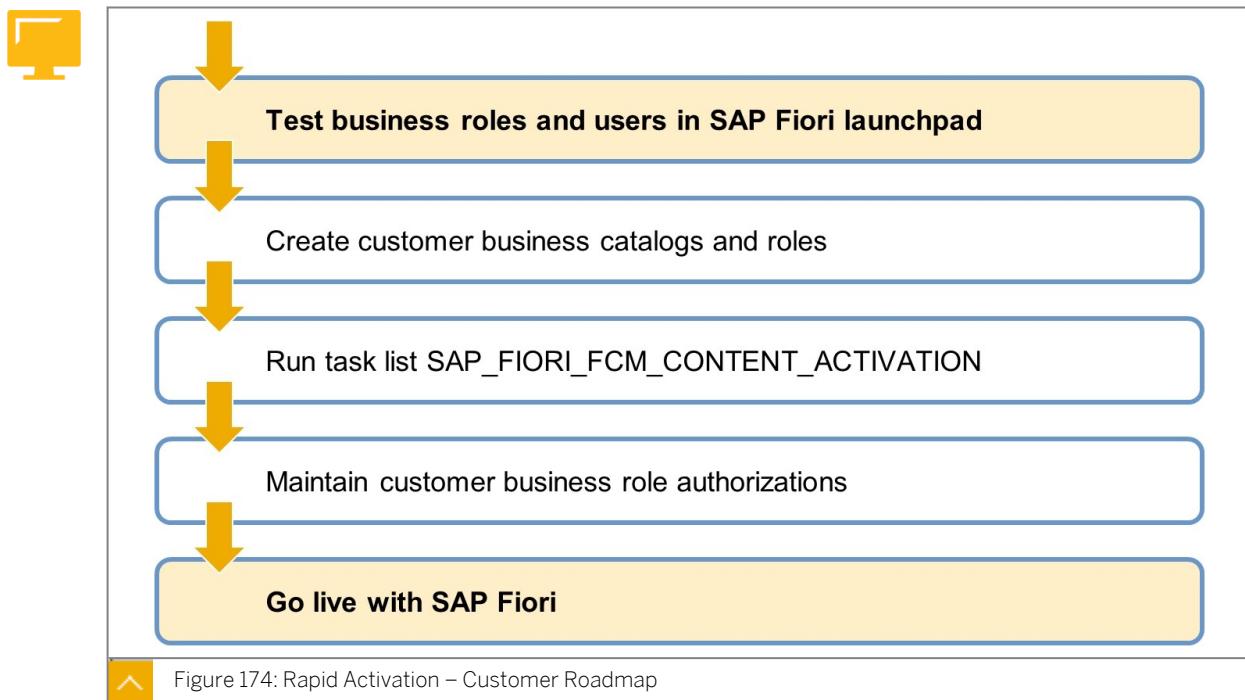


Figure 174: Rapid Activation – Customer Roadmap

The second part of rapid activation for SAP Fiori is about setting up customer-specific catalogs and roles including authorizations. The task list SAP_FIORI_FCM_CONTENT_ACTIVATION activates the content of business roles created by customers in the same way as SAP_FIORI_CONTENT_ACTIVATION does it for business roles shipped by SAP. Both task lists can be run multiple times for multiple business roles. In fact, it is better to activate role by role, rather than activating everything at once. After each task-list run, key-users responsible for the roles can be invited to test their business processes.

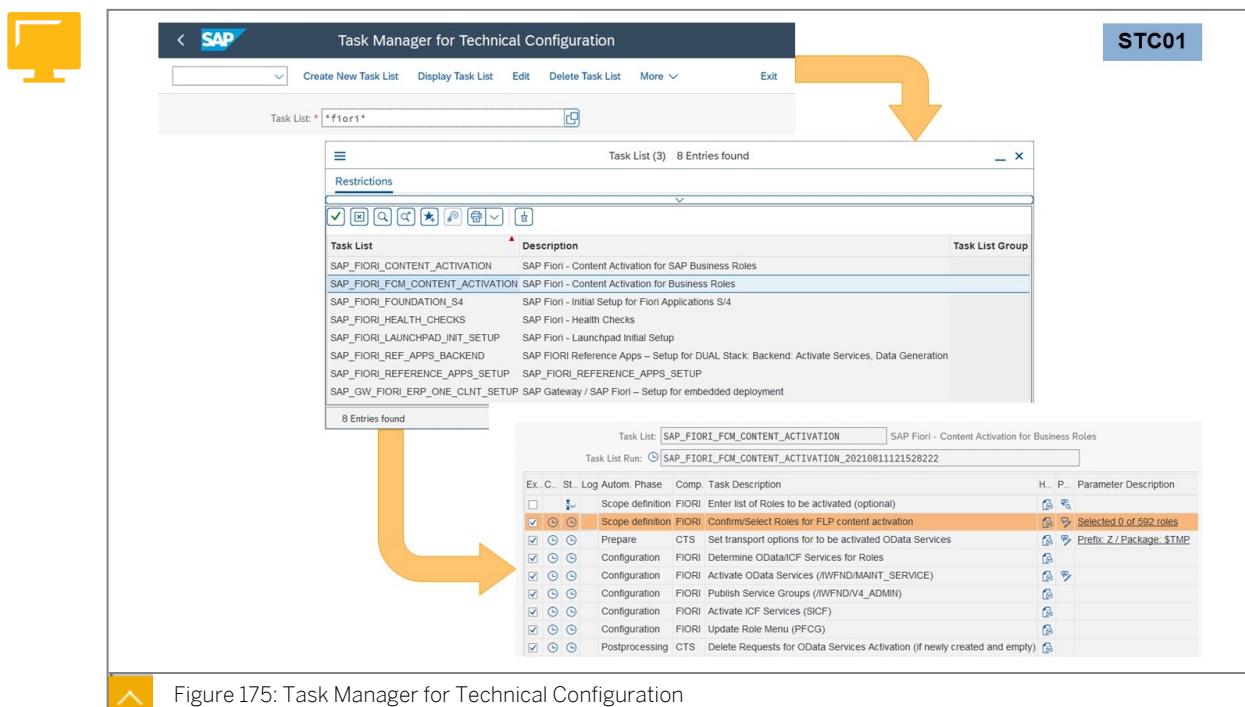


Figure 175: Task Manager for Technical Configuration

The transaction STC01 offers many task lists for a wide variation of technical configuration tasks in an ABAP system. First, search for a task list suitable for the configuration. Second, fill in all parameters in all steps of the task list. Some steps are mandatory, some can be skipped.

and some be activated in addition. Every step is documented and is often connected to other steps. Such dependencies are automatically checked. Finally, the task list can be run in a dialog process or in the background. Start long-running task-list runs in the background.

The screenshot shows the SAP Task List Run Monitor (STC02) interface. At the top, there's a toolbar with buttons for 'Start Search', 'Delete Selection Criterion', 'Selection Screen Help', 'Selection Options', 'More', and 'Exit'. Below the toolbar, there are two main sections: 'Task List Run Status' and 'Task List Run Finalized'. The 'Task List Run Status' section contains several checkboxes for filtering task runs based on their state: 'Waiting to be executed', 'Execution scheduled', 'Running', 'Stopped', 'Manual activities', 'Not finalized', and 'Does not need to be executed'. It also includes fields for 'Created By', 'Created on', 'Changed by', and 'Changed on', each with a date range selector. The 'Task List Run Finalized' section shows a table titled '8 Task List Runs found' with columns for 'St... Fl...', 'Task List Run', 'Task List', 'Task List Variant', 'Created By', and 'Creation Time'. The table lists four entries: SAP_FIORI_CONTENT_ACTIVATION_20210..., SAP_GATEWAY_ACTIVATE_ODATA_SERV..., /WFND/TL_SERVICE_MAINTENANCE_2021..., and SAP_FIORI_FOUNDATION_S4_2021042908. A large orange arrow points downwards from the top left towards the table.

Figure 176: Task-List Run Monitor

All task-list runs are logged in transaction STC02. Every detail of a task-list run from the past and the result of task-list runs running in the background can be checked.



Note:

More information about this topic can be found in UX200 (SAP Fiori – System Administration):

<https://training.sap.com/course/ux200>



LESSON SUMMARY

You should now be able to:

- Explore Rapid Activation for SAP Fiori

Unit 5

Lesson 2

Examining Basic Roles for SAP Fiori



LESSON OBJECTIVES

After completing this lesson, you will be able to:

- Examine Basic Roles for SAP Fiori

Predefined Roles

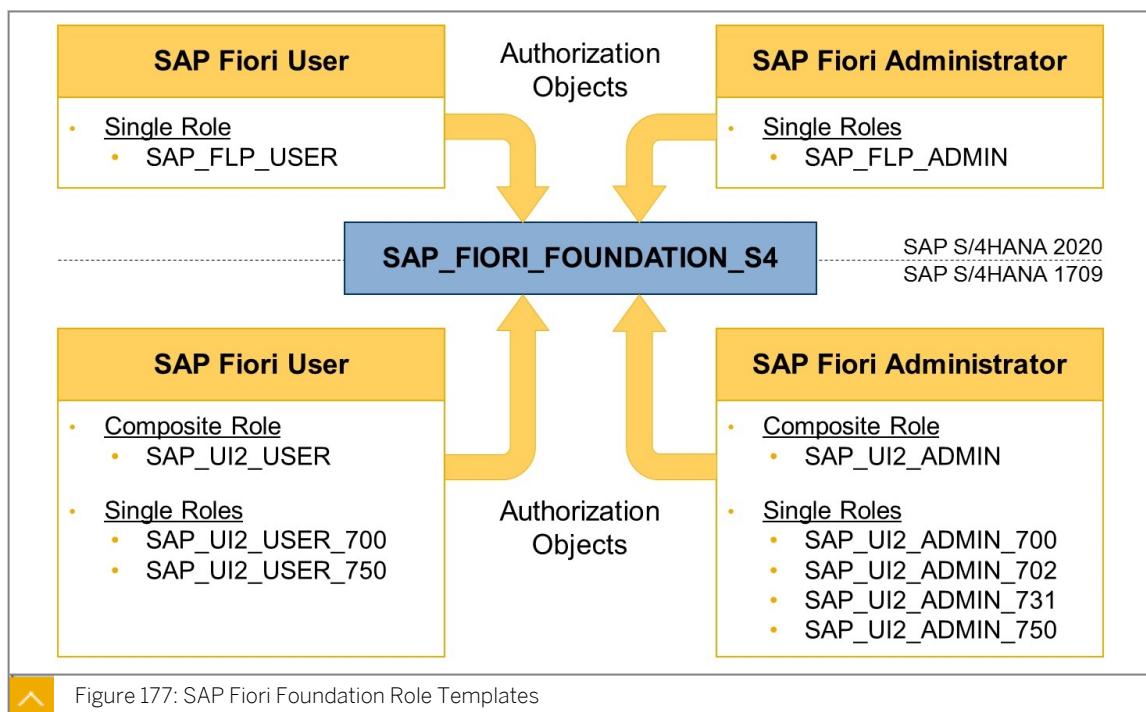
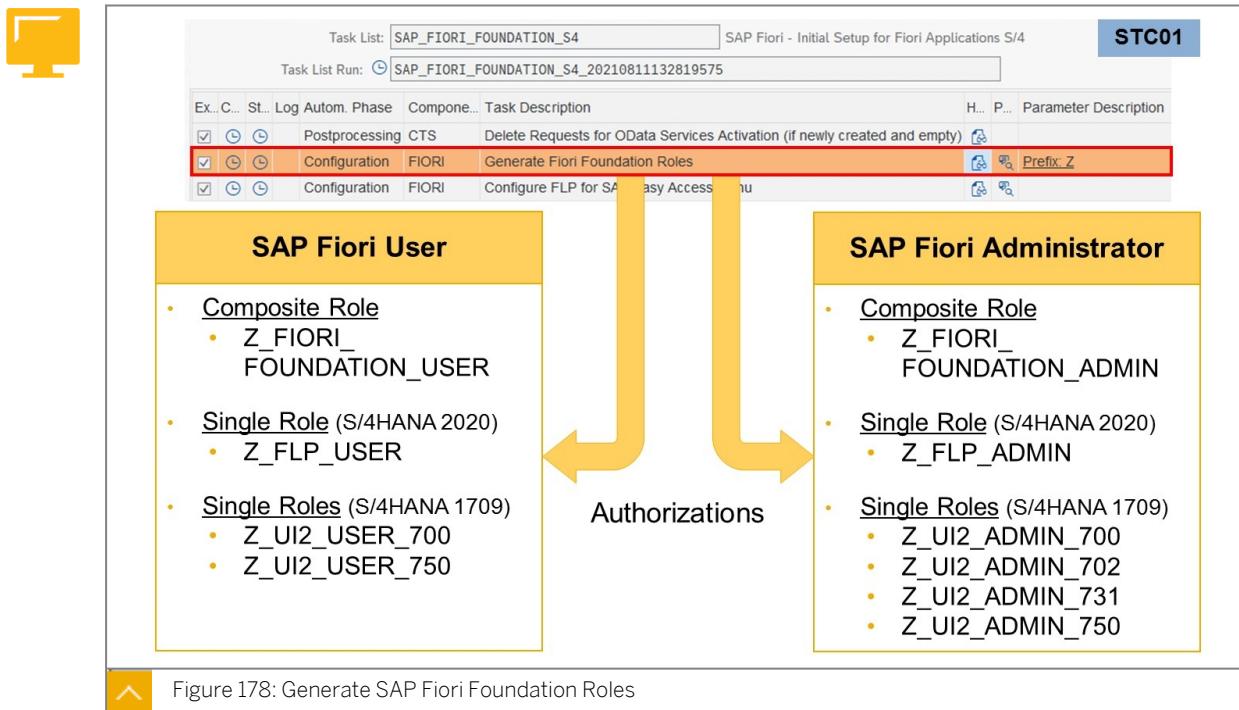


Figure 177: SAP Fiori Foundation Role Templates

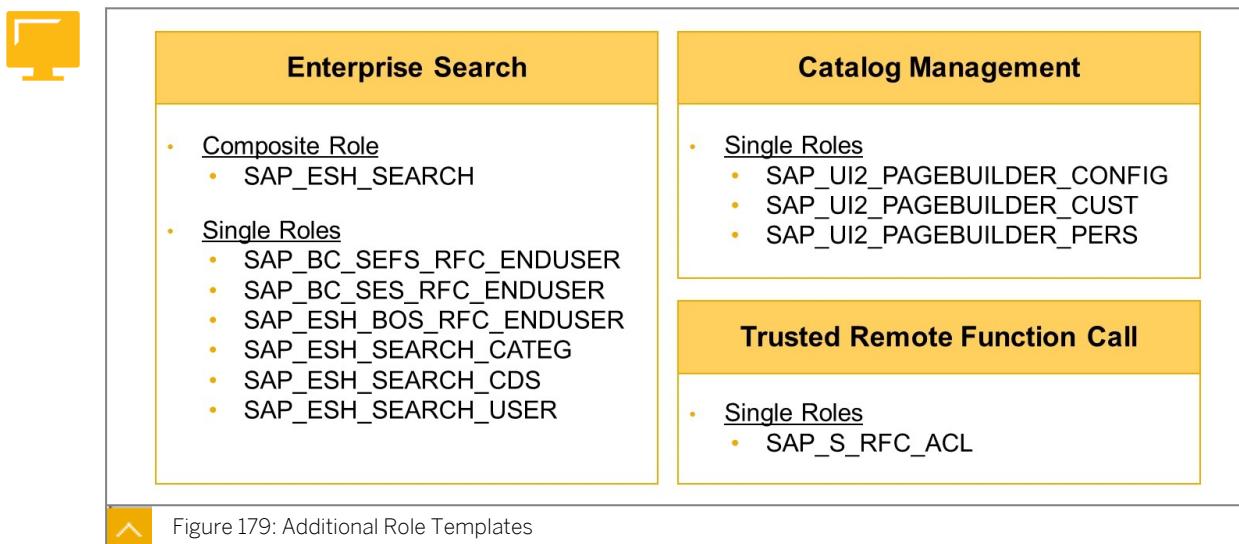
SAP ships foundation role templates for users and administrators of the SAP Fiori launchpad (FLP). They consist of transactions, catalogs, and above all authorization objects. In SAP S/4HANA 1709, several single roles build on each other and contain elements available since a certain release of SAP Fiori visualized as number in the name. The roles ending with "700" include the core authorizations objects for using the SAP Fiori launchpad. The single roles are combined in the composite roles SAP_UI2_USER and SAP_UI2_ADMIN.

Since SAP S/4HANA 2020, all roles are replaced by the single roles SAP_FLP_USER and SAP_FLP_ADMIN.

The task of an administrator is to create from these role templates fully authorized business roles assignable to users. Since rapid activation for SAP Fiori, this task is taken over by the task list SAP_FIORI_FOUNDATION_S4.



The task list generates fully authorized business roles for each role template with a leading prefix ("Z" is the default prefix). Assigning the composite role Z_FIORI_FOUNDATION_USER to a user, allows FLP start(up) and personalization. Assigning the composite role Z_FIORI_FOUNDATION_ADMIN to a user, allows SAP Fiori content administration.



Being authorized to use the FLP does not mean that any app can be started or data can be searched. To do that, additional roles and authorizations are in place. The topic authorizations must be taken seriously. SAP recommends to approach authorization specialists or to educate oneself specifically in the SAP S/4HANA authorization concept.



Note:

More information about this topic can be found in ADM945 (Authorization Concept for SAP Fiori on SAP S/4HANA):

<https://training.sap.com/course/adm945>



LESSON SUMMARY

You should now be able to:

- Examine Basic Roles for SAP Fiori

Configuring SAP Fiori Launchpad

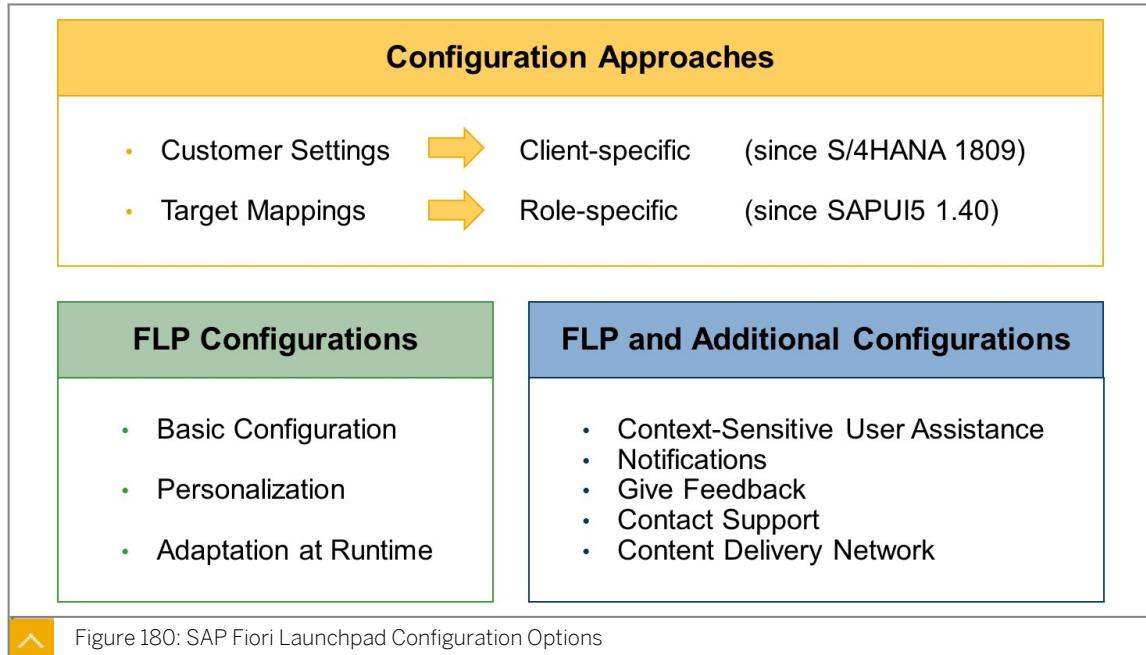


LESSON OBJECTIVES

After completing this lesson, you will be able to:

- Configure SAP Fiori launchpad

Configuration Options



The *SAP Fiori launchpad (FLP)* has many configuration options. There are several basic configurations and also advanced configurations needing other additional technical settings elsewhere. The configurations depend on the release of the software component SAP_UI. There are three approaches to configure the FLP:

Customer settings

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

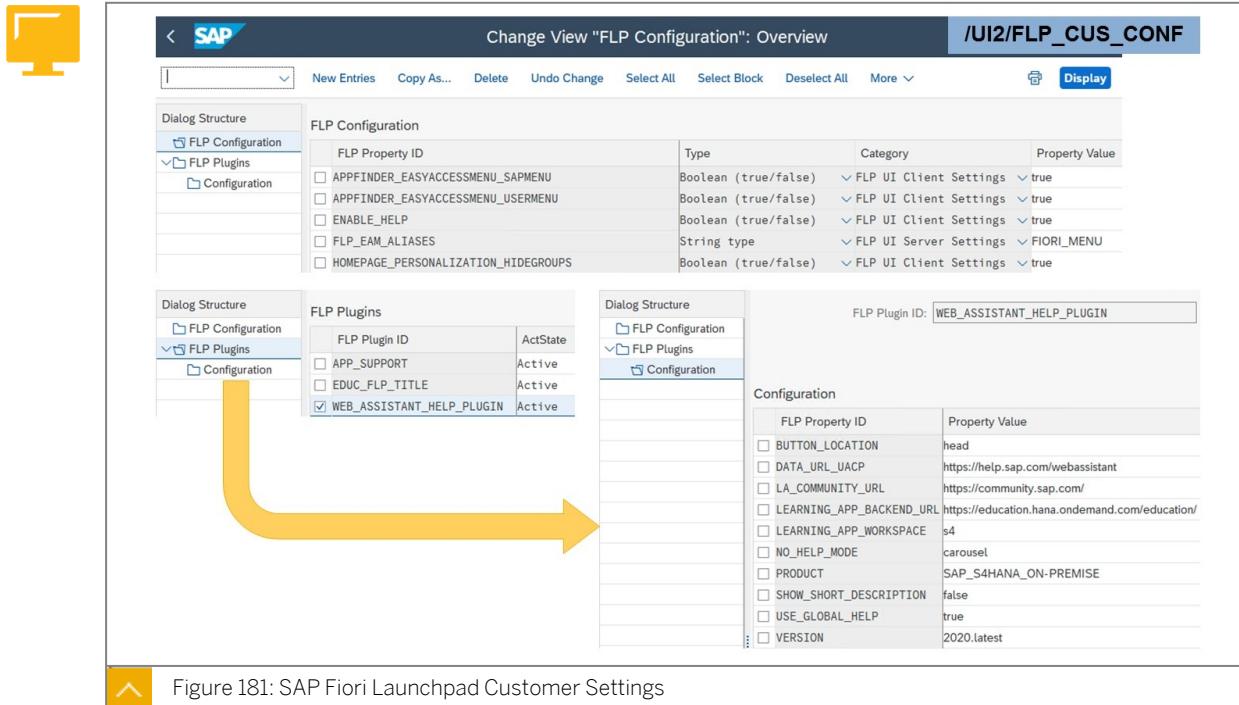
Target Mapping

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

Configuration File (deprecated)

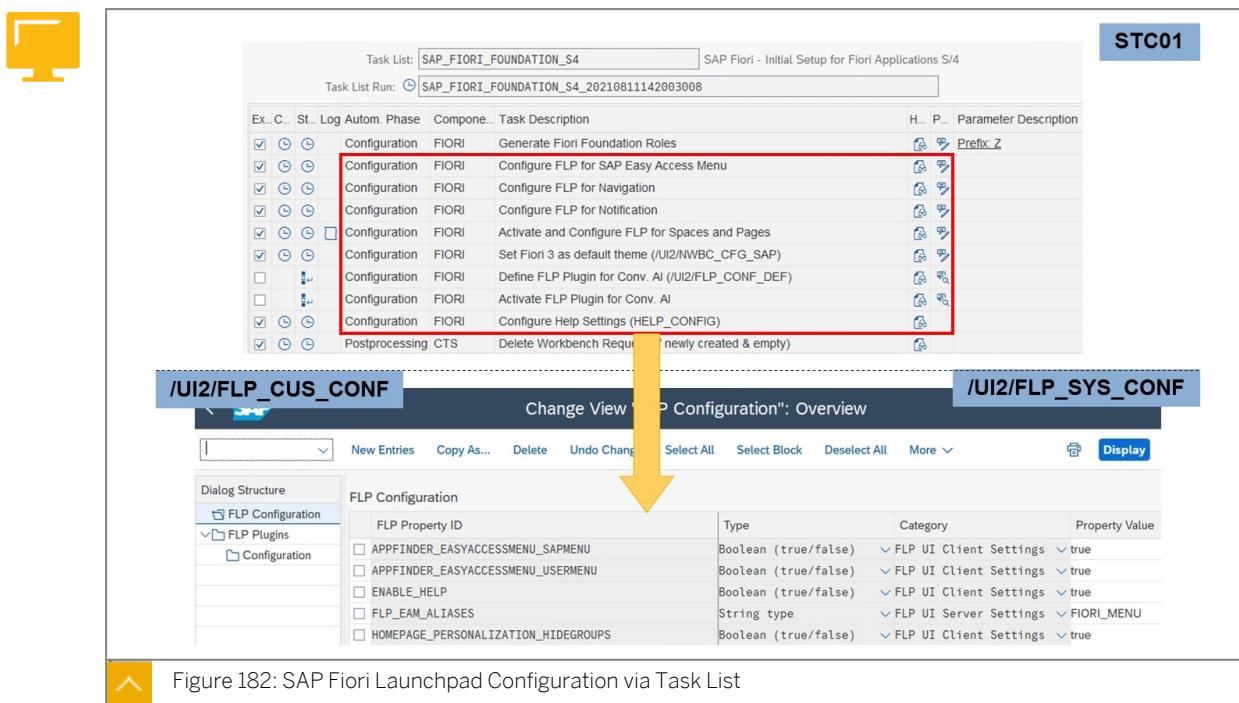
Create and upload a configuration file. This is deprecated since SAPUI5 1.40. These parameters are valid for all client.

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.



FLP Customer Settings (/UI2/FLP_CUS_CONF) are available since SAP S/4HANA 1809. They allow central customizing of the FLP for all users in one client. The value help for properties makes it easy to overview the possible settings. A full release-dependent documentation of all settings can be found in the FLP documentation on <https://help.sap.com/>.

Beside general settings, you can add plugins to the FLP and configure them. For example, the Web assistant plugin provides the context-sensitive user assistance in SAP Fiori apps.



Task lists in transaction STC01 also create configuration settings for the FLP. The task list SAP_FIORI_FOUNDATION_S4 creates multiple entries in *FLP Customer Settings* (/UI2/FLP_CUS_CONF) or *FLP System Settings* /UI2/FLP_SYS_CONF depending on the task parameters you set.



Note:

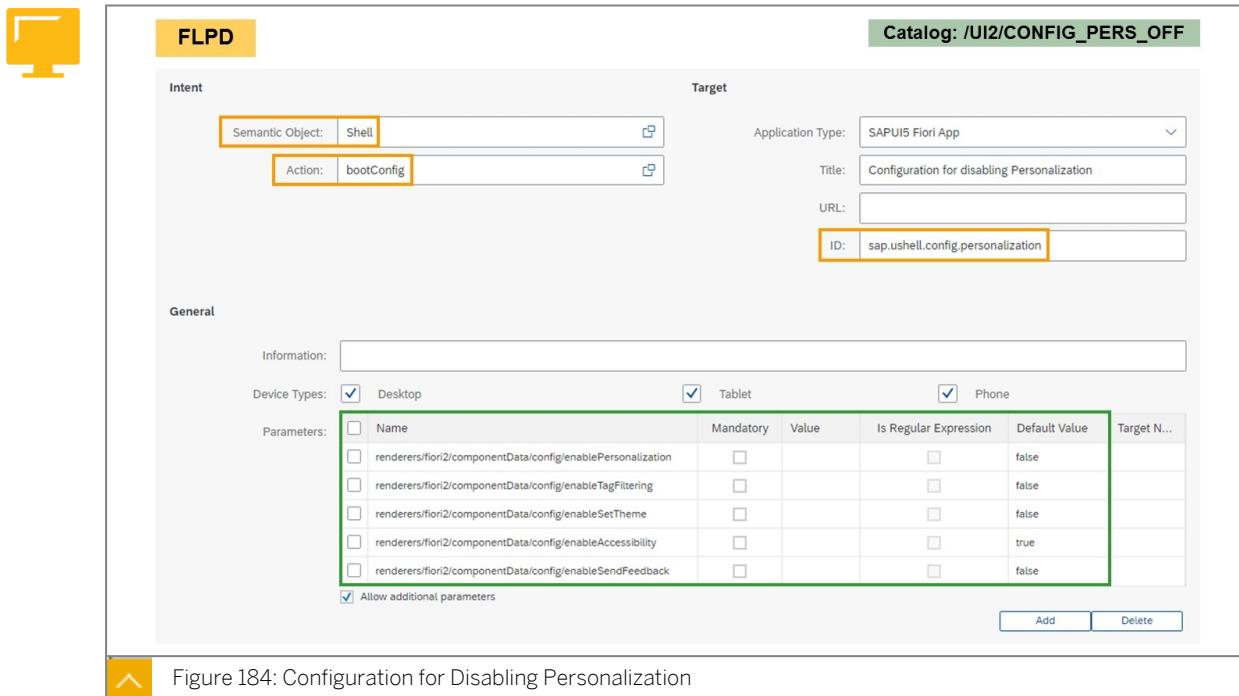
The transaction /UI2/FLP_SYS_CONF allows the configuration of the FLP for all clients. However, these settings are saved in the SAP namespace. They count as modifications, including all the drawbacks.



Name	Mandatory	Value	Is Regular Expression	Default Value
renderers/fiori2/componentData/config/applications/Shell-home/enableHideGroups	<input type="checkbox"/>		<input type="checkbox"/>	true
renderers/fiori2/componentData/config/sizeBehaviorConfigurable	<input type="checkbox"/>		<input type="checkbox"/>	true
apps/inputFieldHistory/enabled	<input type="checkbox"/>		<input type="checkbox"/>	true

Figure 183: Custom Catalog for Basic FLP Configuration

Target mappings for configuring the FLP can be created in any catalog. However, it is recommended that you 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 action **bootConfig** is meant for general for settings for the FLP. 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.



The screenshot shows the SAP Fiori Launchpad configuration interface. At the top left is a yellow icon of a computer monitor with the text 'FLPD'. At the top right is a green bar labeled 'Catalog: /UI2/CONFIG_PERS_OFF'. The main area is divided into 'Intent' and 'Target' sections.

Intent:

- Semantic Object: Shell
- Action: bootConfig

Target:

- Application Type: SAPUI5 Fiori App
- Title: Configuration for disabling Personalization
- URL: (empty)
- ID: sap.ushell.config.personalization

General:

Information: (empty input field)

Device Types: Desktop Tablet Phone

Parameters:

Name	Mandatory	Value	Is Regular Expression	Default Value	Target N...
renderers/flori2/componentData/config/enablePersonalization	<input type="checkbox"/>		<input type="checkbox"/>	false	
renderers/flori2/componentData/config/enableTagFiltering	<input type="checkbox"/>		<input type="checkbox"/>	false	
renderers/flori2/componentData/config/enableSetTheme	<input type="checkbox"/>		<input type="checkbox"/>	false	
renderers/flori2/componentData/config/enableAccessibility	<input type="checkbox"/>		<input type="checkbox"/>	true	
renderers/flori2/componentData/config/enableSendFeedback	<input type="checkbox"/>		<input type="checkbox"/>	false	

Allow additional parameters

Add Delete

Caption: Figure 184: 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 5

Lesson 4

Troubleshooting SAP Fiori Launchpad



LESSON OBJECTIVES

After completing this lesson, you will be able to:

- Maintain SAP Fiori Launchpad Content

Maintenance Transactions



Figure 185: 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 system administrators
- SAP Fiori Launchpad Settings* for content and system administrators
- Setting Up Launchpad Content* for business specialists, content administrators, and developers
- Setting Up Launchpad Layout and Structure* for business specialists and content administrators
- Launchpad Support Tools* for business specialists, content administrators, and developers

- *Launchpad Data Administration* for content and system administrators
- *Exposing Content to Launchpad on SAP Business Technology Platform* for business specialists and content administrators

The screenshot shows two SAP Fiori screens side-by-side. The top screen is titled "SAP Fiori Launchpad Content Aggregator: Client-Specific (Customizing)" and has a URL bar showing "/UI2/FLPCA". It features a search bar, role filters for "SAP_FLP_ADMIN", and checkboxes for "Display OData Services" and "Display ICF Services". A yellow arrow points from the top screen down to the bottom screen. The bottom screen is titled "Launchpad Content Aggregator: Client-Specific (Customizing)" and also has a URL bar showing "/UI2/FLPCA". It includes a toolbar with various icons and a table listing content items. The table columns are: T/TM Match, Tile/Target Mapping ID, Semantic Object, Action, and Title/Subtitle/Information. The data in the table is as follows:

T/TM Match	Tile/Target Mapping ID	Semantic Object	Action	Title/Subtitle/Information
TM only	77725A3DAF0D12684620A6E4AC148FA0	Shell	plugin	Launch App Support Tool
TM only	77725A3DAF0D12684620A6E4AC148FA0	Shell	plugin	Launch App Support Tool
Tile + TM	6B9868670577798014DDDA0EDA8B1C30	FioriLaunchpad	configureClientSpecific	SAP Fiori Launchpad Designer - Client Specific
Tile + TM	36E84F091F481EA483F29BD04ED0B90F	FLPApplication	manage	Manage Launchpad Apps - Cross Client
Tile + TM	2F7EED4A603F90B6280C450AF74D4DA5	FLPBusinessCatalog	manageClientSpecific	FLP Content Manager: Client-Specific
Tile + TM	30D1F4F0EB687D7CB66589F10BD87593	FLPBusinessCatalog	manageCrossClient	FLP Content Manager: Cross-Client
Tile + TM	0763EF2CC5B7B94D78F6B2CC8278BDB3	FioriLaunchpad	configureCrossClient	SAP Fiori Launchpad Designer - Cross Client
Tile + TM	5CC3A7C791218B4DD9F7E95C8801B4B8	FLPPage	manage	Manage Launchpad Pages
Tile + TM	5CC3A7C791218B4DD9F7E95C8801B4B8	FLPPage	manage	Manage Launchpad Pages
Tile + TM	3529868D85ACA95A40A14905DC4C90D9	FLPSpace	manage	Manage Launchpad Spaces
Tile + TM	3529868D85ACA95A40A14905DC4C90D9	FLPSpace	manage	Manage Launchpad Spaces

Figure 186: SAP Fiori Launchpad Content Aggregator

Since SAP S/4HANA 2020, transaction `/UI2/FLPCA` shows all content assigned to business roles. This is very useful for getting an overview about everything a user gets in the *SAP Fiori launchpad (FLP)* when assigning roles:

- Sort by catalogs to show the content in the app finder.
- Sort by semantic object to show the intent-based navigation links.
- Sort by services to show the technical prerequisites.

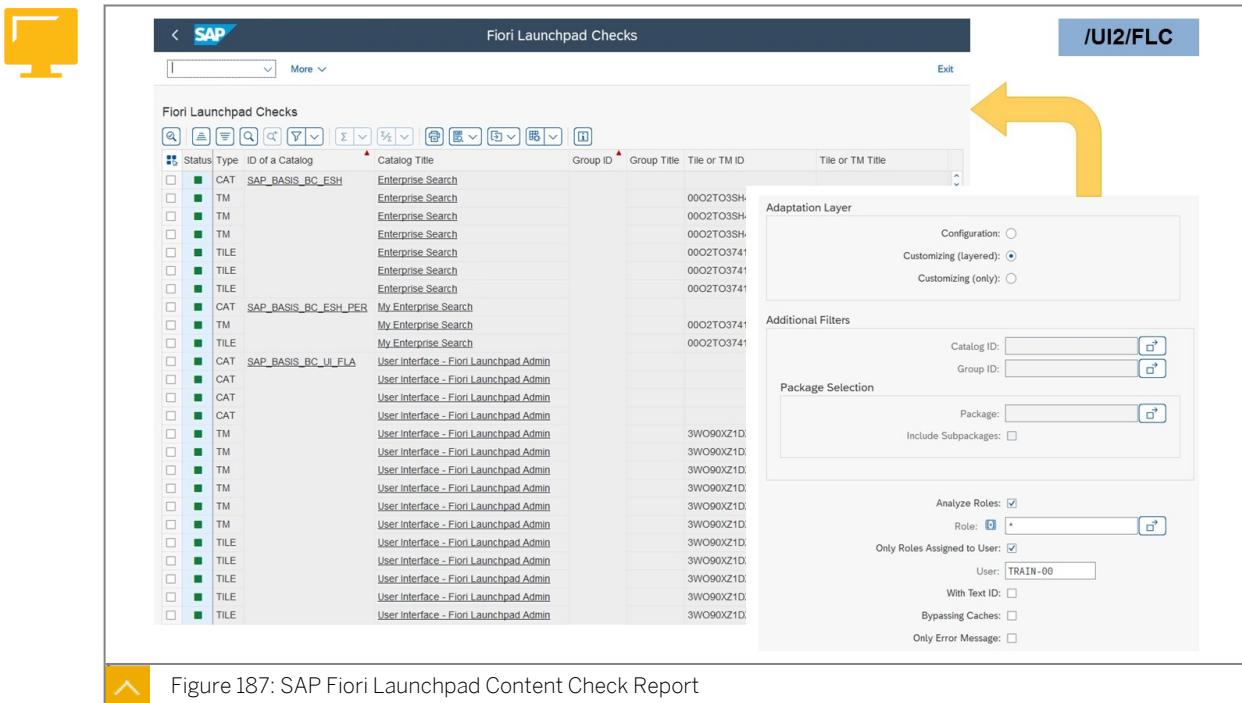


Figure 187: SAP Fiori Launchpad Content Check Report

An important support tool for the FLP 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.

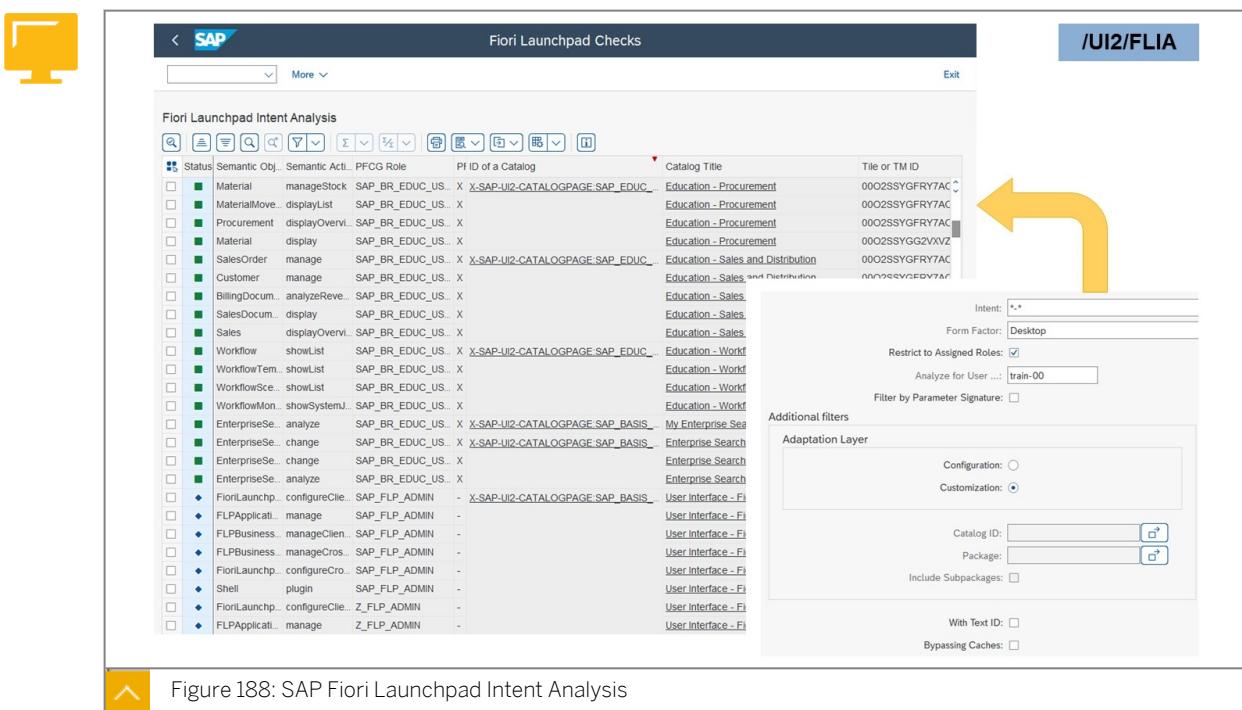
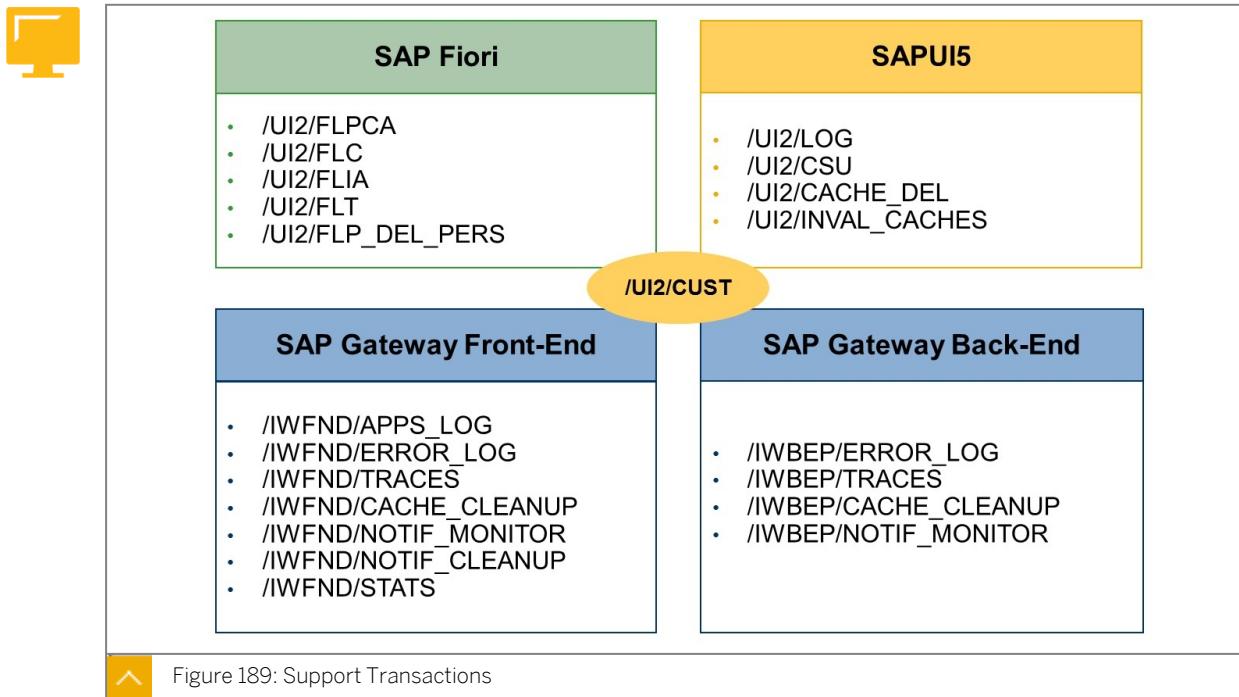


Figure 188: SAP Fiori Launchpad Intent Analysis

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 analysis, 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. It is recommended to restrict the analysis to the assigned roles of a user.

Support Transactions



There are many other transactions available for logging, tracing, cache handling, cleanup and so on, in the areas of SAP Fiori, SAPUI5, and SAP Gateway. There is no need to know them all by heart. Transaction `/UI2/CUST` organizes all of them as a tree in the implementation guide.



Note:

For more information about this topic, please read SAP note [2116090](#).

