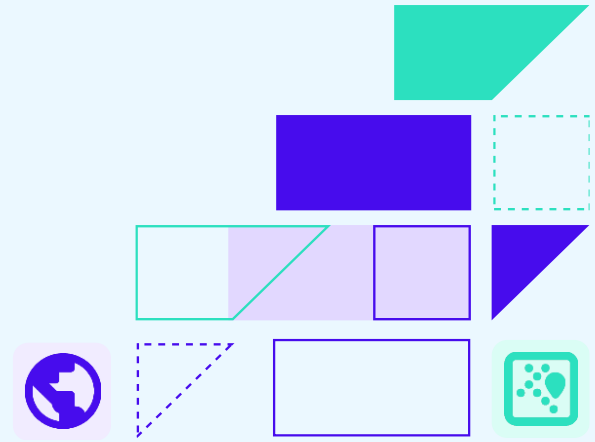
SAP TechEd

SAP TechEd

Use ABAP Cloud for developer extensibility – DT182v

Andre Fischer, Harish Bokkasam – SAP SE
November 3, 2023



Agenda

INTRODUCTION

TODAY’S SCENARIO – OVERVIEW

TODAY’S SCENARIO – EXERCISES

WRAP UP

INTRODUCTION

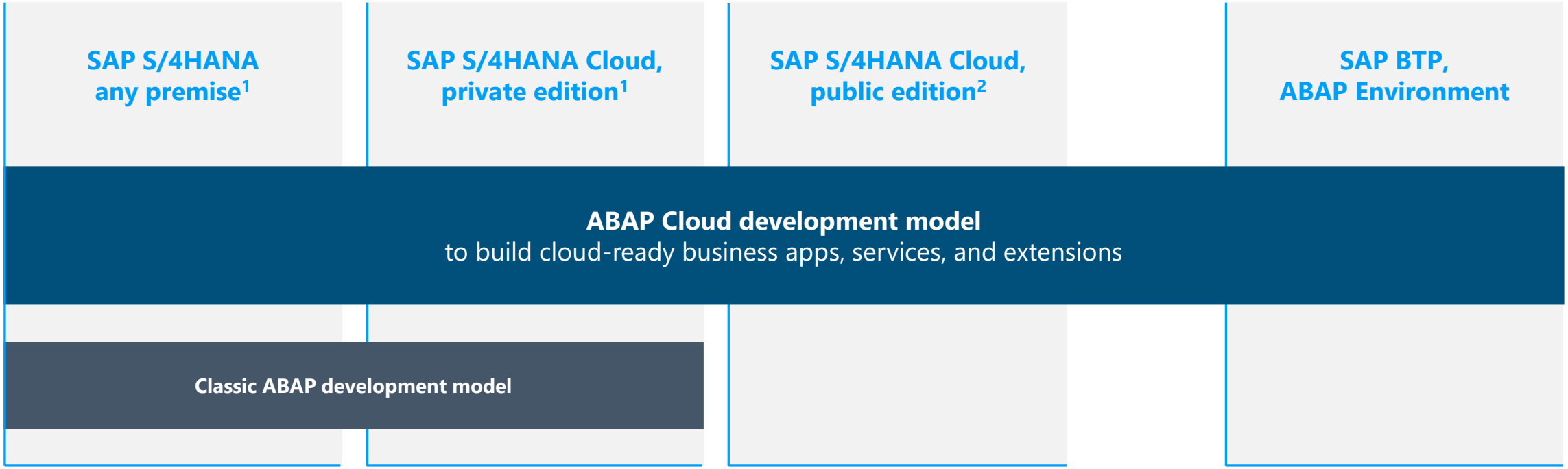
ABAP Cloud

... is the ABAP development model to build cloud-ready business apps, services, and extensions

... comes with SAP Business Technology Platform (SAP BTP) and SAP S/4HANA

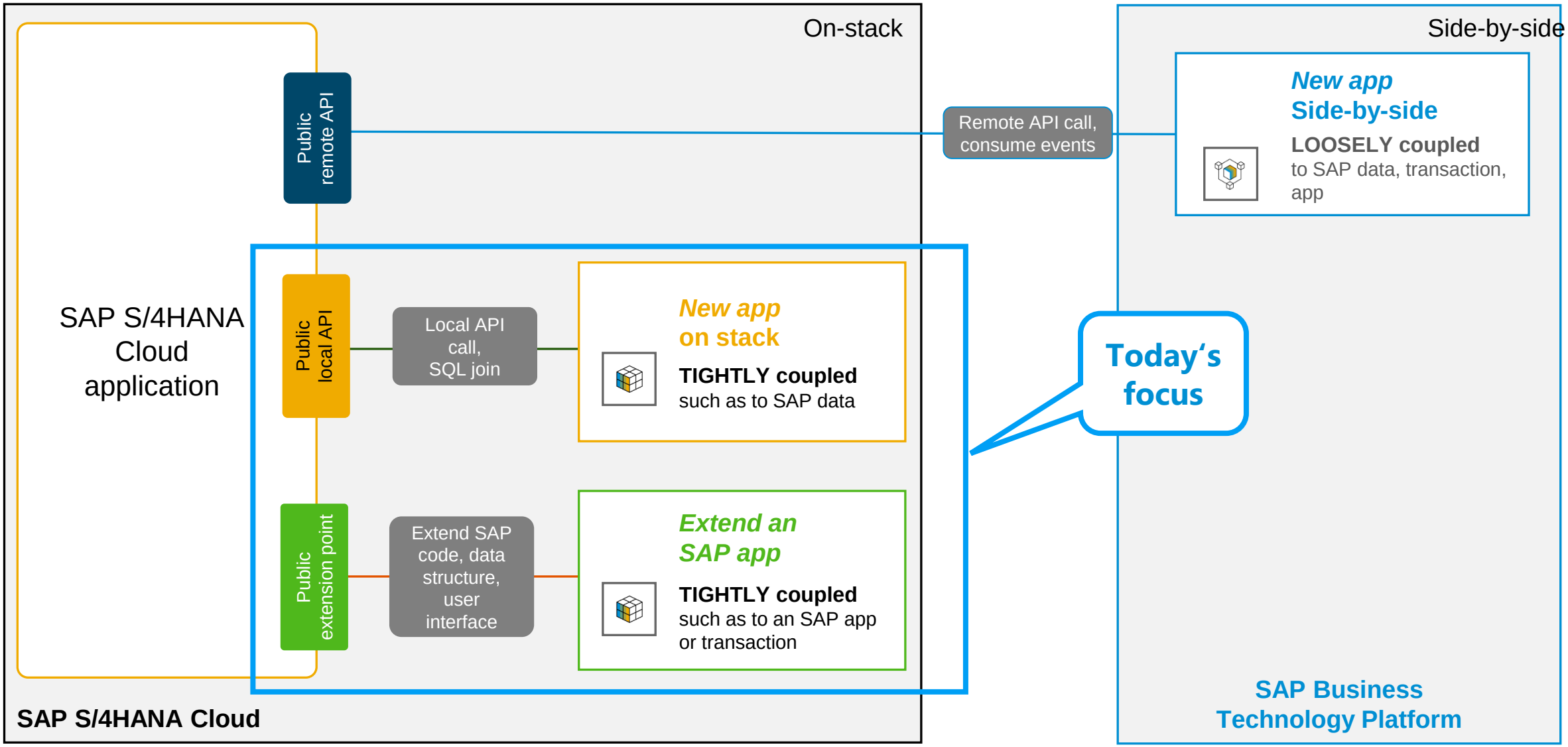
... works with public or private cloud, and even on-premise