App Support

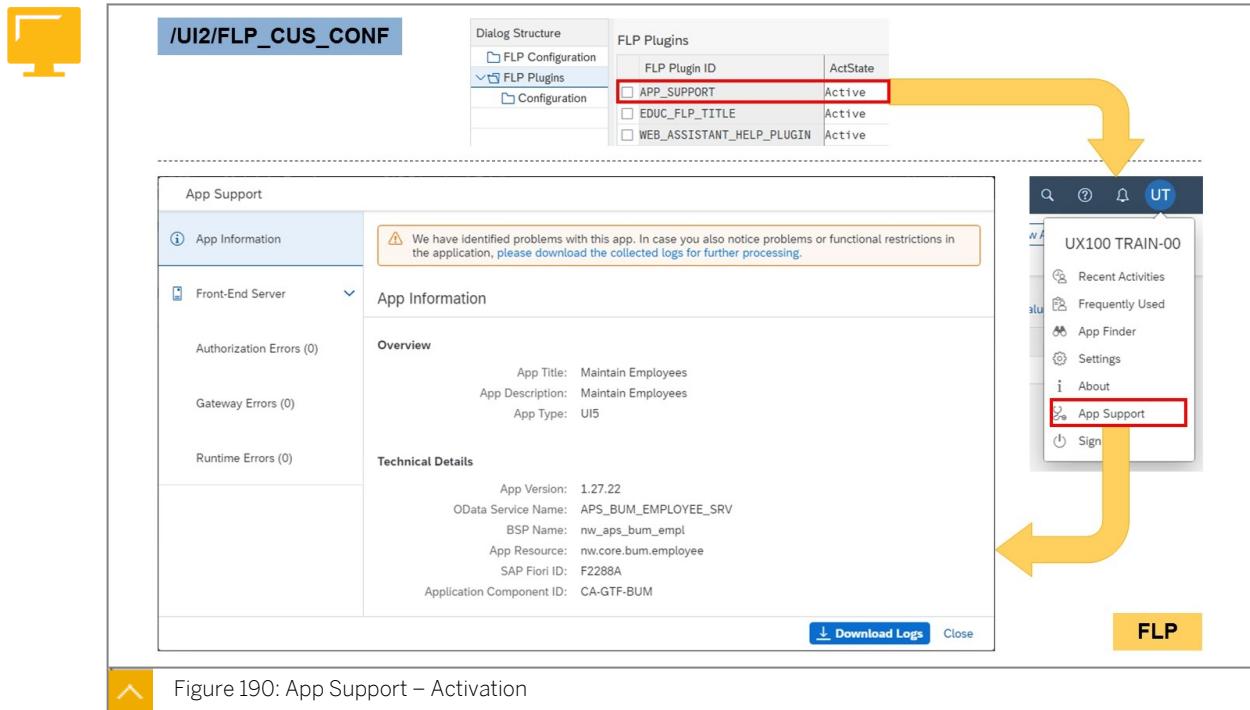
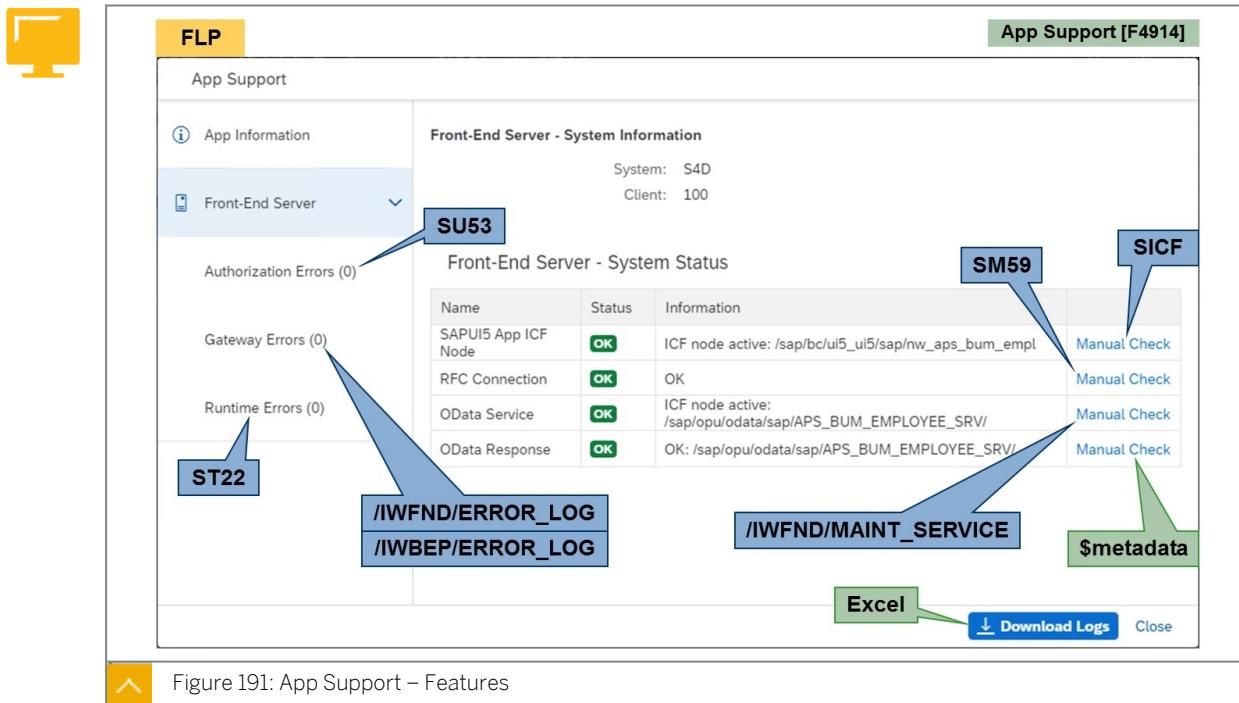


Figure 190: App Support – Activation

Since SAP S/4HANA 2020, the SAPUI5 plugin *App Support* [F4914] can be activated for troubleshooting apps inside the *SAP Fiori launchpad* (FLP). *App Support* is documented in the *SAP Fiori apps reference library*, included in the *SAP_FLP_ADMIN* role, and must be activated like any other SAPUI5 app. By assigning the target mapping or by activating the plugin in the */UI2/FLP_CUS_CONF*, the *App Support* button appears in the user actions menu when launching apps in the FLP.

App Support supports the following app types:

- SAPUI5 Fiori app
- Web Dynpro ABAP application
- WebClient UI application
- ABAP transaction



Beside the app information on the first screen, *App Support* offers plenty of jump offs to additional tools. The *Front-End Server* screen shows the connectivity status of the app. The transactions SM59, SICF, and /IWFND/MAINT_SERVICE can be started for a manual check or a browser showing the metadata of the OData service.

In case of errors, *App Support* distinguishes between authorization, SAP Gateway, and runtime errors. SU53, /IWFND/ERROR_LOG, /IWBEPE/ERROR_LOG, and ST22 are available as jump offs for details.

Finally everything can be downloaded as a Microsoft Excel-file (.xlsx) for documentation or further investigation. Each screen of *App Support* results in an own tab in Microsoft Excel (*App Information*, *Front-End Server*, ...).



LESSON SUMMARY

You should now be able to:

- Maintain SAP Fiori Launchpad Content

Learning Assessment

1. What is the starting point for activating apps with rapid activation for SAP Fiori?

2. Which task lists for SAP Fiori are run only once per ABAP system release?

Choose the correct answers.

- A SAP_FIORI_FOUNDATION_S4
- B SAP_FIORI_CONTENT_ACTIVATION
- C SAP_FIORI_FCM_CONTENT_ACTIVATION
- D SAP_GW_FIORI_ERP_ONE_CLNT_SETUP

3. What is the foundation role shipped by SAP for SAP Fiori users?

4. What is missing in the foundation roles?

5. Which configuration approach is available since SAP S/4HANA 1809?

6. What needs to be created to enable SAP Fiori launchpad customizing per role?

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

Choose the correct answers.

- A /UI2/FLT
- B /UI2/FLIA
- C /UI2/FLC
- D /UI2/FLP
- E /UI2/FLPCA

8. What is the namespace for SAP Gateway front-end transactions?

Learning Assessment - Answers

1. What is the starting point for activating apps with rapid activation for SAP Fiori?

Business roles

2. Which task lists for SAP Fiori are run only once per ABAP system release?

Choose the correct answers.

- A SAP_FIORI_FOUNDATION_S4
- B SAP_FIORI_CONTENT_ACTIVATION
- C SAP_FIORI_FCM_CONTENT_ACTIVATION
- D SAP_GW_FIORI_ERP_ONE_CLNT_SETUP

Correct. The task lists SAP_FIORI_FOUNDATION_S4 and SAP_GW_FIORI_ERP_ONE_CLNT_SETUP are run only once per ABAP system.

3. What is the foundation role shipped by SAP for SAP Fiori users?

SAP_FLP_USER

4. What is missing in the foundation roles?

Authorizations

5. Which configuration approach is available since SAP S/4HANA 1809?

Definition of customer settings using transaction /UI2/FLP_CUS_CONF

6. What needs to be created to enable SAP Fiori launchpad customizing per role?

A business catalog with a target mapping holding the customizing

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

Choose the correct answers.

- A /UI2/FLT
- B /UI2/FLIA
- C /UI2/FLC
- D /UI2/FLP
- E /UI2/FLPCA

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

8. What is the namespace for SAP Gateway front-end transactions?

/IWFND/

Lesson 1

Using the UI Theme Designer

197

Lesson 2

Using SAP Fiori Runtime Authoring

203

Lesson 3

Extending SAP Fiori Applications

207

UNIT OBJECTIVES

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

Unit 6

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 screenshot shows the SAP UI Theme Designer application. On the left, there's a sidebar with 'Target Pages' and 'Applications' sections, where 'SAP Fiori Launchpad' is selected. In the center, a preview window displays a Fiori Launchpad screen titled 'Manage Banks' with a background image of a field of red poppies. The preview window has tabs for 'Original' and 'Preview'. A yellow callout points to the 'Preview' tab with the label 'Preview'. Above the preview window, a yellow callout points to the 'Screen type' dropdown with the label 'Screen type'. To the right of the preview, there's a configuration panel titled 'Education Corn Poppy' under 'User TRAIN-00'. The panel includes sections for 'Company Logo' (set to 'none'), 'Main' (color settings for brand and base colors), 'Highlights' (highlight, focus, and contrast colors), and 'Semantic Colors' (negative color). A vertical toolbar on the far right contains icons for selection, zoom, and other design tools.

Figure 192: UI Theme Designer

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.

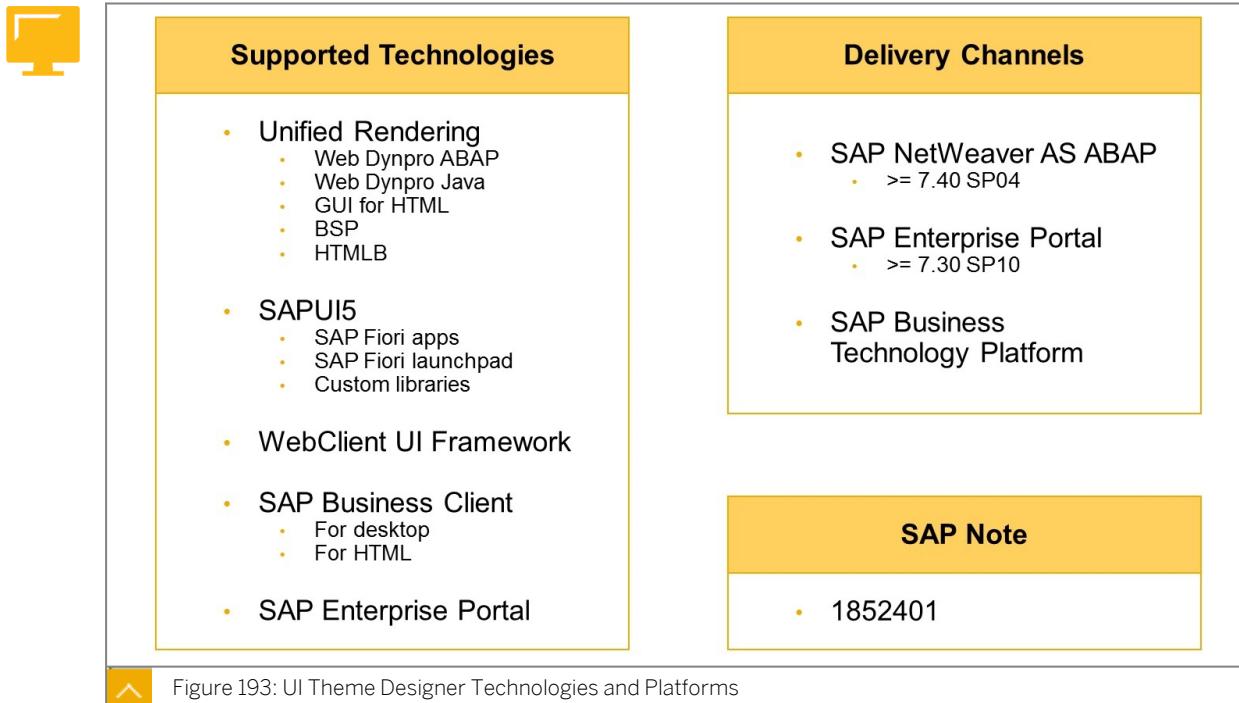


Figure 193: UI Theme Designer Technologies and Platforms

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 Business Technology Platform

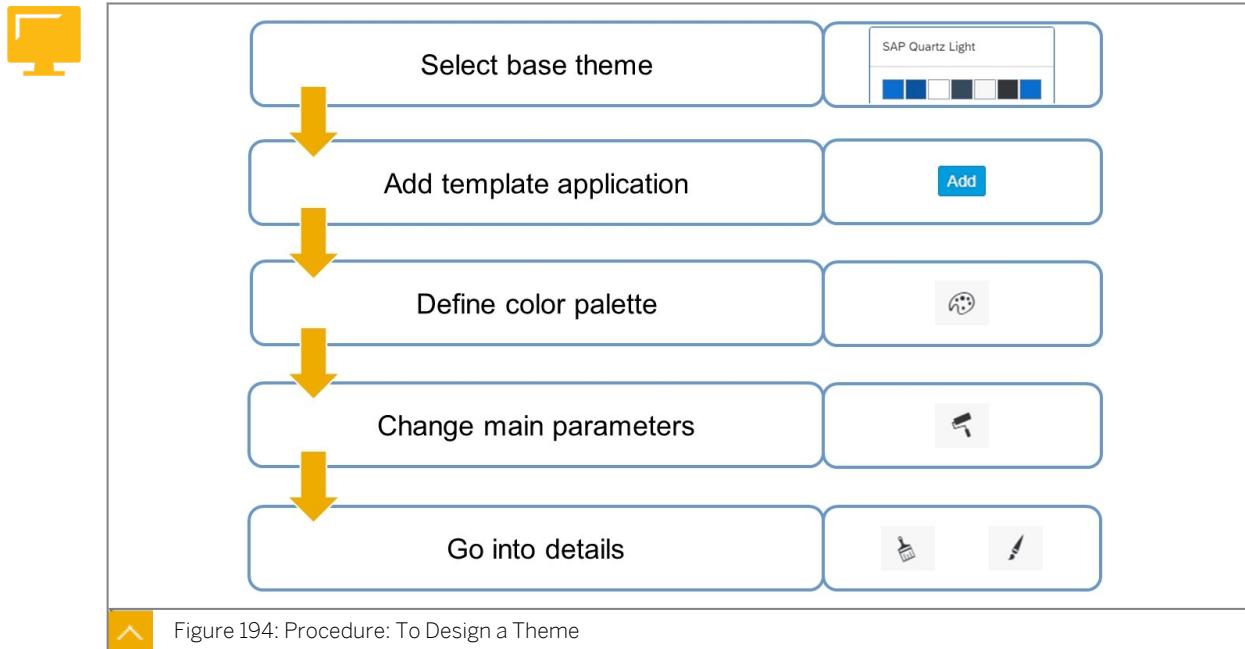
SAP Fiori launchpad → *User Actions Menu* → *Theme Manager* → *Launch Theme Designer*

Prerequisites, installation, and configuration are available in the SAP Note [1852401](#).



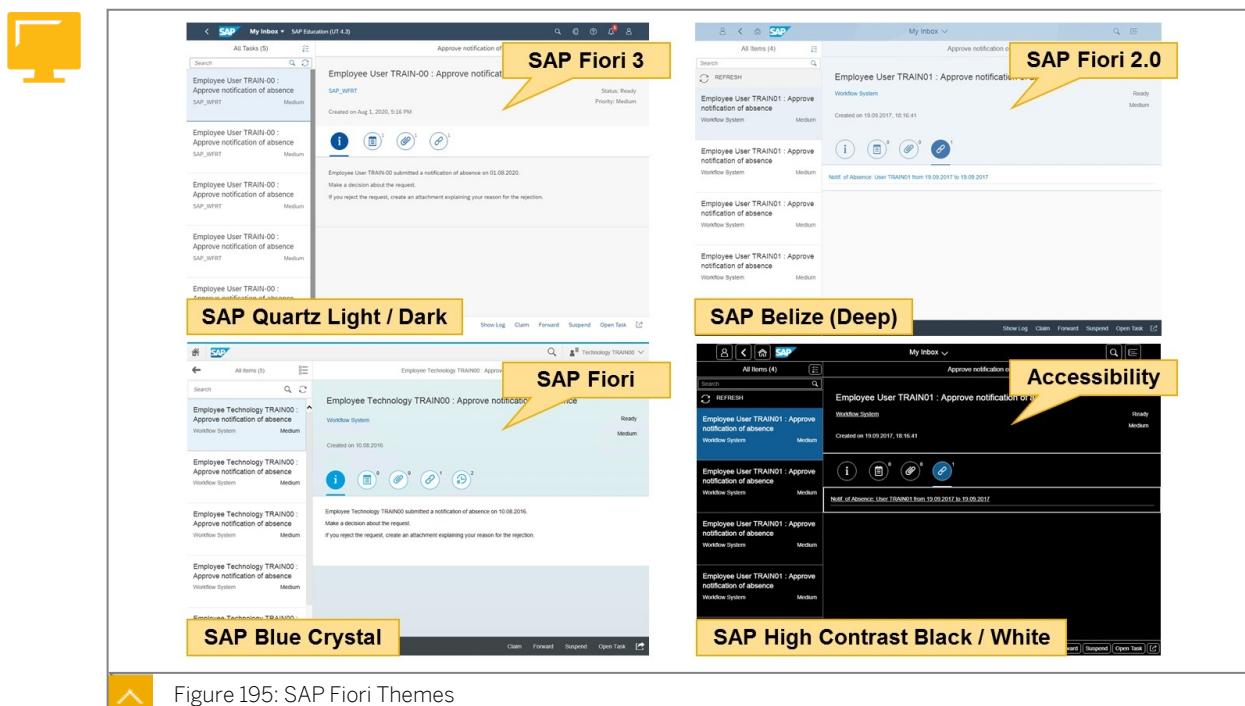
Hint:

When creating themes for the Unified Rendering, it is recommended to always use the latest version. Please check SAP Note [2090746](#) for details.



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 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 delivers the following themes for SAPUI5:

SAP Quartz Dark

- Theme released for SAP Fiori 3
- Initial shipment in SAP S/4HANA Cloud 2002 / SAPUI5 1.72
- Uses own typeface “72”

SAP Quartz Light

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

SAP Belize (Deep)

- 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

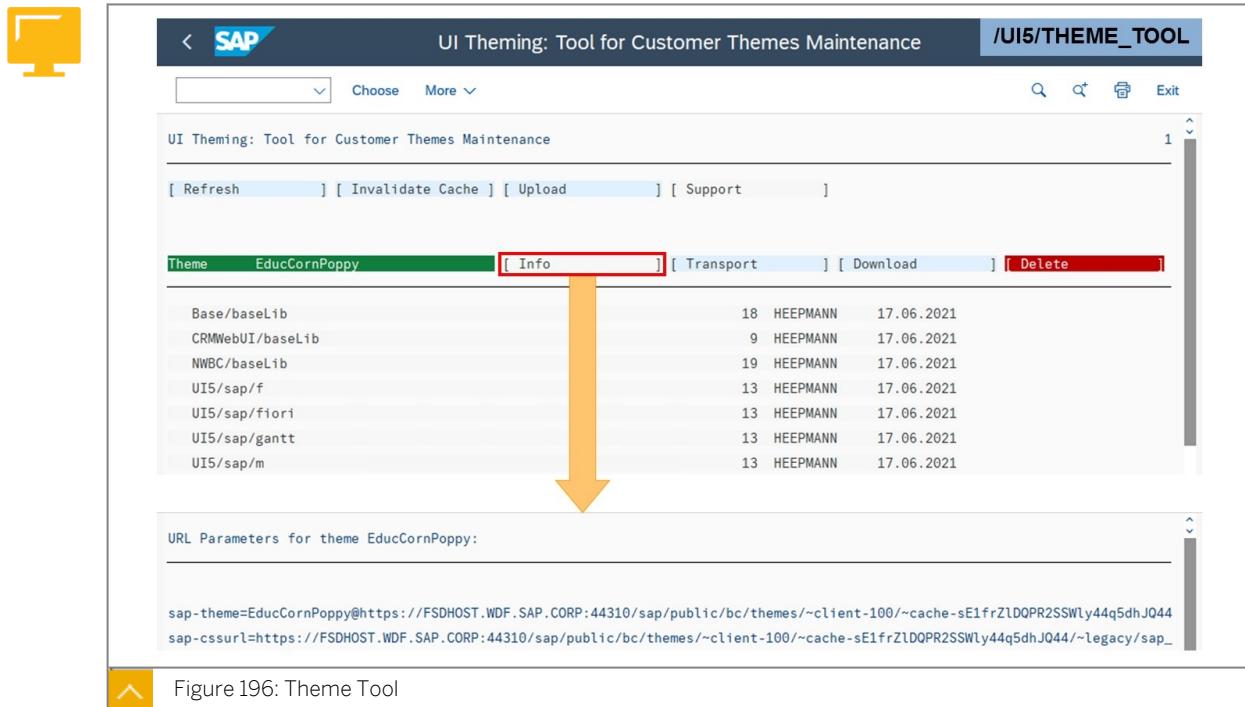


Figure 196: Theme Tool

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. This is also the easiest way to get a URI pointing to a certain theme. Start the tool using the following transaction: /UI5/THEME_TOOL.

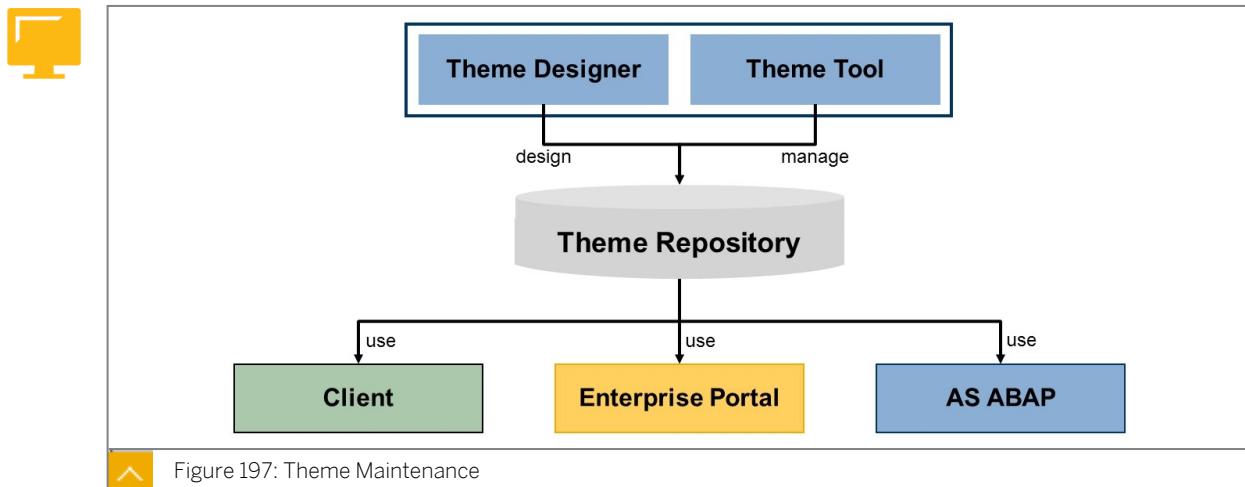


Figure 197: Theme Maintenance

By adding the URI parameter `sap-theme=<theme-id>@<theme-root>`, which points 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 Application Server (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 screenshot illustrates the configuration of the SAP Fiori launchpad theme. It shows the SAP Fiori - Initial Setup for Fiori Applications S/4 interface with the task list SAP_FIORI_FOUNDATION_S4. A specific task, "Set Fiori 3 as default theme (/UI2/NWBC_CFG_SAP)", is highlighted. An arrow points from this task to the NWBC Configuration: Maintenance (SAP) screen, which displays a parameter named THEME with the value sap_fiori_3. To the right, the FLP Configuration screen shows the parameter THEMING_DEFAULT_THEME set to sap_fiori_3 and the parameter THEMING_HIDE_SAP_THEMES set to false. This configuration is noted to be available since SAP UI 7.55.

The central parameter for the *SAP Fiori launchpad* can be set in transaction `/UI2/NWBC_CFG_CUST` per client or in transaction `/UI2/NWBC_CFG_SAP` client-independent. The task list `SAP_FIORI_FOUNDATION_S4` activates the theme SAP Quartz Light or any specified one client-independent. The table entry consists of the following:

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

Theme	Parameter Value
SAP Blue Crystal	sap_bluecrystal
SAP Belize	sap_belize
SAP Quartz Light	sap_fiori_3
SAP Quartz Dark	sap_fiori_3_dark

Since SAP S/4HANA 2020, the parameter `THEMING_DEFAULT_THEME` is available to set the theme in `/UI2/FLP_CUS_CONF`. In addition the parameter `THEMING_HIDE_SAP_THEMES` hides all themes provided by SAP only showing customer-specific ones in the FLP-settings.



LESSON SUMMARY

You should now be able to:

- Use the UI Theme Designer

Unit 6

Lesson 2

Using SAP Fiori Runtime Authoring



LESSON OBJECTIVES

After completing this lesson, you will be able to:

- Use SAP Fiori runtime authoring

Adaptation at Runtime



The screenshot displays two side-by-side Fiori applications, both titled "SAPUI5 Demo App". The left application is for "Manufacturing 1 (DE)" and the right is for "Notebook Basic 15 HT-1000". Both screens show a "General Information" section with various fields like Controlling Area, Cost Center, Name, Description, and Valid From. Above these sections, there are "UI Adaptation" and "Navigation" tabs. A red box highlights a context menu that appears when clicking on the "Control" button in the top navigation bar. This menu includes options: "Rename", "Add: Section", "Remove", "Cut", "Paste", and "Embed Content to: Section". The right application also shows a similar context menu for the "Contact Person" section.

Figure 199: 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 *SAP 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>

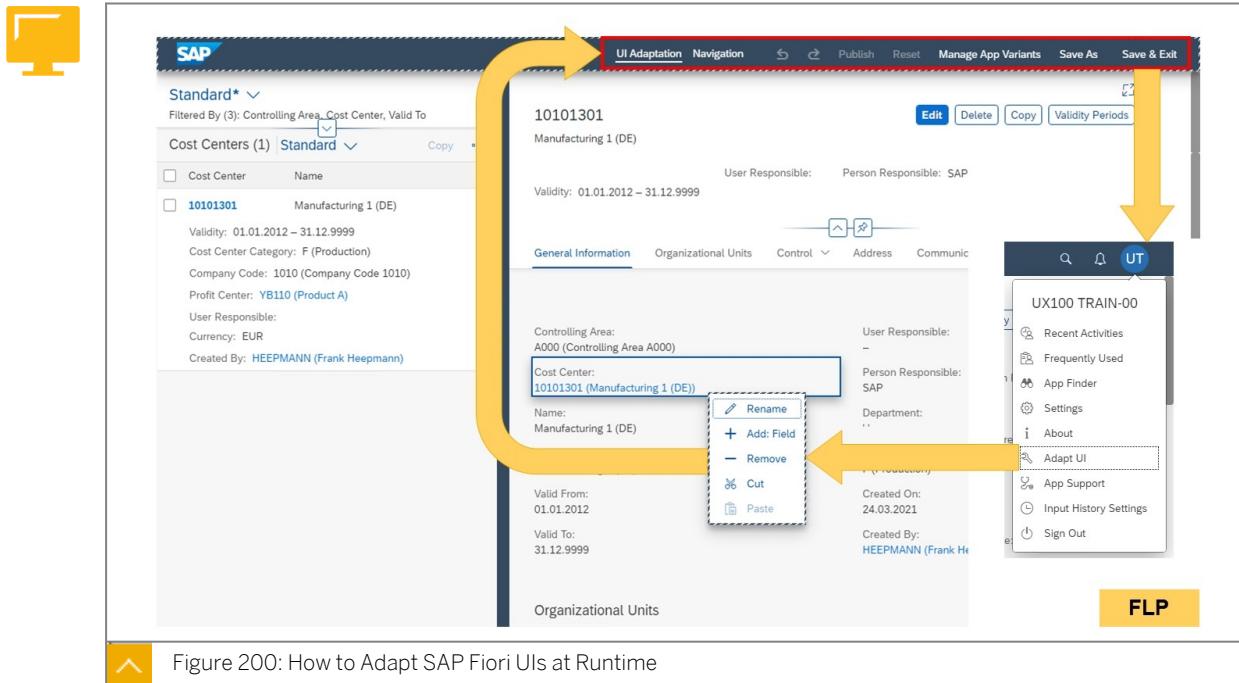


Figure 200: How to Adapt SAP Fiori UIs at Runtime

The UI adaptation mode can be accessed via the *Adapt UI* button in the *User Actions Menu* 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.

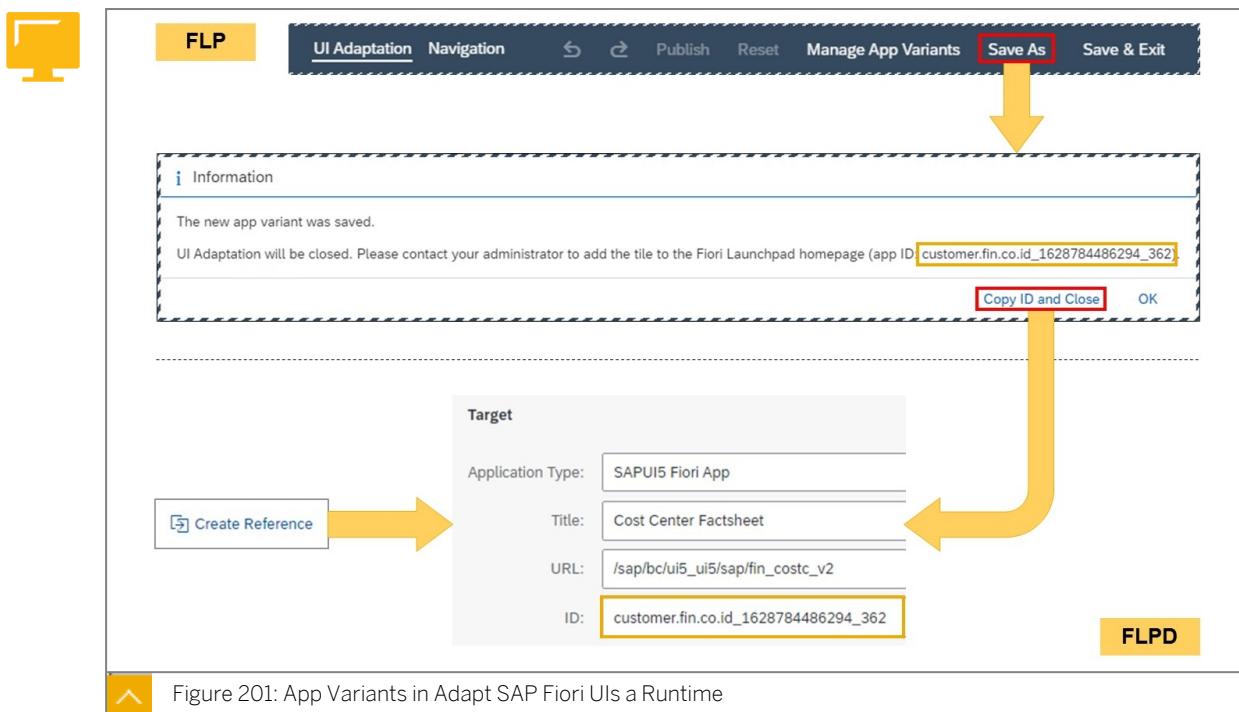


Figure 201: App Variants in Adapt SAP Fiori UIs at Runtime

By choosing *Save As*, the changes can be saved as an app variant. You can 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.



Note:

More information about this topic can be found in S4D425 (Extensibility for SAP S/4HANA):

<https://training.sap.com/course/s4d425>

Runtime Authoring Plugin



Catalog: /UIF/SAP_RTA_PLUGIN

Intent

Semantic Object: Shell

Action: plugin

Target

Application Type: SAPUI5 Fiori App

Title: Runtime Authoring Plug-in

URL:

ID: sap.ushell.plugins.rta

General

Information:

Device Types: Desktop Tablet Phone

Name	Mandatory	Value	Is Regular Expression	Default Value	Target N...
Enter a name					

Allow additional parameters

Add Delete



Figure 202: SAP Runtime Authoring Plugin

To adapt SAP Fiori apps at runtime, the user needs a target mapping to the runtime authoring plugin. No additional settings are necessary 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* user role with a preconfigured target mapping.



LESSON SUMMARY

You should now be able to:

- Use SAP Fiori runtime authoring

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

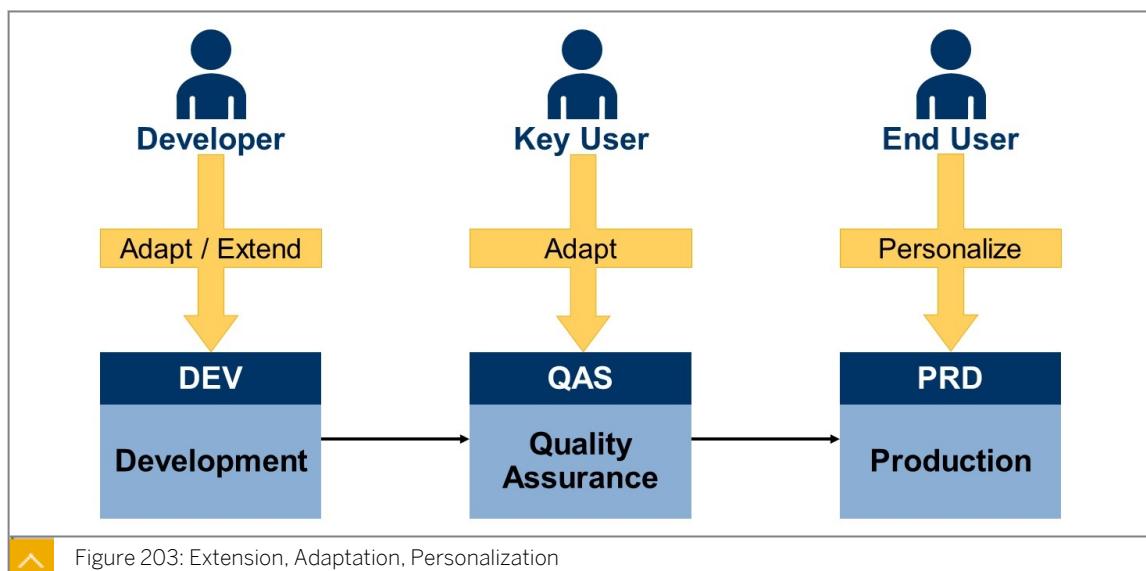
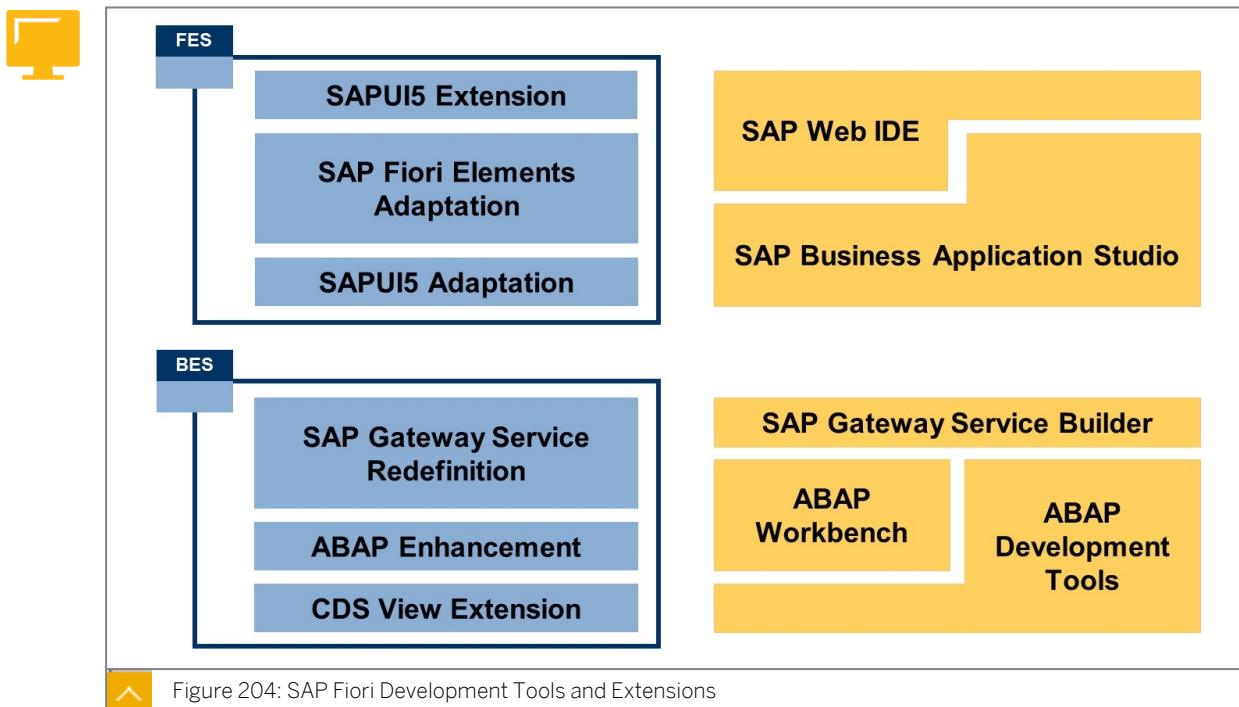


Figure 203: Extension, Adaptation, Personalization

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 personalizes the *SAP Fiori launchpad* with a limited tool set. The changes are only valid for her or himself.
- The key user adapts the *SAP Fiori launchpad* by switching to the adaptation mode. 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 adapts or extends source code with 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 → *Adaptation/Extension*
- SAP Fiori Elements → *Adaptation*

SAP Web IDE is the only tool for extending SAPUI5 the old way. SAP Fiori elements can also be adapted with SAP Web IDE but with restrictions. The SAP Business Application Studio is the main tool for adapting SAPUI5 and SAP Fiori elements.

Technologies used in SAP Fiori in the back-end server providing the business 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

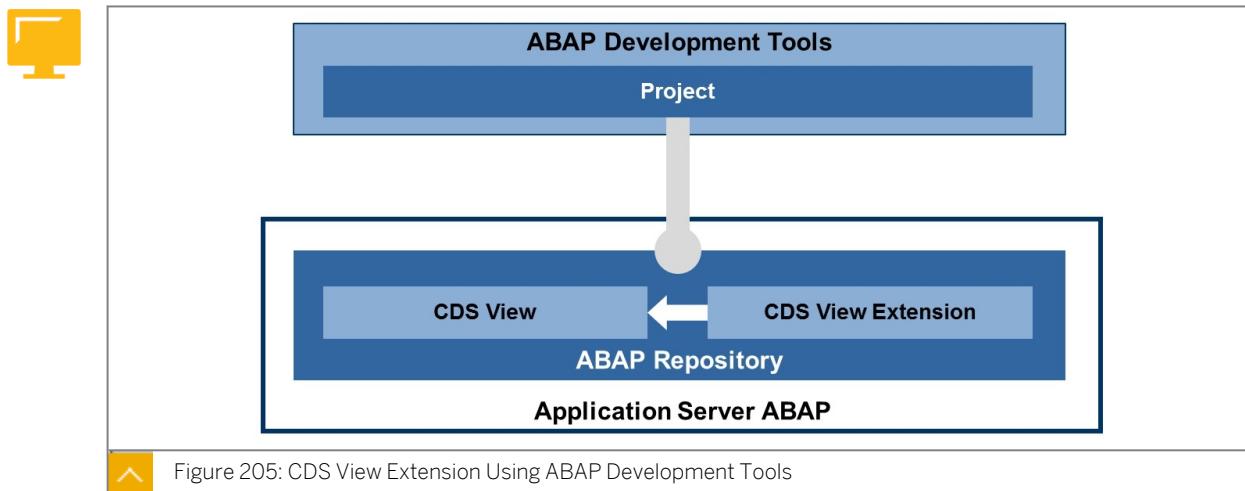


Figure 205: CDS View Extension Using ABAP Development Tools

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 must be created in the customer namespace referencing the original CDS view.

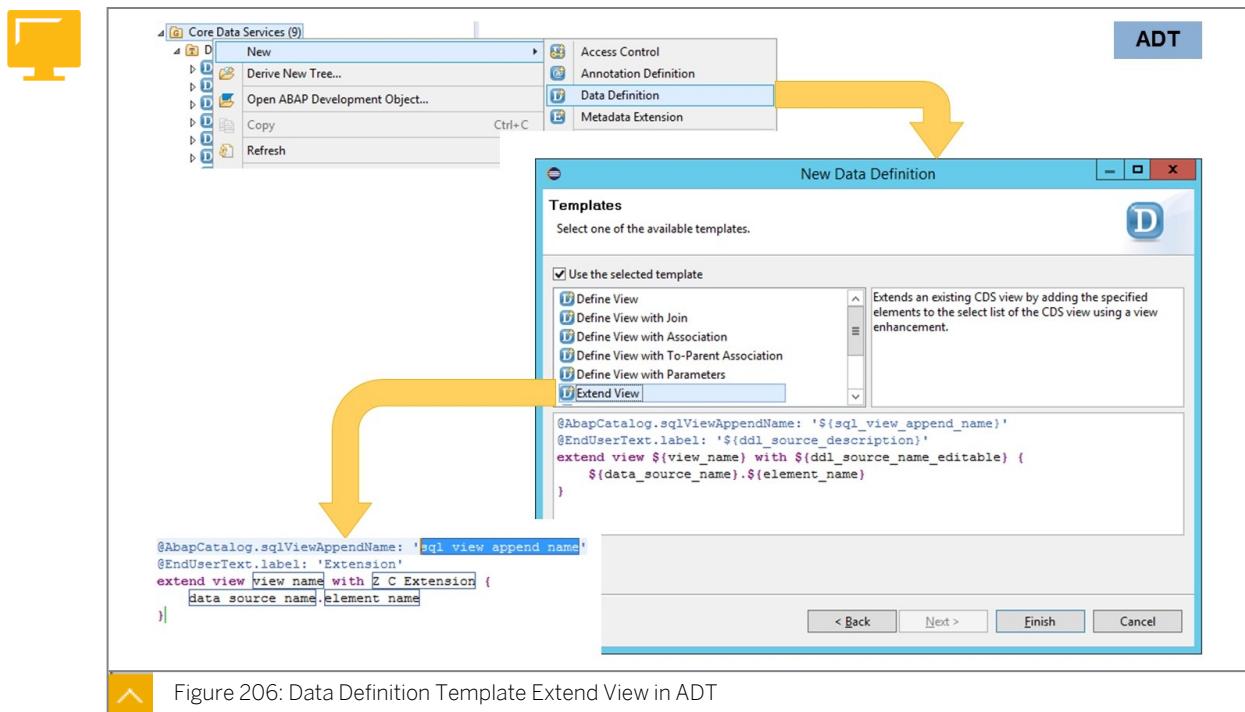


Figure 206: Data Definition Template Extend View in ADT

There is not a special button in ADT to create an extension but there is 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

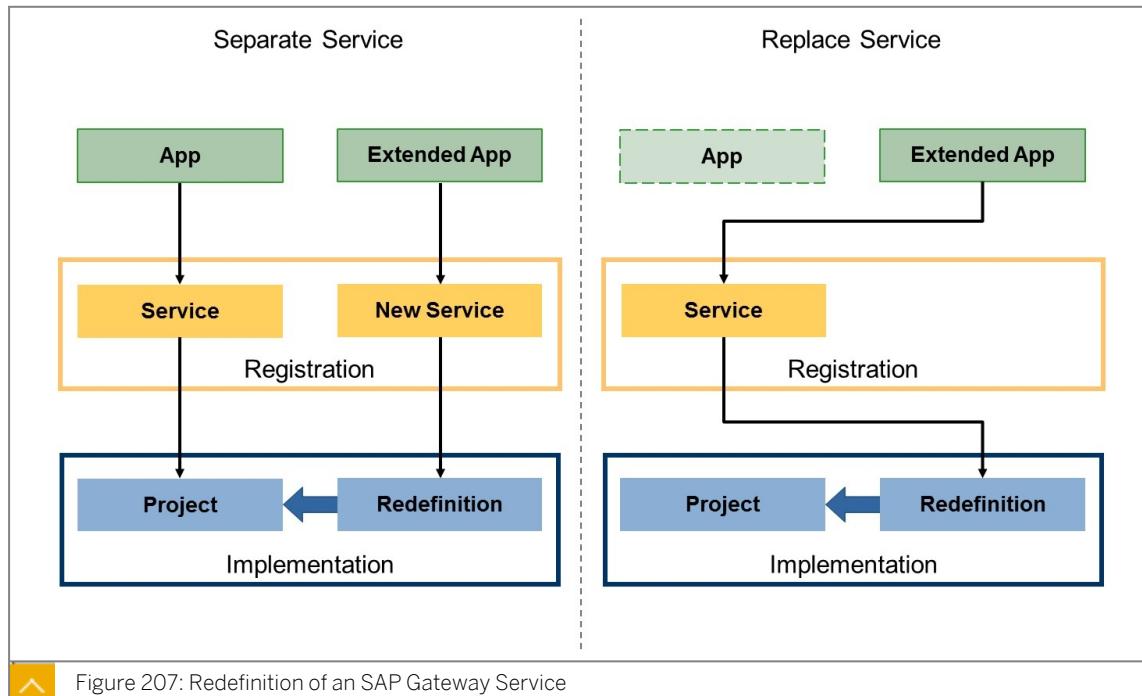


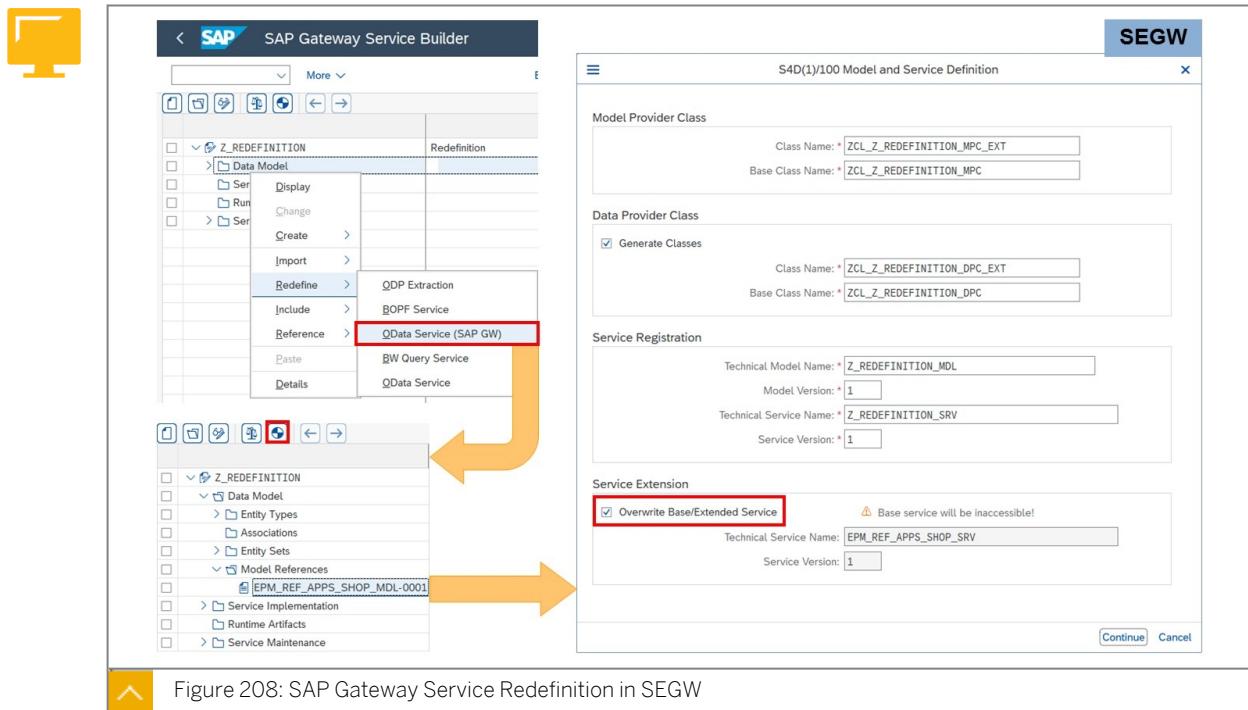
Figure 207: Redefinition of an SAP Gateway Service

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:

- Register a new service and preserve the original one.
- Replace the original service.