ABAP Cloud – One development model for SAP S/4HANA and SAP BTP



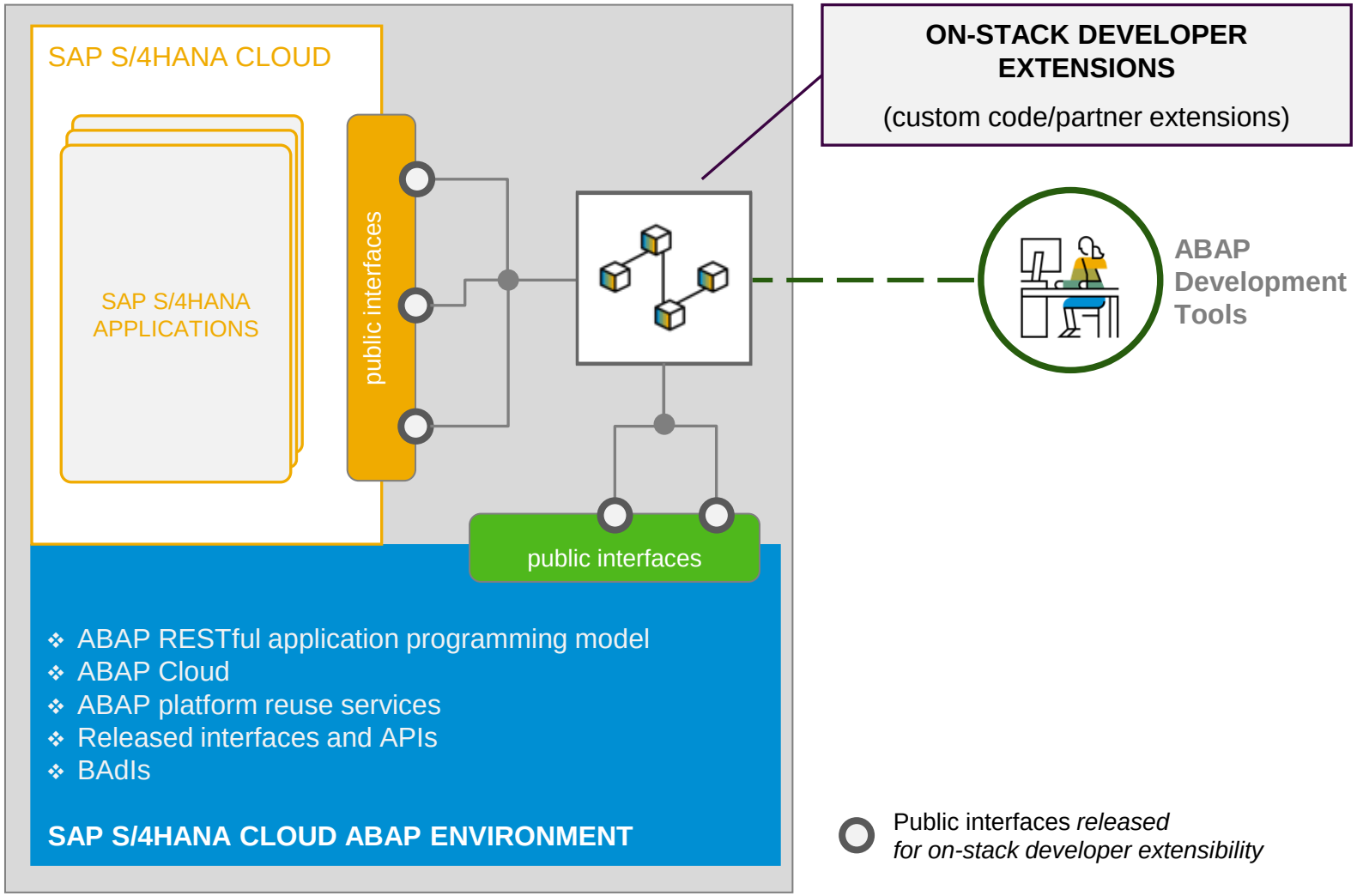
[1] SAP S/4HANA any premise or SAP S/4HANA Cloud, private edition release ≥ 2022
[2] SAP S/4HANA Cloud, public edition release ≥ 2208, 3-system landscape required

SAP S/4HANA Cloud extensibility patterns

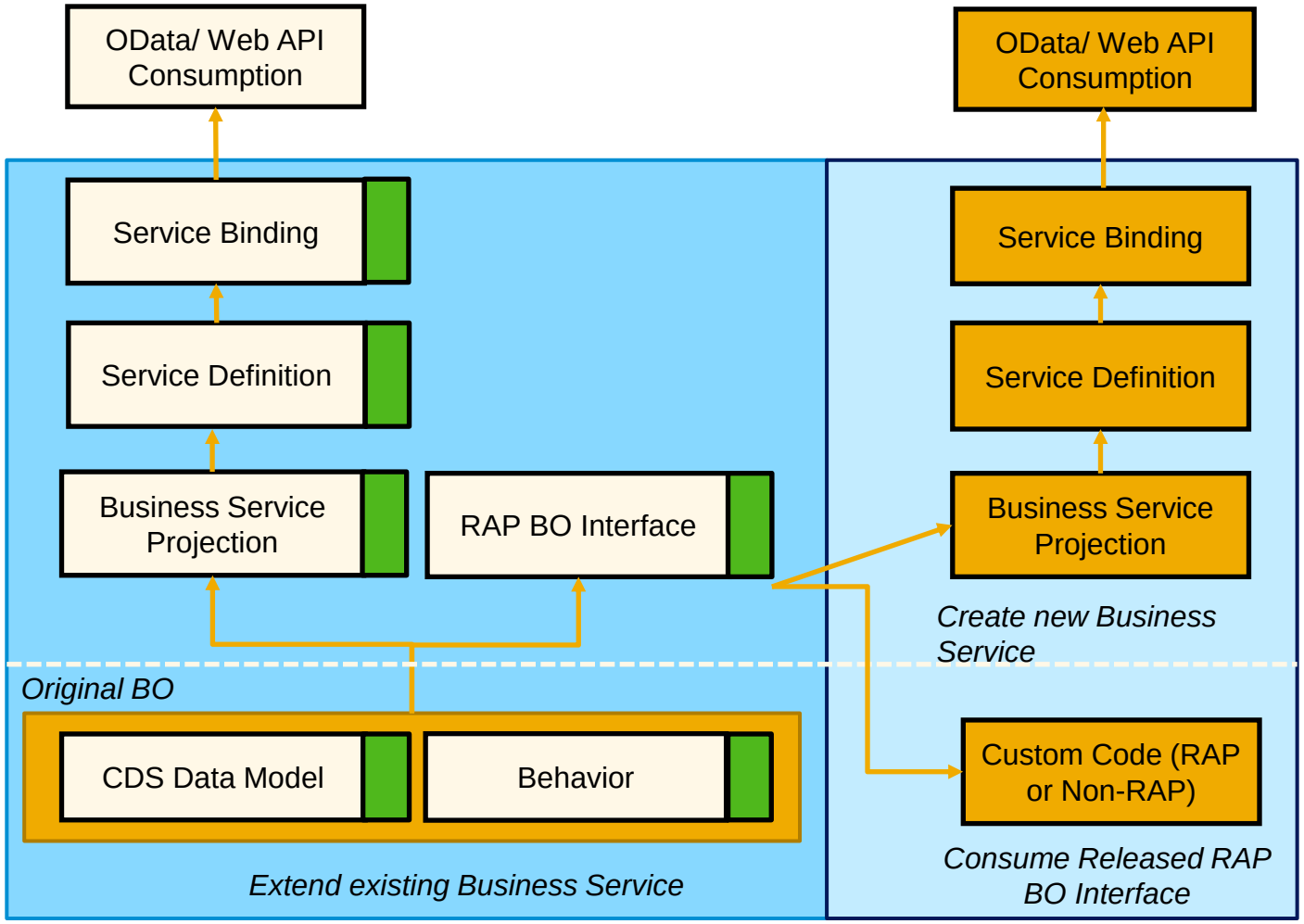


Loosely coupled Tightly coupled

New app on stack and SAP app extension



ON-STACK DEVELOPER EXTENSION OPTIONS



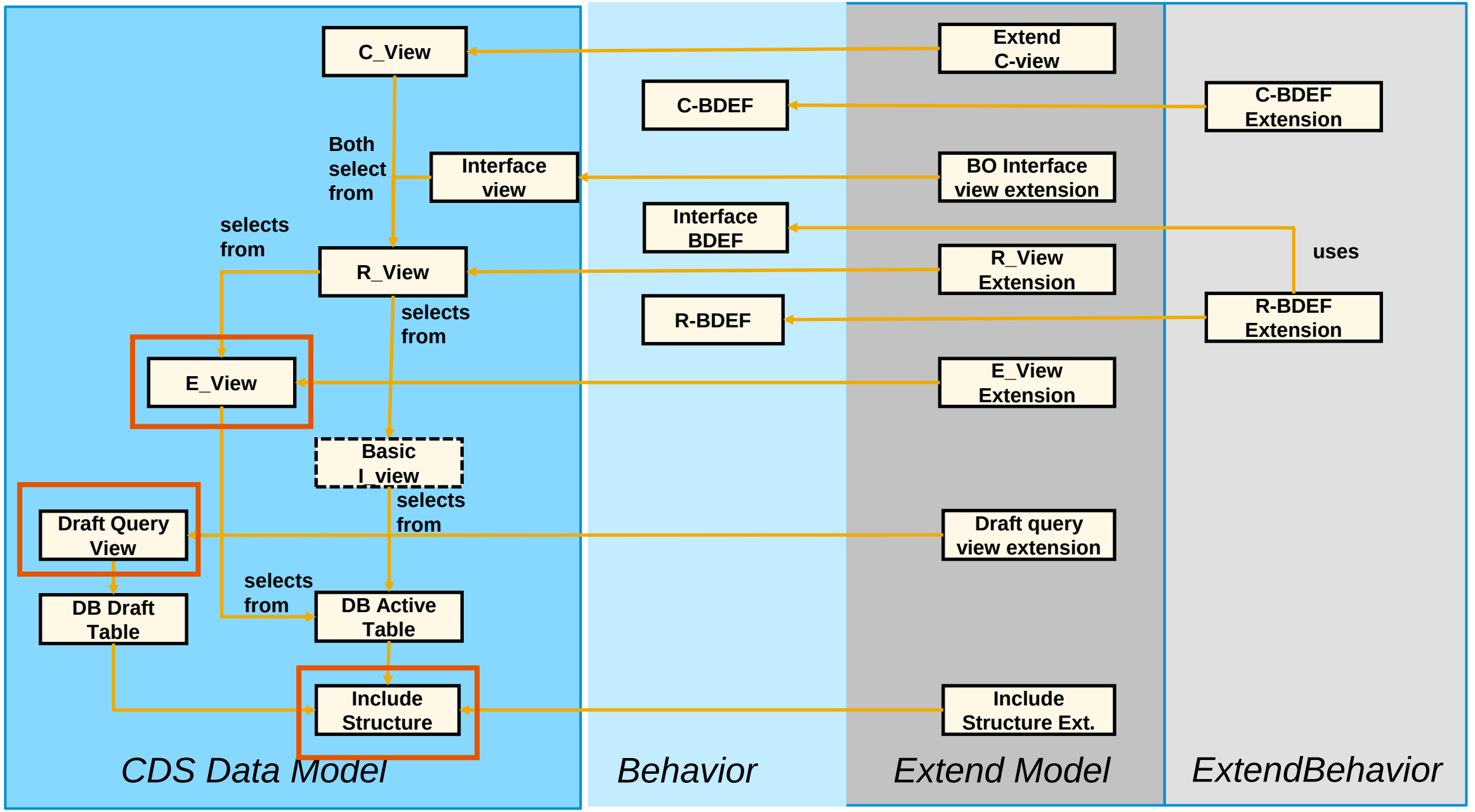
- Create new BO‘s on top of released BO interfaces
- Consume released BO interfaces in custom code
- Extend existing BO‘s