The first option enables 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)*, you can 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>

SAPUI5 Adaptation

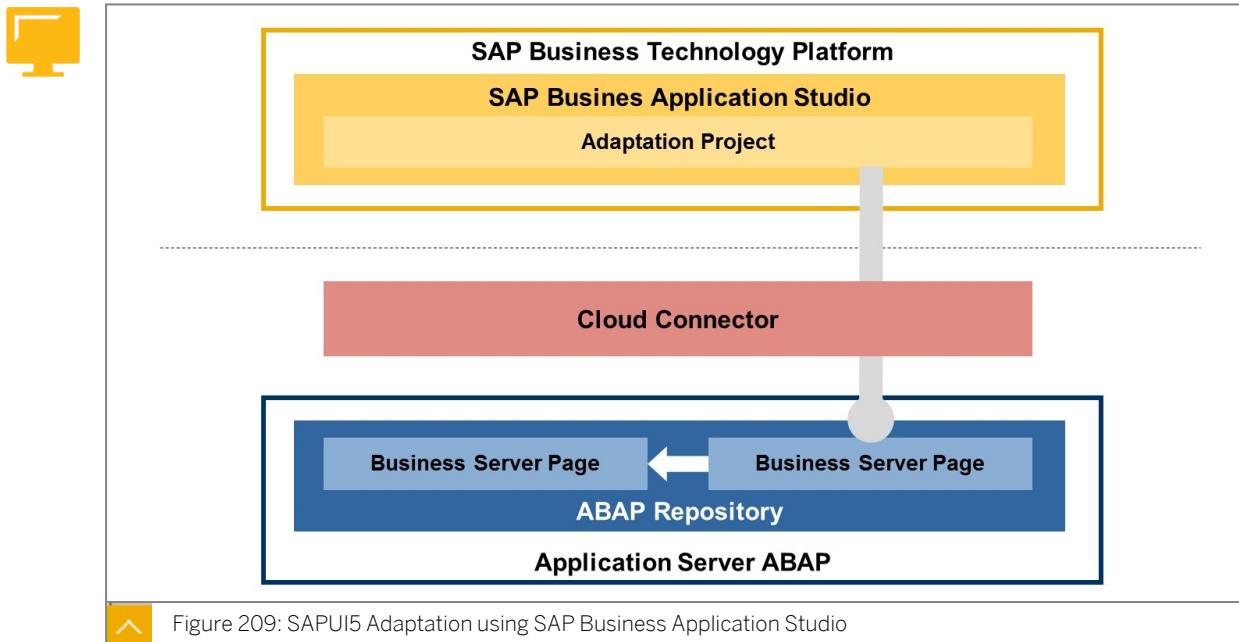


Figure 209: SAPUI5 Adaptation using SAP Business Application Studio

SAP Business Application Studio (BAS) offers a complete tool-set for extending SAP Fiori elements or SAPUI5 freestyle apps. Extensions are made via SAPUI5 flexibility and do not modify the source. This modification-free approach creates app variants, which allows to reference the source application and its artifacts instead of modifying the source artifacts themselves.



Note:

Adaptation projects are supported starting from SAPUI5 version 1.71.

The process involves creating an adaptation project in the BAS connecting to the original business server page (BSP), programming the differences, and uploading the adapted app as a new BSP in the ABAP repository. When calling the adapted 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 adapted app.



Caution:

The *Cloud Connector* is mandatory when connecting on-premise systems to BAS.

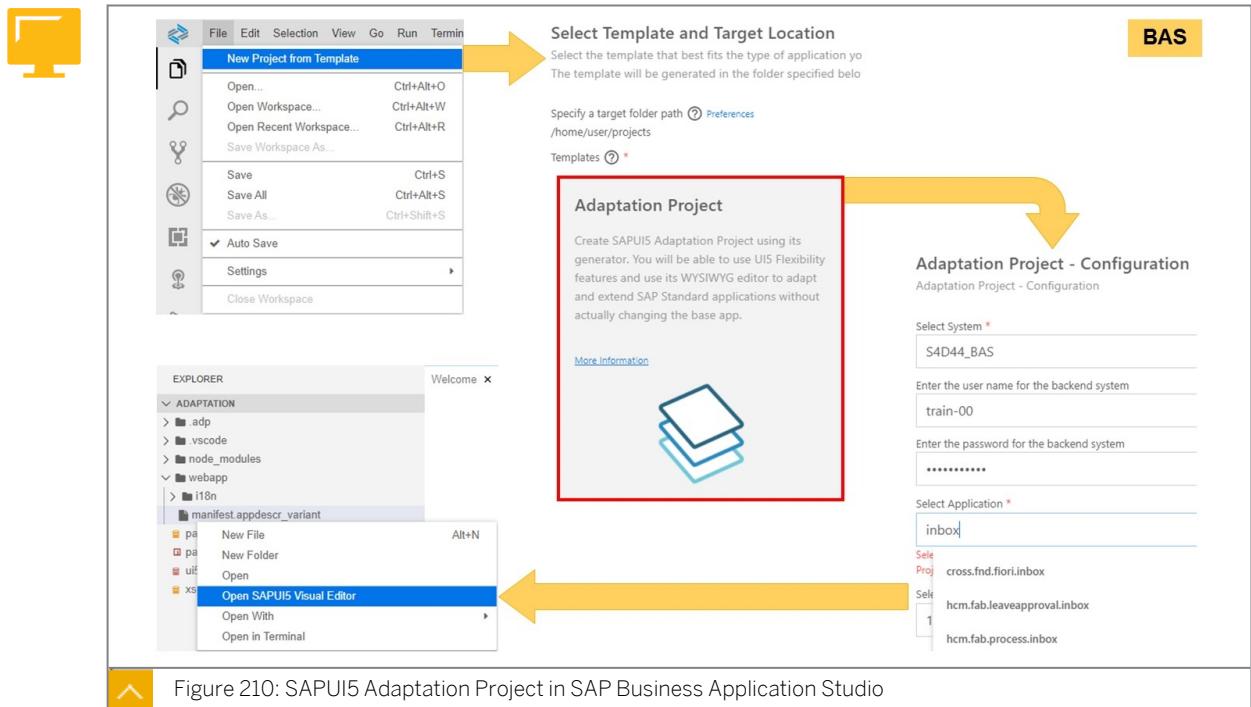


Figure 210: SAPUI5 Adaptation Project in SAP Business Application Studio

Creating an adaptation project is done through the corresponding template referencing the original app in a source system. 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 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.

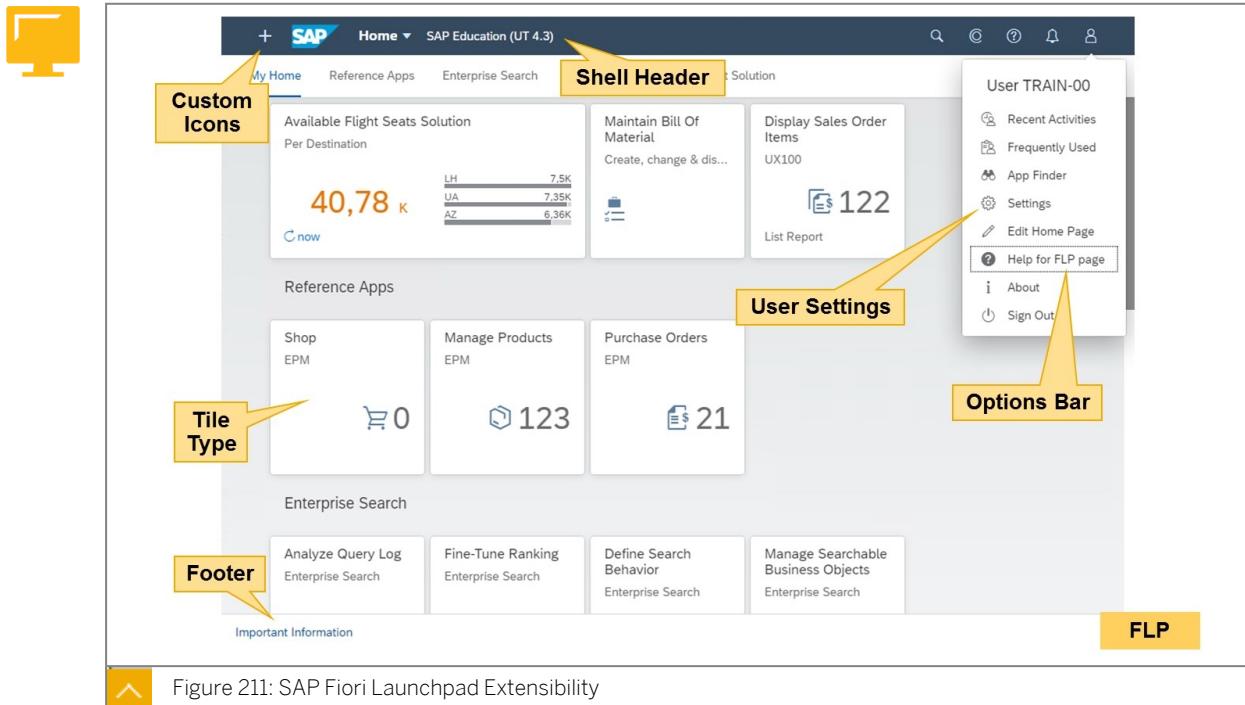


Note:

More information about this topic can be found in UX402 (Advanced SAPUI5 Development):

<https://training.sap.com/course/ux402>

SAP Fiori Launchpad Plugins



The SAP Fiori launchpad (FLP) can be extended in its functionality through 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

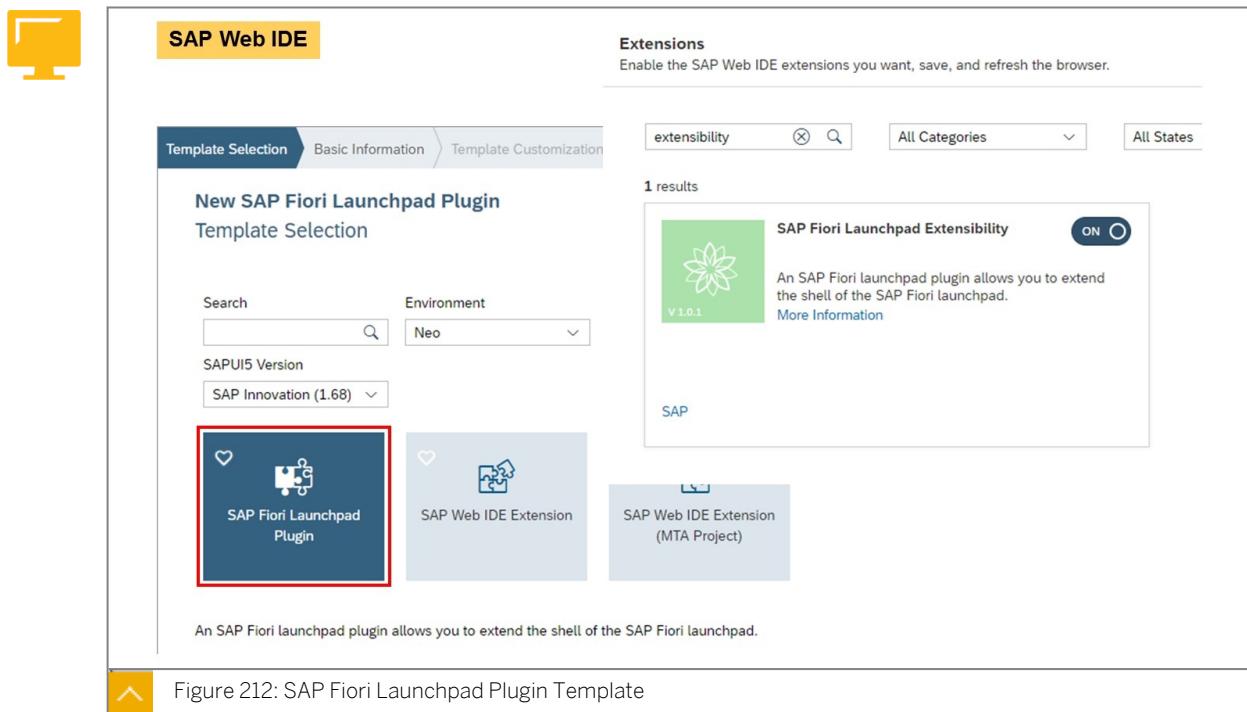


Figure 212: SAP Fiori Launchpad Plugin Template

The SAP Web IDE Full-Stack in the SAP Business Technology Platform (BTP) 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.

Generally, there are no views in a plugin. 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();