= ***New app*** **on stack**

= ***Extension***

INTERNAL – SAP and Partners Only

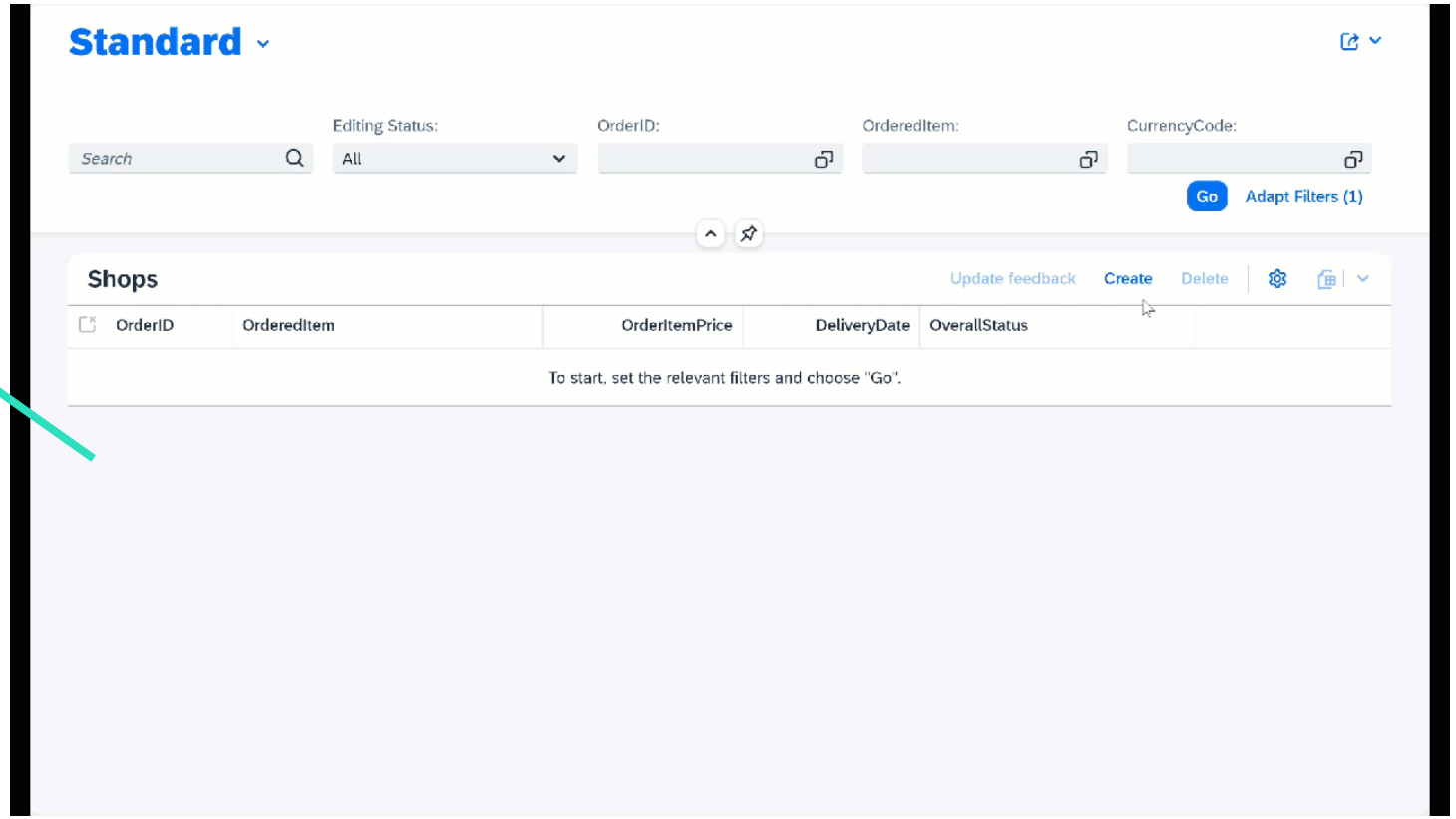
RAP business object extension in detail



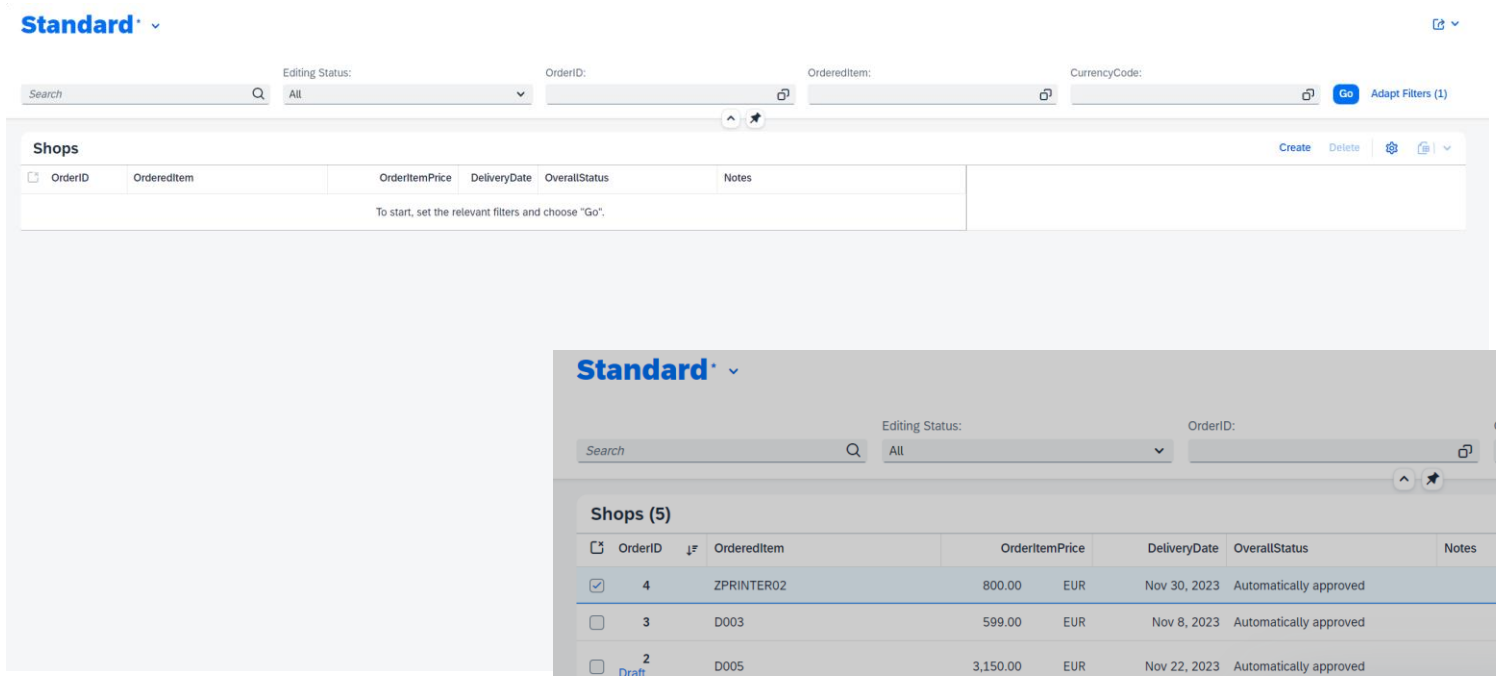
TODAY’S SCENARIO – OVERVIEW

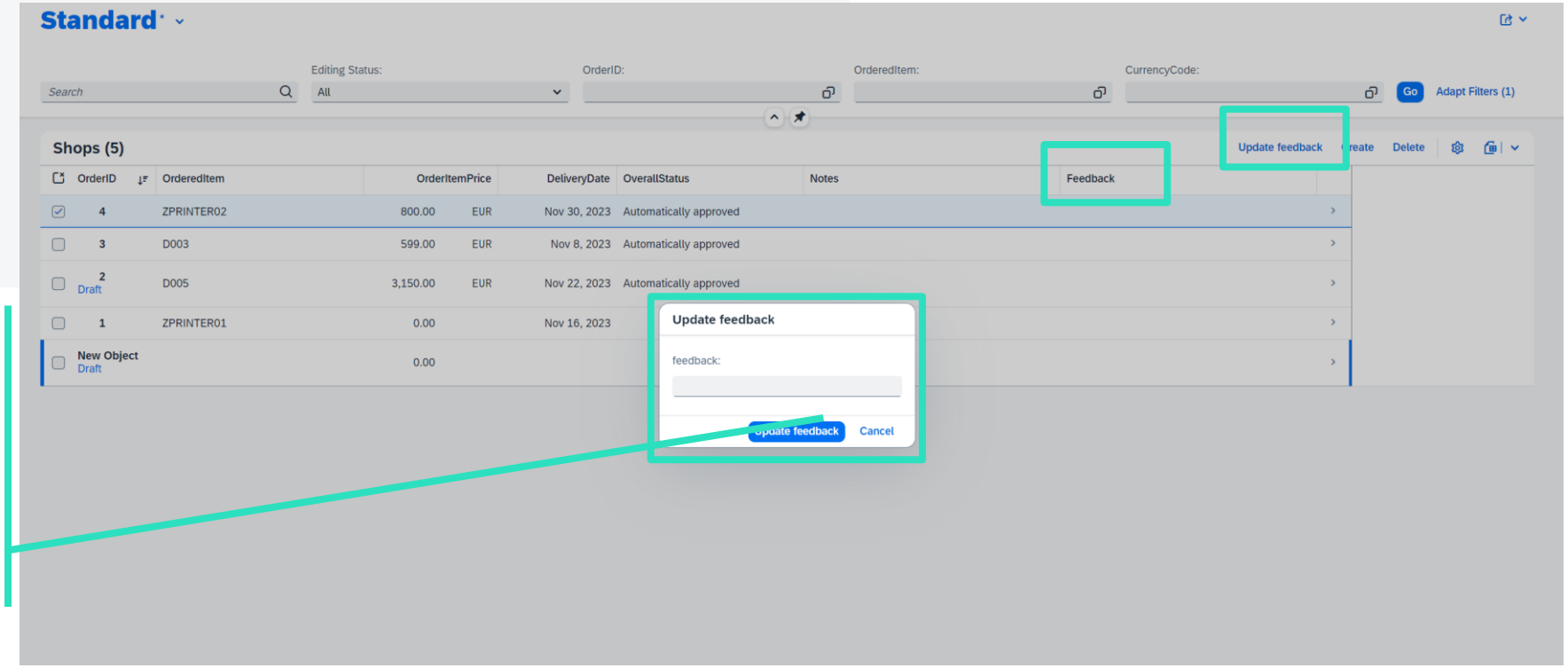
Extended Onlineshop App – Behavior extension