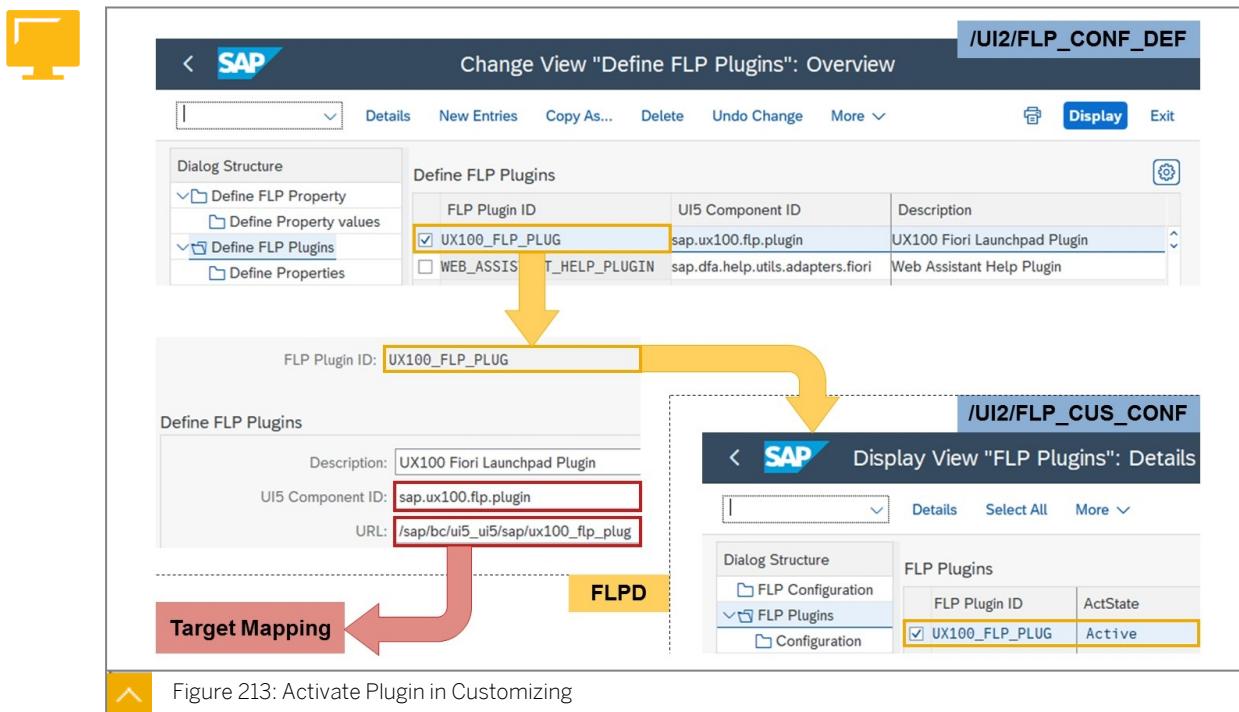
    /**
     * Set header title
     */
    rendererPromise.then(function (oRenderer) {
        oRenderer.setHeaderTitle("SAP Education (UT 4.3)");
    });
},
...

```



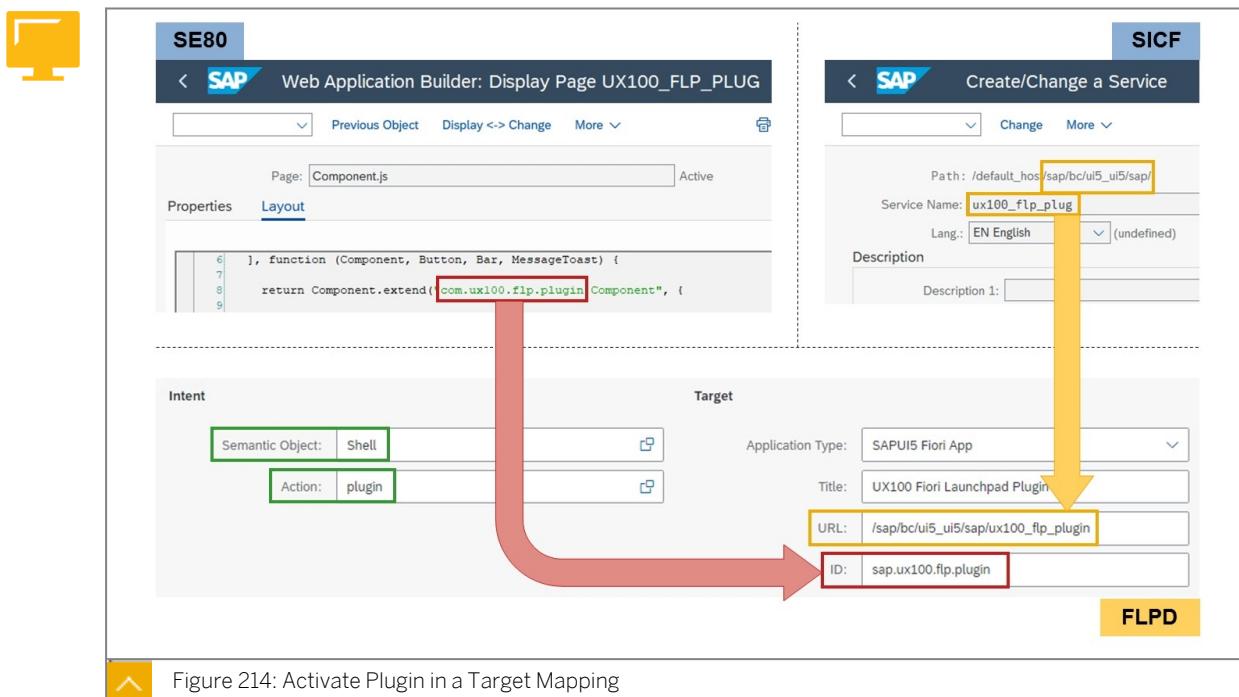
Note:

Plugins are loaded and instantiated asynchronously. When multiple plugins are loaded, do not assume any specific instantiation order of the components.



Every plugin must 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) similar to 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 their own branch in the structure of the transaction. This enables the user to add additional configuration to the activation, which can be used in the plugin during runtime.



Like other FLP configurations, you can also 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 a semantic object and **plugin** as an action.



LESSON SUMMARY

You should now be able to:

- Extend SAP Fiori applications
- Extend SAP Fiori launchpad

Learning Assessment

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 Business Technology Platform
- D SAP NetWeaver Application Server Java
- E SAP Business Intelligence

2. Which design template is used for SAP Fiori 3?

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

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

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

Choose the correct answers.

- A Target mapping to runtime authoring plugin
- B SAP Fiori catalog SAP_CA_BC_SSB
- C User role SAP_UI_FLEX_KEY_USER
- D SAP Fiori catalog /UIF/SAP_RTA_PLUGIN
- E User role SAP_S RFCACL

6. 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 Adaptation/Extension
- D ABAP Enhancement
- E SAP Fiori Elements Adaptation

7. Which tool must be used to extend CDS Views?

8. What is used to extend the SAP Fiori launchpad?

9. How can you activate an extension for the SAP Fiori launchpad (FLP)?

Choose the correct answers.

- A FLP Content Manager (/UI2/FLPCM_CUST)
- B Target Mapping (/UI2/FLPD_CUST)
- C Customer settings (/UI2/FLP_CUS_CONF)
- D Definition of FLP settings (/UI2/FLP_CONF_DEF)

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 Business Technology 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 Business Technology Platform.

2. Which design template is used for SAP Fiori 3?

SAP Quartz

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

Theme repository in front-end server

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

Start app → User Action Menu → Adapt UI

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

Choose the correct answers.

- A Target mapping to runtime authoring plugin
- B SAP Fiori catalog SAP_CA_BC_SSB
- C User role SAP_UI_FLEX_KEY_USER
- D SAP 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 SAP Fiori catalog /UIF/SAP_RTA_PLUGIN.

6. 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 Adaptation/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 Adaptation/Extension and SAP Fiori Elements Adaptation.

7. Which tool must be used to extend CDS Views?

ABAP Development Tools (ADT)

8. What is used to extend the SAP Fiori launchpad?

SAPUI5 plugins

9. How can you activate an extension for the SAP Fiori launchpad (FLP)?

Choose the correct answers.

- A FLP Content Manager (/UI2/FLPCM_CUST)
- B Target Mapping (/UI2/FLPD_CUST)
- C Customer settings (/UI2/FLP_CUS_CONF)
- D Definition of FLP settings (/UI2/FLP_CONF_DEF)

Correct. Extensions for the FLP can be activated via a target mapping or in the customer settings.

Lesson 1

Examining SAP Fiori Mobile

227

Lesson 2

Examining SAP Fiori for iOS

235

UNIT OBJECTIVES

- Examine SAP Fiori Mobile
- Examine SAP Fiori for iOS

Unit 7

Lesson 1

Examining SAP Fiori Mobile



LESSON OBJECTIVES

After completing this lesson, you will be able to:

- Examine SAP Fiori Mobile

Consumption Options

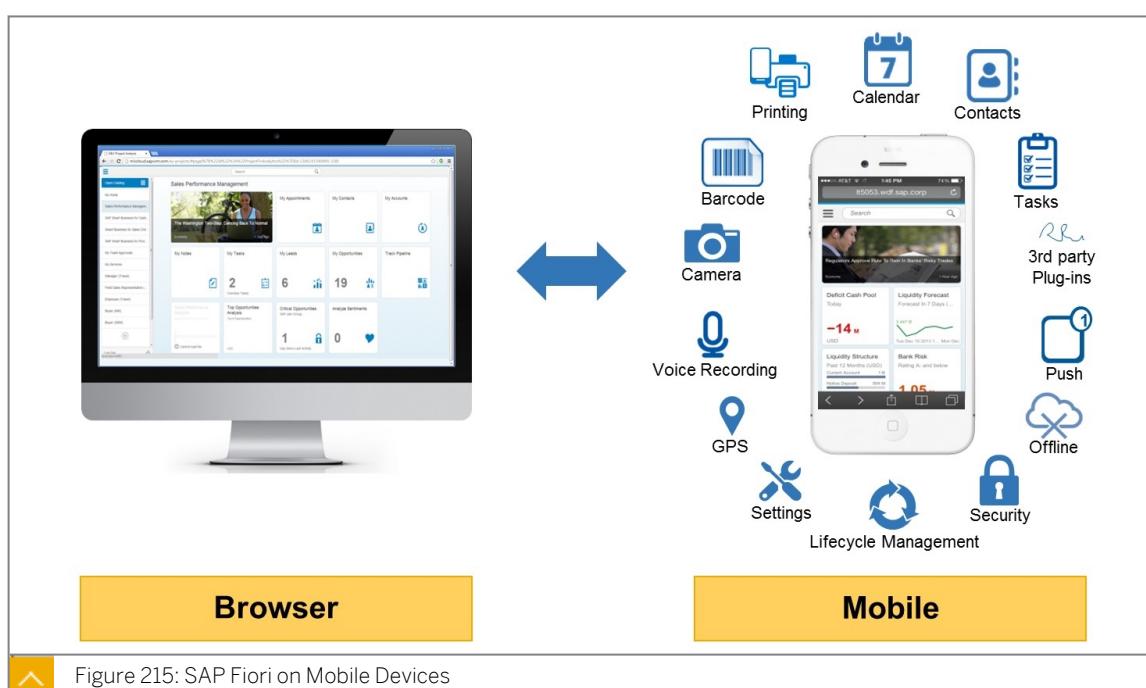


Figure 215: SAP Fiori on Mobile Devices

Using SAP Fiori in a mobile environment involves more than just opening the *SAP Fiori launchpad* in the browser. A mobile device has many 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 such as lifecycle management, security, or app settings can get complicated, and even dangerous, if they are not handled properly.

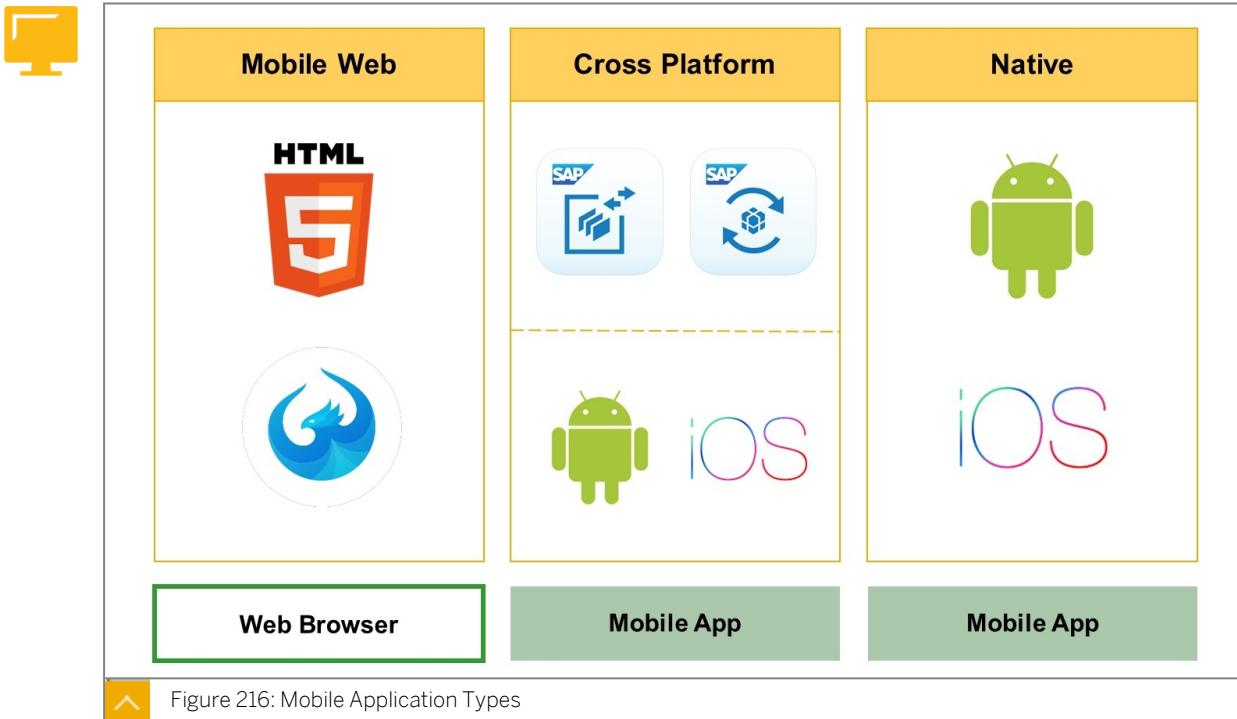


Figure 216: Mobile Application Types

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

Cross platform apps are apps created by “low-level” code. The application definition is represented using metadata or scripts instead of source code. The “low-level” code is dynamically interpreted on the client or by the server to render a native user interface (UI). The code is created using special tools provided by a development framework.

SAP offers two types of cross platform development approaches depending on the use case:

- SAP Mobile Cards is a script-based application development platform for card UIs.
- Mobile Development Kit (MDK) is a metadata-based application development platform.

Lastly, native apps are apps developed for a certain runtime environment using its native development environment. They can only run in this one environment, 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.

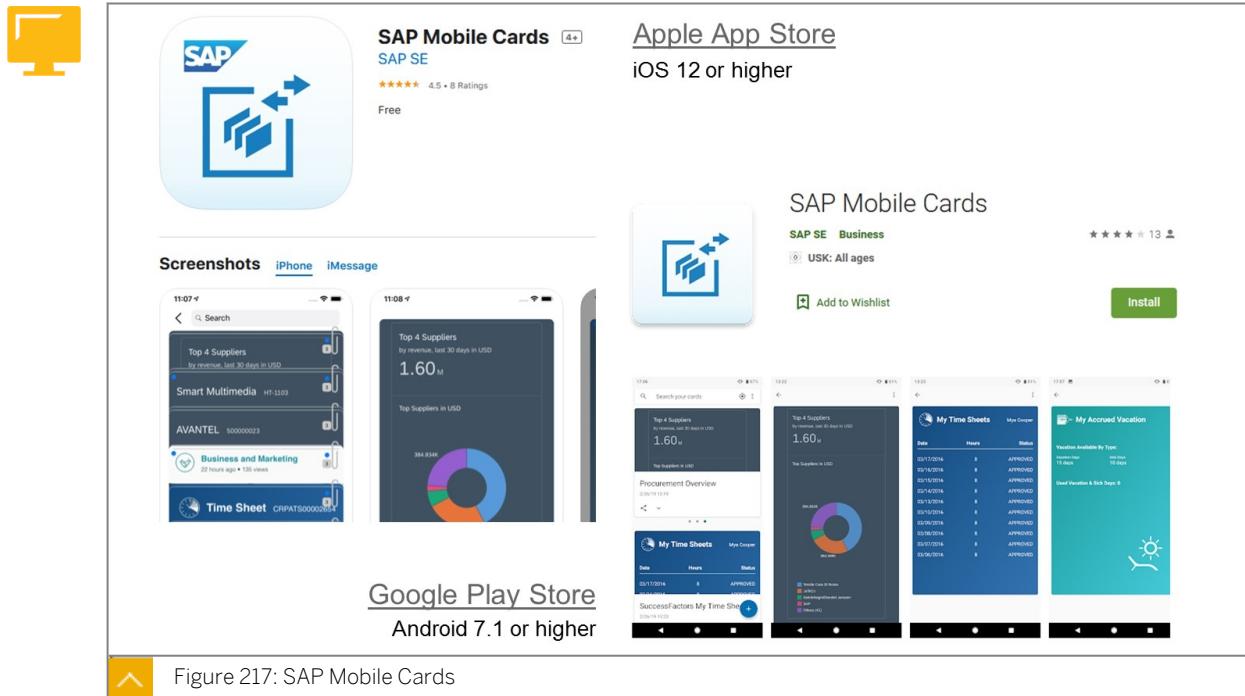


Figure 217: SAP Mobile Cards

SAP Mobile Cards is a feature within SAP Mobile Services. It provides customers access to a micro app platform to publish data into a consumer grade “wallet” or “passbook” style app. Using SAP Mobile Cards, companies can create simple, yet highly valuable quick-win apps.

The SAP Mobile Cards app is available for Apple iOS and Google Android. It provides the frame for very simple apps and lean processes such as approvals or bits of valuable data.

Apple App Store:

<https://apps.apple.com/us/app/sap-mobile-cards/id1168110623>

Google Play Store:

<https://play.google.com/store/apps/details?id=com.sap.content2go>

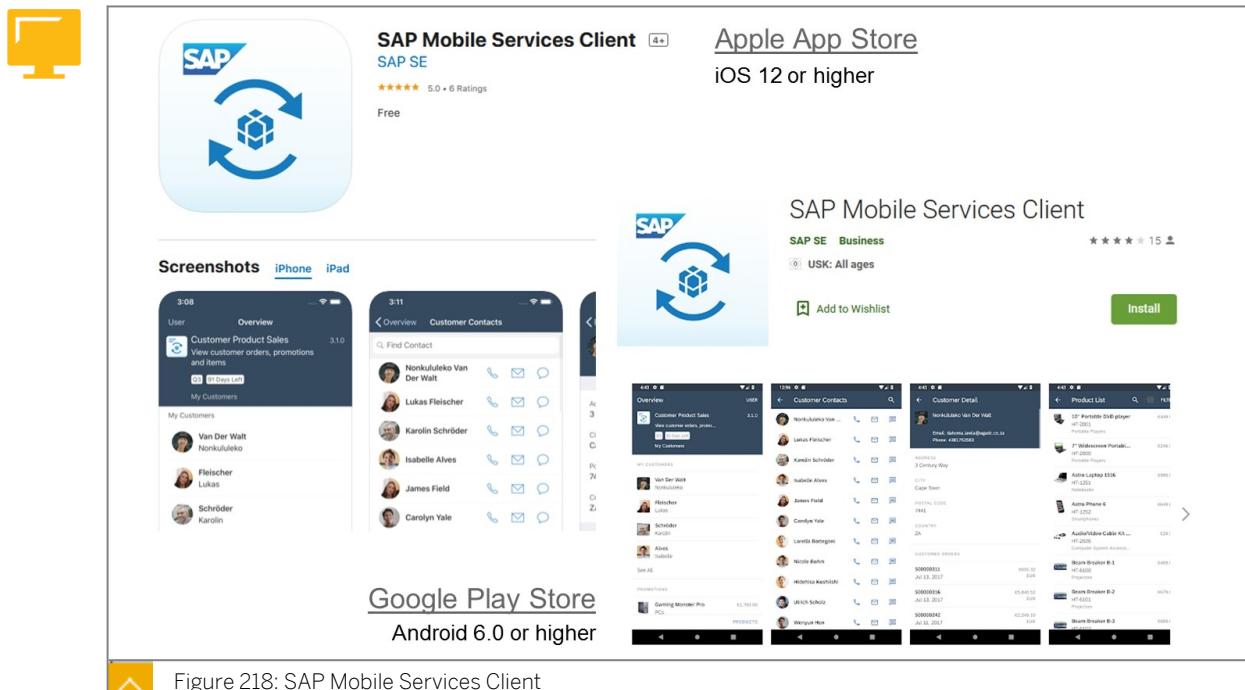


Figure 218: SAP Mobile Services Client

Mobile Development Toolkit (MDK) enables developers to quickly build mobile apps with a native look and feel in a visual and declarative way. The SAP Mobile Services provides the runtime whereas the *SAP Business Application Studio (BAS)* or *SAP Web IDE* provides the development environment.

The generic client for running MDK apps on Apple iOS and Google Android is the *SAP Mobile Services Client*. It is the frame for running complex cross platform native apps such as field services or asset maintenance.

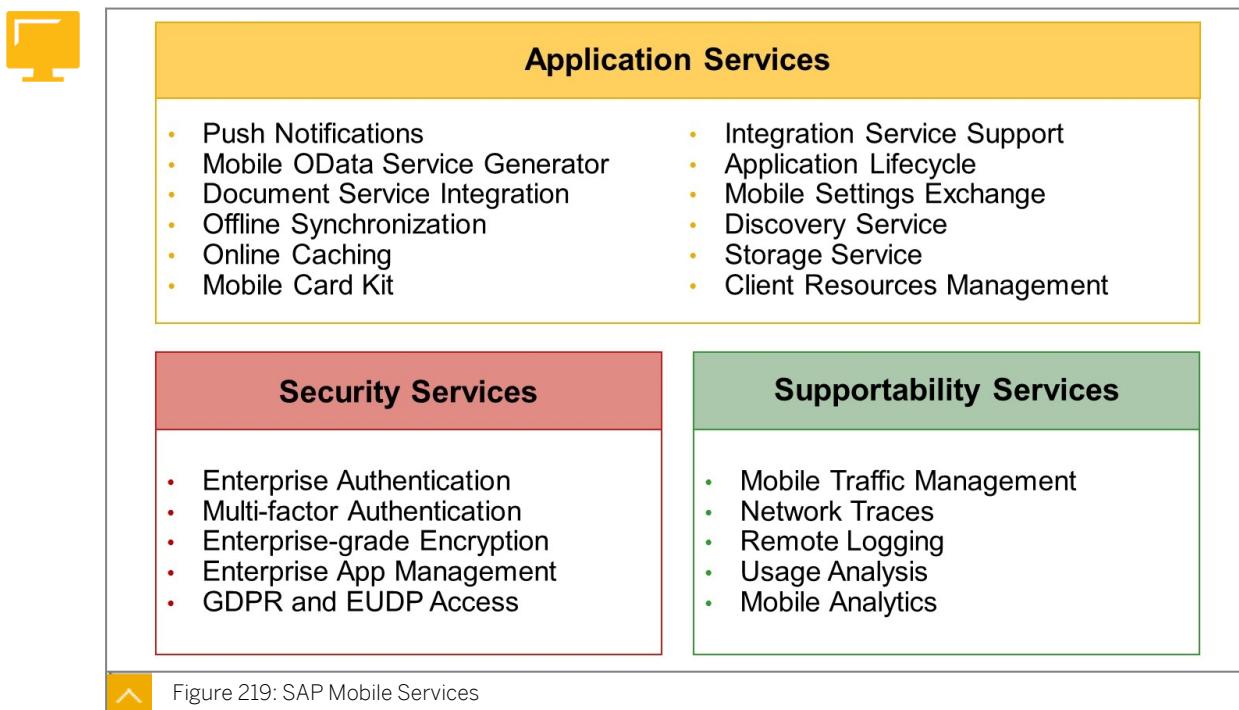
Apple App Store:

<https://apps.apple.com/us/app/sap-mobile-services-client/id1413653544>

Google Play Store:

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

SAP Mobile Services



The SAP Mobile Services for SAP Business Technology Platform (BTP) 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). Developing and providing apps is a major part.

The service areas of SAP 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 can directly connect to an *SAP Fiori launchpad*. All other application services come in handy for certain apps or scenarios.

Security

In the security area, the SAP Fiori mobile apps mainly benefit 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 SAP Fiori mobile apps. The analytical features are useful on a more generic level.



Note:

The SAP Mobile Platform (SMP) was the on-premise solution of SAP for a Mobile Application Development Platform (MADP) and the precursor of the SAP Mobile Services. The SMP is no longer available and maintenance ended in 2020.

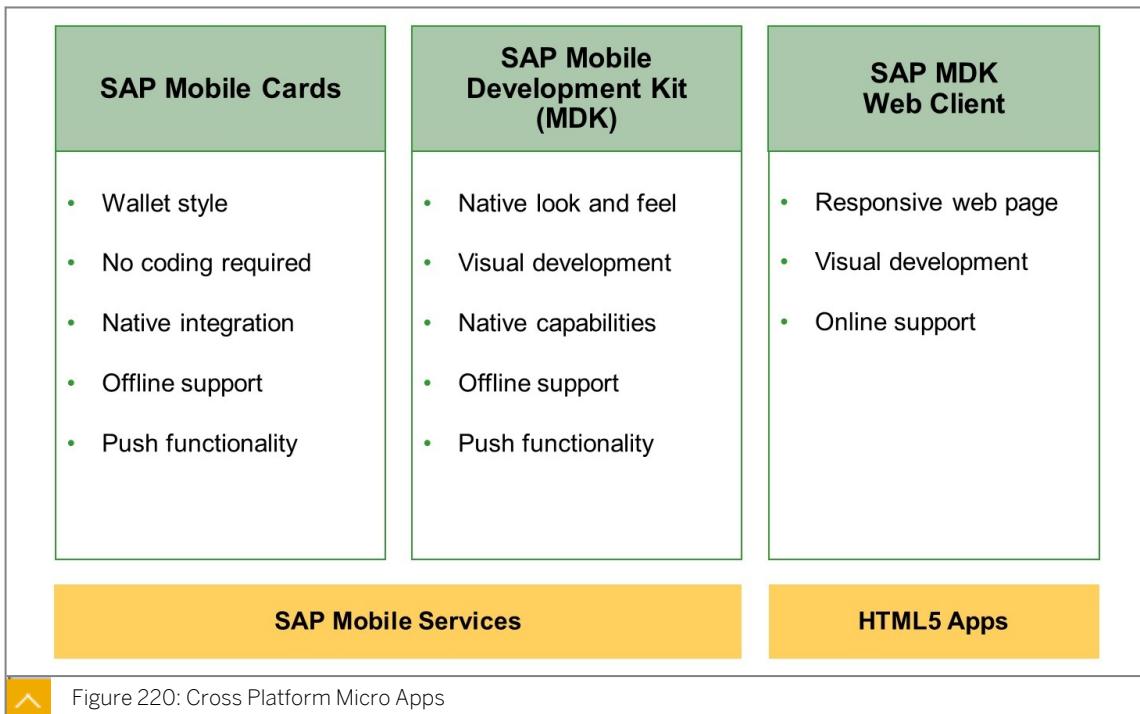


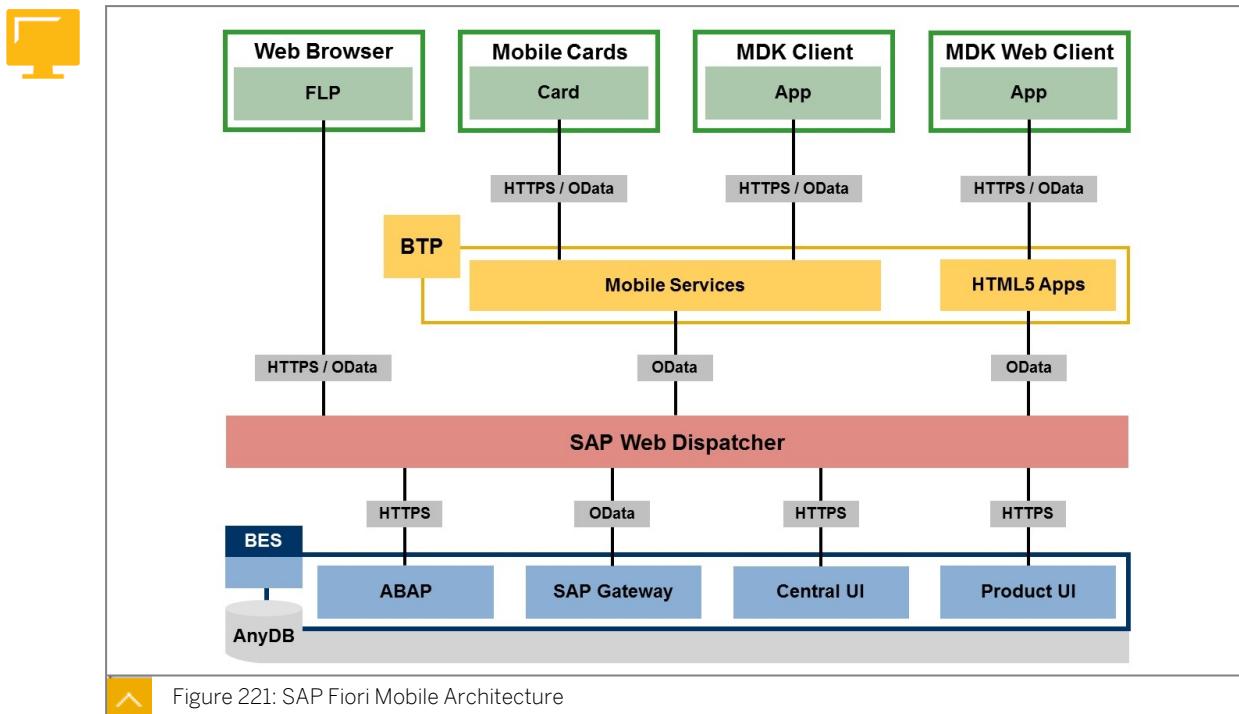
Figure 220: Cross Platform Micro Apps

SAP Mobile Cards provide a simple “wallet” or “passbook” style user experience (UX). No coding is required because the content is directly deployed from SAP Fiori elements apps. Other sources may need some low scripting code such as SAP Ariba, SAP Fieldglass, or SAP SuccessFactors. SAP Mobile Cards offer several native integrations such as maps, location, email, SMS, voice calling, and smart watches. It provides full offline and push notification support.

SAP Mobile Development Kit (MDK) provides fast and reliable native look and feel apps. The visual development approach simplifies the development and lifecycle management following the guidelines of SAP Fiori. The MDK offers access to all native capabilities of the mobile device. It provides full offline and push notification support.

The SAP MDK Web Client expands MDK capabilities to desktop complementing existing MDK mobile clients. You can run the same metadata across mobile and desktop platforms. MDK web apps are deployed as HTML5-apps in the SAP BTP and can be accessed in web browsers.

Due to the nature of web apps, it supports only online use case for OData connections.
SAP MDK Web Client is planned to release in October 2020.



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

- The web browser on the mobile device accesses the *SAP Fiori launchpad (FLP)* via the SAP Web Dispatcher such as a desktop browser.
- The SAP Mobile Cards access the SAP Mobile Services, which provide the UI of the card and forward the OData request to the BES.
- SAP MDK Client connects to SAP Mobile Services for initial on-boarding, receives the metadata, interprets it to create the UI using native SDK controls and forwards the OData request to the BES.
- SAP MDK Web Client allows you to run MDK metadata in a browser intended for desktop consumption. The deployed metadata is published as HTML5 app, which connects to the BES via SAP Connectivity service.

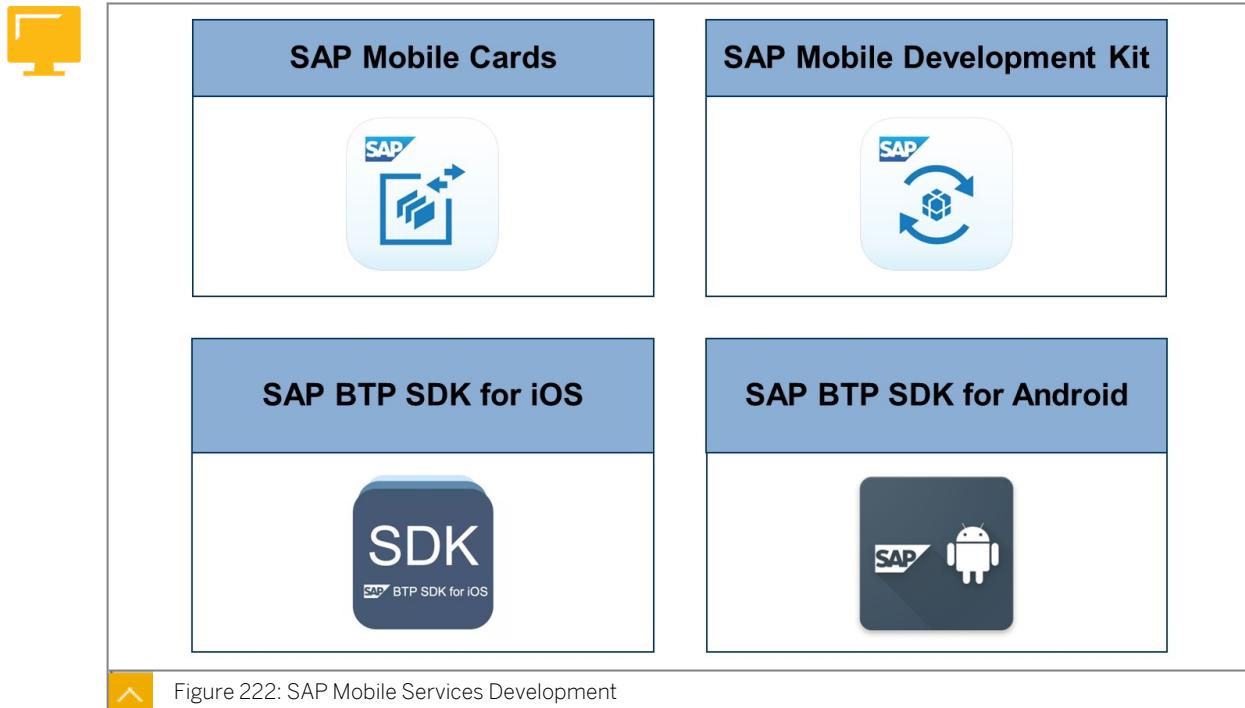


Figure 222: SAP Mobile Services Development

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

SAP Mobile Cards

The SAP Mobile Cards are consumer grade “wallet” or “passbook” style apps. These are very simple apps such as 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 Mobile Development Kit (MDK)

The MDK 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 Business Application Studio* and *SAP Web IDE*. It offers a code-free tool and a WYSIWYG editor, and app templates including support for offline, push, geo-location and more. SAP MDK Web Client allows you to run the same metadata in web browsers (for online use case).

SAP BTP SDKs for Android and iOS

The SAP BTP SDKs for Android and iOS 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 7

Lesson 2

Examining SAP Fiori for iOS

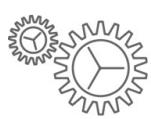


LESSON OBJECTIVES

After completing this lesson, you will be able to:

- Examine SAP Fiori for iOS

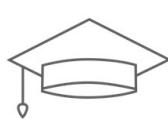
Native SAP Fiori Apps



Processes



SDK



Academy



Apps

Figure 223: 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 reimagined workflows



Note:

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

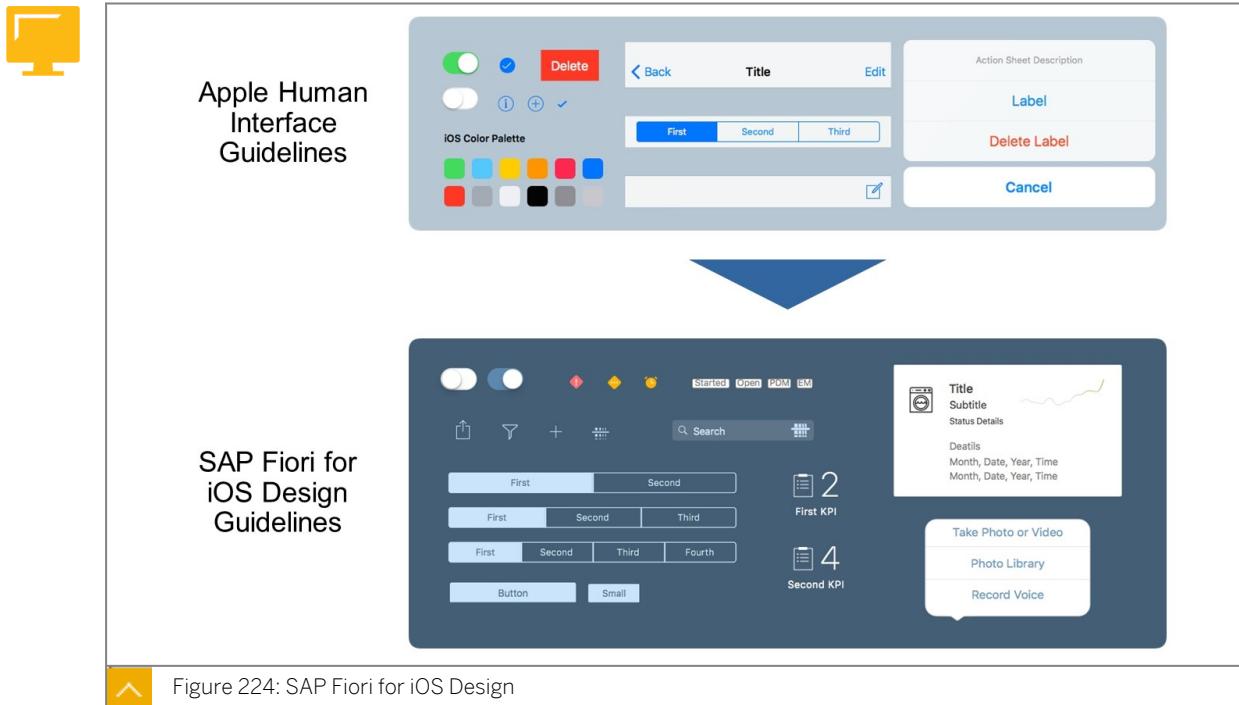


Figure 224: SAP Fiori for iOS Design

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, and 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/>.

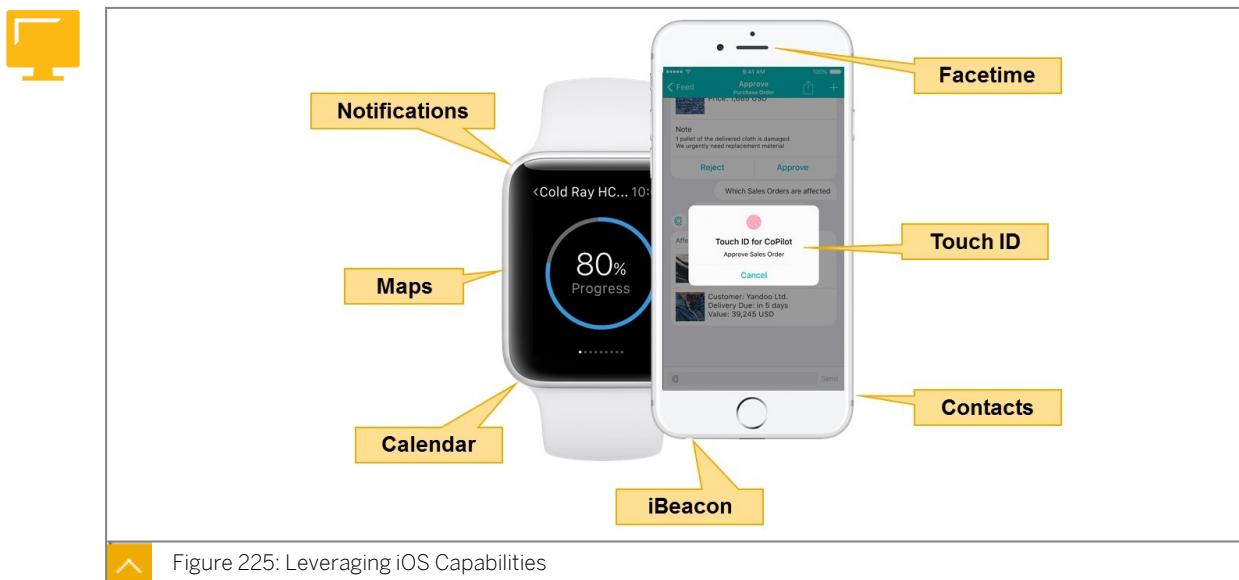


Figure 225: Leveraging iOS Capabilities

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 such as payments or approvals

iBeacon

Provides location-specific information such as 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

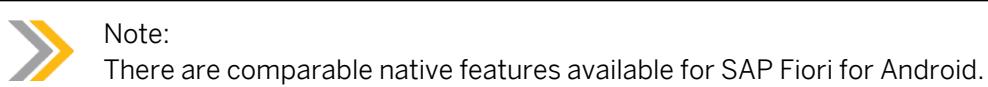
Enables ad-hoc communication, for example, between maintenance technicians and technical back-office experts

Calendar

Creates or changes meeting request directly in the app

Contacts

Displays linked contacts to data sets and start communication





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 actions and topics in your customer projects anywhere and anytime. This app allows consultants 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 your work.

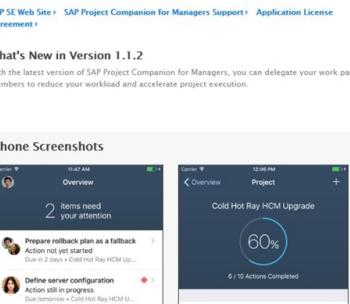
[SAP SE Web Site](#) • [SAP Project Companion for Consultants Support](#) • [Application License Agreement](#) • [More](#)

What's New in Version 1.1.2
With the latest version of SAP Project Companion for Consultants, consultants managers in work packages delegated to them. By taking responsibility for what they contribute to project management and execution.

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

[View More by This Developer](#)

iPhone Screenshots



iPhone Screenshots

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

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

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 workload and accelerate project execution.

Customer Ratings
This application hasn't received enough ratings to display a summary.

Compatibility: Requires iOS 10.0 or later. Compatible with iPhone, iPad, and iPod touch.

Customer Ratings
This application hasn't received enough ratings to display a summary.

Compatibility: Requires iOS 10.0 or later. Compatible with iPhone, iPad, and iPod touch.

Customer Ratings
This application hasn't received enough ratings to display a summary.

Apple App Store
iOS 10.x or higher

The first enterprise app for iOS is the *SAP Project Companion*, which is available in one flavor each for consultants and managers. It digitizes the processes that a project manager must 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 Business Technology Platform (BTP) and leverages Touch ID to access the mobile and security services in SAP BTP.

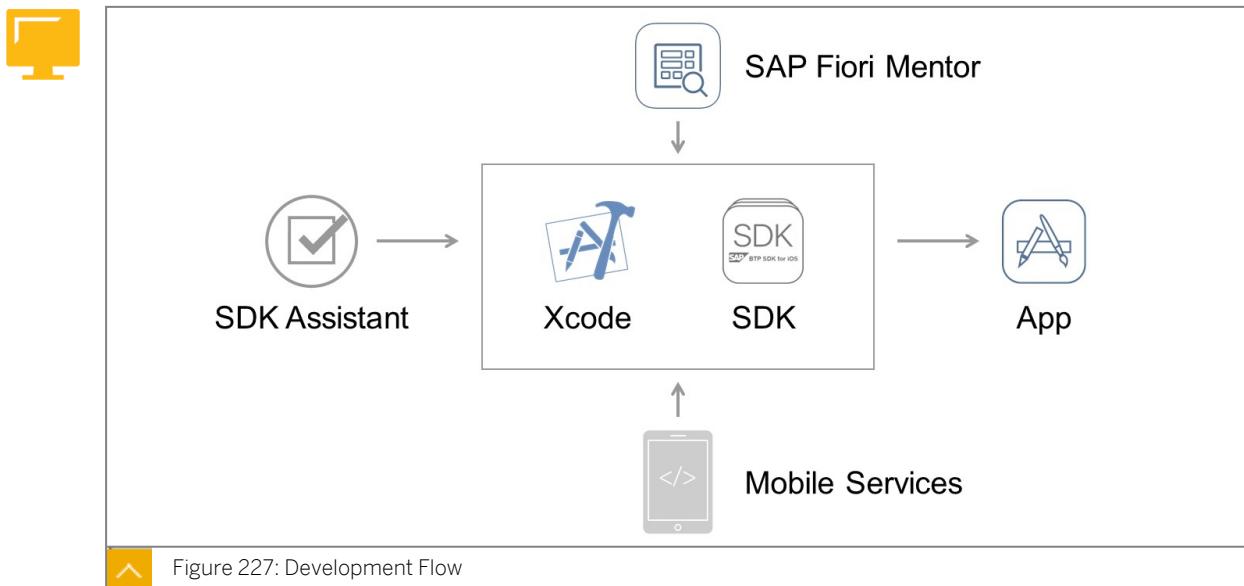
SAP Project Companion for Consultants in Apple App Store:

[https://itunes.apple.com/de/app/sap-project-companion-for-consultants/id1183388005?
mt=8](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](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 BTP SDK Assistant for iOS*, which generates projects and code to ease integration into the SAP BTP. These projects are used by Xcode and the SAP BTP 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 SAP 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 Mobile Services to accelerate developing analytics, push notifications, do online and offline syncing, and more.

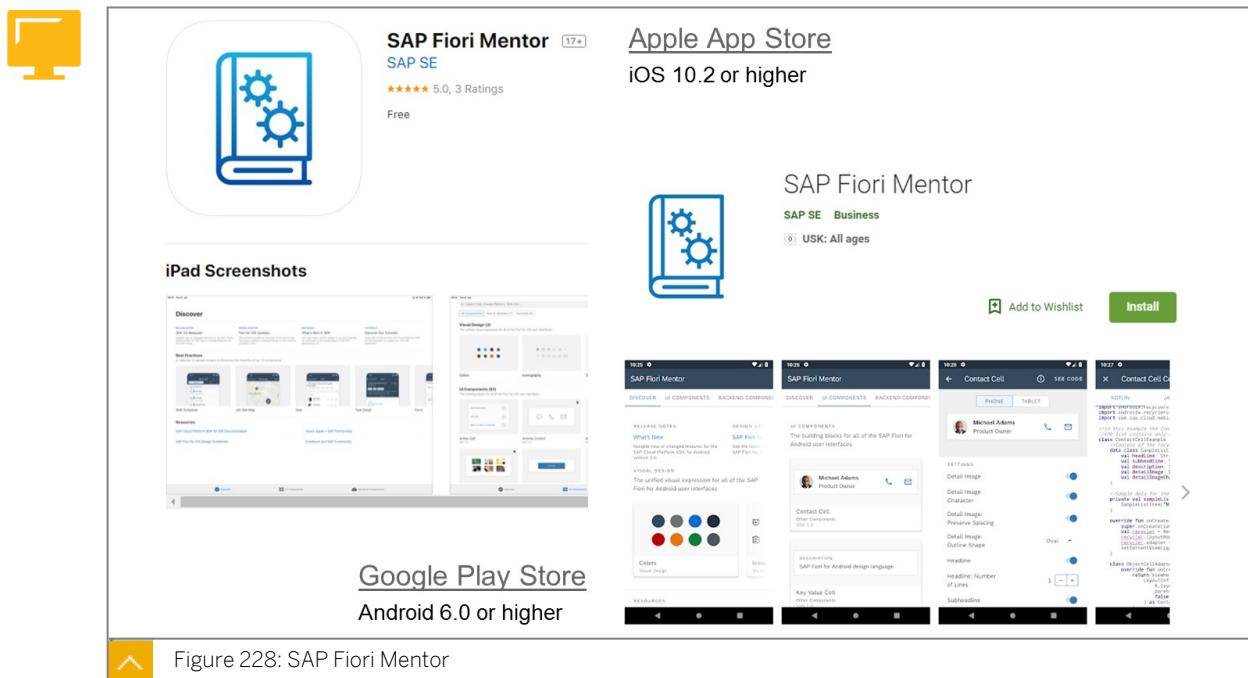


Figure 228: SAP Fiori Mentor

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 SAP 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>

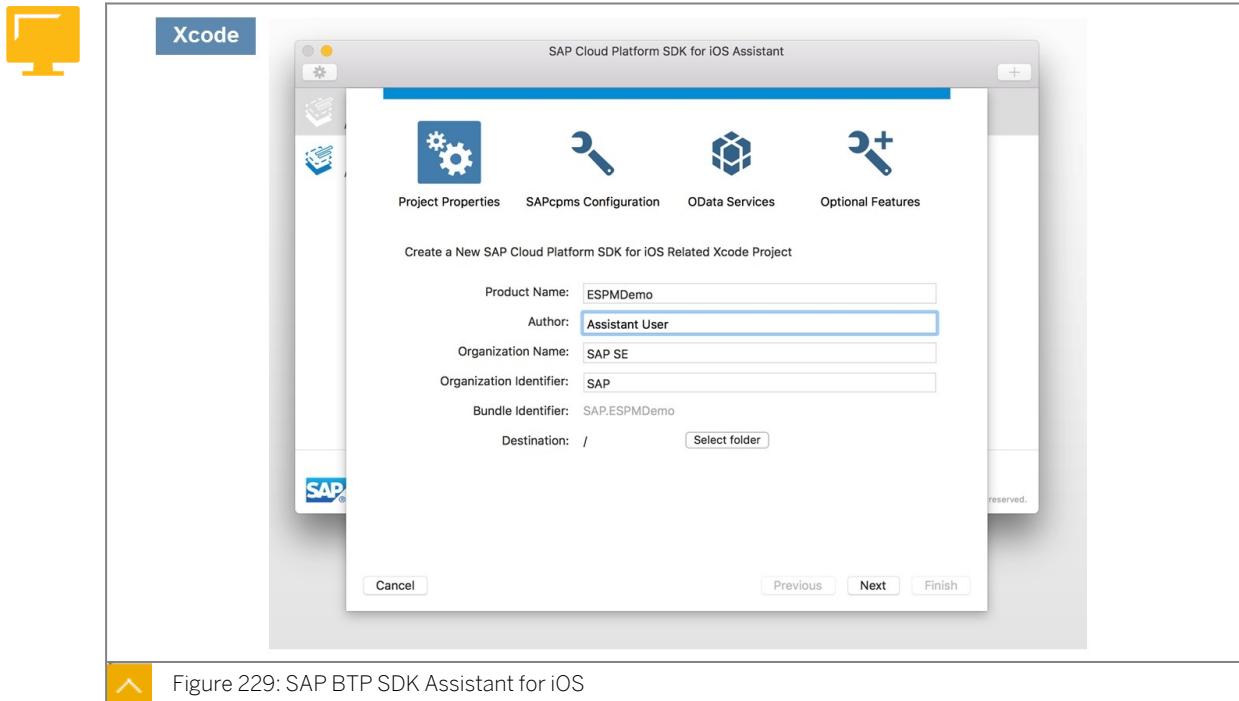


Figure 229: SAP BTP SDK Assistant for iOS

The SAP BTP SDK Assistant for iOS 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 Mobile Services configuration.



Note:

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

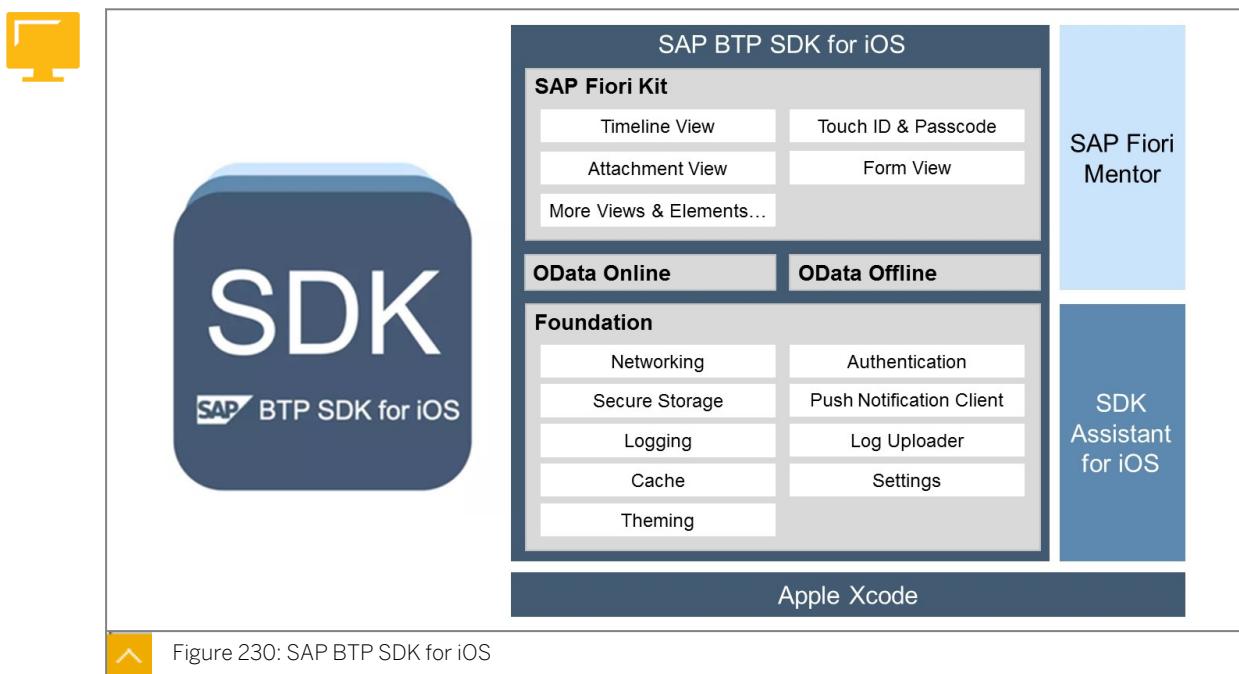
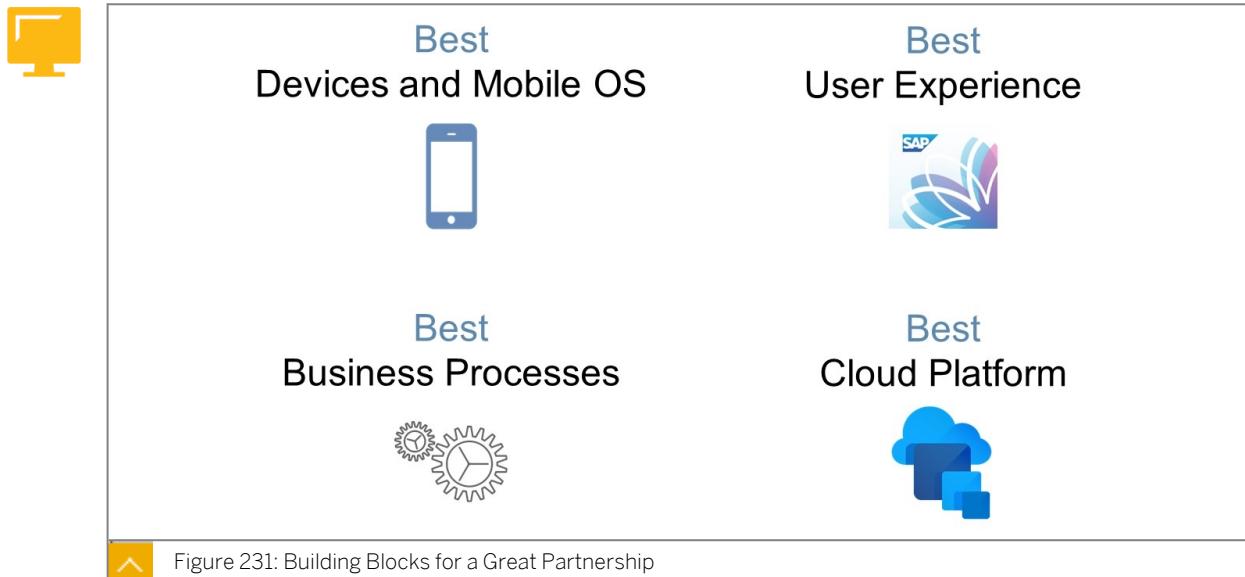


Figure 230: SAP BTP SDK for iOS

The SAP BTP SDK for iOS is the core in terms of developing SAP Fiori for iOS apps. It is a native UI layer that implements the SAP 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 BTP, 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

- Apple and SAP both focus on best developer experience and productivity
- SAP BTP SDK for iOS enables tight interaction between iOS device and SAP BTP services
- Leverage SAP BTP with many high value services for enterprise mobility



LESSON SUMMARY

You should now be able to:

- Examine SAP Fiori for iOS

Learning Assessment

1. Which mobile application type can be created using the elements on the left?

Match the item in the first column to the corresponding item in the second column.

Swift
Metadata
SAPUI5

Native App
Cross Platform
Mobile Web App

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 Mobile Services

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

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

5. What is SAP BTP SDK Assistant for iOS?

Learning Assessment - Answers

1. Which mobile application type can be created using the elements on the left?

Match the item in the first column to the corresponding item in the second column.

Swift
Metadata
SAPUI5

Native App
Cross Platform
Mobile Web App

Correct. Developing Swift results in a native app, SAPUI5 results in a mobile web app, and metadata is the source for a cross platform app.

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 Mobile Services

Correct. To improve SAP Fiori for mobile SAP 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 BTP SDK Assistant for iOS?

Plugin for Apple Xcode to create ready-to-run Xcode projects for SAP Fiori

Lesson 1

Exploring SAP Business Technology Platform Services

249

Lesson 2

Exploring SAP Launchpad Service

257

UNIT OBJECTIVES

- Explore SAP Business Technology Platform Services
- Explore SAP Launchpad Service

Unit 8

Lesson 1

Exploring SAP Business Technology Platform Services

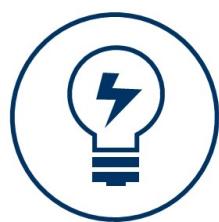


LESSON OBJECTIVES

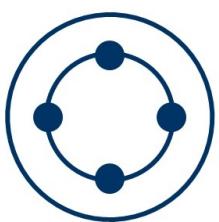
After completing this lesson, you will be able to:

- Explore SAP Business Technology Platform Services

Services for SAP Fiori



Empower
the line of
business



Connect your
enterprise to the
digital economy



Deliver open,
agile, flexible
apps



Customize
SAP
Applications

<https://cloud.sap.com>

Figure 232: SAP Business Technology Platform

SAP Business Technology Platform (BTP) 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 BTP delivers a variety of open, agile, flexible, and customizable apps, as well as an environment to develop and operate own apps.



New Solution



Cloud Extension

On-Premise Extension



Figure 233: SAP Business Technology Platform – Use Cases

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

New Solution

SAP BTP 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

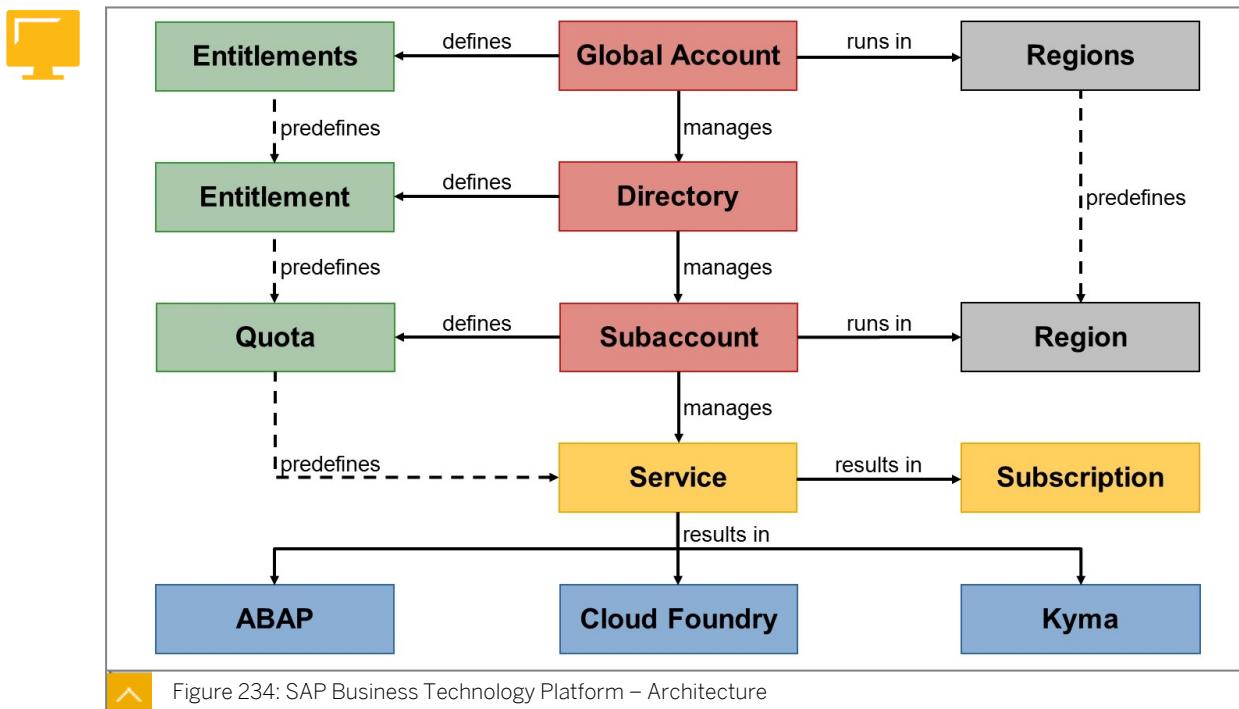
SAP BTP can be used to extend existing SAP cloud solutions and to integrate them with on-premise systems.

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

On-Premise Extension

SAP BTP 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



The SAP BTP is organized in global accounts hosted by multiple cloud infrastructure providers in different regions. On global account level, entitlements are defined assigning entities and services including billing information. Directory is a hierarchical element that allows grouping of subaccounts with entitlements. These entitlements can be configured by

defining quota plans on subaccount level, which are running in a certain region. Services defined as entitlement and sized as quota then result in subscriptions like SAP Business Application Studio and environments. SAP BTP provides the following environments:

SAP BTP, Cloud Foundry Environment

The SAP BTP, Cloud Foundry environment enables development of new business applications and business services, supporting multiple runtimes, programming languages, libraries, and services. It leverages a multitude of buildpacks including community innovations and self-developed buildpacks.

SAP BTP, Kyma Environment

The SAP BTP, Kyma environment is a fully managed Kubernetes runtime based on the open-source project "Kyma" that allows developers to extend SAP solutions with serverless functions and combine them with containerized microservices.

SAP BTP, ABAP Environment

Within the SAP BTP, Cloud Foundry environment or SAP BTP, Kyma environment, you can create a new (name) space for ABAP development. This is what is referred to as the SAP BTP, ABAP environment. It allows creation of extensions for ABAP-based products, for example SAP S/4HANA Cloud, and develop new cloud applications. It is also possible to transform existing ABAP-based custom code or extensions to the cloud.

SAP BTP, Neo Environment (only existing customers)

The SAP BTP, Neo environment allows development of HTML5, Java, and SAP HANA Extended Application Services (XS) applications. Especially SAPUI5 can be used to develop rich user interfaces for modern web-based business applications.



Development	Service Integration	Process Integration
<ul style="list-style-type: none"> • ABAP environment • HTML5 Application Repository • Kyma runtime • SAP BAS • SAP HANA Cloud • Transport Management 	<ul style="list-style-type: none"> • API Management • Cloud Management • Connectivity • Destination • Event Mesh • Integration Suite • Master Data Integration 	<ul style="list-style-type: none"> • Business Rules • Intelligent Robotic Process Automation • Job Scheduling Service • Process Visibility • Workflow • Workflow Management
User Experience	Documents	Security
<ul style="list-style-type: none"> • Launchpad Service • Mobile Services • UI Theme Designer • UI5 flexibility for key users • Web Analytics 	<ul style="list-style-type: none"> • Document Classification • Document Translation • Document Information Extraction • Print Service 	<ul style="list-style-type: none"> • Authorization and Trust Management • Credential Store • Personal Data Manager • Usage Data Management
▲ Figure 235: SAP Business Technology Platform – Services		

SAP CP offers many services in many different areas. The following services may be relevant for SAP Fiori:

Development

SAP Business Application Studio (BAS) is a cloud development tool. SAPUI5-apps build with BAS can be deployed to the HTML5 Application Repository. The ABAP environment,

Kyma runtime, and SAP HANA Cloud provide additional data and programs. The Transport Management allows the transport between subaccounts.

Service Integration

Integration Suite is the central service for integration in SAP BTP. Connectivity allows the use of a Destination to connect to other systems. API Management facilitates unified access to channels and interfaces and can be combined with Cloud Management providing additional management features. Event Mesh connects applications, services, and systems across landscapes. Master Data Integration (MDI) offers a central access layer for data sharing and distribution.

Process Integration

Intelligent Robotic Process Automation (iRPA) is an AI-driven (Artificial Intelligence) process automation, whereas Business Rules, Process VisibilityWorkflow, and Workflow Management are about process management. The Job Scheduling Service defines and manages jobs one-time or in recurring schedules.

User Experience

The central Launchpad service provides *SAP Fiori launchpads* in SAP BTP. Mobile Services offer everything concerning mobile access, development, and management. The tools *UI Theme Designer* and *UI5 flexibility for key users* allow changes to the design and behavior of SAPUI5-apps. Web Analytics enables the collection and analysis of website data.

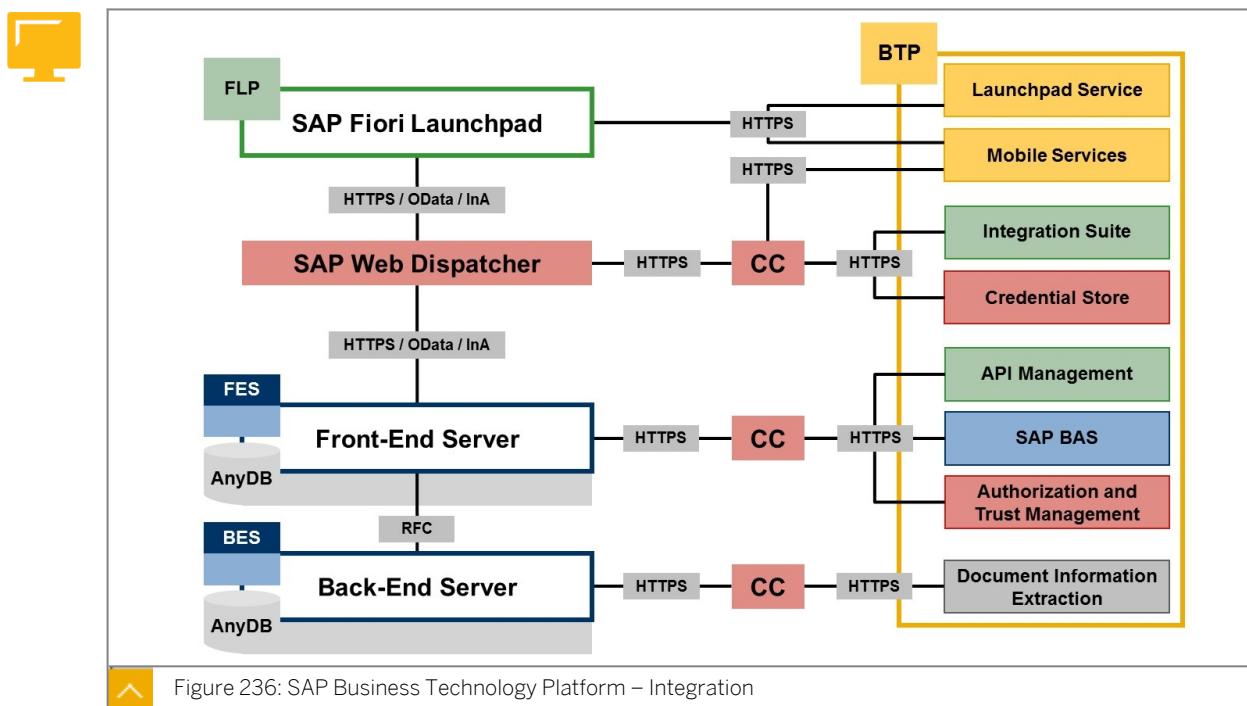
Documents

Document Classification, Document Translation, and Document Information Extraction provide features around documents described in their names. The Print Service manages print queues, connect print clients, and monitor print status.

Security

Authorization & Trust Management configures trust to identity providers for authentication and manages authorization models. Credential Store provides a secure repository for passwords and keys. Personal Data Manager helps customers to stay compliant with data protection and privacy regulations. Usage Data Management collects information about the usage of services and applications on all the entity levels.

Cloud Connector



When integrating SAP Fiori for an on-premise landscape with SAP BTP services, the connectivity settings depend on the nature of the service, and which system connects to which service. SAP Launchpad service and SAP Mobile Services enhance the FLP directly. SAP Integration Suite and SAP Credential Store enhance the FLP indirectly with additional features called via routing rules of the SAP Web Dispatcher. The FES benefits from tools like API Management, SAP Business Application Studio (BAS), and SAP Authorization and Trust Management service. You can also extend data management of the BES with SAP Document Information Extraction service for example. All connections rely on the Cloud Connector. You can install as many Cloud Connectors as necessary to fulfill security and connectivity specifications of your landscape.

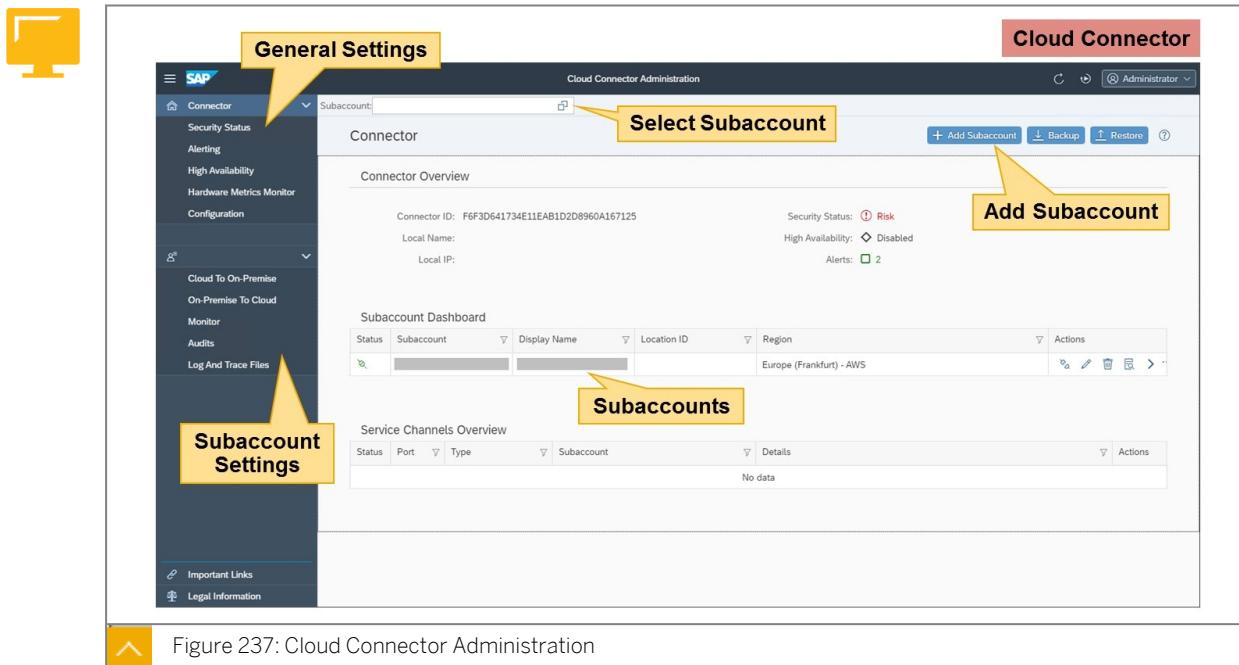


Figure 237: Cloud Connector Administration

The Cloud Connector is a middle-ware available free-of-charge on <https://tools.hana.ondemand.com/#cloud>. It provides a secure tunnel through the internet between the SAP BTP and an on-premise system landscape. It is installed in the on-premise system landscape on connectivity layer and manages landscape access based on subaccounts.

The *Cloud Connector Administrator* is the central tool for managing the Cloud Connector. In the standard setup, it can be accessed via <https://localhost:8443/>. The entry page shows a list of subaccounts and their connection status. Selecting a subaccount at the top allows to check the details of this subaccount using the subaccount settings on the left. In this details, the on-premise systems and their services are defined, which are approved to be used with SAP BTP.

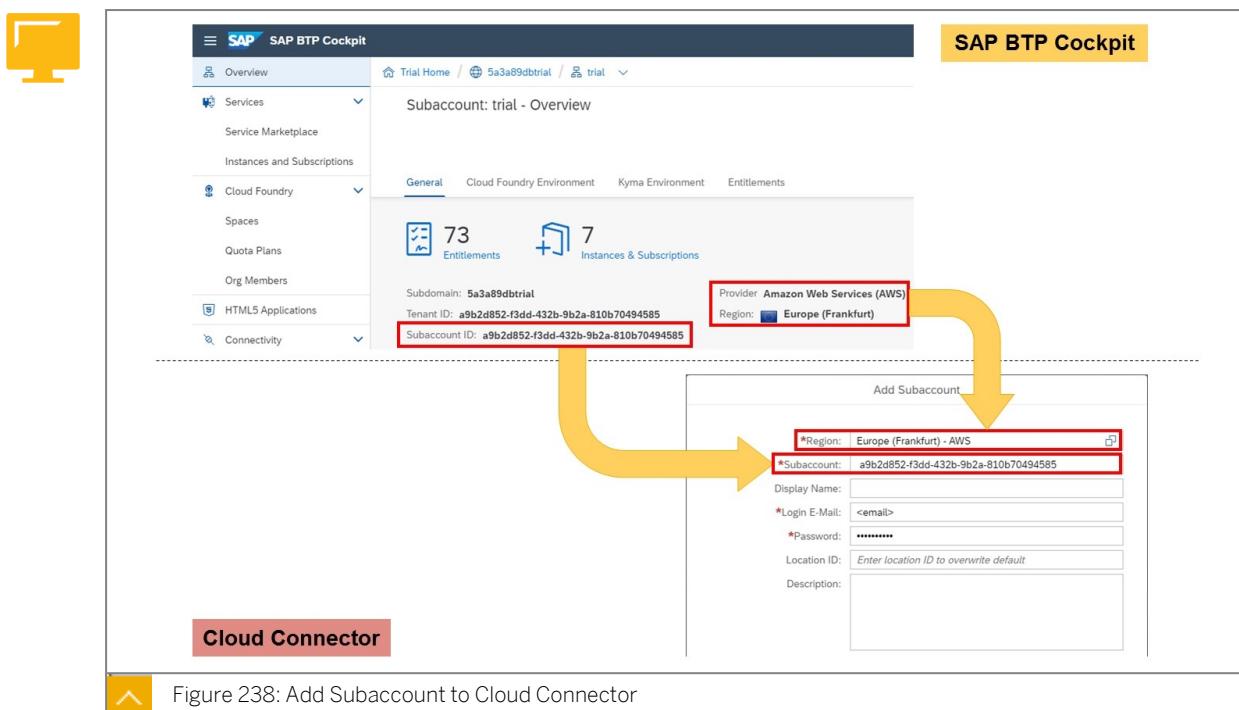


Figure 238: Add Subaccount to Cloud Connector

To add a subaccount to the Cloud Connector, region, subaccount ID, and user/password are needed. The region is written on the tile of the subaccount. The subaccount ID can be seen when choosing *More Info* on the tile or in the overview inside the subaccount.

**Hint:**

If your region is not available in the Cloud Connector, you should update the Cloud Connector.

**LESSON SUMMARY**

You should now be able to:

- Explore SAP Business Technology Platform Services

Exploring SAP Launchpad Service



LESSON OBJECTIVES

After completing this lesson, you will be able to:

- Explore SAP Launchpad Service

SAP Launchpad Service Architecture

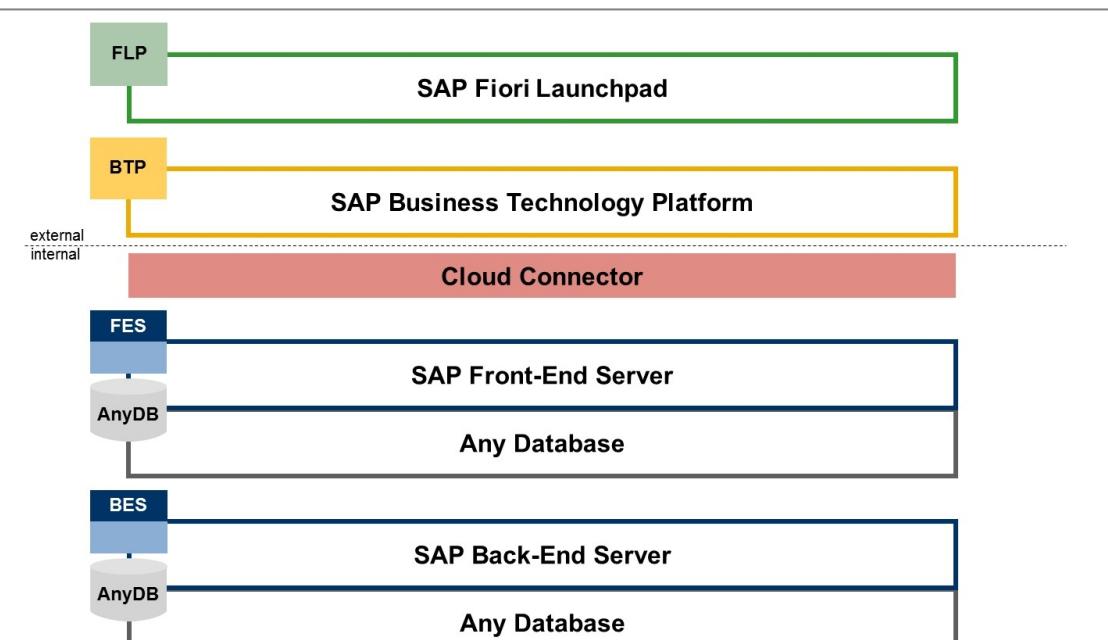
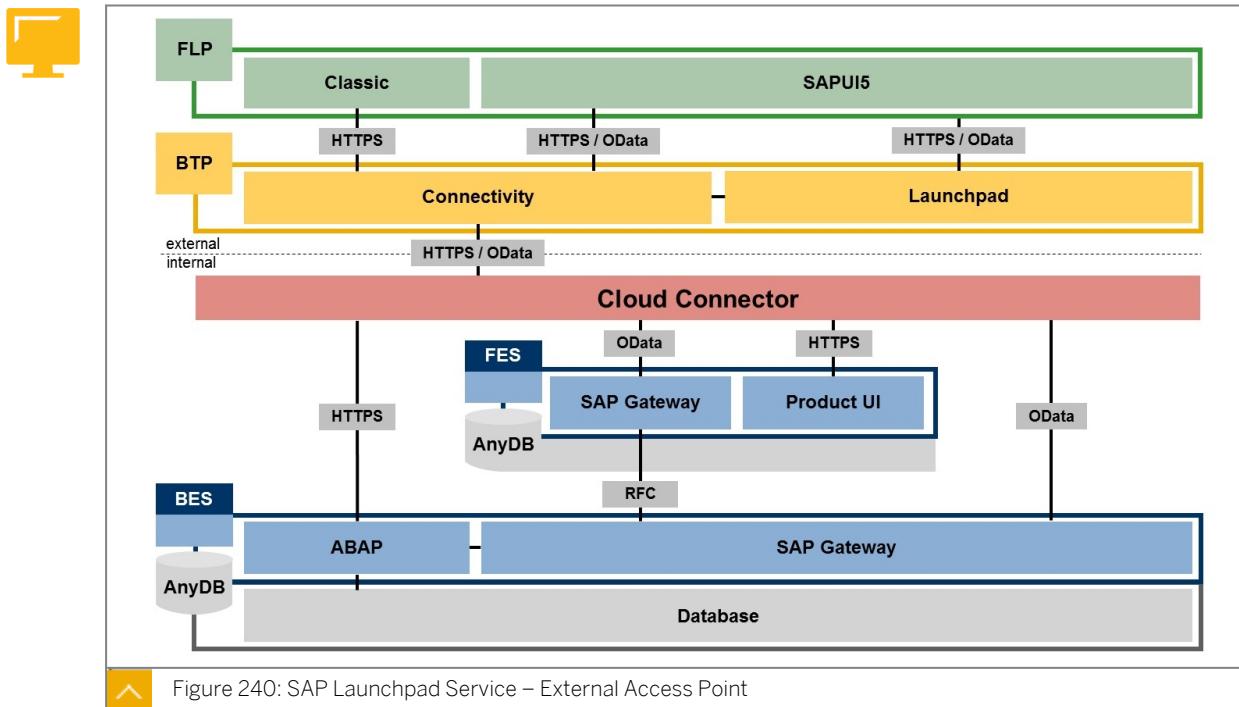
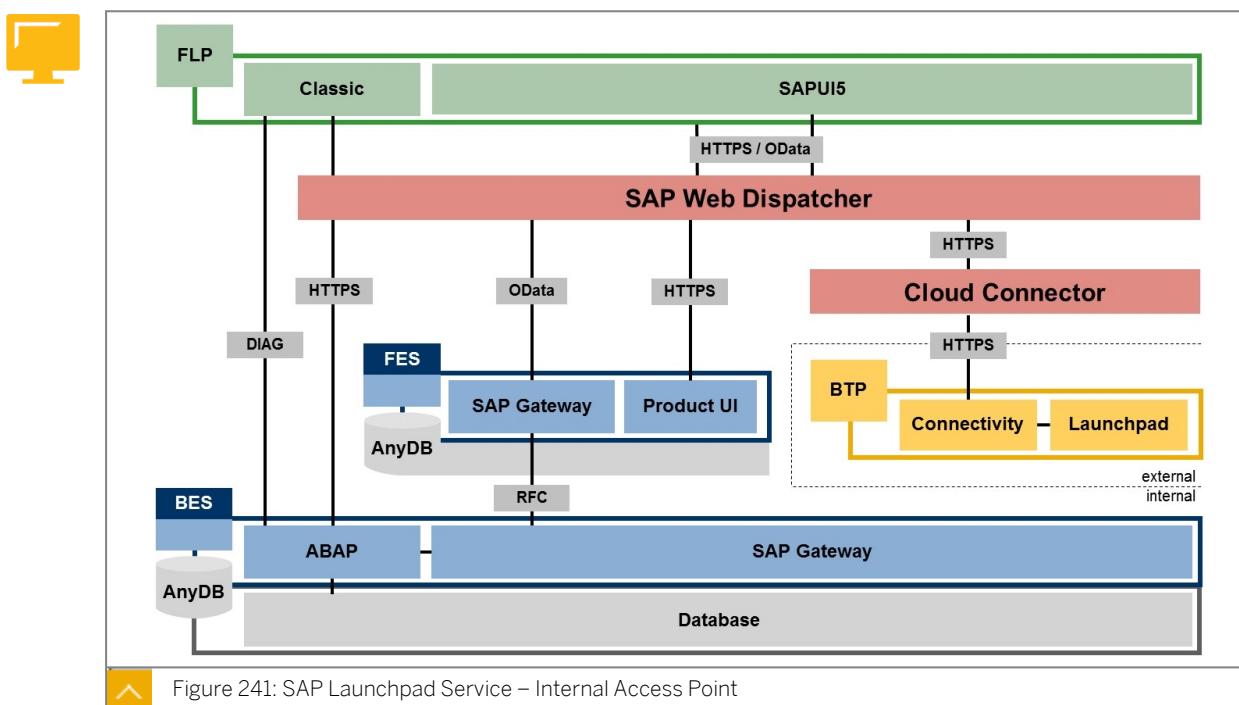


Figure 239: SAP Launchpad Service – System Landscape

SAP Business Technology Platform (BTP) runs in the Internet. Therefore, clients need to connect to the Internet to consume SAP Launchpad service. When using apps on mobile devices, the client itself is in most cases already in the Internet. When using desktop computers, it depends on the setup of the system landscape.



Communication between client and back-end server (BES) is based on HTTPS and OData. When SAP BTP is used as an external access point (EAP), client requests from the SAP Fiori launchpad (FLP) are processed by SAP Launchpad service for the FLP itself or forwarded by SAP Connectivity service via the Cloud Connector to the customer landscape. Here the customer can directly connect the BES via the Cloud Connector with SAP BTP or use a front-end server (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.



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 BTP can be used just for the UI parts as an internal access point (IAP). The FLP in the client connects to an SAP Web Dispatcher provided in the company network. This SAP Web Dispatcher routes the HTTPS request for SAP Launchpad service via a Cloud Connector to the SAP BTP. However, all OData requests are routed inside of the company network to an FES. No company data exits the corporate network in this scenario. However, the company can still benefit from the newest features of SAP Fiori in the SAP BTP.

SAP Launchpad Service Content

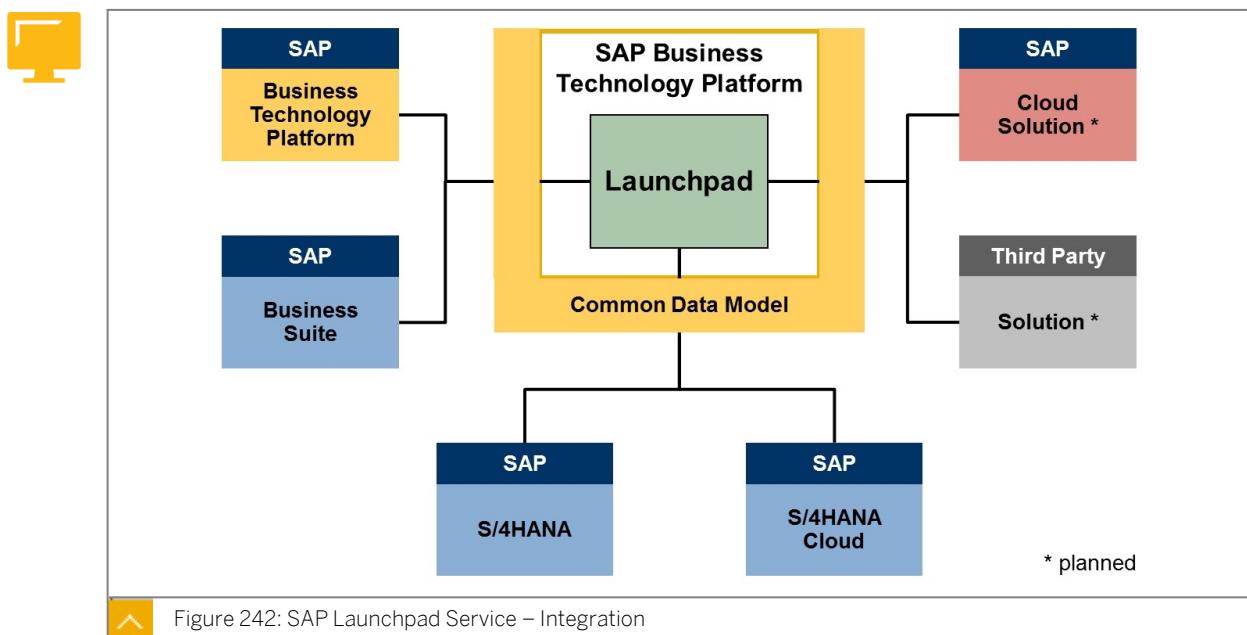


Figure 242: SAP Launchpad Service – Integration

SAP Launchpad service provides a central point of access to SAP, custom-built, and third-party applications. It offers an intuitive user experience based on SAP Fiori design, a simplified access to apps and services, and a personalized home page with role-based content. The following scenarios can be distinguished:

UX integration

Build a central point of access to SAP and third-party solutions (both cloud and on-premise), in particular accessing multiple SAP S/4HANA systems from one common launchpad.

Extensions

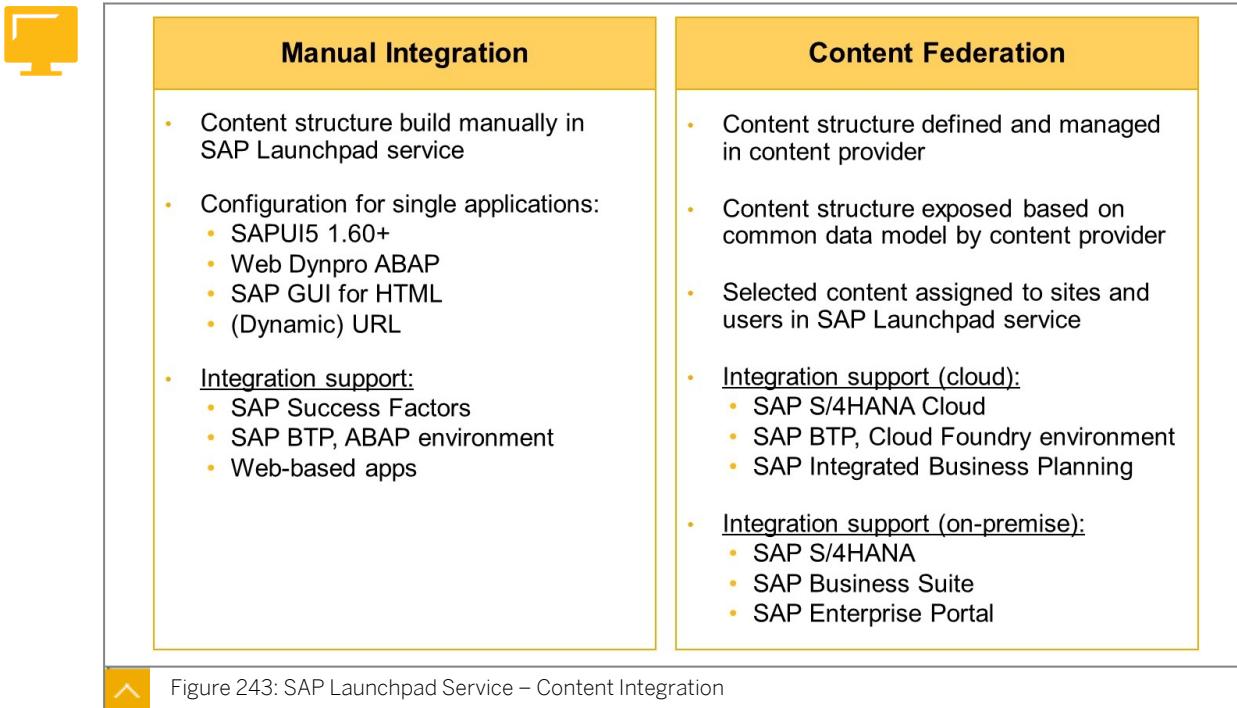
Provide an easy access to custom applications and extensions build on SAP BTP.

External-facing scenarios

Grant employees or business partners secure access to selected applications and services over the Internet on desktop and mobile devices outside the corporate network.

Cloud transition

Elevate the SAP Fiori experience to the cloud, especially for customers running an application-centric SAP Enterprise Portal.



The SAP Launchpad service provides administration capabilities for integrating business applications and managing of related content such as roles and tiles. The administrator of the SAP Launchpad service builds content structure manually and manages configuration for each application. The following application templates are available:

- SAPUI5 1.60+
- Web Dynpro ABAP
- SAP GUI for HTML
- (Dynamic) URL

For selected providers, the SAP Launchpad service offers content federation for an advanced integration of business content. This makes the day-to-day operation and maintenance of the content administrator much more efficient:

- The content administrator of the provider (for example SAP S/4HANA) manages application configuration and content structure using the native tools.
- The content administrator of the provider exposes configuration and content structure based on the common data model (CDM) format.
- The SAP Launchpad service administrator selects relevant content from the provider and assigns roles to launchpad sites and users.

Manual integration	Content federation (cloud)	Content federation (on-premise)
SAP Success Factors	SAP S/4HANA Cloud 2108	SAP S/4HANA with SAP FES 2020
SAP BTP, ABAP environment	SAP BTP, Cloud Foundry for HTML5 apps	SAP Business Suite with SAP FES 6.0 SP04

Manual integration	Content federation (cloud)	Content federation (on-premise)
Web-based apps	SAP Integrated Business Planning (IBP)	SAP Enterprise Portal 7.50 SPS 19

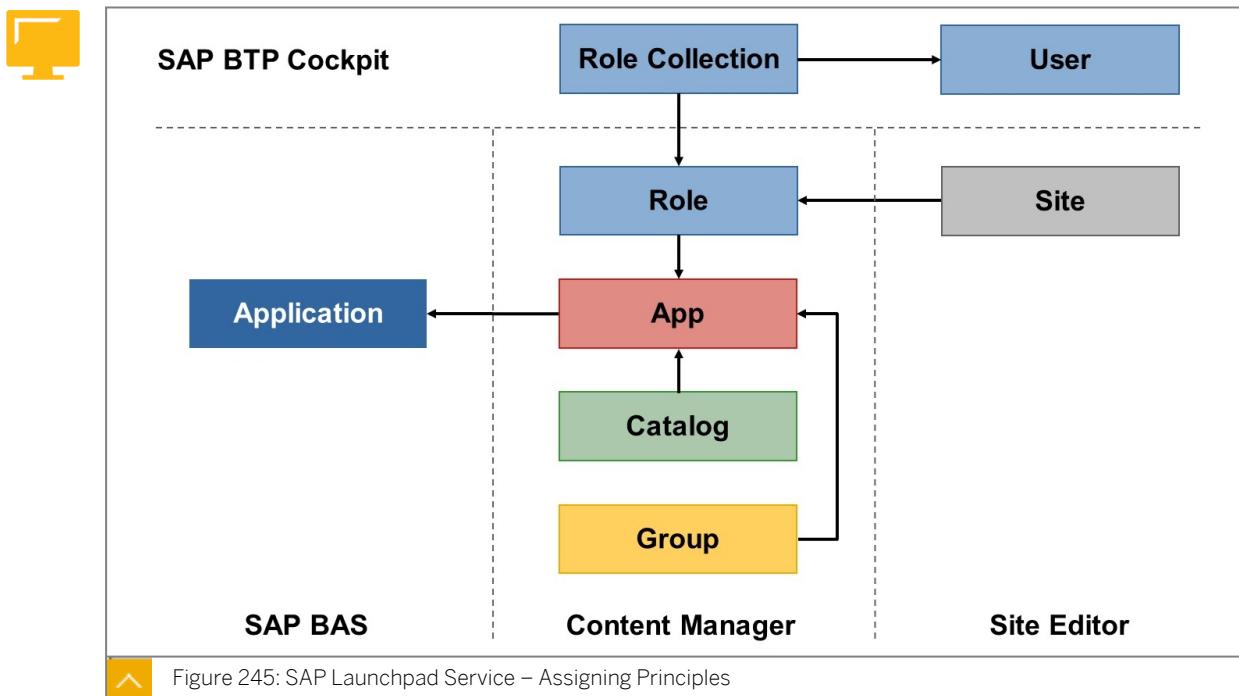
The figure shows the SAP Launchpad Service - Site Manager interface. It consists of three main panels:

- Site Directory:** A panel on the left for managing sites. It includes a "Create Site" button, an "Import Site" button, and two site preview cards: "Development Central" and "Shopping Market".
- Content Manager:** A central panel for managing content. It shows a list of items (11) with columns for Title, Description, ID, Provider, and Status. Items listed include Computer Shop, Everyone, Manage Products, Product List, SAP Business Appl., SAP Web IDE, Shop Developer, Shop Development, and Shop Management.
- Provider Manager:** A panel on the right for managing providers. It shows a table for Content Providers with columns for Title & Description, ID, Destination, Last ... (Last Update), Status, and Actions. One provider listed is "HTML5 Apps" with ID "saas_ap_prouter", Destination "HTML5 App Repository", Last Update "Nov 2, 2020, 4:19:18 PM", and Status "Active".

Yellow callout boxes on the right side of the interface identify the three panels: "Site Manager", "Content Manager", and "Provider Manager".