Add e.g. side effects



Extended Onlineshop App – Data model extension





New field:
Feedback
New Action:
Update Feedback

Solution packages

ZRAP630SOL and ZRAP630_SOL_EXT

- ZRAP630_SOL (19) *#Generated RAP630 tutorial package*
  - Business Services (2)
    - Service Bindings (1)
      - ZRAP630UI_SHOP_O4_SOL *Service binding for Shop*
    - Service Definitions (1)
      - ZRAP630UI_SHOP_SOL *Service definition for Shop*
  - Cloud Communication Management (1)
  - Cloud Identity and Access Management (1)
  - Core Data Services (10)
    - Behavior Definitions (3)
      - ZRAP630C_SHOPTP_SOL *Behavior for ZRAP630C_ShopTP_SOL*
      - ZRAP630I_SHOPTP_SOL *Behavior for ZRAP630I_ShopTP_SOL*
      - ZRAP630R_SHOPTP_SOL *Behavior for ZRAP630R_ShopTP_SOL*
    - Data Definitions (6)
      - ZRAP630C_SHOPTP_SOL *Projection View for Shop*
      - ZRAP630E_SHOP_SOL *Extension include view for Shop*
      - ZRAP630I_SHOPTP_SOL *Interface Projection View for Shop*
      - ZRAP630I_VH_PRODUCT_SOL *Value help for products*
      - ZRAP630R_SHOPTP_SOL *CDS View for Shop*
      - ZRAP630R_SHOP_D_SOL *Draft query view for Shop*
    - Metadata Extensions (1)
      - ZRAP630C_SHOPTP_SOL *MDE for Shop*
  - Dictionary (3)
    - Database Tables (2)
      - ZRAP630SH00D_SOL *Draft table for entity ZRAP630R_ShopTP_SOL*
      - ZRAP630_ASHOPSOL *Table for entity ZRAP630R_ShopTP_SOL*
    - Structures (1)
      - ZRAP630SSHOP_SOL *Extension include for Shop*
  - Source Code Library (2)
    - Classes (2)
      - ZRAP630BP_R_SHOPTP_SOL *Behavior implementation*
      - ZRAP630_CL_VH_PRODUCT_SOL *Value help for products*

Two packages contain the solution

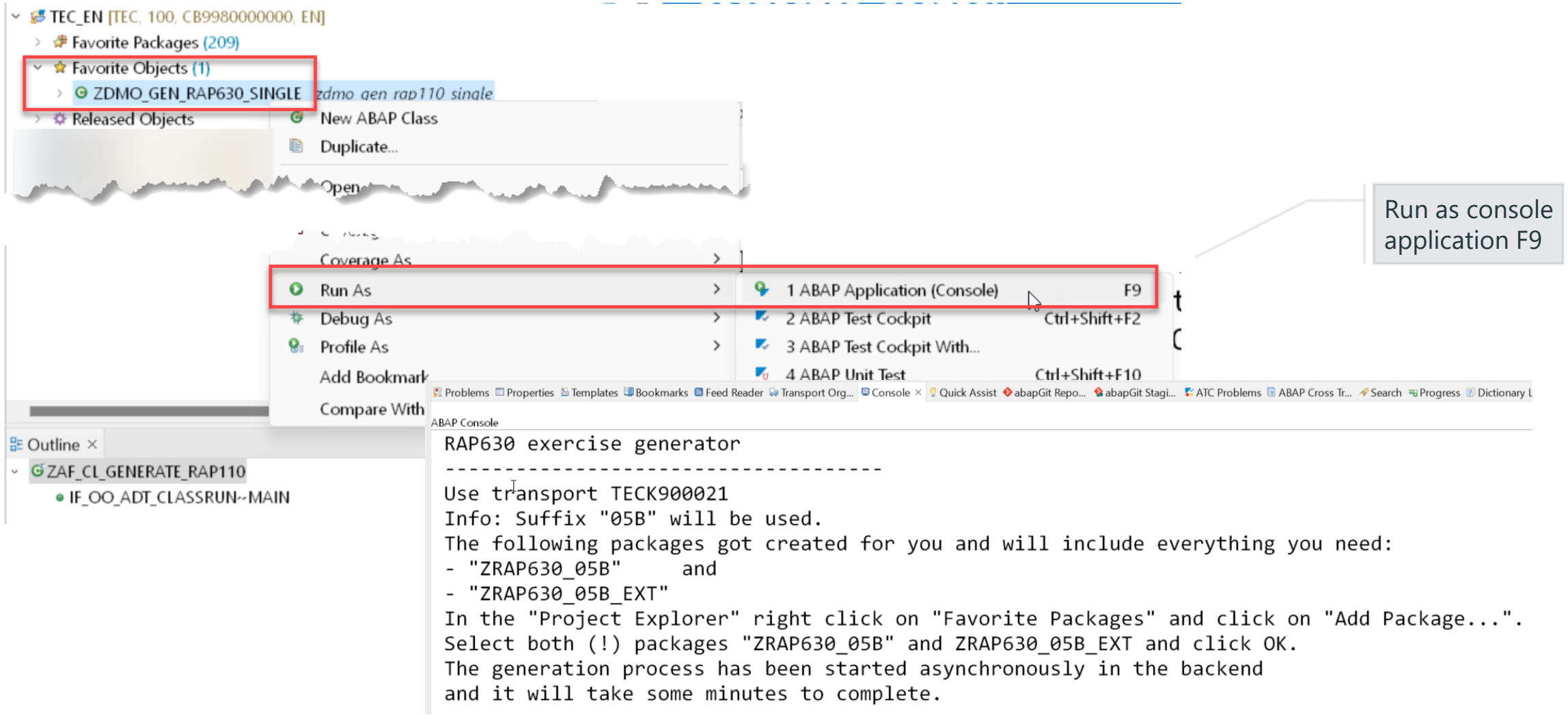
- ZRAP630_SOL
  Base BO
- ZRAP630_SOL_EXT
  Extension objects

- ZRAP630_SOL_EXT (10) *Solution for RAP630 / DT 182v*
  - Core Data Services (8)
    - Behavior Definitions (2)
      - ZRAP630C_EXT_SHOPTP_EXT *Behavior extension for projection bdef*
      - ZRAP630R_EXT_SHOPTP_SOL *Extension for ZRAP630R_ShopTP_SOL*
    - Data Definitions (6)
      - ZRAP630C_EXT_SHOP_SOL *Extension for C-CDS view*
      - ZRAP630E_EXT_SHOP_SOL *Extension for Extension Include View*
      - ZRAP630I_EXT_SHOP_SOL *Extension for I-CDS view*
      - ZRAP630R_EXT_SHOP_D_SOL *Extension for R-CDS Draft view*
      - ZRAP630R_EXT_SHOP_SOL *Extension for R-CDS view*
      - ZRAP630_A_FEEDBACK_SOL *Pass feedback as a parameter*
  - Dictionary (1)
    - Structures (1)
      - ZRAP630EXTSSHOP_SOL *Extend Extension Include Structure*
  - Source Code Library (1)
    - Classes (1)
      - ZBP_ZRAP630R_SHOP_X_FBK_SOL *Behavior Implementation for ZRAP630R_EXT_*