Figure 244: SAP Launchpad Service – Site Manager

The *Site Manager* is the central frame for administrating the SAP Launchpad service. The *Site Directory* enables management of (launchpad) sites like create, edit, and delete, and provides the *Site Editor* for configuring the details of a site. The *Content Manager* allows creating and managing apps, roles, catalogs, and groups manually or exploring and selecting content from providers. These providers are managed by the *Provider Manager*.



The app is the central element in the content model in SAP Launchpad service. It combines tile and target mapping for accessing an application and is mapped to roles, catalogs, and groups:

- Mapped to a catalog enables the app to be available in the app finder of a launchpad site.
- Mapped to a group enables the app to be available on the home page of a launchpad site.
- Mapped to a role enables the app and all mapped catalogs and groups to be assigned to a launchpad site by mapping the role to the site.

Defining a role in the *Content Manager* generates a role collection having the same name in the *SAP BTP cockpit*. This role collection can then be assigned to users allowing the users access to the site and all mapped apps, catalogs, and groups.

The screenshot shows the SAP Launchpad Service interface for managing an application named "Computer Shop". The interface is divided into three main sections: Properties, Navigation, and Visualization.

Properties Tab:

- General:** Title: Computer Shop, Description: -, ID: 9ba84544-9ebd-49af-8dd2-6300a0503a38, Open App: In place.
- Configuration:** System: No System, App UI Technology: URL, URL: <https://ui5.sap.com/test-resources/sap/m/demokit/cart/webapp/index.html>, Parameters: No data.
- Intent:** Semantic Object: ComputerOrder, Action: create.
- Parameters:** Name, Default Value, Rename To, Required: No data.
- Visualization:** Allow additional parameters checked.

Content Manager: A yellow button at the bottom center of the visualization section.

Navigation Tab:

Visualization Tab: Selected. General settings include:

- Visualization Type: Static App Launcher, Size: 1x1.
- Title: Computer Shop, Subtitle: -, Information: -, Icon: -, Supported Devices: Desktop, Tablet, Mobile.
- Parameters: No data.

Figure 246: SAP Launchpad Service – App

The mandatory fields in the properties of the app are *Title*, *System*, *App UI Technology*, and fields for defining the target depending on the selected UI technology. In the navigation *Semantic Object*, *Action*, and optional parameters are defined. The visualization offers settings for creating a tile as static or dynamic app-launcher for a supported set of devices.



LESSON SUMMARY

You should now be able to:

- Explore SAP Launchpad Service

Learning Assessment

1. What is SAP Business Technology Platform?

2. What establishes a secure connection between on-premise systems and SAP Business Technology Platform?

3. How is the system landscape setup called when the SAP Business Technology Platform is only providing the *SAP Fiori launchpad*, no business data?

4. What combines tile and target mapping in the SAP Launchpad service?

Learning Assessment - Answers

1. What is SAP Business Technology Platform?

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

2. What establishes a secure connection between on-premise systems and SAP Business Technology Platform?

Cloud Connector

3. How is the system landscape setup called when the SAP Business Technology Platform is only providing the *SAP Fiori launchpad*, no business data?

Internal Access Point

4. What combines tile and target mapping in the SAP Launchpad service?

App

Lesson 1

Getting Further Information

269

UNIT OBJECTIVES

- Get further information

Getting Further Information



LESSON OBJECTIVES

After completing this lesson, you will be able to:

- Get further information

Further Information

You can find further information here:



- Administration and configuration courses:
 - ADM103 (System Administration II of SAP S/4HANA and SAP Business Suite):
<https://training.sap.com/course/adm103>
 - ADM945 (Authorization Concept for SAP S/4HANA):
<https://training.sap.com/course/adm945>
 - BIT602 (SAP Workflow with SAP Fiori):
<https://training.sap.com/course/bit602>
 - HA200 (SAP HANA – Installation and Administration):
<https://training.sap.com/course/ha200>
 - UX200 (SAP Fiori – System Administration):
<https://training.sap.com/course/ux200>



- UI development courses:
 - UX399 (Web Development Basics (JavaScript, HTML5 and CSS3)):
<https://training.sap.com/course/ux399>
 - UX400 (Developing UIs with SAPUI5):
<https://training.sap.com/course/ux400>
 - UX402 (Advanced SAPUI5 Development):
<https://training.sap.com/course/ux402>
 - UX403 (SAP Fiori Elements Development):
<https://training.sap.com/course/ux403>
 - UX410 (Developing SAP Fiori UIs):
<https://training.sap.com/course/ux410>



- Back-end development courses:
 - GW100 (SAP Gateway – Building OData Services):
<https://training.sap.com/course/gw100>
 - S4DEV (Introduction to Application Programming on SAP S/4HANA):
<https://training.sap.com/course/s4dev>
 - S4D400 (Introduction to ABAP Programming on SAP S/4HANA):
<https://training.sap.com/course/s4d400>
 - S4D425 (Extensibility for SAP S/4HANA):
<https://training.sap.com/course/s4d425>
 - S4D430 (Building Views in ABAP Core Data Services (CDS)):
<https://training.sap.com/course/s4d430>



- Development communities:
 - ABAP Development:
<https://community.sap.com/topics/abap>
 - ABAP Extensibility:
<https://community.sap.com/topics/abap-extensibility>
 - SAP Business Application Studio:
<https://community.sap.com/topics/business-application-studio>
 - SAP Gateway:
<https://community.sap.com/topics/gateway>
 - SAPUI5 Development:
<https://community.sap.com/topics/ui5>



- Integration communities:
 - Mobility:
<https://community.sap.com/topics/mobile-platform>
 - SAP Business Client:
<https://community.sap.com/topics/business-client>
 - SAP Business Technology Platform:
<https://community.sap.com/topics/business-technology-platform>
 - SAP Conversational AI:
<https://community.sap.com/topics/conversational-ai>
 - SAP Portal and SAP Launchpad Service:
<https://community.sap.com/topics/portal>



- User experience communities:
 - SAP Fiori :
<https://community.sap.com/topics/fiori>
 - SAP Fiori Elements:
<https://community.sap.com/topics/fiori-elements>
 - SAP Fiori Tools:
<https://community.sap.com/topics/fiori-tools>
 - SAP Screen Personas:
<https://community.sap.com/topics/screen-personas>
 - UI Theme Designer:
<https://community.sap.com/topics/ui-theme-designer>



LESSON SUMMARY

You should now be able to:

- Get further information

Lesson 1

Examining SAP Business Suite

275

Lesson 2

Creating Classic Target Mappings

285

Lesson 3

Configuring Analytical Applications

293

Lesson 4

Integrating SAP Fiori in SAP Enterprise Portal

301

Lesson 5

Integrating SAP Workflow in SAP Fiori

309

Lesson 6

Using SAP Screen Personas

315

UNIT OBJECTIVES

- Examine Architecture of SAP Business Suite
- Examine Principal Apps
- Examine Content Model for SAP Business Suite
- Create LPD_CUST target mappings
- Create Web Dynpro target mappings
- Create transaction target mappings
- Enable SAP Fiori Smart Business
- Model SAP Fiori KPI Apps
- Integrate SAP Fiori in SAP Enterprise Portal

- Integrate SAP Workflow in SAP Fiori
- Use SAP screen personas

Examining SAP Business Suite



LESSON OBJECTIVES

After completing this lesson, you will be able to:

- Examine Architecture of SAP Business Suite
- Examine Principal Apps
- Examine Content Model for SAP Business Suite

Landscape of SAP Fiori for SAP Business Suite

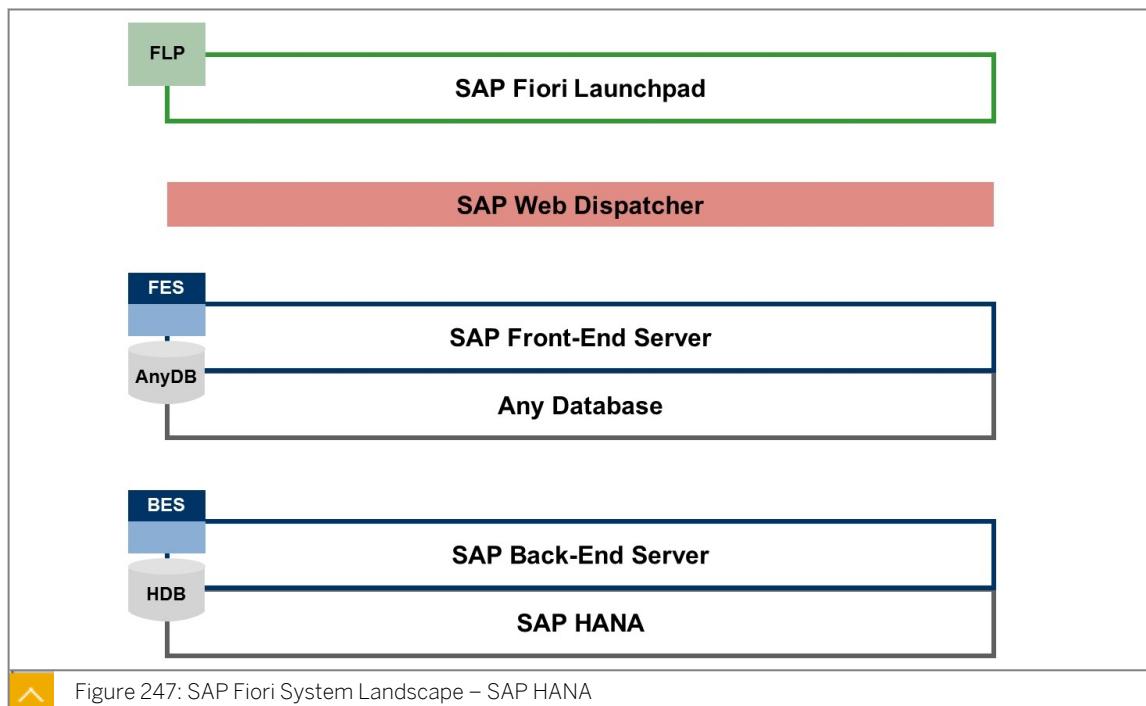
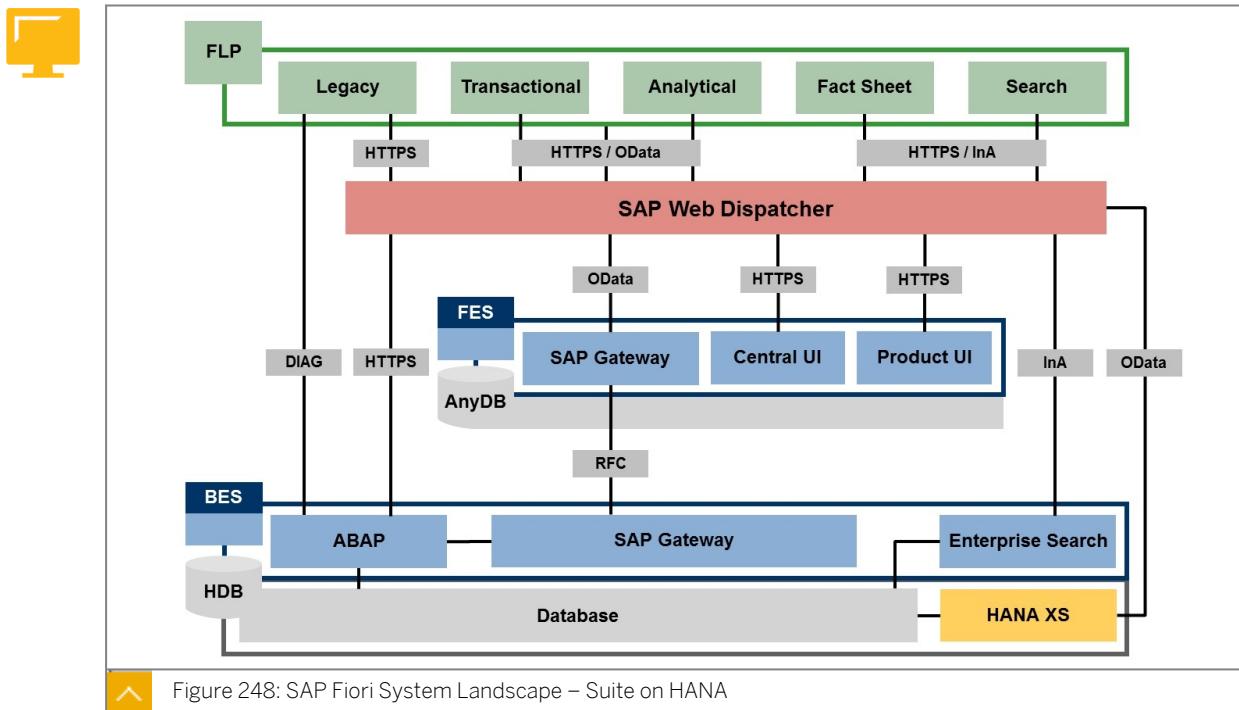


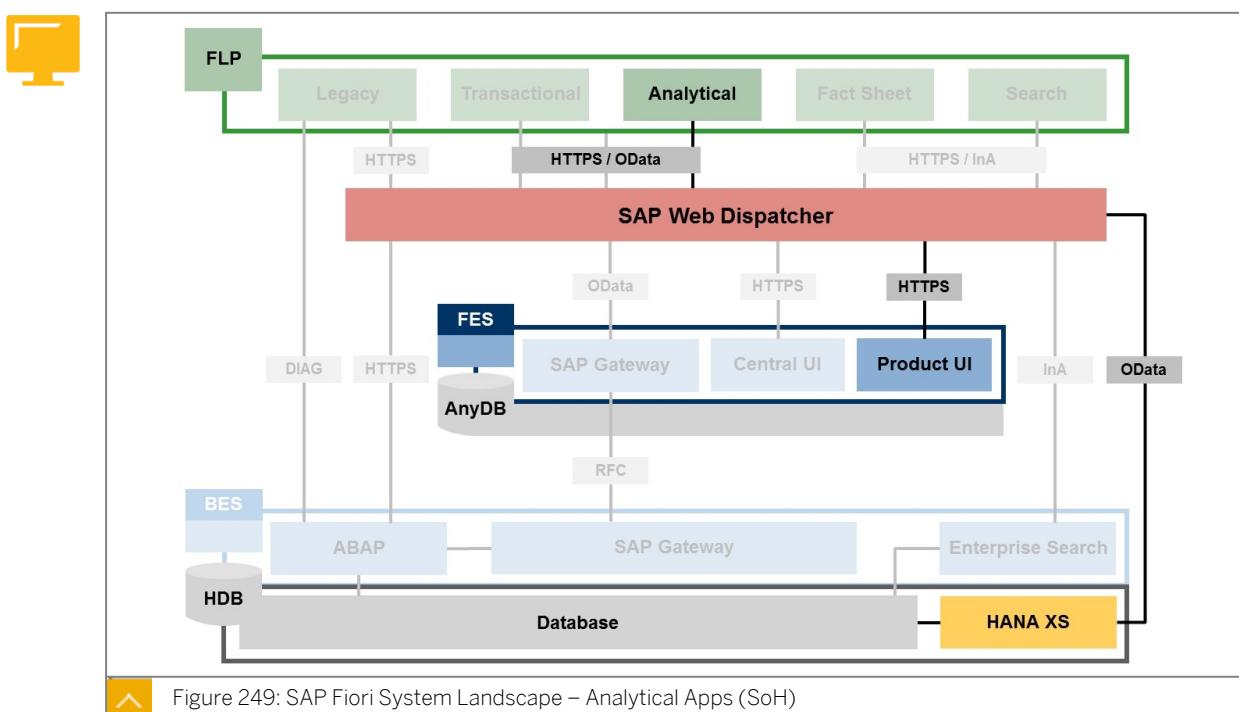
Figure 247: SAP Fiori System Landscape – SAP HANA

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, in a Suite on HANA (SoH), analytical and fact sheet apps are now available together with the SAP Fiori search on back-end data. The SAP Web Dispatcher becomes more important because analytical apps get their data directly from the SAP HANA extended application services (XS). 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.

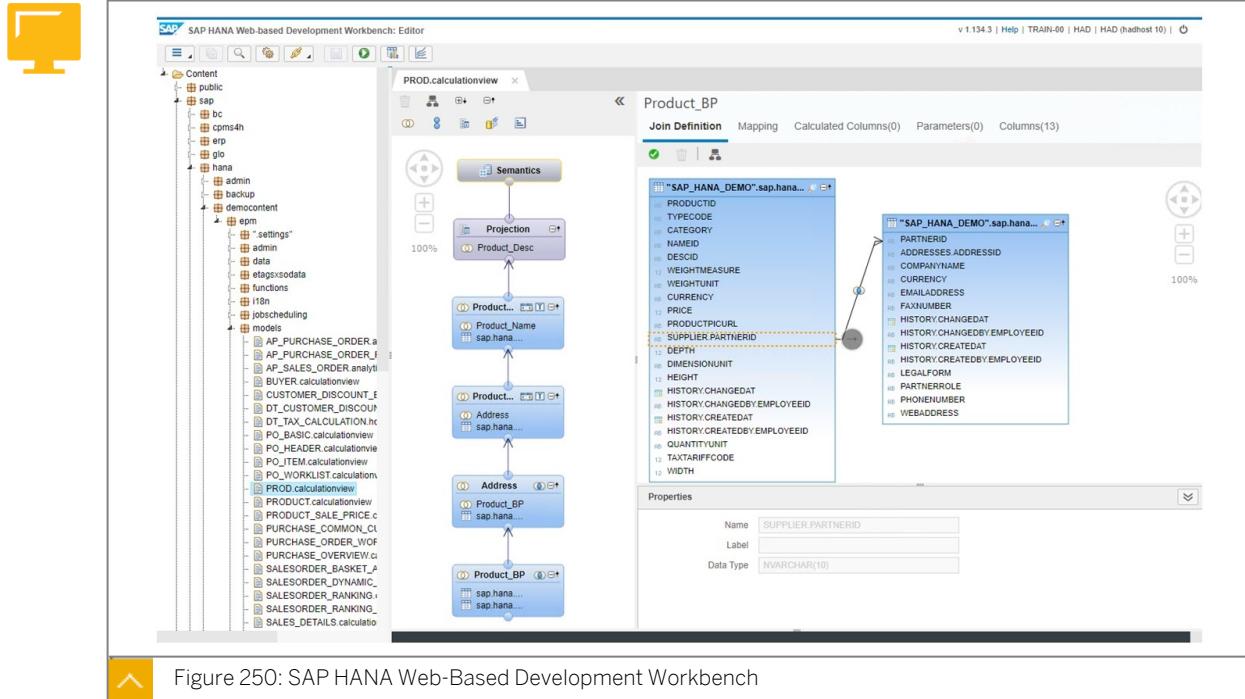


Figure 250: SAP HANA Web-Based Development Workbench

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* enables you to develop entire applications in a Web browser without having to install any development tools. It 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 must be installed separately.



My Quotation Pipeline [F0028]

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.

HANA XS Components

Product Version	SAP SMART BUSINESS FOR ERP 1.0 SAP Smart Business 1.0 for SAP ERP
Support Package Stack	05 (01/2015)
Product Instance(s)	Quotation Apps
Software Component Version	HCO_HBA_A_QTPI_S 100
Prerequisite for installation	SAP SMART BUSINESS FOR ERP 1.0 - SPS 05 (01/2015) is an <i>Add On</i> 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
qtpi_xsodata	sap.hba.apps.qtpi.s.odata

SAP HANA Roles

Role	Package
sap.hba.apps.qtpl.s.roles:qtpl	sap.hba.apps.qtpl.s.roles



Figure 251: SAP Analytical App based on SAP HANA XS

In this example from the *SAP Fiori apps reference library*, you see HANA XS components are part of the installation. These must 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.

SAP HANA Enterprise Search

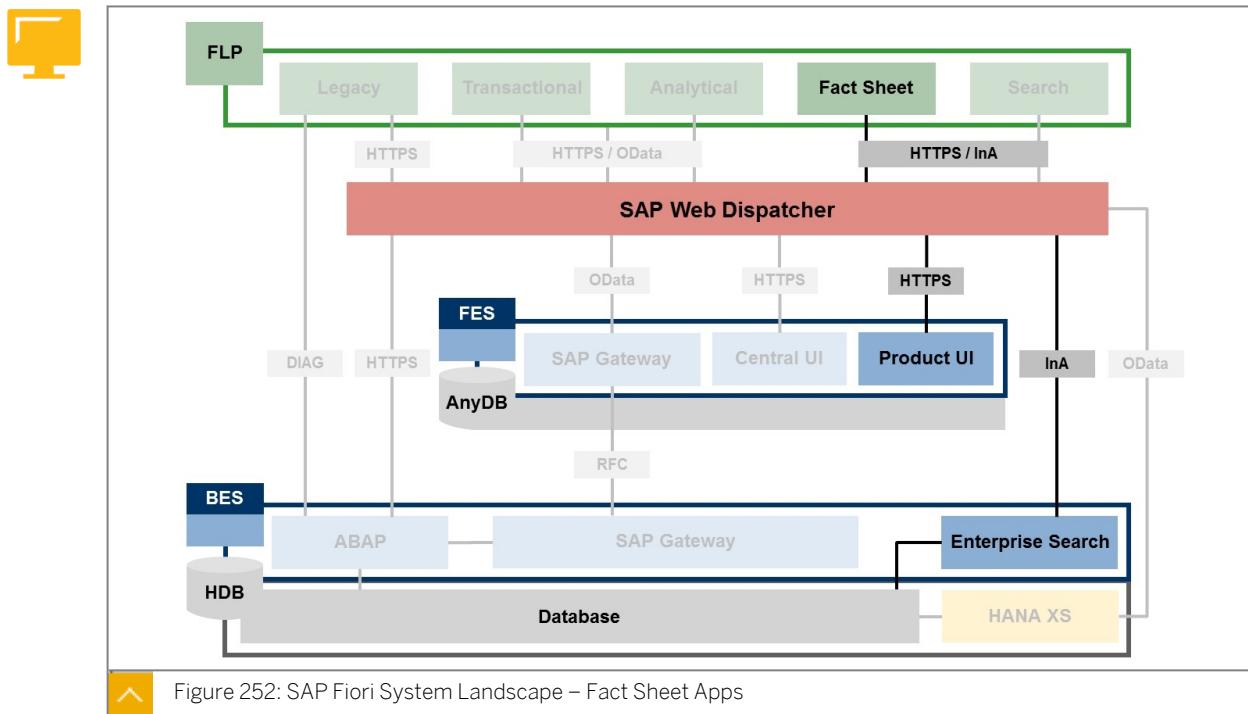


Figure 252: SAP Fiori System Landscape – Fact Sheet Apps

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. This protocol 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.

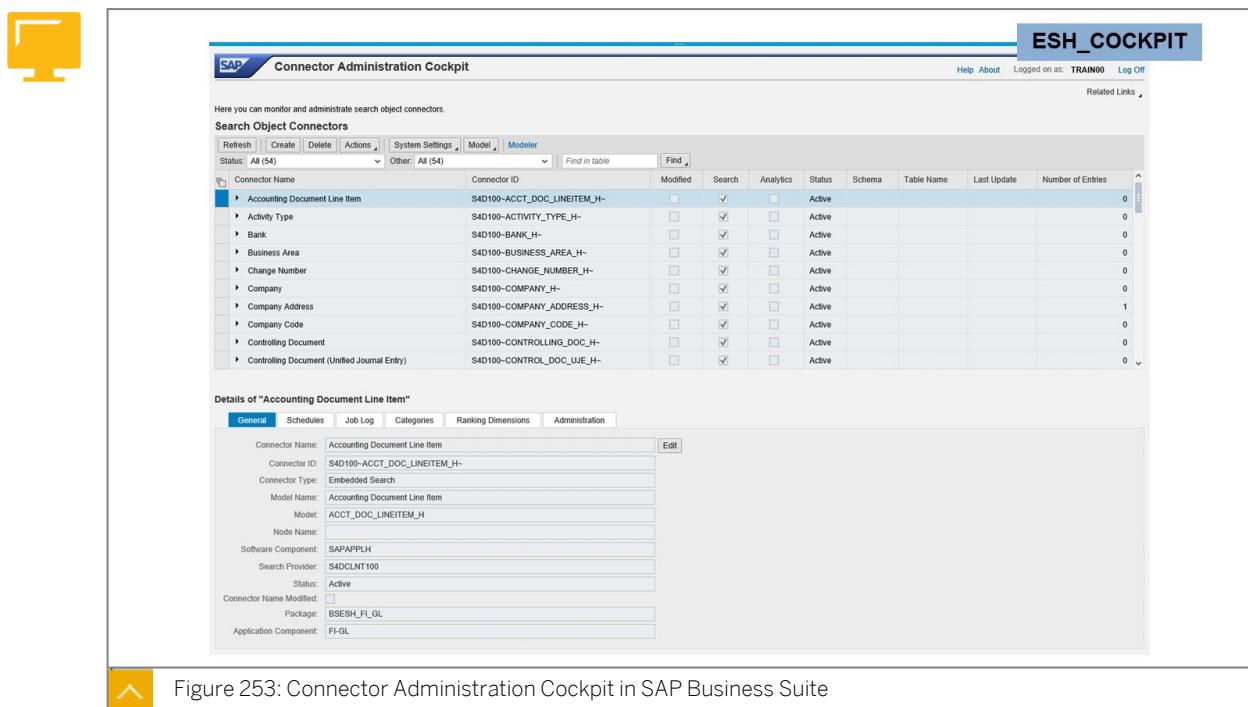


Figure 253: Connector Administration Cockpit in SAP Business Suite

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.

The screenshot shows the SAP Fiori Apps Reference Library interface. At the top right, it says "Bill of Material [F0640]". On the left, there's a yellow computer monitor icon. Below the title, there are two sections: "Installation" and "Configuration".

Installation: A note states: "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." Below this is a table for "Back-End Components (ABAP)" with the following data:

Product Version	EHP7 FOR SAP ERP 6.0 SAP enhancement package 7 for SAP ERP 6.0
Support Package Stack	09 (07/2015)
Product Instance(s)	Master Data Governance
Software Component Version	SAP_APPL 617

Configuration: A note says: "The following sections list app-specific data required to configure the app:". Under "OData Services", it says: "The following OData services must be activated on the front-end server. Users require PFCG authorization for the front-end and back-end systems." Below this is a table:

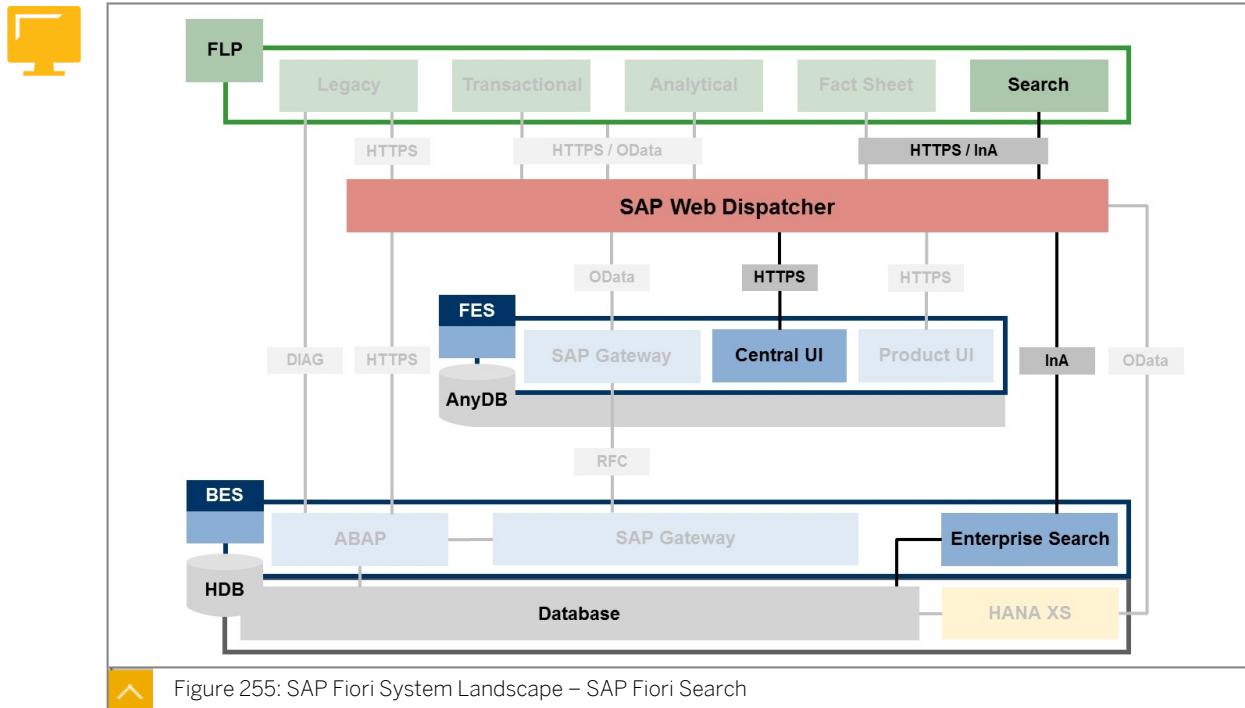
oData Service	Version	Back-End Authorization Role (PFCG)
CB_BILLOFMATERIAL_SRV	001	SAP_PLM_BILLOFMATERIAL_APP

Under "Search Connector", it says: "The following search connectors need to be activated in the back-end system." Below this is another table:

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

At the bottom left is a yellow arrow pointing up, and at the bottom right is the caption: "Figure 254: 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.

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:

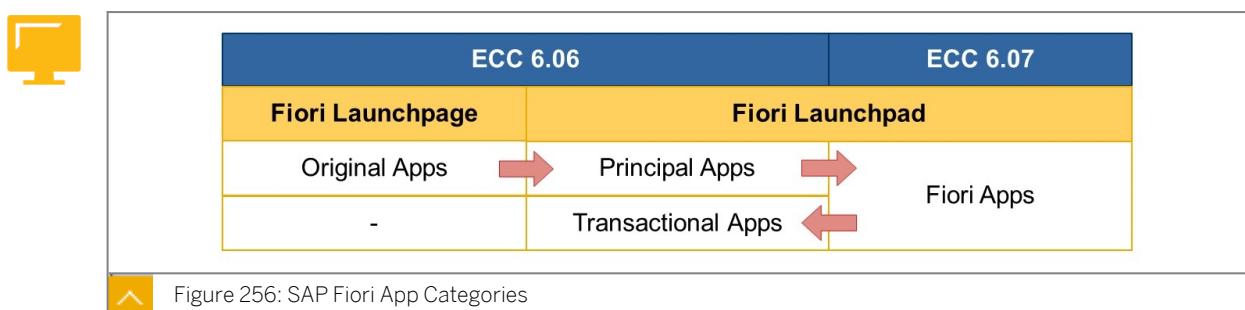
Bad Performance

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.

Principal Apps

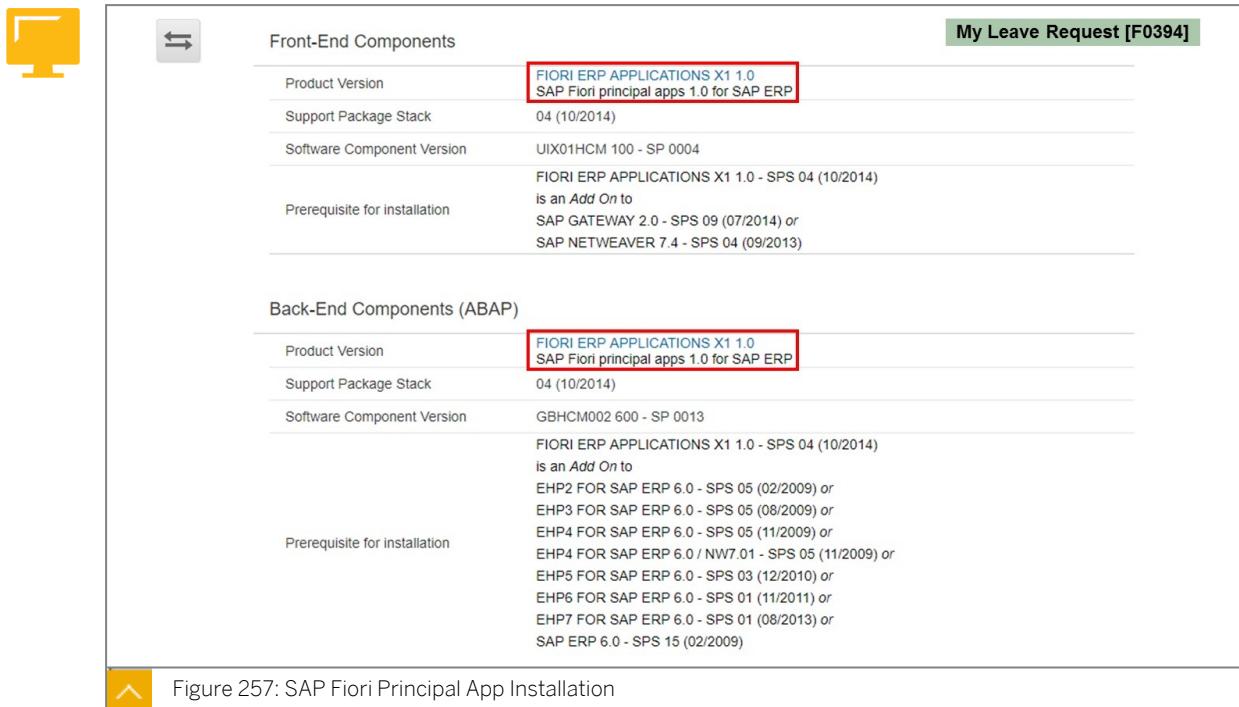


In 2013, the first 25 SAP Fiori apps were developed for ECC 6.06. These so-called original apps are completely different in their architecture compared to current SAP Fiori apps. They are stand-alone transactional apps, which can be embedded in the SAP Fiori launchpage. The launchpage is simply a webpage offering links visualized as tiles to start the original apps. No

customizing or any other adaptation is possible. The original apps and the *SAP Fiori launchpage* are deprecated and no longer available.

With ECC 6.07, the SAP Fiori apps as we know them currently were introduced, including the *SAP 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 *SAP 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.



The screenshot shows the SAP Fiori Principal App Installation screen. At the top right, it says "My Leave Request [F0394]". On the left, there's a yellow icon of a computer monitor with arrows. The main content is divided into two sections: "Front-End Components" and "Back-End Components (ABAP)".

Front-End Components:

Product Version	FIORI ERP APPLICATIONS X1 1.0 SAP Fiori principal apps 1.0 for SAP ERP
Support Package Stack	04 (10/2014)
Software Component Version	UIX01HCM 100 - SP 0004
Prerequisite for installation	FIORI ERP APPLICATIONS X1 1.0 - SPS 04 (10/2014) is an Add On to SAP GATEWAY 2.0 - SPS 09 (07/2014) or SAP NETWEAVER 7.4 - SPS 04 (09/2013)

Back-End Components (ABAP):

Product Version	FIORI ERP APPLICATIONS X1 1.0 SAP Fiori principal apps 1.0 for SAP ERP
Support Package Stack	04 (10/2014)
Software Component Version	GBHCM002 600 - SP 0013
Prerequisite for installation	FIORI ERP APPLICATIONS X1 1.0 - SPS 04 (10/2014) is an Add On to EHP2 FOR SAP ERP 6.0 - SPS 05 (02/2009) or EHP3 FOR SAP ERP 6.0 - SPS 05 (08/2009) or EHP4 FOR SAP ERP 6.0 - SPS 05 (11/2009) or EHP4 FOR SAP ERP 6.0 / NW7.01 - SPS 05 (11/2009) or EHP5 FOR SAP ERP 6.0 - SPS 03 (12/2010) or EHP6 FOR SAP ERP 6.0 - SPS 01 (11/2011) or EHP7 FOR SAP ERP 6.0 - SPS 01 (08/2013) or SAP ERP 6.0 - SPS 15 (02/2009)

In this example of the *SAP Fiori apps 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 SAP Fiori apps. Most of them have a version 2 in the title or have been renamed.

Content Model for SAP Business Suite

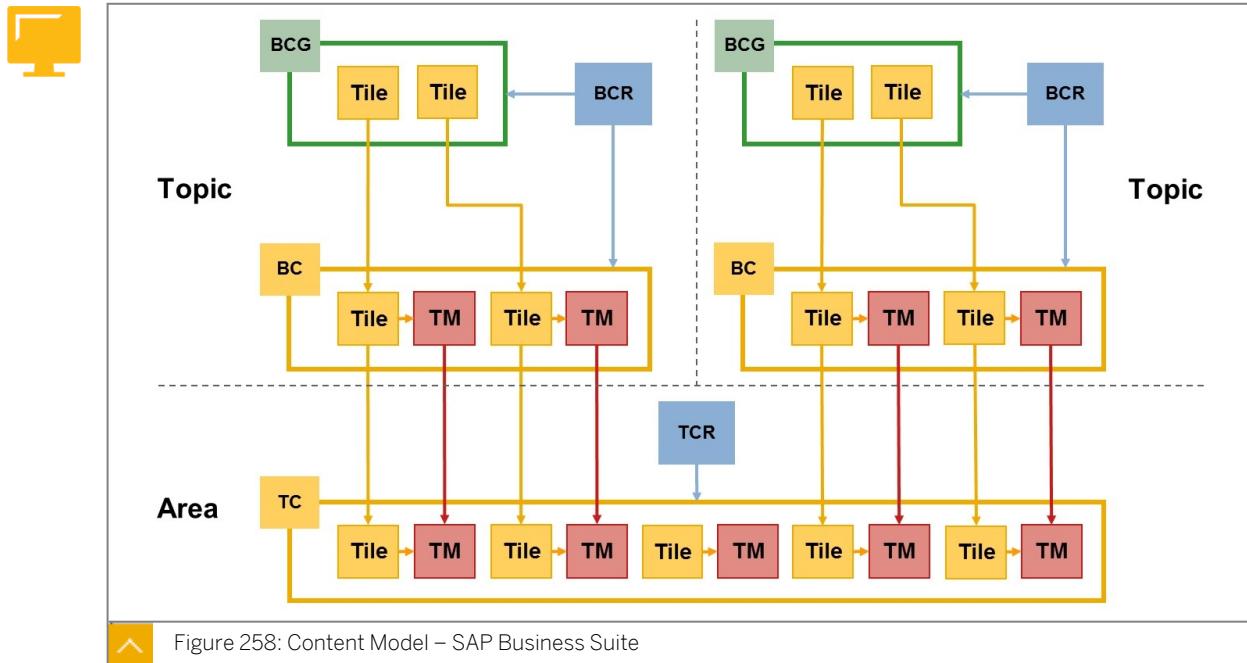


Figure 258: Content Model – SAP Business Suite

In the content model for SAP Business Suite , 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).

The screenshot shows the SAP Fiori Content Model configuration interface. It includes sections for Technical Configuration, Business Catalog(s), Business Group(s), and Business Role(s). The interface features a sidebar with a monitor icon and a 'My Team Calendar [F1309]' button. The 'Technical Configuration' section lists items like SAP_HR_TC_T, HCM, and SAP_HR_TCR_T. The 'Business Catalog(s)' section lists SAP_HR_BC_EMPLOYEE_T with a description of 'Employee (HCM) - Content'. The 'Business Group(s)' section lists SAP_HR_BCG_EMPLOYEE_T with a description of 'Employee (HCM)'. The 'Business Role(s)' section lists SAP_HR_BCR_EMPLOYEE_T with a description of 'Employee - HR Info'. Buttons for 'Extend Apps Selection' are visible in each section.

Figure 259: Content Model – SAP Business Suite Example

In this example from the *SAP Fiori apps 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 abbreviation makes the difference.



LESSON SUMMARY

You should now be able to:

- Examine Architecture of SAP Business Suite
- Examine Principal Apps
- Examine Content Model for SAP Business Suite

Creating Classic Target Mappings



LESSON OBJECTIVES

After completing this lesson, you will be able to:

- Create LPD_CUST target mappings
- Create Web Dynpro target mappings
- Create transaction target mappings

LPD_CUST Mapping Elements

The screenshot shows a configuration interface for a SAPUI5 application named "MDG_APPROVE_CR". The interface includes sections for "SAPUI5 Application", "Technical Configuration", and a summary table.