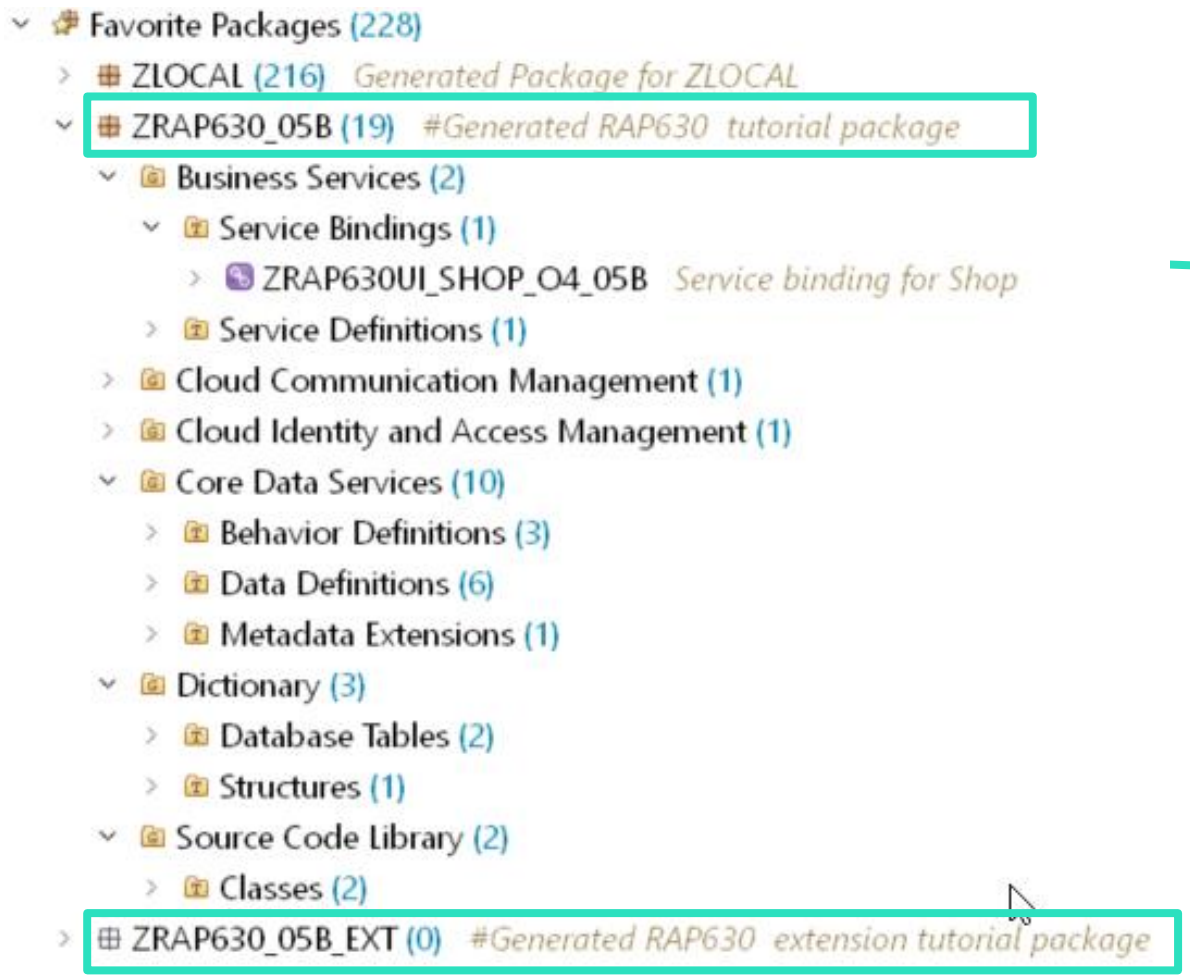
TODAY'S SCENARIO – EXERCISES

Generate your own extensible RAP BO

Run ZDMO_GEN_RAP630_SINGLE via F9



Use ABAP Cloud for developer extensibility

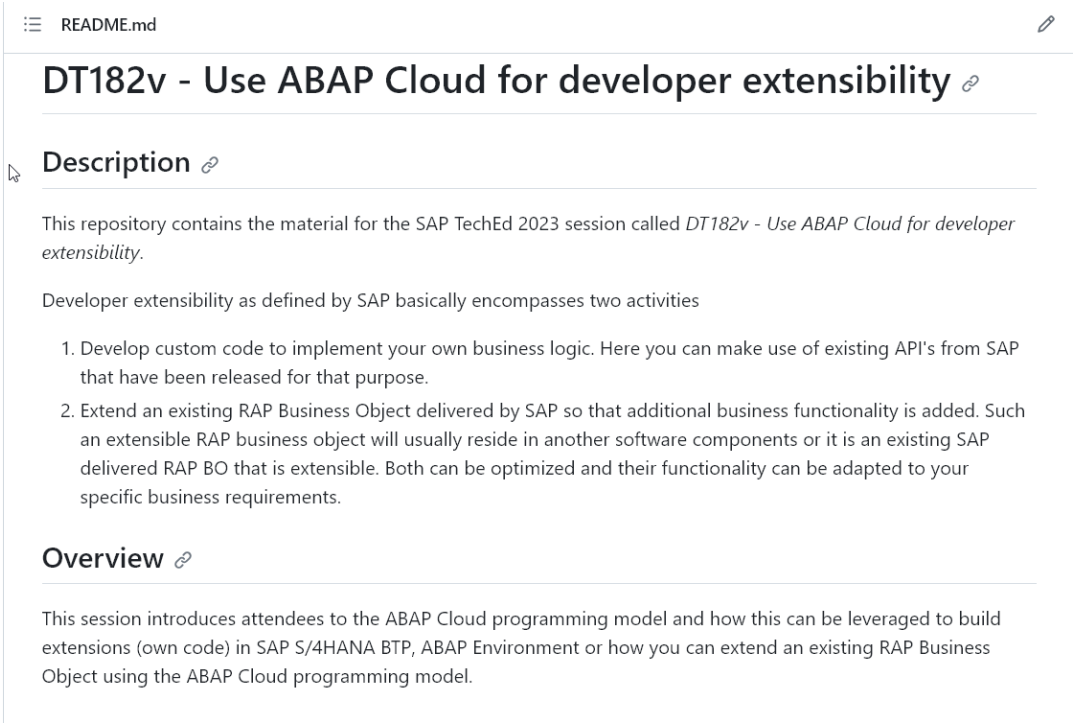


Two packages have been generated:

- ZRAP630_###
  Base BO
- ZRAP630_###_EXT
  Extension objects

The DT182v hands-on exercises

https://github.com/SAP-samples/teched2023-DT182v



README.md

DT182v - Use ABAP Cloud for developer extensibility

Description

This repository contains the material for the SAP TechEd 2023 session called *DT182v - Use ABAP Cloud for developer extensibility*.

Developer extensibility as defined by SAP basically encompasses two activities

1. Develop custom code to implement your own business logic. Here you can make use of existing API's from SAP that have been released for that purpose.
2. Extend an existing RAP Business Object delivered by SAP so that additional business functionality is added. Such an extensible RAP business object will usually reside in another software components or it is an existing SAP delivered RAP BO that is extensible. Both can be optimized and their functionality can be adapted to your specific business requirements.