SAPUI5 Application

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

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 Configuration

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 260: Technical Configuration Using LPD_CUST

In this example from the *SAP Fiori apps 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.

 **FLPD**

SAP_MDG_TC_T: Approve Master Data

Intent

Semantic Object:	ChangeRequest	<input type="button" value=""/>
Action:	approveChangeRequest	<input type="button" value=""/>

Target

Application Type:	SAP Fiori App using LPD_CUST	<input type="button" value=""/>
Launchpad Role:	UIMDG001	<input type="button" value=""/>
Launchpad Instance:	TRANSACTIONAL	<input type="button" value=""/>
Application Alias:	ApproveChangeRequest	<input type="button" value=""/>
Application ID:	<input type="text"/>	

 Figure 261: 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.

 **LPD_CUST**

Overview of Launchpads

Role	Instance	Description	Re...	Em...	FP...	Change	Display	Delete	Copy	Transp...	Alias M...	Transla...	User Date	
IWP_FILE_R	IWP_FILE_R	Local Reporting Renovation	<input type="checkbox"/>	SAP 07.11.2012	<input type="checkbox"/>									
IWP_LR_DP	IWP_LR_DP	Local Reporting Renovation	<input type="checkbox"/>	SAP 07.11.2012	<input type="checkbox"/>									
IWP_REN_LR	IWP_REN_LR	Local Reporting Renovation	<input type="checkbox"/>	SAP 07.11.2012	<input type="checkbox"/>									
LUMIRA	ANALYTICS	Lumira Analytics	<input type="checkbox"/>	SAP 13.05.2014	<input type="checkbox"/>									
MISMATCH_...	MISMATCH_INSTANCE	Customization for the error application	<input type="checkbox"/>	SAP 16.09.2010	<input type="checkbox"/>									
SPBUIT0	NEWSITEM	News	<input type="checkbox"/>	SAP 11.03.2014	<input type="checkbox"/>									
UIAPF170	FACTSHEETS	SAP SFIN Launchpad for Fact Sheets	<input type="checkbox"/>	SAP 01.04.2015	<input type="checkbox"/>									
UIAPF170	TRANSACTIONAL_CM	SAP SFIN Launchpad for CM-Transactions	<input type="checkbox"/>	SAP 19.11.2015	<input type="checkbox"/>									
UIAPF170	TRANSACTIONAL_CO	SAP SFIN Launchpad for CO-Transactions	<input type="checkbox"/>	SAP 19.11.2015	<input type="checkbox"/>									
UIAPF170	TRANSACTIONAL_FI	SAP SFIN Launchpad for FI-Transactions	<input type="checkbox"/>	SAP 19.11.2015	<input type="checkbox"/>									
UIAPF170	TRANSACTIONAL_FO	SAP SFIN Launchpad for FO-Transactions	<input type="checkbox"/>	SAP 19.11.2015	<input type="checkbox"/>									
UIAPF170	TRANSACTIONAL_GLO	SAP SFIN Launchpad for Globalization	<input type="checkbox"/>	SAP 19.11.2015	<input type="checkbox"/>									
UICORE	CA	Applications - CA	<input type="checkbox"/>	SAP 20.11.2015	<input type="checkbox"/>									
UICORE	CMD	Applications - Central Master Data	<input type="checkbox"/>	SAP 20.11.2015	<input type="checkbox"/>									
UICORE	EAM	Applications - EAM	<input type="checkbox"/>	SAP 20.11.2015	<input type="checkbox"/>									
UICORE	GTS	Applications - Global Trade Services	<input type="checkbox"/>	SAP 19.11.2015	<input type="checkbox"/>									
UICORE	LO	Applications - Settlement Management	<input type="checkbox"/>	SAP 19.11.2015	<input type="checkbox"/>									
UICORE	PLM	Applications - PLM	<input type="checkbox"/>	SAP 19.11.2015	<input type="checkbox"/>									
UICORE	PRC	Applications - Procurement	<input type="checkbox"/>	SAP 19.11.2015	<input type="checkbox"/>									
UICORE	PS	Applications - PS	<input type="checkbox"/>	SAP 19.11.2015	<input type="checkbox"/>									
UICORE	QM	Applications - Quality Management	<input type="checkbox"/>	SAP 19.11.2015	<input type="checkbox"/>									
UICORE	SCM	Applications - SCM	<input type="checkbox"/>	SAP 19.11.2015	<input type="checkbox"/>									
UICORE	SD	Applications - SD	<input type="checkbox"/>	SAP 19.11.2015	<input type="checkbox"/>									
UIHR001	TRANSACTIONAL	Launchpad for HCM Fiori Transactional ap	<input type="checkbox"/>	SAP 27.05.2015	<input type="checkbox"/>									
UIILM001	DAAG_APPS	Data Aging Application	<input type="checkbox"/>	SAP 23.05.2014	<input type="checkbox"/>									
UIILM001	TRANSACTIONAL	ILM Data Archiving Job Mon lpd	<input type="checkbox"/>	SAP 01.04.2014	<input type="checkbox"/>									
UIMDC001	TRANSACTIONAL	Master Data Consolidation Launchpad	<input type="checkbox"/>	SAP 02.09.2014	<input type="checkbox"/>									
UIMDG001	TRANSACTIONAL	MDG Fiori Launchpad	<input type="checkbox"/>	SAP 21.10.2015	<input type="checkbox"/>									

 Figure 262: 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.

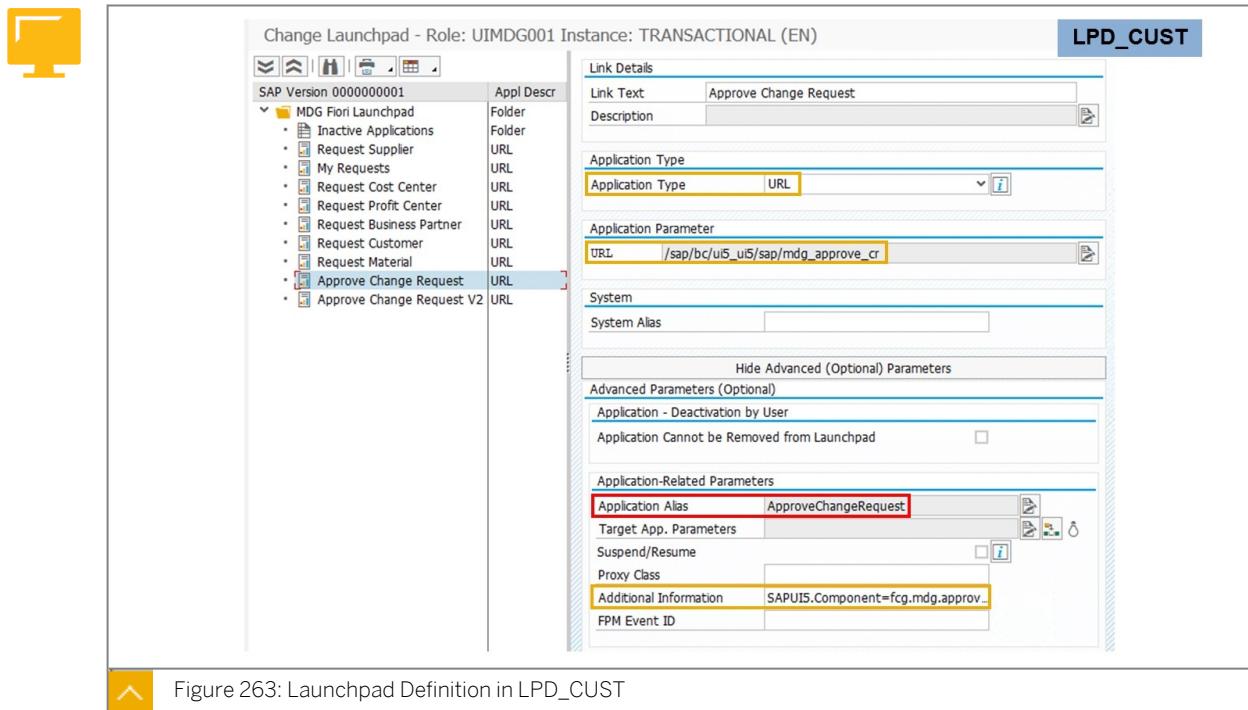


Figure 263: Launchpad Definition in LPD_CUST

Each link has an application type and, depending on the type, several additional parameters. The application type of SAPUI5 is URL. Here are the additional parameters:

- The URL of the ICF node
- The path to the component for the SAPUI5 app
- 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.

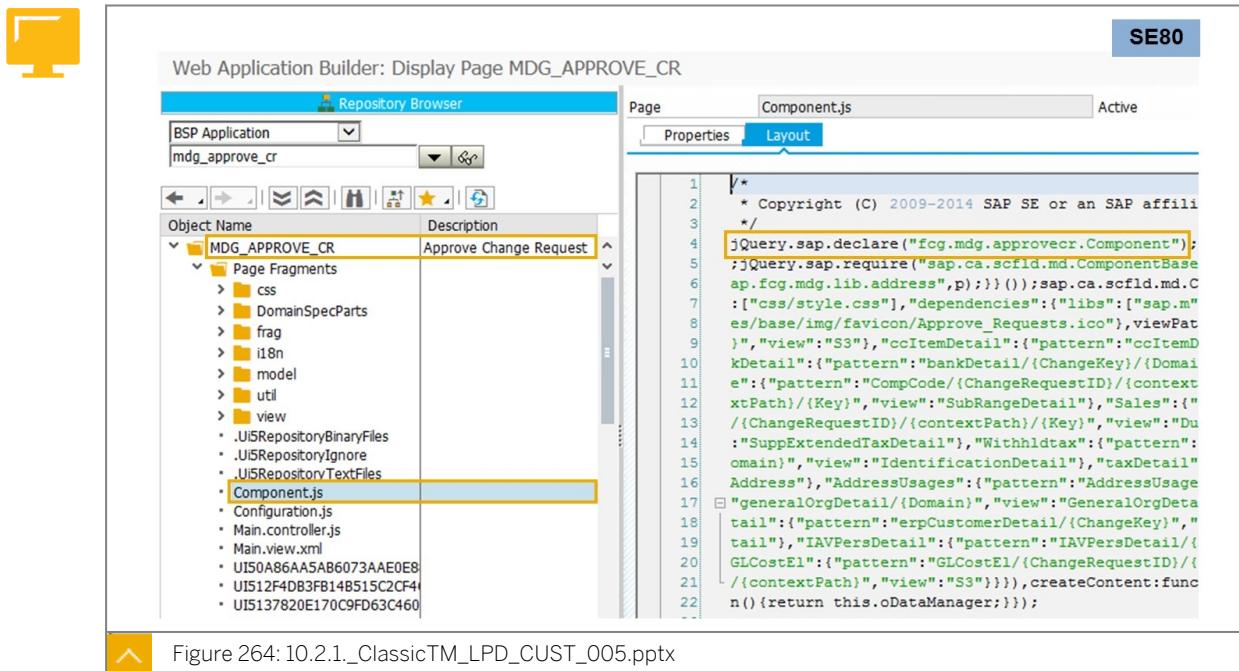


Figure 264: 10.2.1_ClassicTM_LPD_CUST_005.pptx

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`.

Web Dynpro Mapping Elements

The screenshot shows the SAP Fiori Launchpad. At the top right, it says "Upload Document for Bank Account [WDA_FCLM_BAM_UPLOAD_DOC (APP_CFG_FCLM_BAM_UPLOAD_DOC)]". Below this, the page title is "SAP Fiori Launchpad". A note says "You require the following data to give users access to the app in the SAP Fiori launchpad." The "Technical Configuration" section lists the following data:

System Alias	S4FIN
WebDynpro Application	WDA_FCLM_BAM_UPLOAD_DOC
WebDynpro Configuration	APP_CFG_FCLM_BAM_UPLOAD_DOC

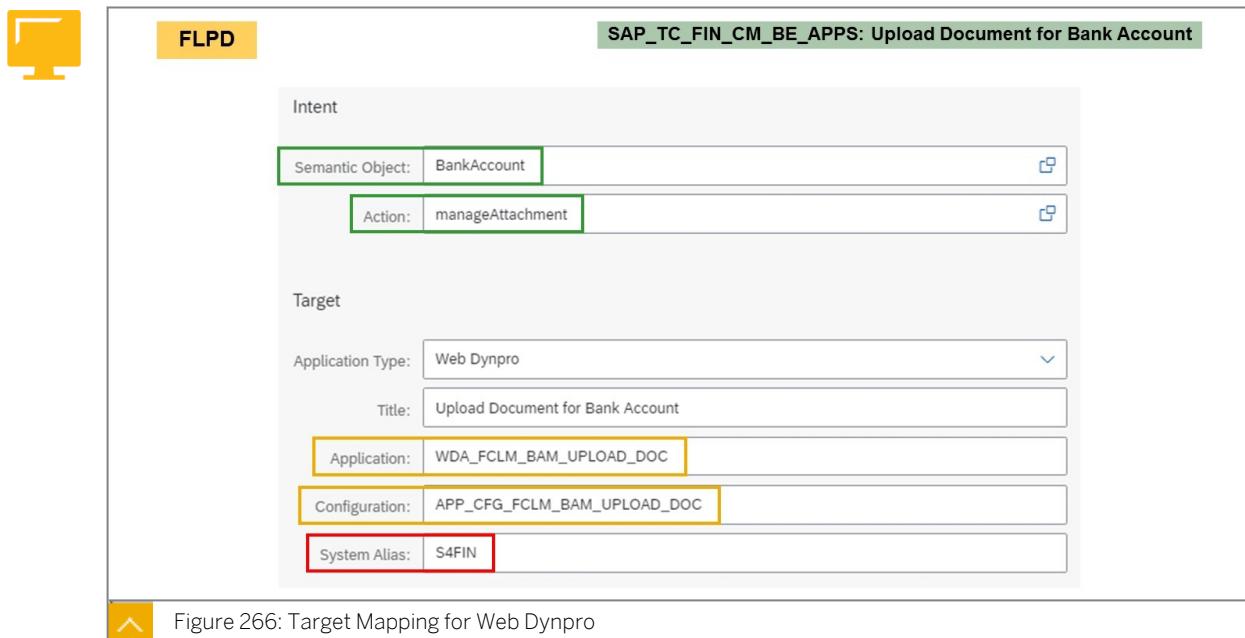
The "Target Mapping(s)" section shows a table:

Semantic Object	Semantic Action	Parameter Key	Parameter Value
BankAccount	manageAttachment	sap-system	S4FIN

In the screenshot, the "System Alias" row, the "WebDynpro Application" row, and the "WebDynpro Configuration" row are highlighted with orange boxes. The "Parameter Value" column for the target mapping row also has a red box around "S4FIN".

Figure 265: Technical Configuration for Web Dynpro

In this example from the *SAP Fiori apps 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.

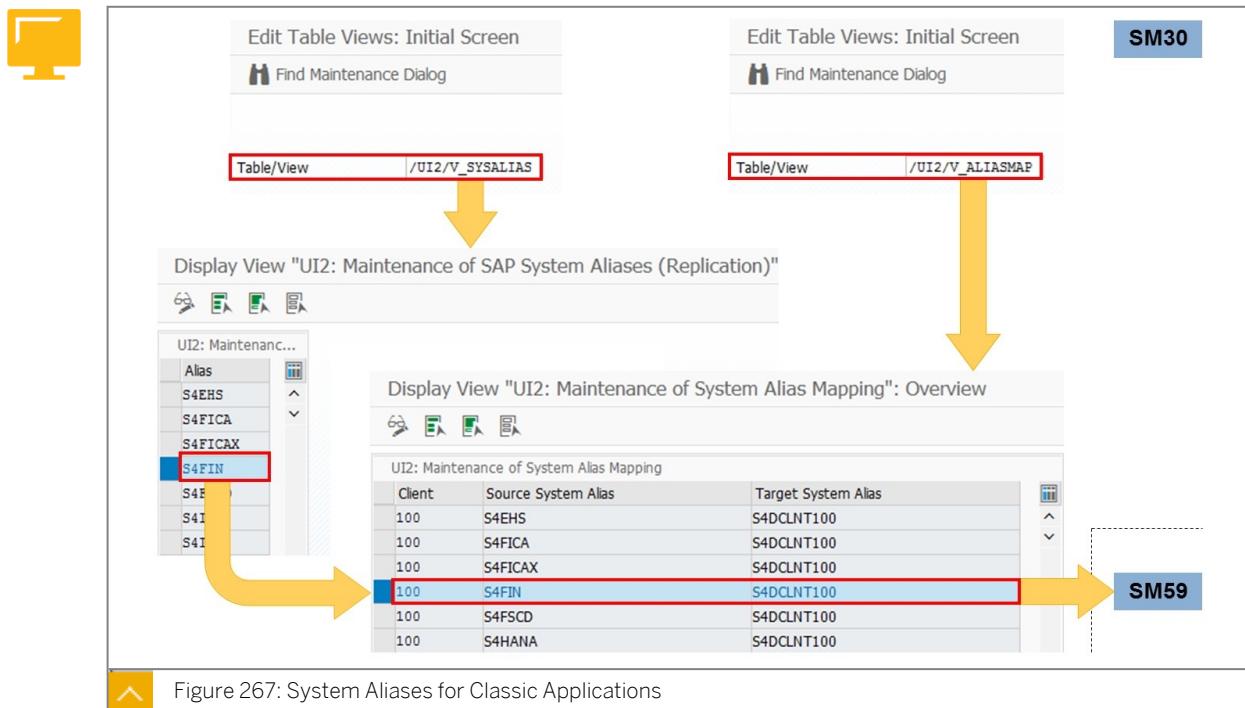


The intent can be found in the target mapping. As target the application type Web Dynpro is set. The Web Dynpro application and configuration is visible in the target mapping and in the system alias pointing to the system holding the application.

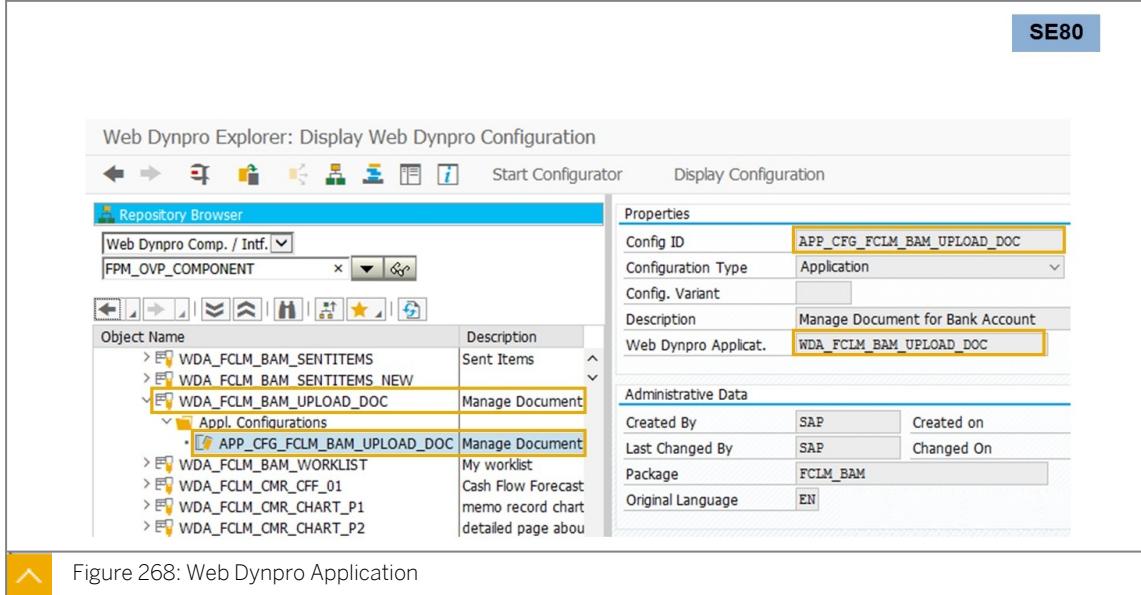


Caution:

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



System aliases for classic applications 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 such as 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.

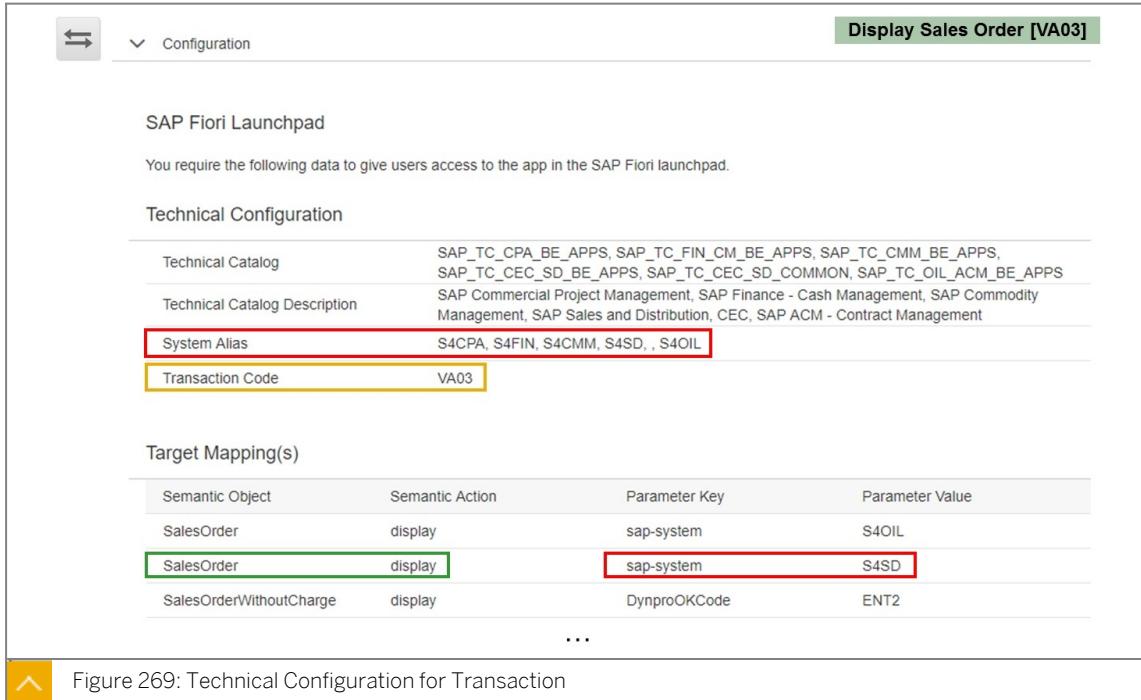


The screenshot shows the 'Web Dynpro Explorer: Display Web Dynpro Configuration' interface. In the top right corner, there is a blue button labeled 'SE80'. The main window has a toolbar with various icons. Below the toolbar is a 'Repository Browser' pane containing a tree view of objects under 'Web Dynpro Comp. / Intf.' and 'FPM_OVP_COMPONENT'. One node, 'WDA_FCLM_BAM_UPLOAD_DOC', is selected and highlighted in yellow. To the right of the tree view is a 'Properties' pane. The 'Config ID' field is set to 'APP_CFG_FCLM_BAM_UPLOAD_DOC'. The 'Configuration Type' is 'Application'. The 'Description' is 'Manage Document for Bank Account'. The 'Web Dynpro Applat.' is 'WDA_FCLM_BAM_UPLOAD_DOC'. The 'Object Name' column lists several nodes with their descriptions: 'WDA_FCLM_BAM_SENTITEMS' (Sent Items), 'WDA_FCLM_BAM_SENTITEMS_NEW' (Manage Document), 'WDA_FCLM_BAM_UPLOAD_DOC' (Manage Document), 'WDA_FCLM_BAM_WORKLIST' (My worklist), 'WDA_FCLM_CMRA_CFF_01' (Cash Flow Forecast), 'WDA_FCLM_CMRA_CHART_P1' (memo record chart), and 'WDA_FCLM_CMRA_CHART_P2' (detailed page abou). The 'Properties' pane also includes sections for 'Administrative Data' with fields like 'Created By' (SAP), 'Last Changed By' (SAP), 'Package' (FCLM_BAM), and 'Original Language' (EN).

Figure 268: Web Dynpro Application

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

Transaction Mapping Elements



The screenshot shows the 'SAP Fiori Launchpad' configuration for 'Display Sales Order [VA03]'. At the top left is a yellow monitor icon. The top right shows the transaction code 'VA03'. The main area is titled 'Configuration' with a back arrow icon. Below it is a section for 'Technical Configuration' with the following details:

- Technical Catalog: SAP_TC_CPA_BE_APPS, SAP_TC_FIN_CM_BE_APPS, SAP_TC_CMM_BE_APPS, SAP_TC_CEC_SD_BE_APPS, SAP_TC_CEC_SD_COMMON, SAP_TC_OIL_ACM_BE_APPS
- Technical Catalog Description: SAP Commercial Project Management, SAP Finance - Cash Management, SAP Commodity Management, SAP Sales and Distribution, CEC, SAP ACM - Contract Management
- System Alias: S4CPA, S4FIN, S4CMM, S4SD, , S4OIL (highlighted with a red box)
- Transaction Code: VA03 (highlighted with a yellow box)

Below this is a 'Target Mapping(s)' table:

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

Figure 269: Technical Configuration for Transaction

In this example from the SAP Fiori apps 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.

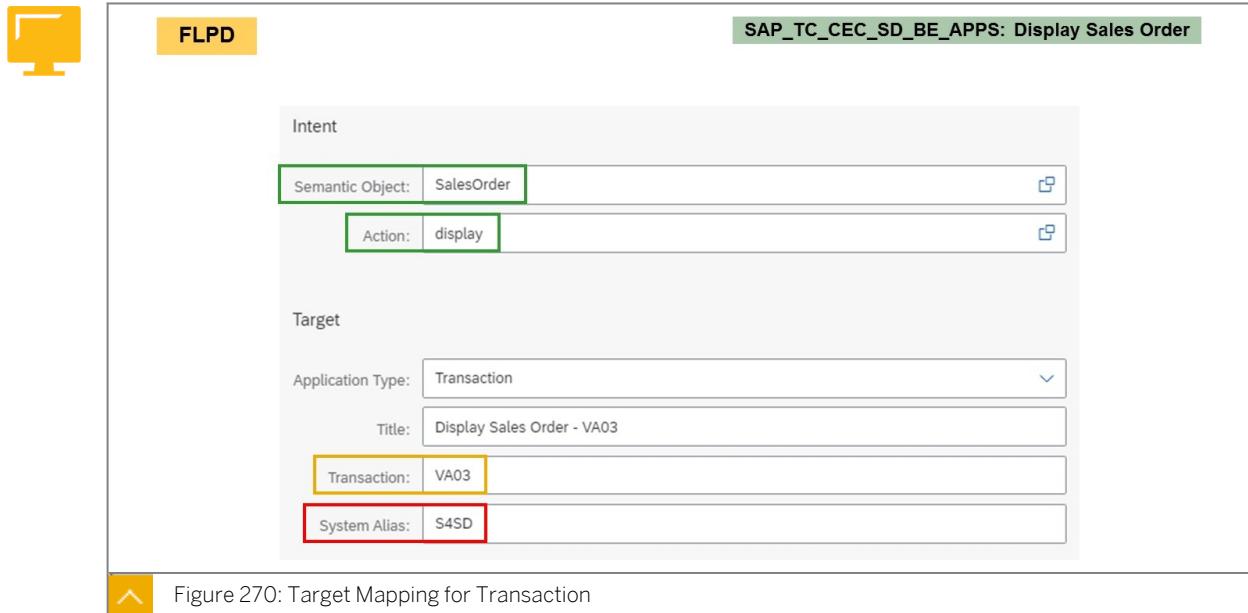


Figure 270: Target Mapping for Transaction

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

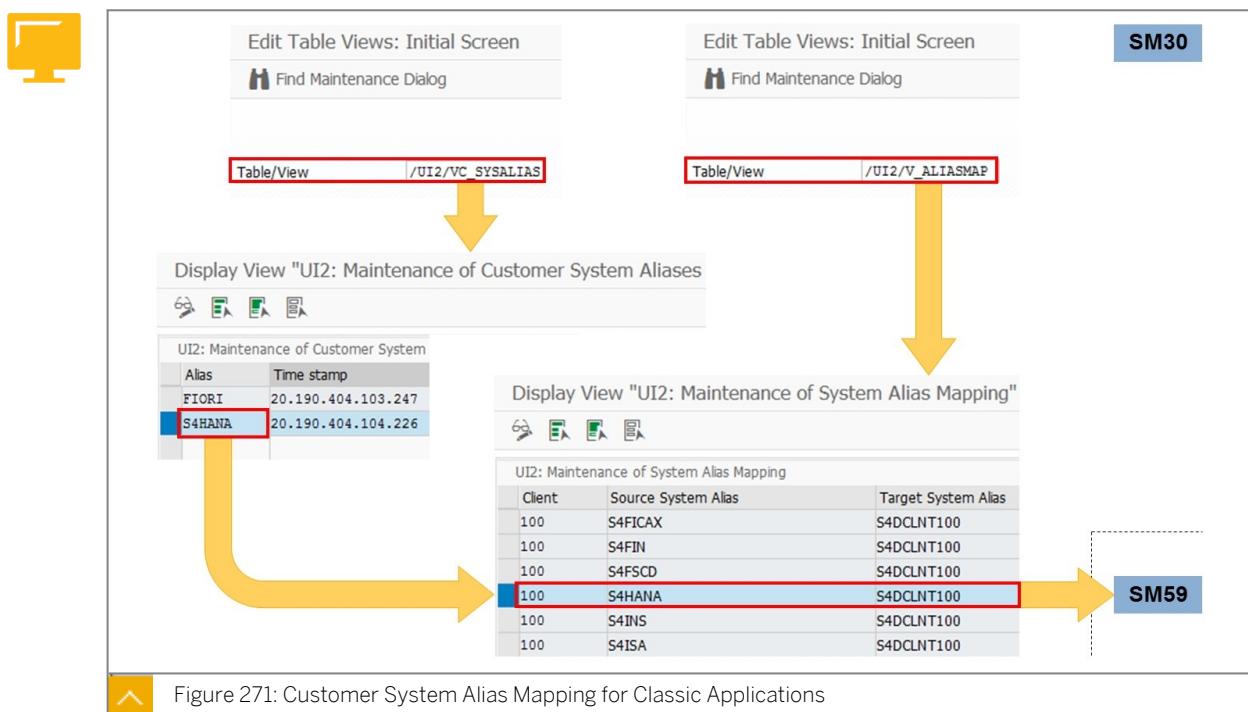
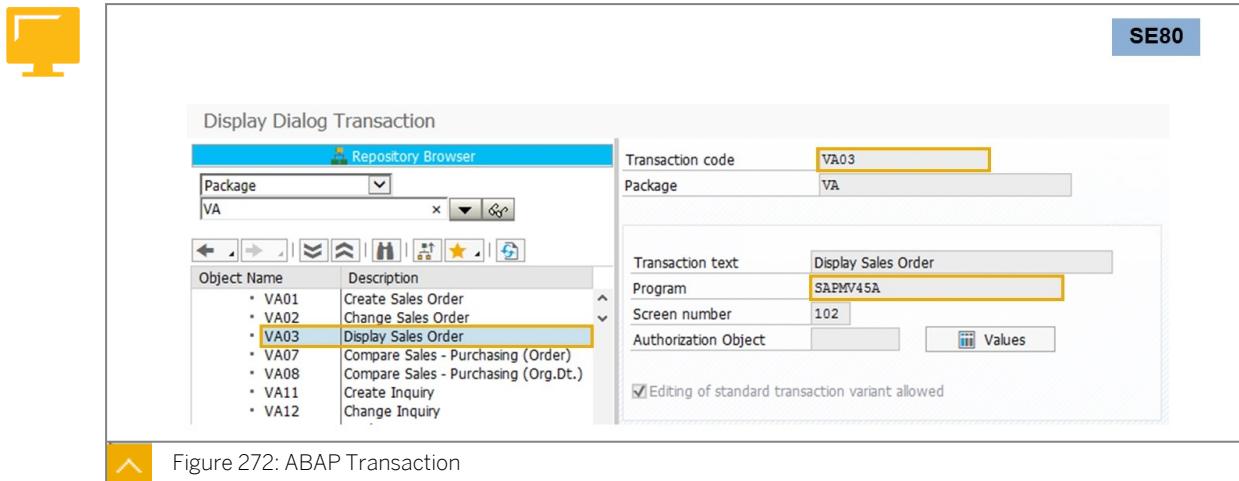


Figure 271: Customer System Alias Mapping for Classic Applications

Transactions are — such as Web Dynpro applications — classic applications and, therefore, use the same system aliases viewable in the maintenance view /UI2/V_ALIASMAP. 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 LPD_CUST target mappings
- Create Web Dynpro target mappings
- Create transaction target mappings

Unit 10

Lesson 3

Configuring Analytical Applications

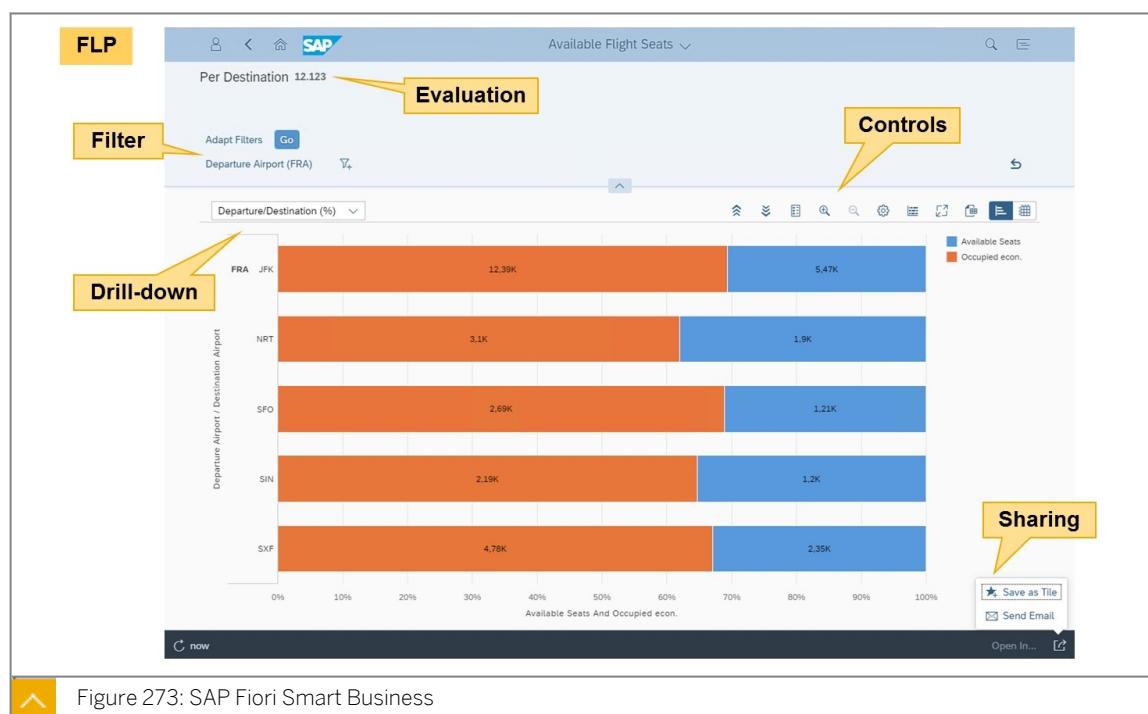


LESSON OBJECTIVES

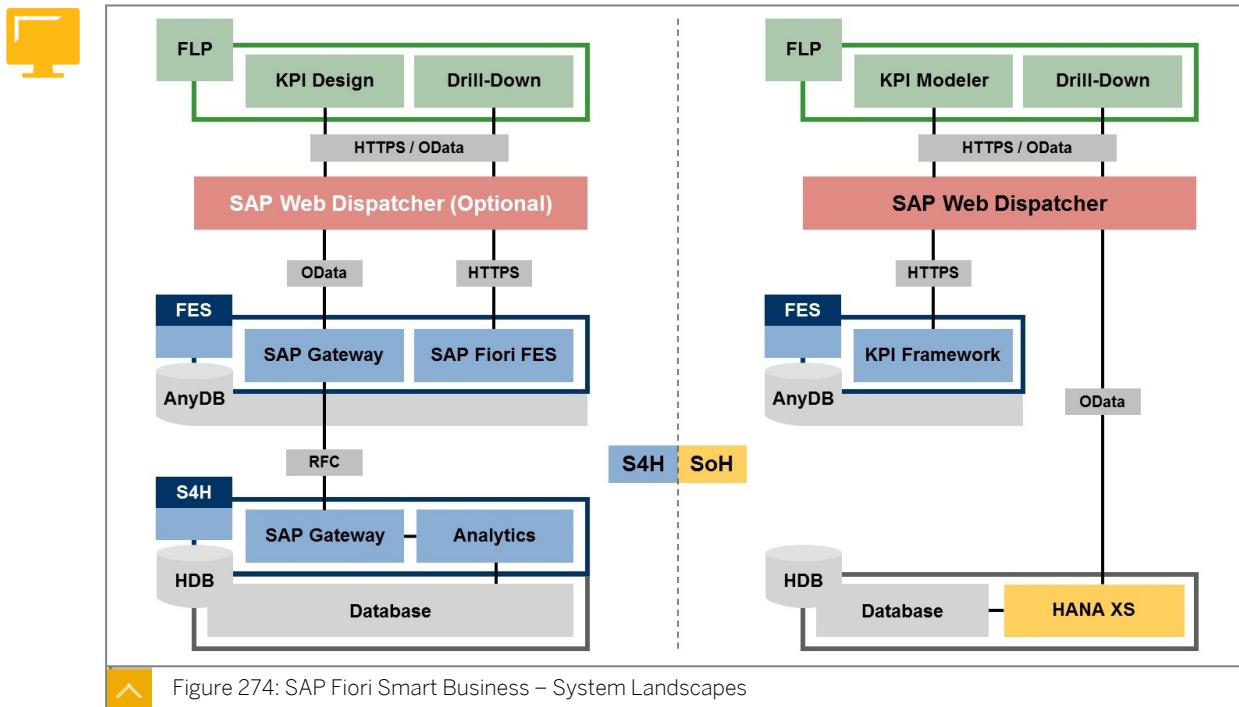
After completing this lesson, you will be able to:

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

SAP Fiori Smart Business



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 SAP 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. Therefore, an SAP Web Dispatcher is mandatory, and provides routing rules as well as additional users and authorizations in SAP HANA.

The screenshot shows the SAP Fiori Smart Business configuration interface. At the top, there are tabs for S4H (selected) and SoH. A green button labeled "Create KPI [F0817]" is visible. Below the tabs, a section titled "Configuration" is expanded, with a note stating: "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 (–), Technical PFCG Role (–), Semantic Object (FioriApplication), Action (createSBKPIIS4HANA), and SAPUI5 Application (SB_APPS_KPIS1). 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). To the right of these fields, corresponding URLs are listed: /UI2/SAP_KPIMOD_TC_S, /UI2/SAP_KPIMOD_TCG_S, /UI2/SAP_KPIMOD_TCR_S, FioriApplication, createSBKPI, –, –, and –.

Figure 275: SAP Fiori Smart Business – Configurations

In this example from the *SAP Fiori apps 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), SoH, and FLP. The main area is titled "KPI Design / KPI Modeler" and contains a grid of six tiles. The tiles are: "KPI Workspace" (S/4HANA), "Create KPI" (S/4HANA), "Create Evaluation" (S/4HANA), "Create Tile" (S/4HANA), "Configure Drill-Down" (S/4HANA), and "Manage KPI Associations" (S/4HANA). To the right of this grid, there is a separate column titled "KPI Modeler" containing two tiles: "Manage KPI Authorizations" (SoH) and "Migration Tool" (SoH). The "Manage KPI Authorizations" tile has a note: "Grant access to evaluations".

Figure 276: KPI Design / KPI Modeler

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 necessary 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 SAP Fiori catalogs. In an SoH environment, additional authorizations in SAP HANA are necessary. 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, 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 UISKPIO1 is no longer supported.



The screenshot shows the SAP Fiori Launch Pad (FLP) interface. A yellow arrow points from the FLP icon at the top left to the 'Analytical Queries' catalog. Inside the catalog, a second yellow arrow points from the main view to the 'View Details' screen of a specific query. The 'View Details' screen displays the query details, including its name, application component, and annotations.

Column Name	Data Element	Column Description	Column Data Type	Length
AIRPORTFROM	S_FROMAIRP	Departure airport	CHAR	3
AIRPORTTO	S_TOAIRP	Destination airport	CHAR	3
FLIGHTDATE	S_DATE	Flight date	DATS	8
AIRLINE	S_CARR_ID	Airline Code	CHAR	3
FLIGHTCONNECTION	S_CONN_ID	Flight Connection Number	NUMC	4
AIRCRAFTTYPE	S_PLANETYE	Aircraft Type	CHAR	10
CURRENCY	S_CURRCODE	Local currency of airline	CUKY	5
MAXIMUMNUMBEROFESEATS	S_SEATSMAX	Maximum capacity in economy class	INT4	10
NUMBEROFCOCKUPIEDSEATS	S_SEATSOCC	Occupied seats in economy class	INT4	10
NUMBEROFAVAILABLESEATS			INT4	10
FLIGHTPRICE	S_PRICE	Airfare	CURR	15

Annotation details from the 'View Details' screen:

- ABAPCATALOG.SQVIEWNAME: 'CUX100FLBYAIRPO'
- ANALYTICS QUERY: true
- ENDUSERTEXT LABEL: 'UX100: Flights by Airport Query'
- ODATA.PUBLISH: true
- VDM.VIEWTYPE: #CONSUMPTION



Figure 277: 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 screenshot shows the SAP KPI Workspace interface. On the left, a sidebar lists various KPIs under the heading 'All KPIs (183)'. The KPIs listed include 'Available Flight Seats', 'Excess Component Consumption', 'Production Execution Duration', 'Scrap Reason', 'Operation Scrap', 'Material Scrap', 'Manage Commodity Order Requests', and 'Manage Fill Packets'. Each KPI entry shows a count of evaluations (e.g., 1). A yellow box labeled 'KPIs' points to this sidebar. On the right, a detailed view of the 'Available Flight Seats' KPI is shown. It includes tabs for 'Information', 'Data Source', and 'Associations'. The 'Data Source' tab is active, displaying details like 'Description: Number of available flight seats', 'Tags:', 'Properties:', 'Goal Type: Minimizing', and 'Additional Languages: Additional Languages'. Below this, the 'Associations' tab shows a table with one row: 'Evaluations (1)', 'Evaluation', 'Status', and 'Description'. The 'Evaluation' column shows 'Per Destination', 'Status' is 'Active', and 'Description' is 'Number of available flight seats'. A yellow box labeled 'Associations' points to the associations table. At the bottom, there are buttons for 'Edit', 'Copy', 'Add Evaluation', 'Add as Favorite', and 'Delete'. A yellow box labeled 'Evaluations' points to the 'Evaluations (1)' section. The top right corner of the main area has a 'Details' button.

Figure 278: KPI Workspace

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.

The screenshot shows the SAP KPI Evaluation interface. On the left, a sidebar titled 'Per Destination' shows a table for 'Available Flight Seats' with three rows: 'Value 1' (1550M), 'Value 2' (219.2M), and 'Value 3' (66.46M). A yellow box labeled 'Thresholds' points to this table. On the right, there are sections for 'Input Parameters and Filters', 'Navigation Intents', and 'Tiles'. The 'Tiles' section displays a tile for 'Available Flight Seats Per Destination' with a value of '0.0 M' and a unit of 'USD, Actual'. A yellow box labeled 'Tiles' points to this tile. Below the tiles, there is a 'Drill-Down Configuration' section. A yellow box labeled 'Drill-Downs' points to this section. The top right corner of the main area has a 'Details' button.

Figure 279: KPI Evaluation

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 SAP Business Suite and at least two additional ones in S4H. They are defined as part of an evaluation and saved in a SAP 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.

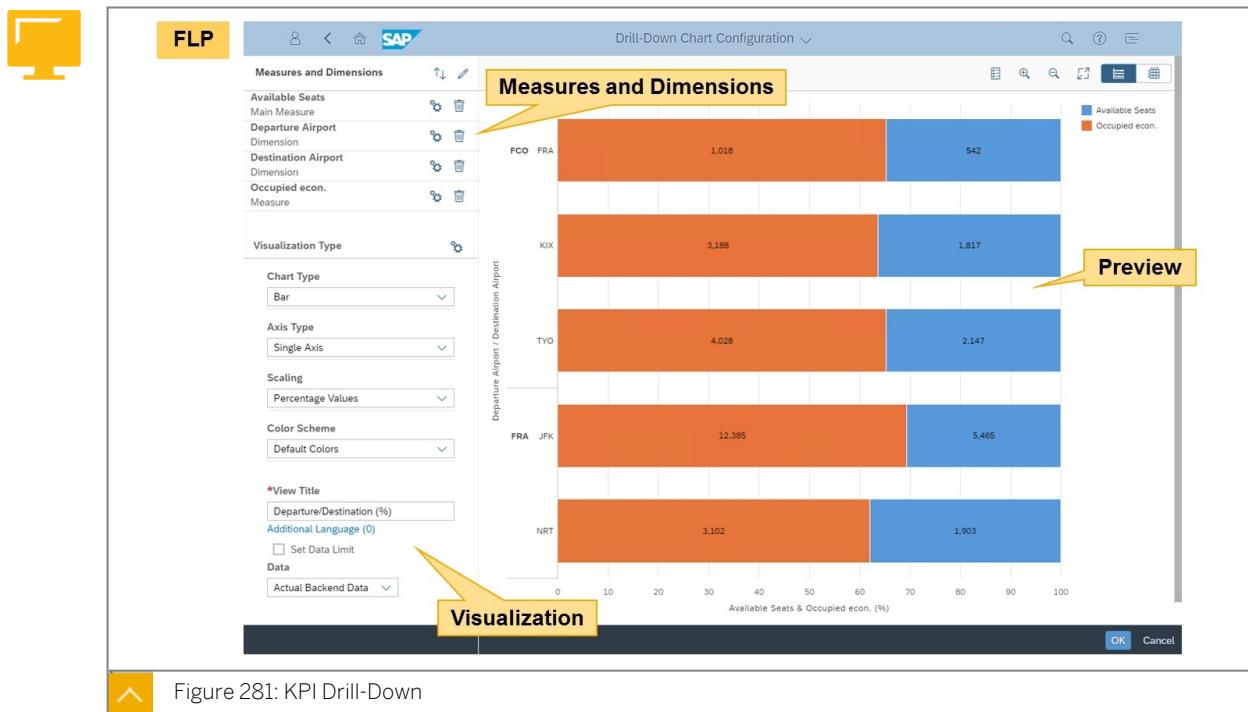


Figure 281: KPI Drill-Down

The generic KPI drill-down enables customers to visualize their business data in an SAP Fiori 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.

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

Integrating SAP Fiori in SAP Enterprise Portal

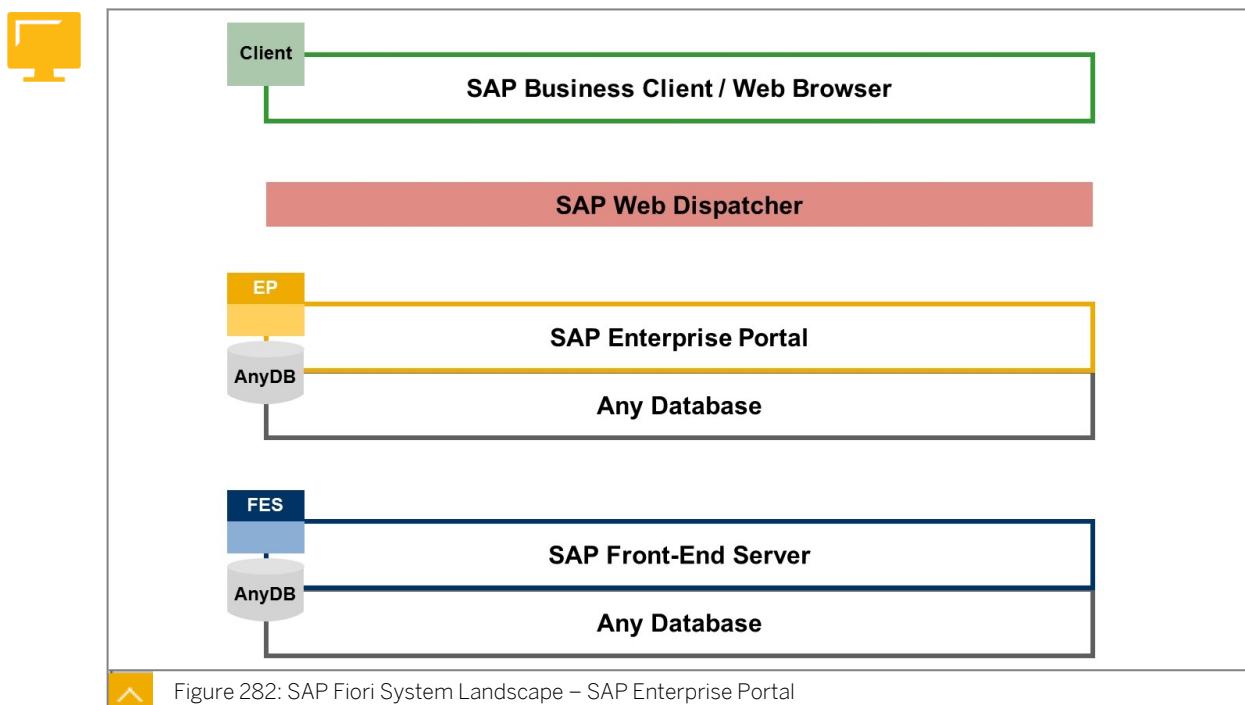


LESSON OBJECTIVES

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. However, it is still the source of all data for SAP Fiori apps.

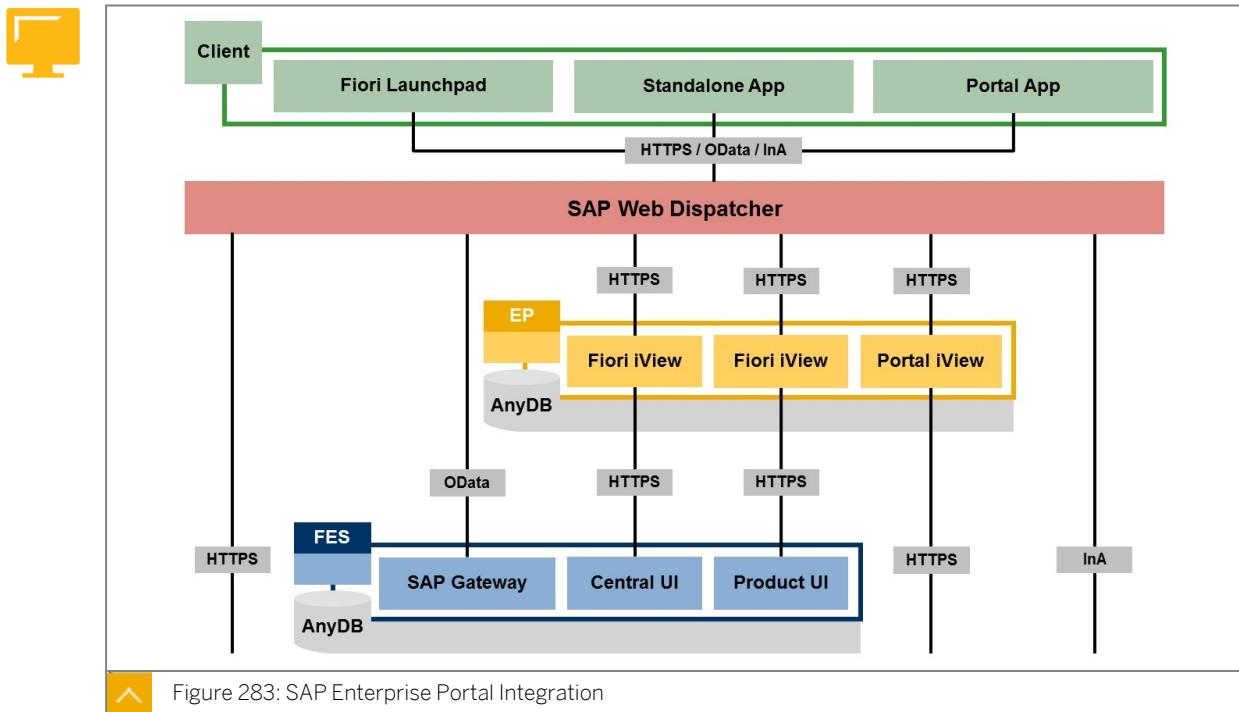


Figure 283: SAP Enterprise Portal Integration

Apps running in the client can be distinguished as follows: the *SAP Fiori launchpad*, a SAP 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 SAP Web Dispatcher. The FLP and SAP Fiori standalone apps can be embedded in the EP through SAP Fiori iViews.

SAP Fiori iView Mode

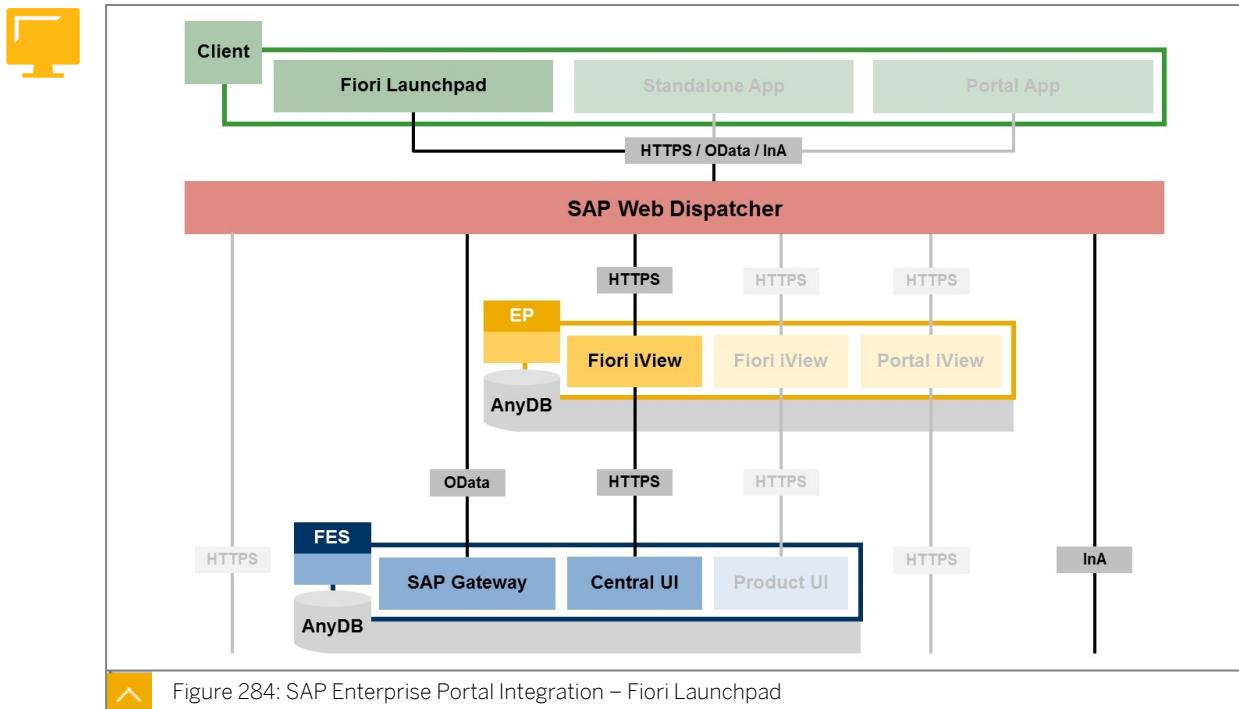
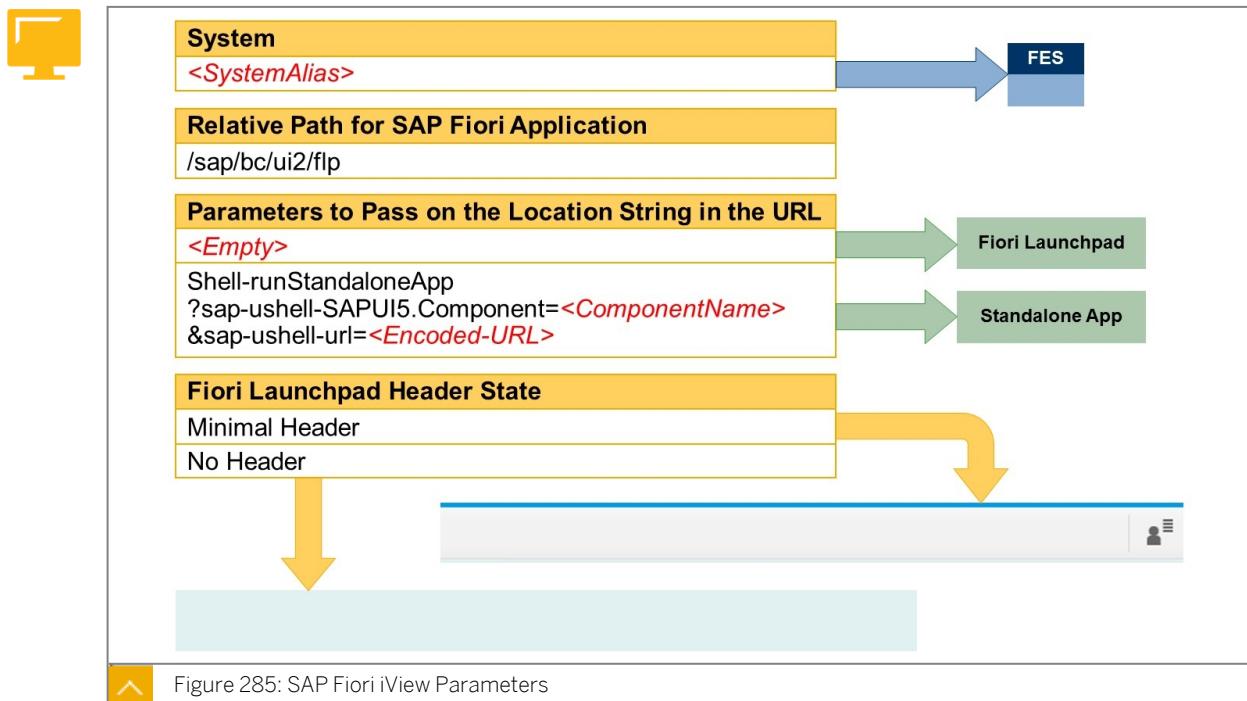


Figure 284: SAP Enterprise Portal Integration – Fiori Launchpad

When using the FLP in the client, OData and InA requests are routed to the FES and BES. However, the UI is embedded in a SAP 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 SAP Fiori standalone, a SAP 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 SAP Fiori app without any FLP frame.

SAP Fiori Standalone Mode

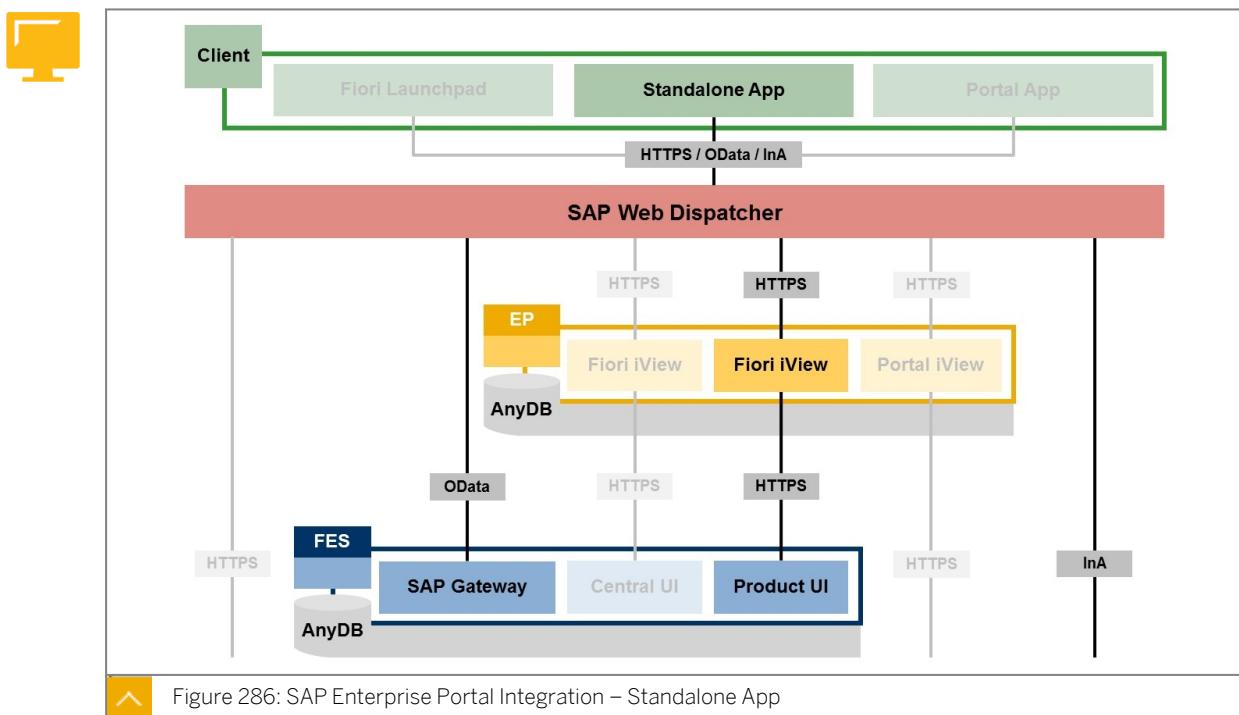


Figure 286: SAP Enterprise Portal Integration – Standalone App

You can embed a single SAP 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 SAP 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.

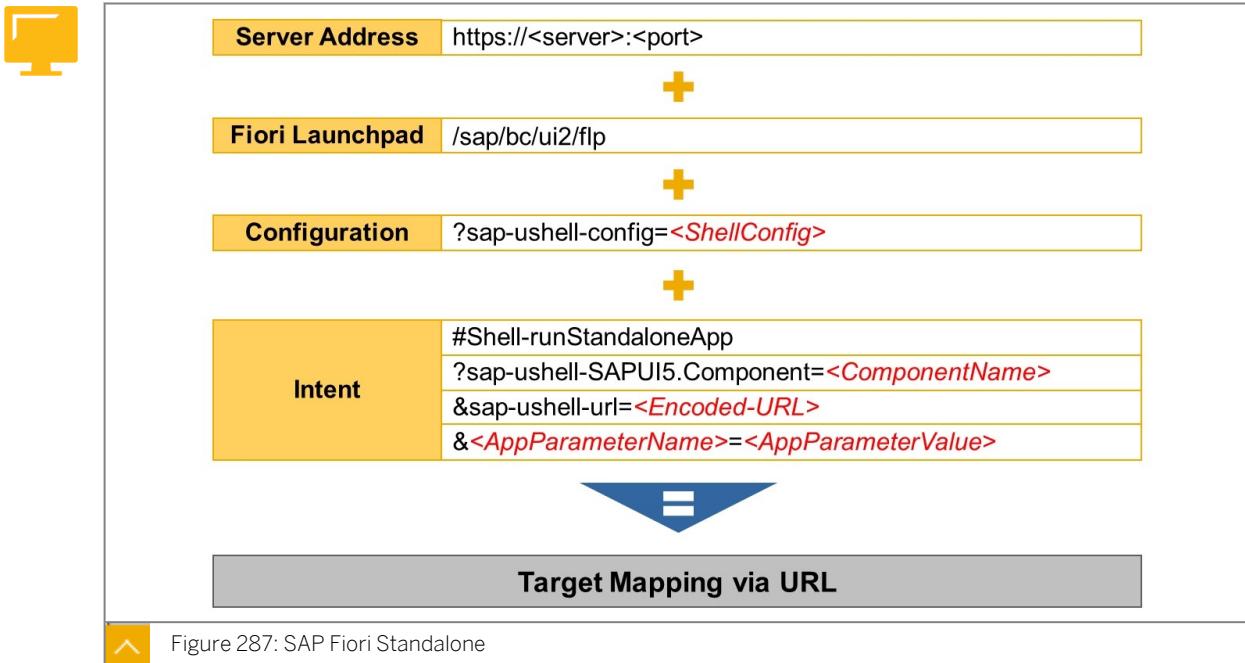


Figure 287: SAP Fiori Standalone

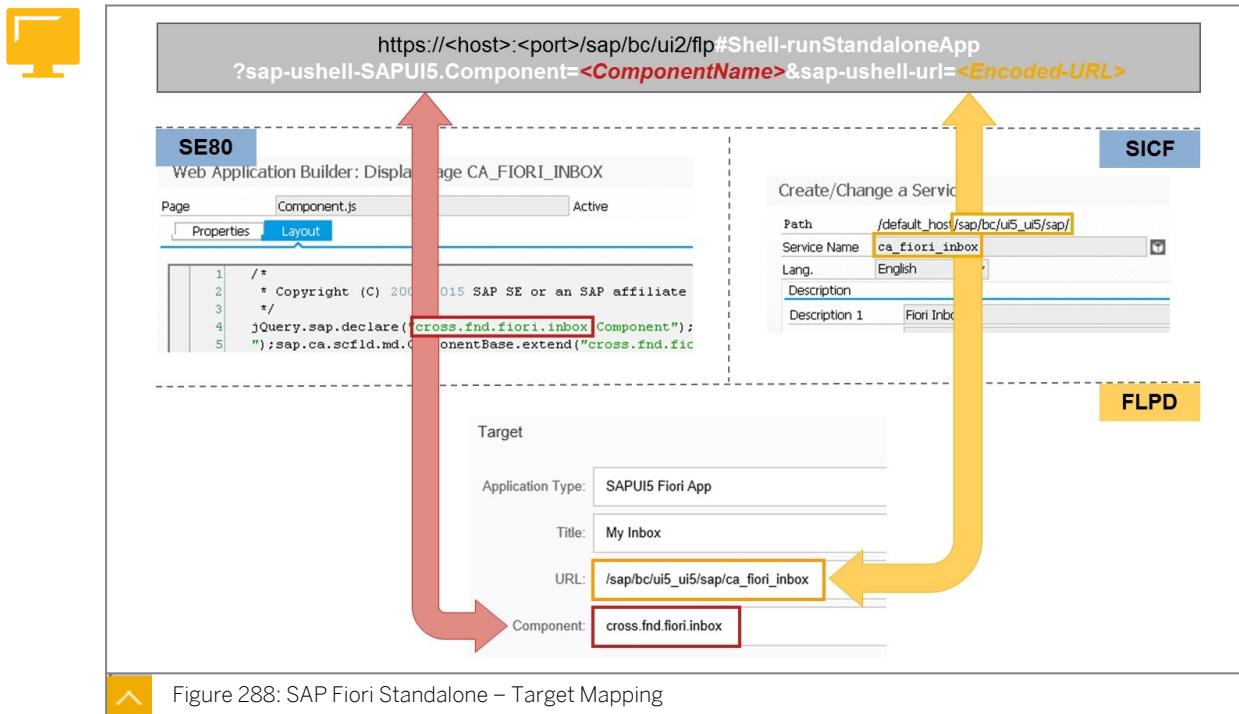
Calling a SAP Fiori app in standalone mode can be done by building a URL containing all information that is normally defined as a target mapping in a SAP Fiori catalog, that is, the SAPUI5 component, the ICF node, and optional parameters of the app on the FES. These are set as URL parameters of the `#Shell-runStandaloneApp`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



Note:

As some of the features that have been introduced for the *SAP Fiori launchpad* since then do not support this standalone mode, it is recommended to use it in development scenarios only. Most notably, this applies to cache busting and the application index dependency management.

Specifying the path to the application component as a URL parameter does not allow 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.

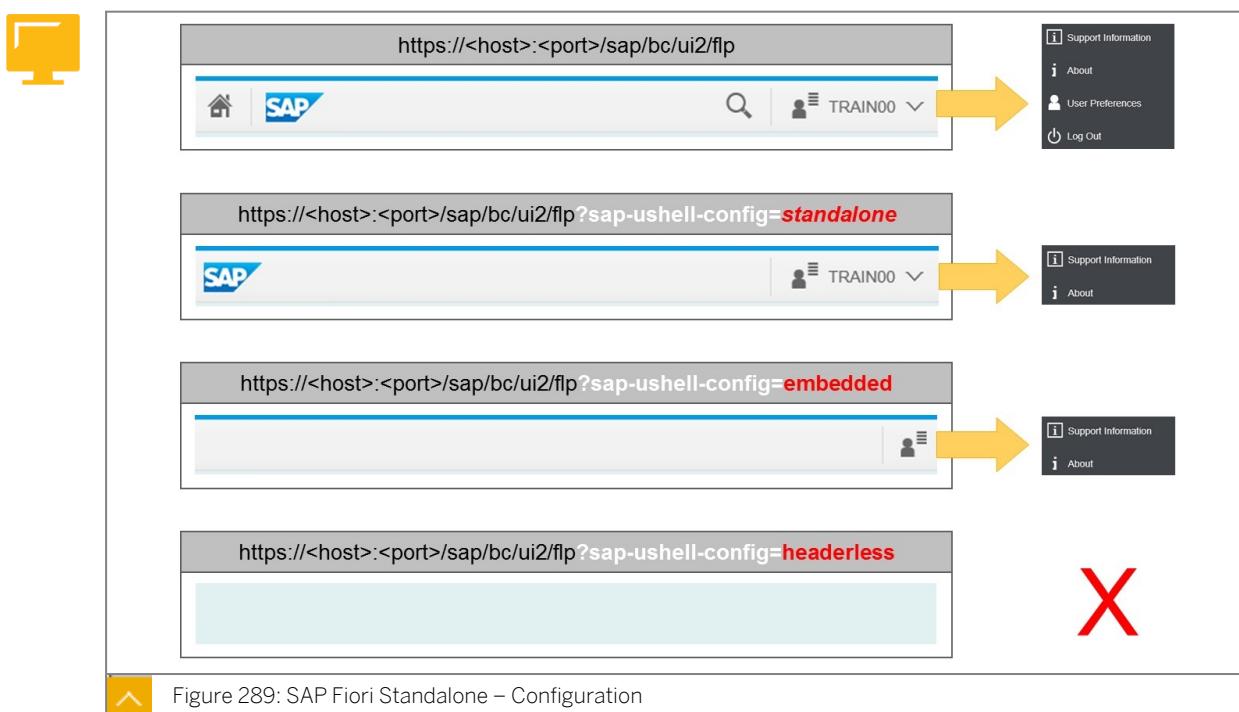


The elements of the target mapping can be taken from the system getting the path for the SAPUI5 component from the BSP application using SE80 and the path of the ICF node from the app using SICF. If the app is delivered by SAP, another option is to search for the suitable target mapping in the SAP Fiori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



Note:

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



When calling a SAP Fiori app standalone, you can configure the appearance and functions of the FLP header bar by with 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 SAP Fiori iView.

headerless

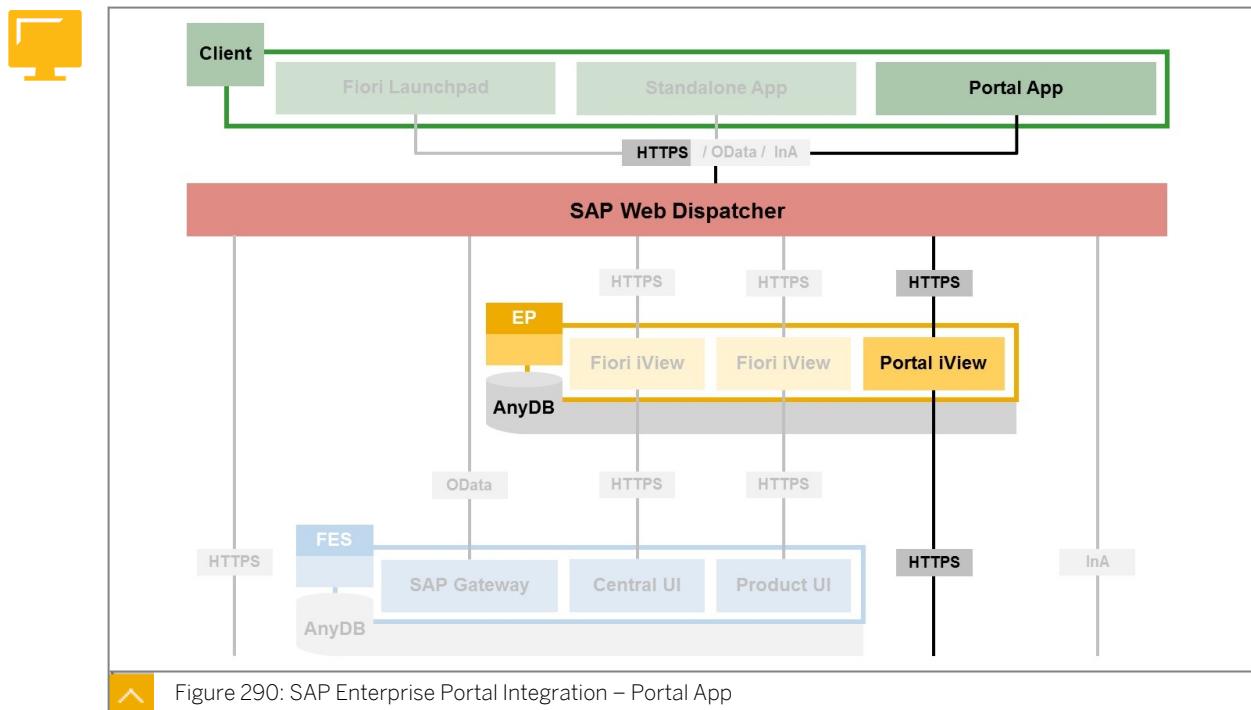
The header bar disappears completely showing only the app. Headerless matches the **No Header** value for the *Fiori Launchpad Header State* parameter of a Fiori iView.



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 SAP Fiori. In particular, if there is already an EP available in the customer landscape, all configured iViews and other technologies can be combined with SAP Fiori. In addition, Web Dynpro and ABAP transactions can be made available for users using portal configurations instead of SAP 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 10

Lesson 5

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 Business Suite interface for managing SAP Workflow Apps. It includes sections for Selected Apps, SAP Fiori Launchpad, ICF Nodes for SAPUI5 Applications, and OData Services.

Selected Apps:

Leading Product Version	App Name	Action
SAP Fiori for request approvals 1.0	Approve Requests	> X
SAP Fiori for request approvals 2.0	My Inbox	> X

SAP Fiori Launchpad:

Technical Catalog	PFCG Role for Technical Catalog	App Name	Semantic Object	Action
SAP_FND_TC_T	SAP_FND_TCR_T	My Inbox	WorkflowTask	displayInbox
SAP_FND_TC_T_X1	SAP_FND_TCR_T_X1	Approve Requests		approveRequests

ICF Nodes for SAPUI5 Applications:

SAPUI5 Application	App Name
CA_ALL_APV	Approve Requests
CA_FIORI_INBOX	My Inbox

OData Services:

oData Service	Version	App Name
IWPGW/TASKPROCESSING	0001	Approve Requests
	0002	My Inbox

Figure 291: SAP Workflow Apps – SAP Business Suite

One of the first integrations of SAP Fiori was with SAP Workflow. Many apps are available that are in some way connected 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, which 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 similar to 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). The app uses an updated version of the central workflow Gateway service /IWPGW/TASKPROCESSING. This Gateway service already provided SAP Workflow data before SAP Fiori, and can be installed using the IW_PGW add-on. During 2015, both apps were offered in parallel. However, currently there is no further ongoing development for *Approve Requests*.

The screenshot shows the SAP Workflow App interface for SAP S/4HANA. In the top right corner, there is a button labeled "My Inbox [F0862]". The main area is divided into sections:

- Selected Apps:** A table showing app configurations. One row is highlighted with a green border: "SAP Fiori for SAP S/4HANA 1511" under "App Name" and "My Inbox" under "First Release". Another row is marked with a red warning icon: "SAP Fiori for request approvals 1.0" under "App Name" and "Approve Requests" under "First Release".
- Aggregated Configuration Requirements:** A table showing requirements. One row is highlighted with a green border: "SAP_FND_TC_T" under "Technical Catalog" and "My Inbox" under "App Name".
- ICF Nodes for SAPUI5 Applications:** A table showing ICF nodes. One row is highlighted with a green border: "CA_FIORI_INBOX" under "SAPUI5 Application" and "My Inbox" under "App Name".
- OData Services:** A table showing OData services. One row is highlighted with a green border: "/WPGW/TASKPROCESSING" under "oData Service" and "0002" under "Version".

Figure 292: SAP Workflow App – SAP S/4HANA

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 SAP S/4HANA instead of an add-on as before. The same applies for the SAP Gateway workflow service, which is now also part of the software component SAP Fiori Front-End Server 2.0, such as the *My Inbox* app.

My Inbox

The screenshot shows the configuration dialog for the "My Inbox" app. It has several tabs:

- General:** Fields include Title ("My Inbox"), Subtitle ("All Items"), Keywords ("Workflow, Inbox, Approval"), and Icon ("sap-icon://complete").
- Dynamic Data:** Fields include Service URL ("/sap/opu/odata/IWPGW/TASKPROCESSING;v=2;m...") and Refresh Interval in Seconds ("0").
- Navigation:** Fields include Use semantic object navigation (checkbox checked), Semantic Object ("WorkflowTask"), Action ("displayInbox"), and Parameters ("allItems=true").
- Intent:** Fields include Semantic Object ("WorkflowTask") and Action ("displayInbox").
- Target:** Fields include Application Type ("SAPUI5 Fiori App"), Title ("My Inbox"), URL ("sap/bc/ui5/ui5/sap/ca_fiori_inbox"), and ID ("cross.fnd.fiori.inbox").

Figure 293: My Inbox Configuration

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.

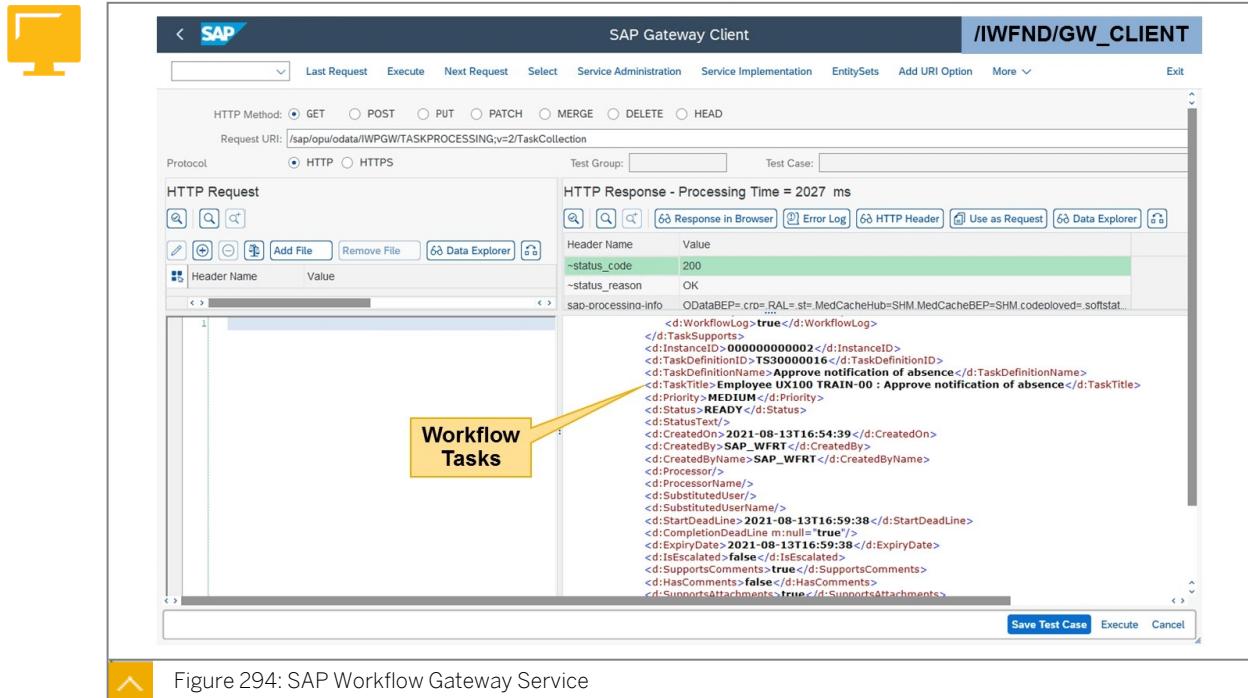


Figure 294: SAP Workflow Gateway Service

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.

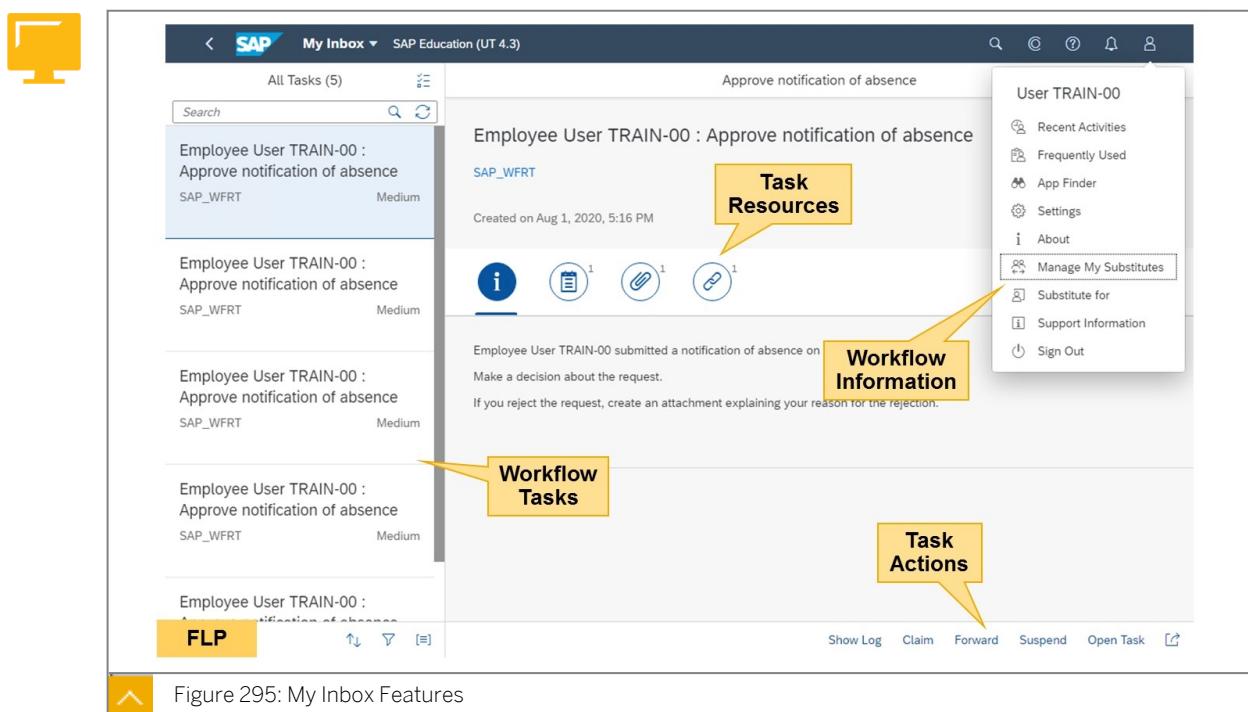
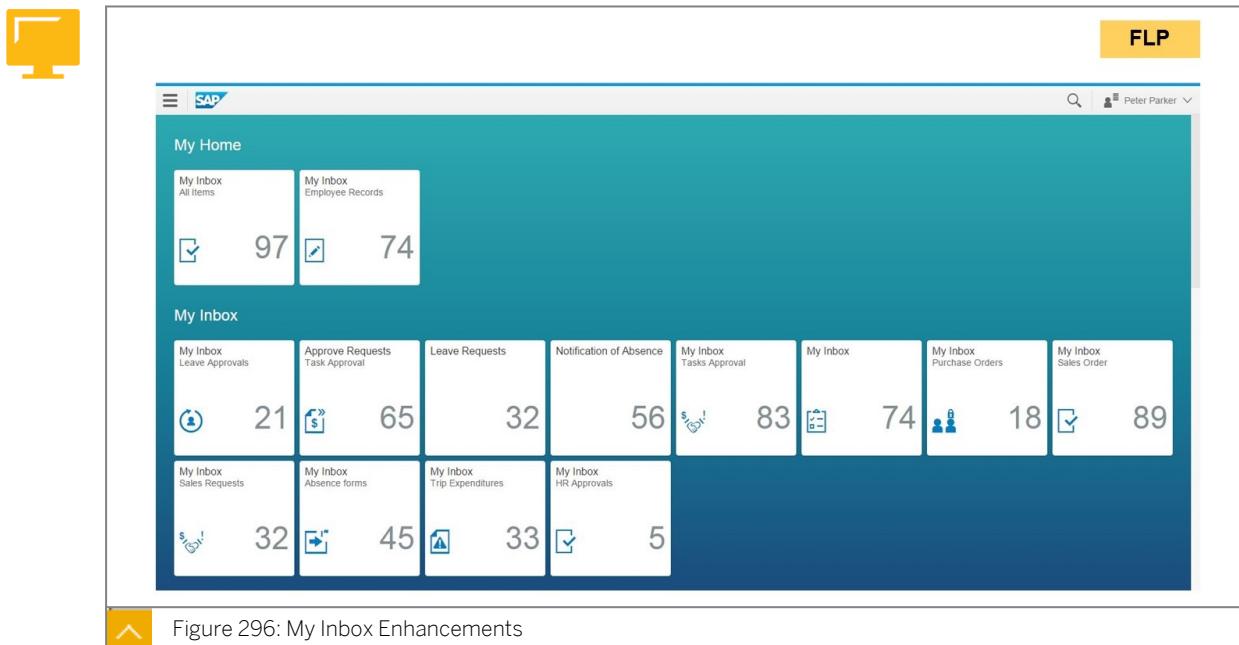


Figure 295: My Inbox Features

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 SAP Fiori.



By using SAPUI5 extensions, you can 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 such as 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

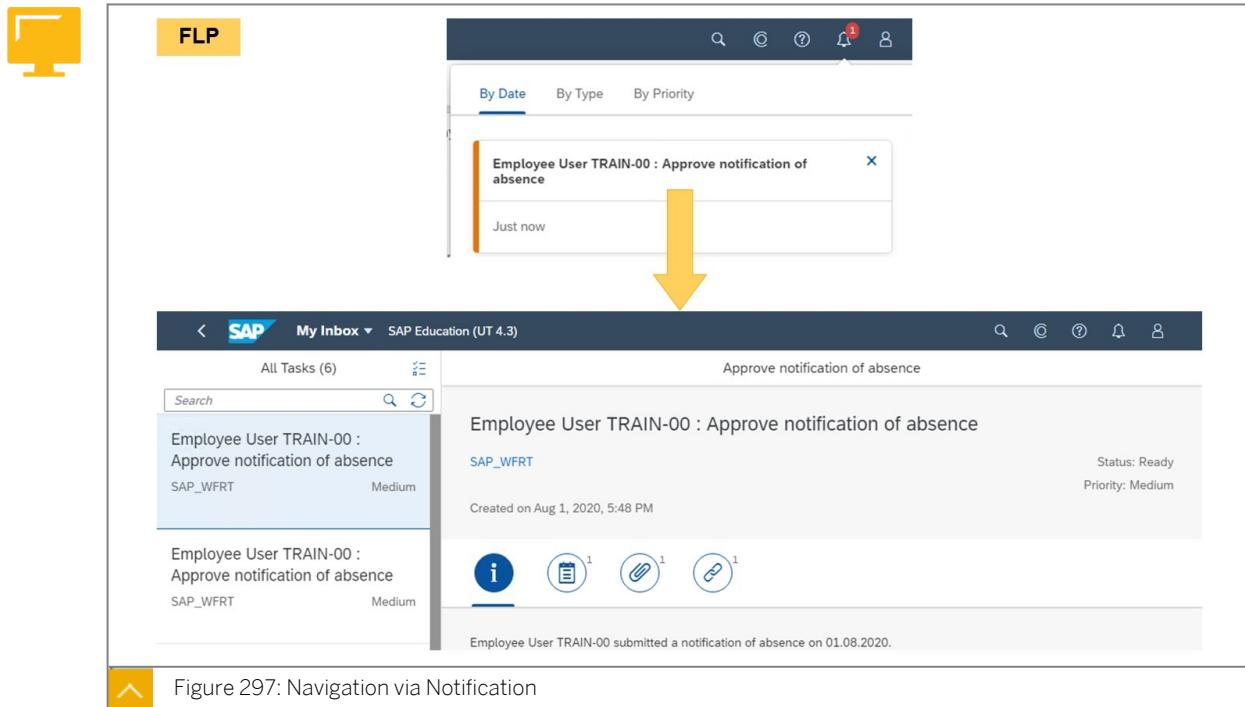


Note:

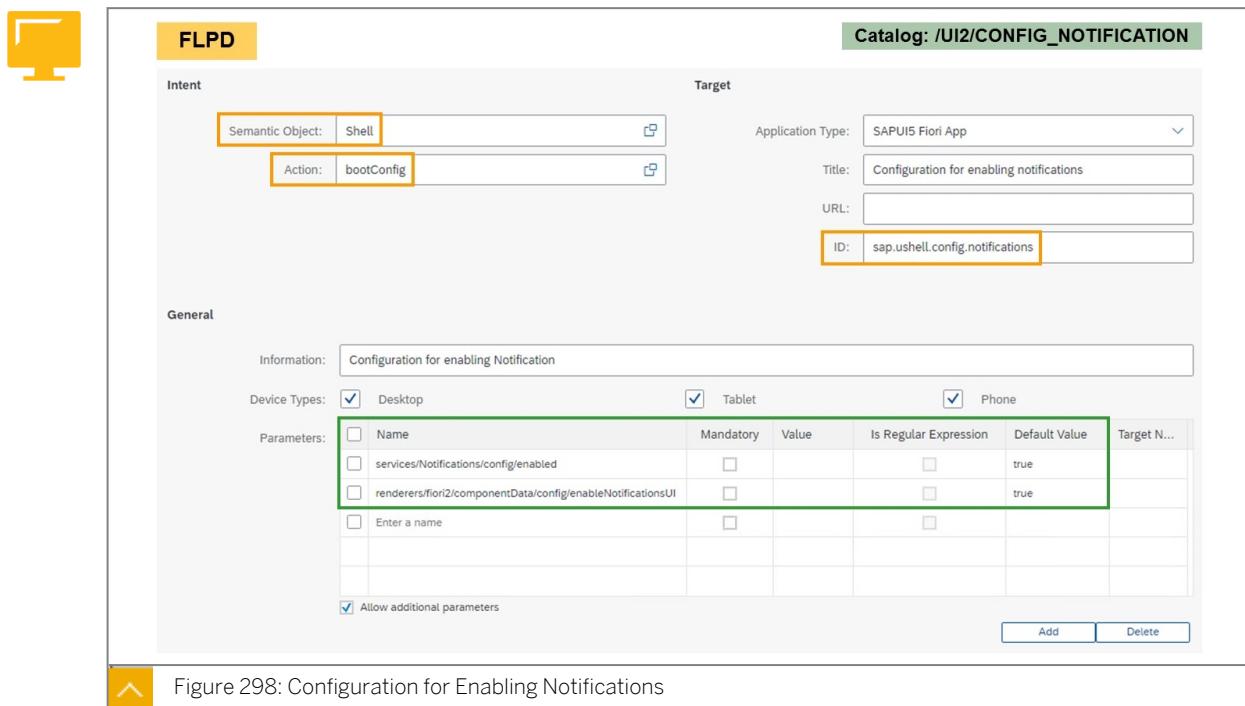
More information about this topic can be found in BIT602 (SAP Workflow with SAP Fiori):

<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 corner, 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 must be configured to send notifications for workflow tasks, and the FES must be configured to receive notifications from the BES. This must 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 10

Lesson 6

Using SAP Screen Personas



LESSON OBJECTIVES

After completing this lesson, you will be able to:

- Use SAP screen personas

Potential of SAP Screen Personas

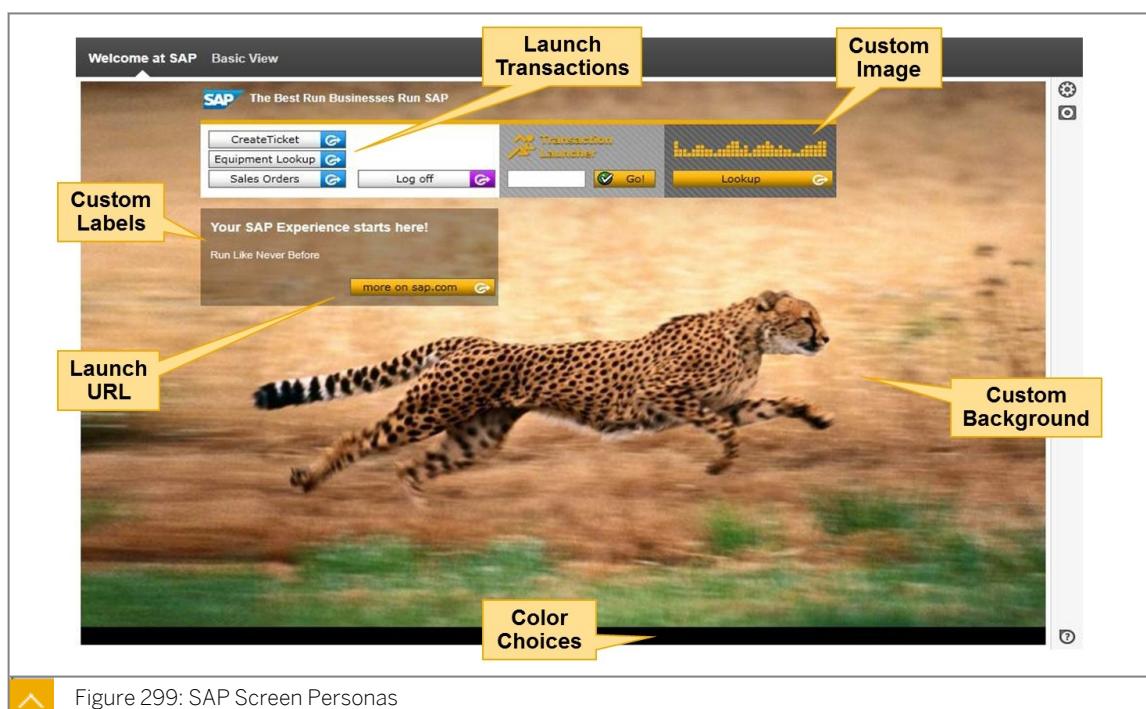


Figure 299: SAP Screen Personas

The SAP Screen Personas product enables companies to improve user productivity through adaptation 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. This is because changes to the user experience are made without programming a single line of code.

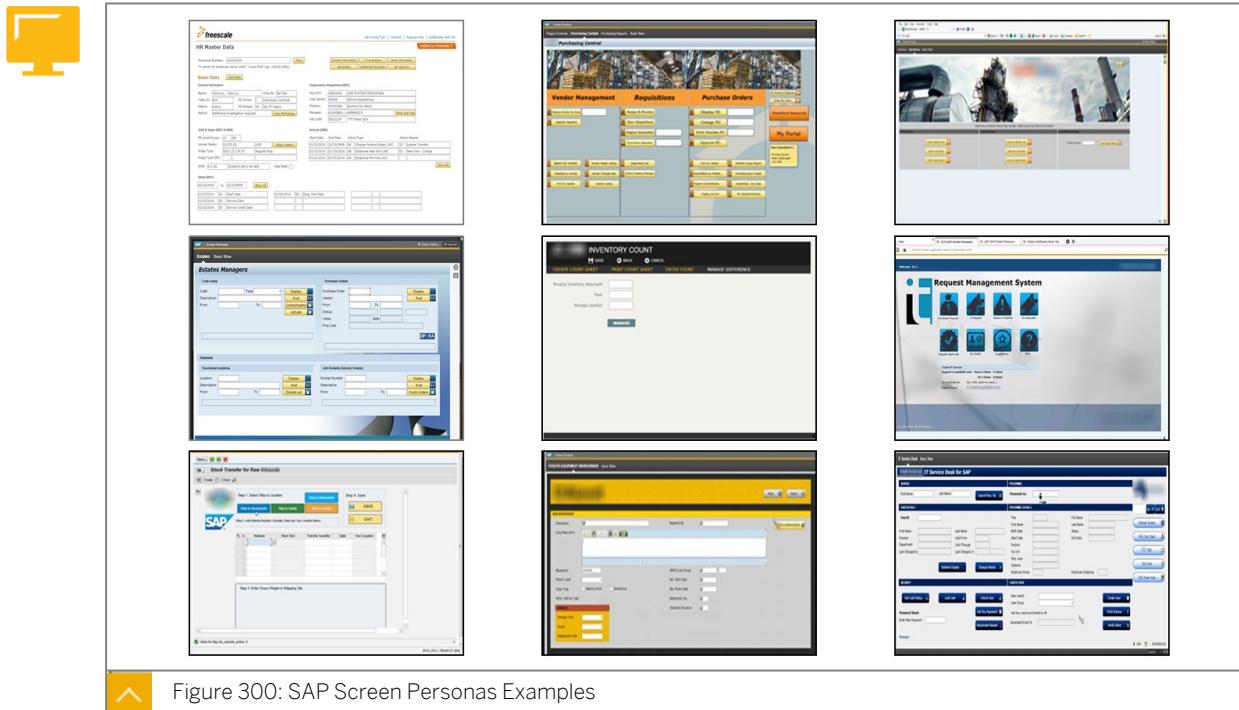


Figure 300: SAP Screen Personas Examples

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>

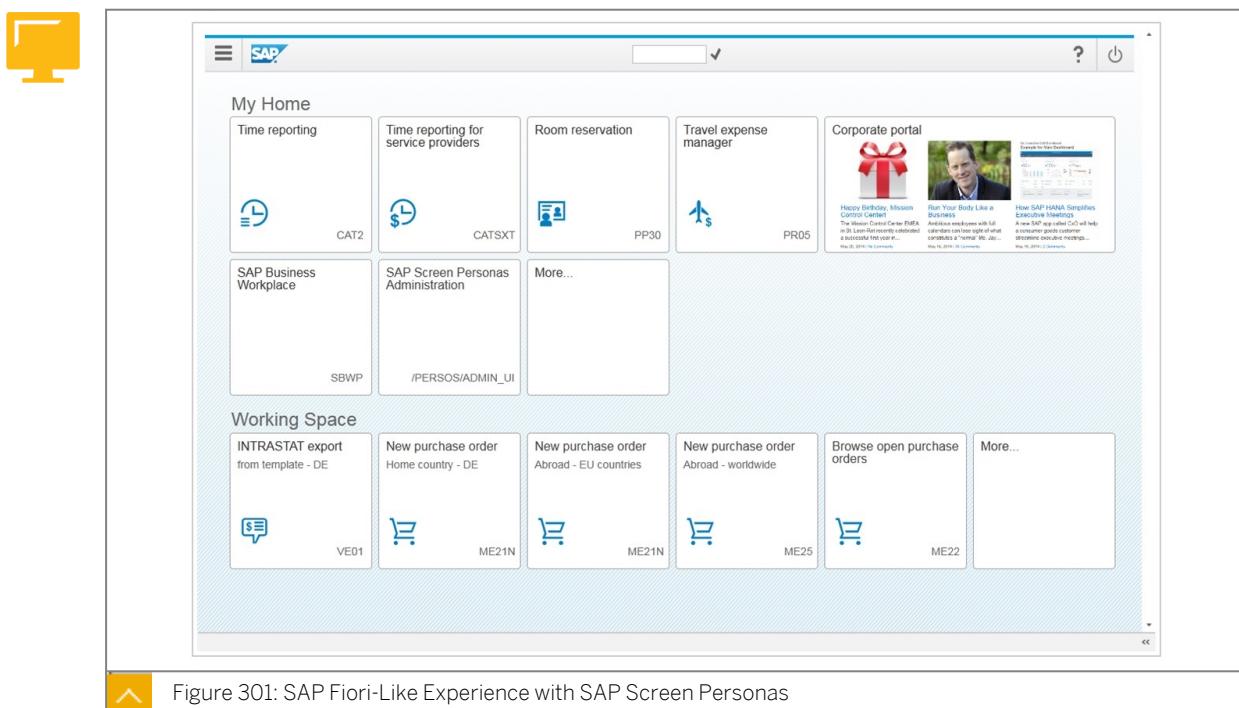


Figure 301: SAP Fiori-Like Experience with SAP Screen Personas

Using SAP Screen Personas, you can modify any SAP GUI interface into a design much like SAP Fiori. At first glance, this looks like a SAP 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 SAP Fiori style guide is available at <http://www.sapscreenpersonas.com>.



Figure 302: Run Simple with SAP Screen Personas 3.0

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 transaction 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

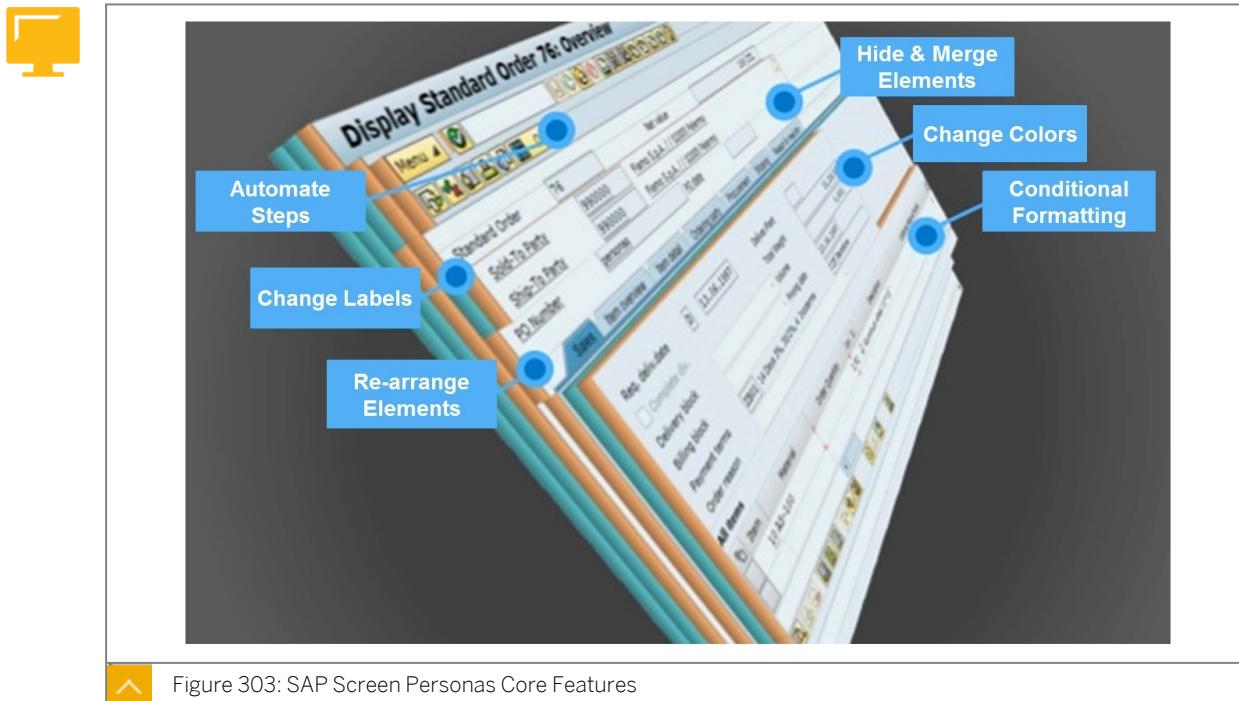


Figure 303: SAP Screen Personas Core Features

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. Macros can be created through JavaScript to automate steps such as 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](#))

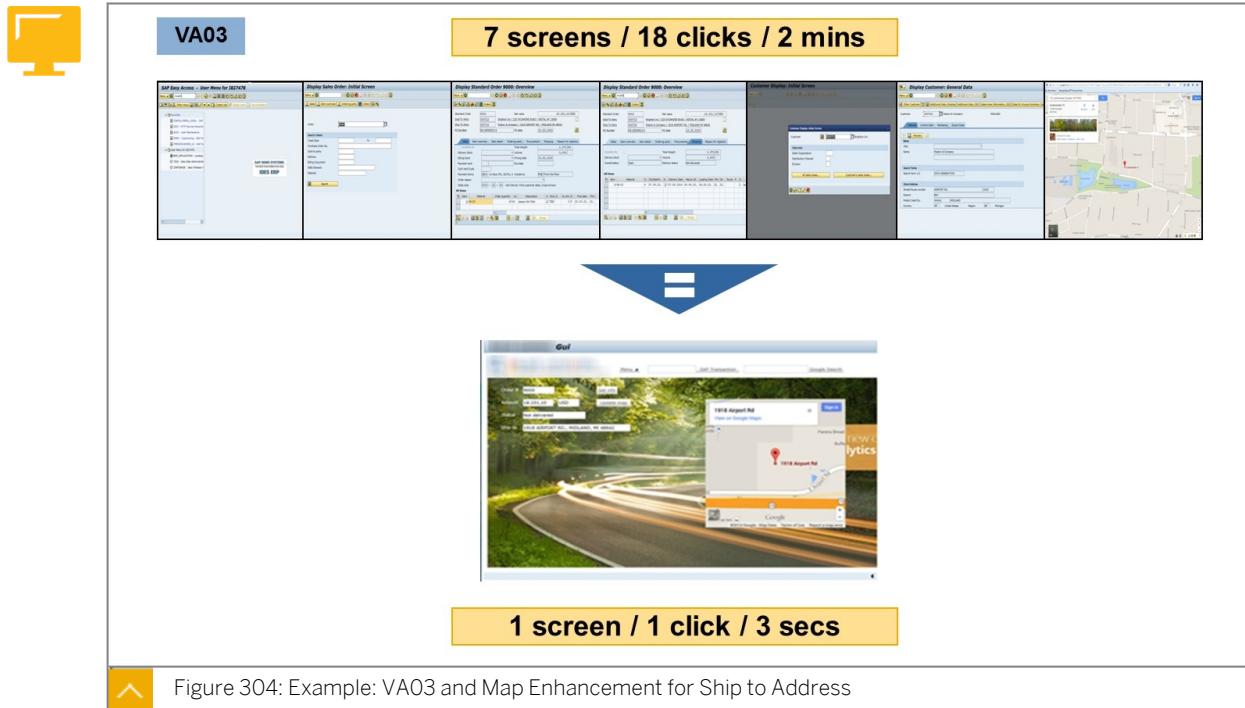


Figure 304: Example: VA03 and Map Enhancement for Ship-to Address

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 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.

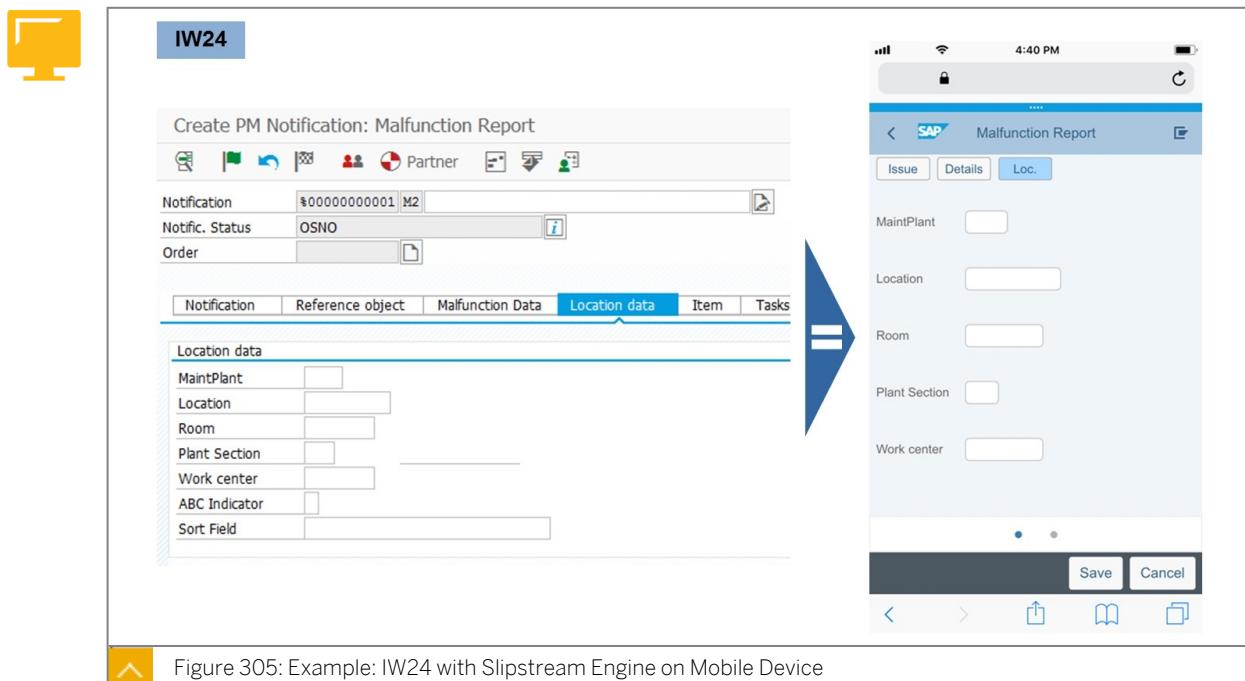


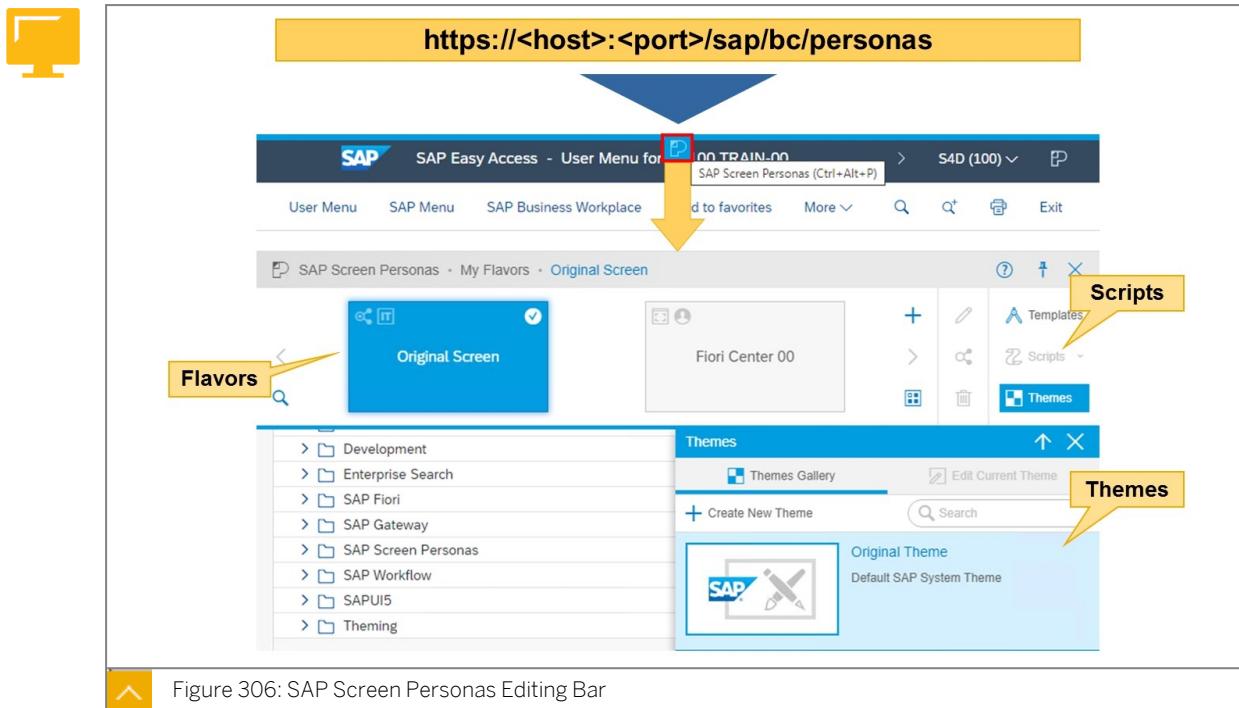
Figure 305: Example: IW24 with Slipstream Engine on Mobile Device

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. It enables it 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. However, 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.



Hint:

For starting the SAP Screen Personas editor with Slipstream Engine, use ICF service `/sap/bc/se/m`.

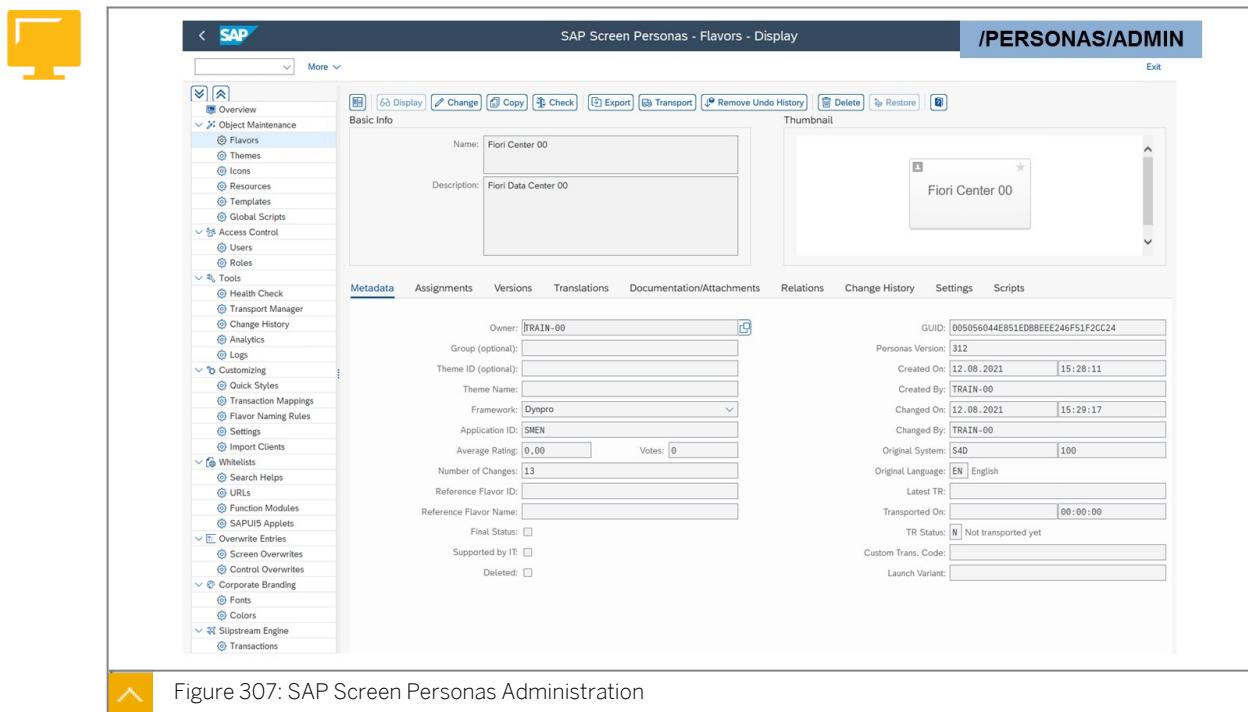
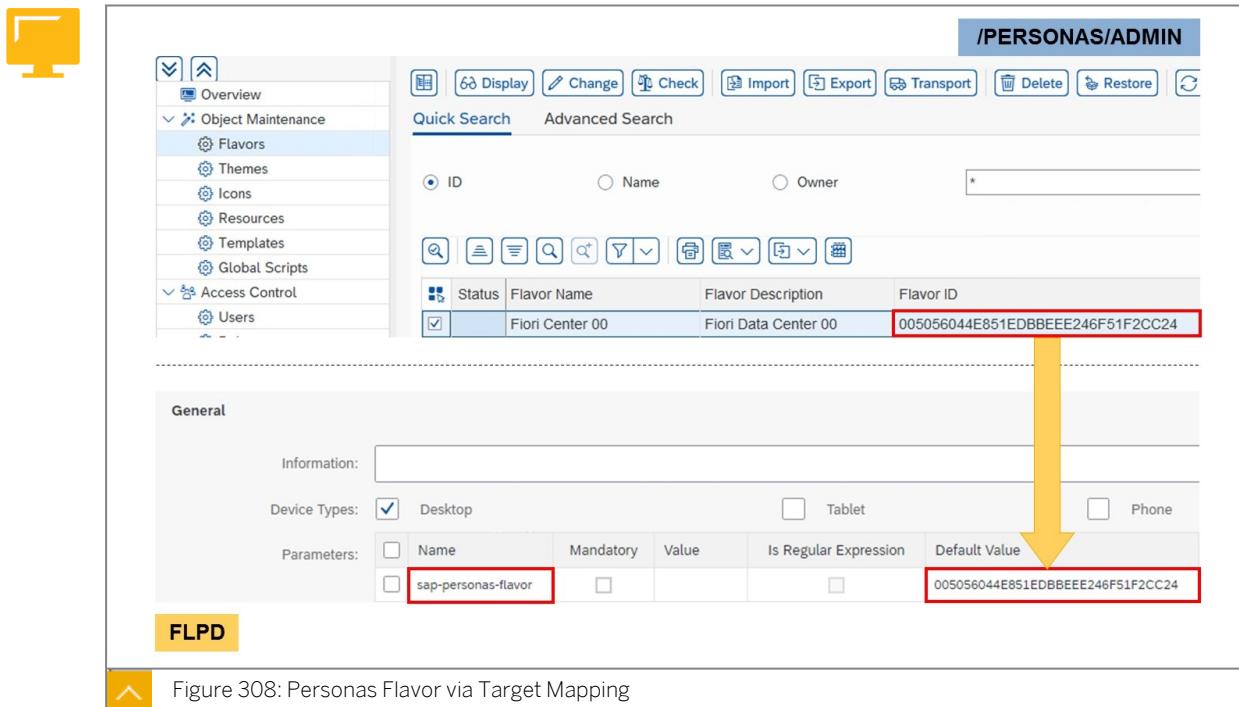


Figure 307: SAP Screen Personas Administration

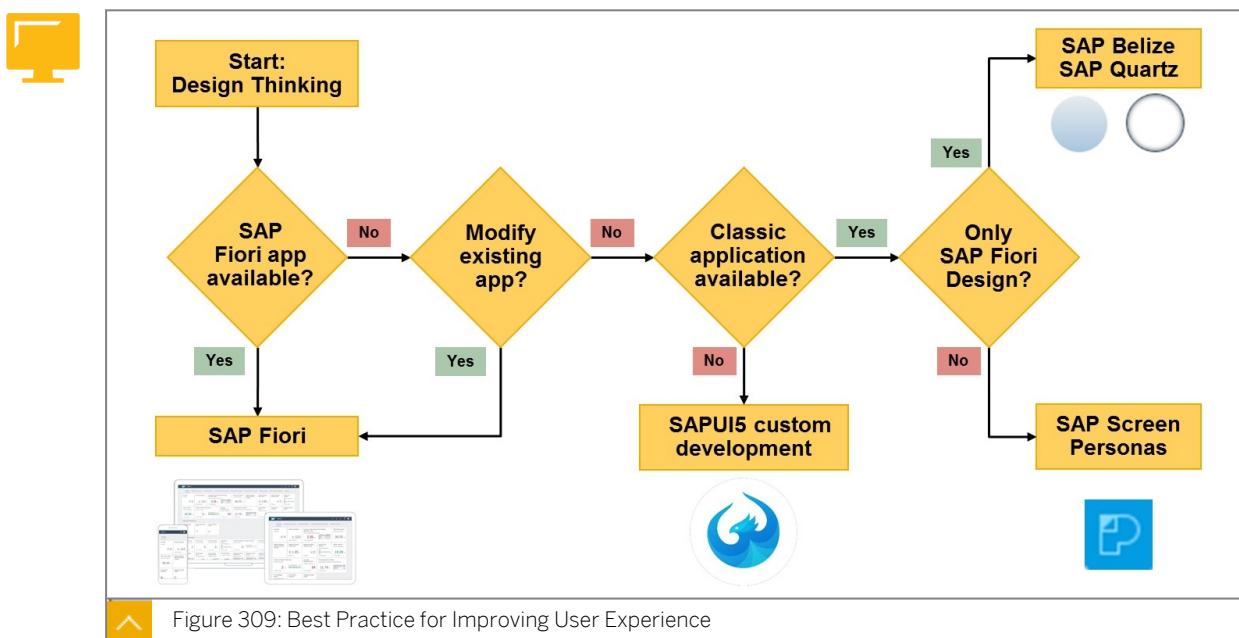
Several transactions are available to administer SAP Screen Personas, but the central transaction is **/PERSONAS/ADMIN**, which offers a dashboard to access all other transactions. It is divided into the following sections:

- *Object Maintenance*
- *Access Control*
- *Tools*
- *Customizing*
- *Whitelists*
- *Overwrite Entries*
- *Corporate Branding*
- *Slipstream Engine*

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



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.



Flavors or themes are not shipped by SAP. SAP Screen Personas is a tool that enables 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.



LESSON SUMMARY

You should now be able to:

- Use SAP screen personas

Learning Assessment

1. Which system part provides the OData services for analytical apps?

2. What is the data source definition for the SAP Fiori search?

3. What is needed to define a target mapping for a Web Dynpro ABAP application?

Choose the correct answers.

- A Web Dynpro window
- B Web Dynpro component
- C System alias
- D Web Dynpro application
- E Web Dynpro configuration

4. What is needed to define a target mapping for an ABAP transaction?

Choose the correct answers.

- A Ok-code
- B Transaction code
- C Dynpro number
- D System alias

5. Which tool is used in an SAP Business Suite environment to model KPI?

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

7. What does a KPI consist of?

Choose the correct answers.

- A Evaluation
- B Authentication
- C Drill-down
- D Tile
- E Indicator

8. Which part of a KPI is responsible for the visualization?

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

10. 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

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

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

13. What is SAP Screen Personas?

14. Which UI clients does SAP Screen Personas support?

Choose the correct answers.

- A SAP Business Client
- B SAP Mobile Services Client
- C SAP GUI for Windows
- D SAP GUI for HTML
- E SAP GUI for Java

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

Learning Assessment - Answers

1. Which system part provides the OData services for analytical apps?

SAP HANA extended application services (XS)

2. What is the data source definition for the SAP Fiori search?

Search connectors

3. What is needed to define a target mapping for a Web Dynpro ABAP application?

Choose the correct answers.

- A Web Dynpro window
- B Web Dynpro component
- C System alias
- D Web Dynpro application
- E Web Dynpro configuration

Correct. The system alias, Web Dynpro application and configuration are needed for a target mapping for Web Dynpro ABAP.

4. What is needed to define a target mapping for an ABAP transaction?

Choose the correct answers.

- A Ok-code
- B Transaction code
- C Dynpro number
- D System alias

Correct. The transaction code and system alias are needed for a target mapping for an ABAP transaction.

5. Which tool is used in an SAP Business Suite environment to model KPI?

KPI Modeler

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

KPI Design

7. What does a KPI consist of?

Choose the correct answers.

- A Evaluation
- B Authentication
- C Drill-down
- D Tile
- E Indicator

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

8. Which part of a KPI is responsible for the visualization?

Drill-down

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

SAP Fiori iView

10. 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.

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

My Inbox

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

Front-End Server (FES)

13. What is SAP Screen Personas?

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

14. Which UI clients does SAP Screen Personas support?

Choose the correct answers.

- A SAP Business Client
- B SAP Mobile Services 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.

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

Flavors