Overview

This session introduces attendees to the ABAP Cloud programming model and how this can be leveraged to build extensions (own code) in SAP S/4HANA BTP, ABAP Environment or how you can extend an existing RAP Business Object using the ABAP Cloud programming model.

01

Access to the exercises:
https://github.com/SAP-samples/teched2023-DT182v

02

Set up your development environment.

1. Install and set up your ABAP Development Tools for Eclipse
2. Create an SAP BTP ABAP Environment Trial User
3. Create an ABAP Cloud Project

03

Move at your own speed!

04

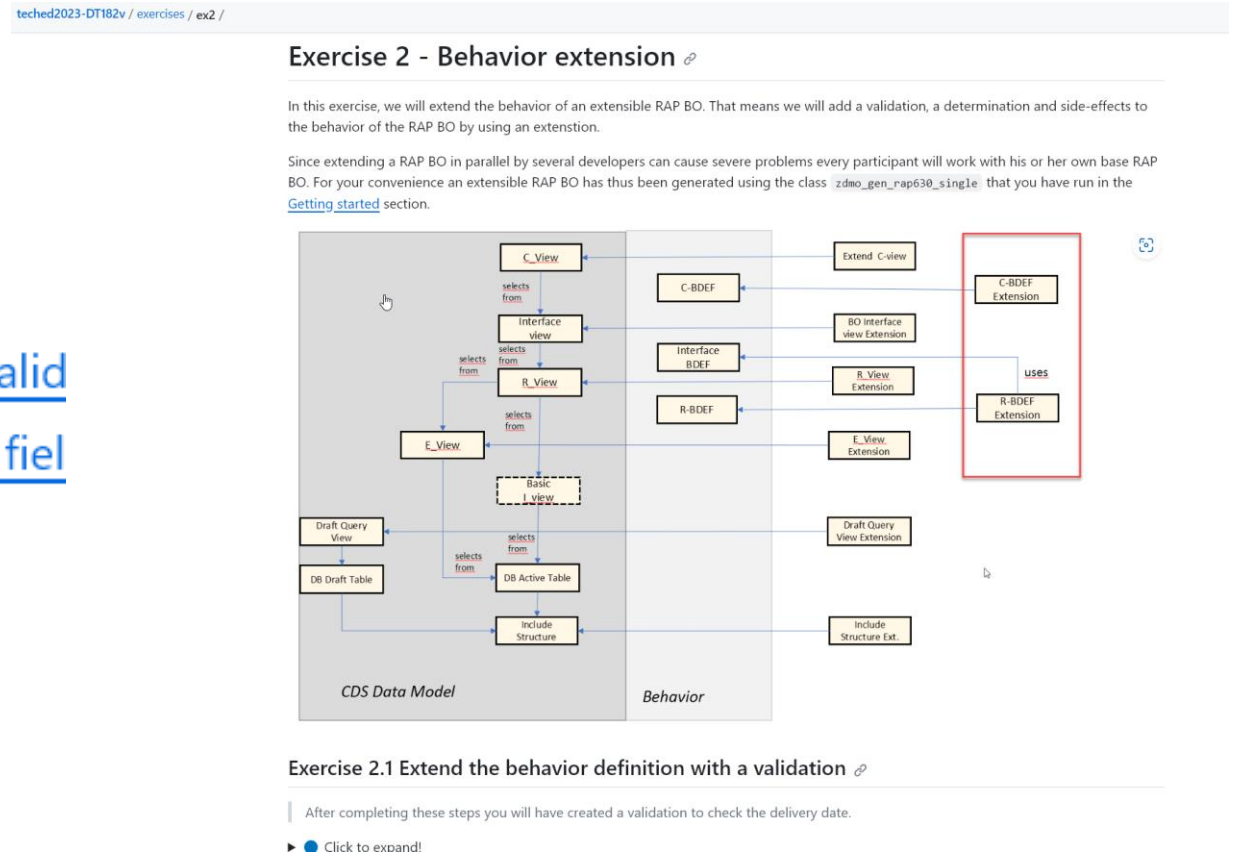
Feel free to ask questions via GitHub issues!

▶ Provide screenshots to facilitate the support

The DT182v hands-on exercises – Working on GitHub (1)

Follow the step-by-step instructions provided on GitHub.

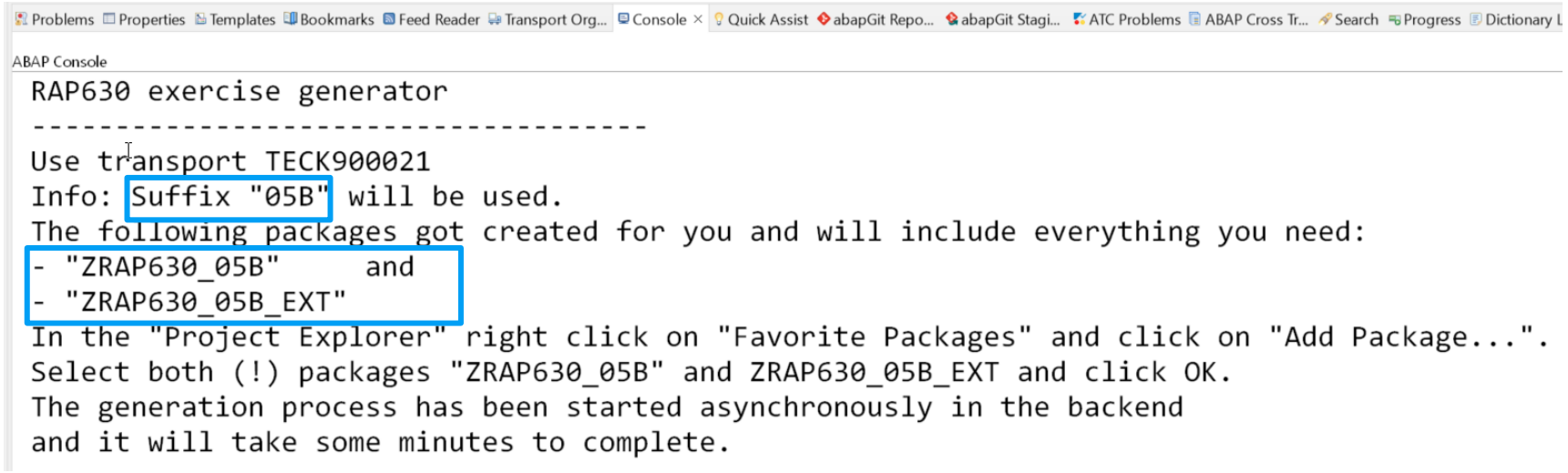
- Getting Started
- Exercise 1 - Consume a released RAP BO
- Exercise 2 - Extend the behavior (determinations, valid
- Exercise 3 - Extend the data model (add additional fiel
- Exercise 4 - Extend the behavior (add an action)



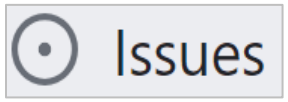
The DT182v hands-on exercises – Working on GitHub (2)

###

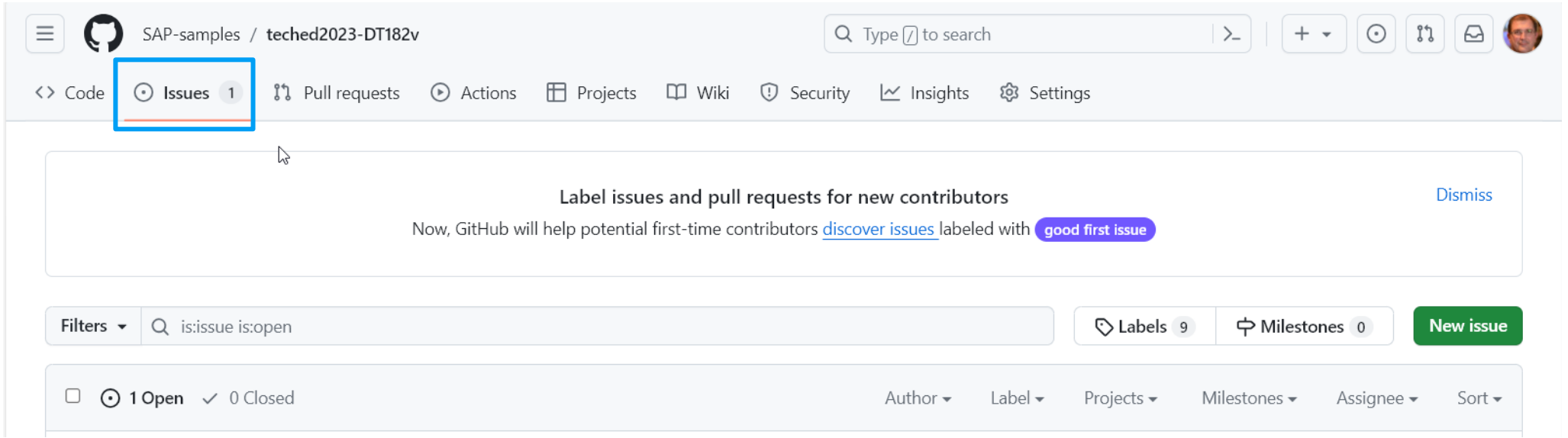
Always replace all occurrences of ### with your assigned suffix in code snippets



The DT182v hands-on exercises – Working on GitHub (3)



Open GitHub Issues to get help from the SAP experts.



SUMMARY

Key takeaways

ABAP Cloud is the development model provided to build cloud-ready enterprise services, apps, and extensions on SAP BTP ABAP environment and all editions[1] of SAP S/4HANA, in the cloud and on-premises.

The **ABAP RESTful application programming model** (RAP) is at the **heart of ABAP Cloud** for efficiently building **transactional** OData-based services and SAP Fiori apps with built-in cloud qualities.

RAP best support **SAP HANA** and **SAP Fiori elements**.

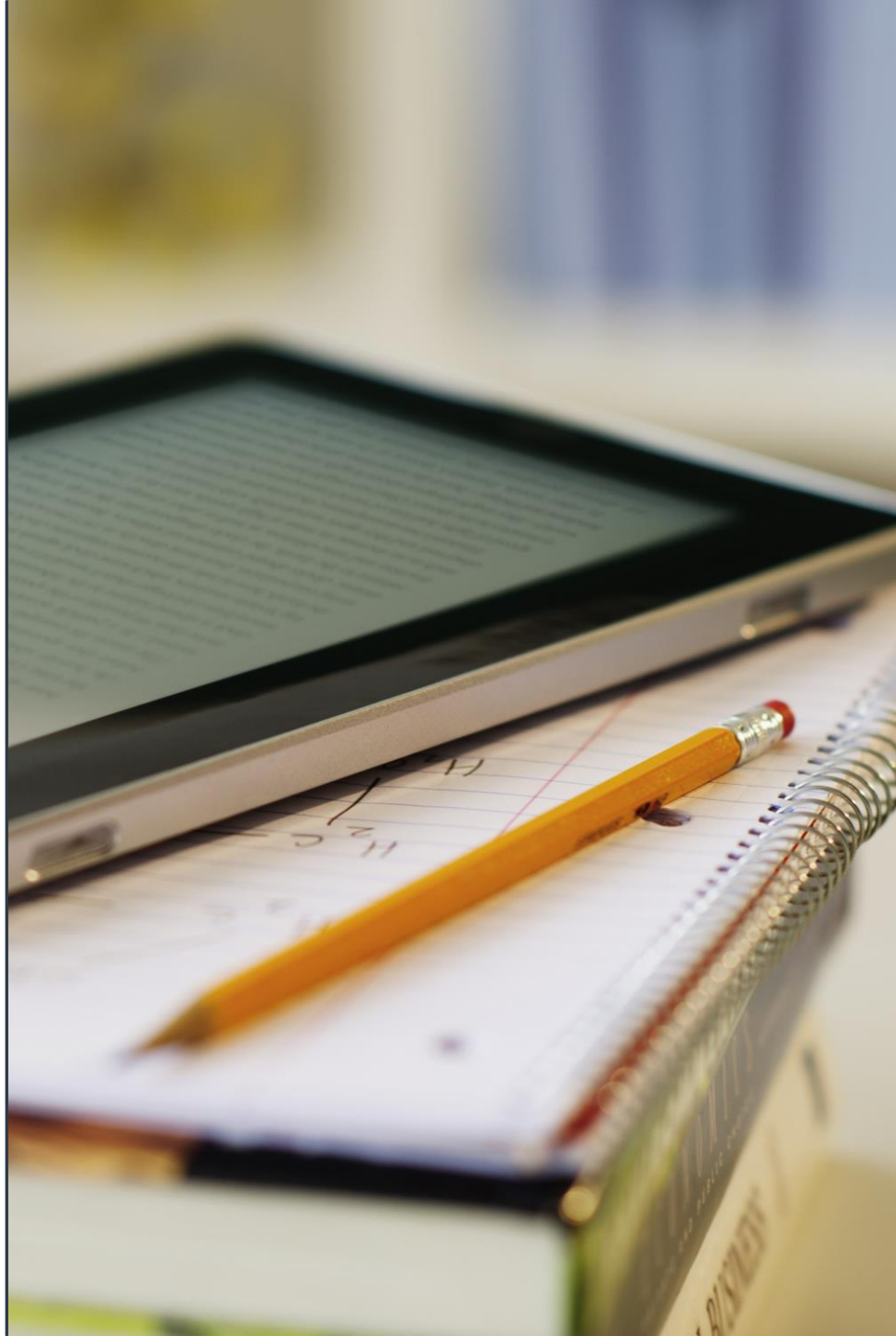
RAP is available on **SAP BTP ABAP Environment**, **SAP S/4HANA Cloud**, and **SAP S/4HANA** as of edition 1909.

The **RAP feature set** is enhanced **quarterly** in SAP BTP ABAP environment, **twice a year** in SAP S/4HANA Cloud, and **every two years** in SAP S/4HANA and SAP S/4HANA Cloud, private edition.

What's New: SAP BTP ABAP environment | SAP S/4HANA | SAP S/4HANA Cloud

What's Next: ABAP Platform Roadmap Information for all SAP products

[1] SAP S/4HANA any premise or SAP S/4HANA Cloud, private edition release ≥ 2022
SAP S/4HANA Cloud, public edition release ≥ 2208, 3-system landscape required



Thank you

Contact information:

andre.fischer@sap.com
Product Manager for ABAP Platform

harish.bokkasam@sap.com
Product Manager for ABAP Platform

© 2023 SAP SE or an SAP affiliate company. All rights reserved. See Legal Notice on www.sap.com/legal-notice for use terms, disclaimers, disclosures, or restrictions related to SAP Materials for general audiences.

APPENDIX

COMPACT GLOSSARY

About some of the used RAP features – Compact Glossary

→ BACK TO MAIN PRESENTATION

UI Semantics with CDS Annotations

Various UI-specific CDS annotations can be used to solve the most common UI layout tasks in SAP Fiori elements apps built with RAP. UI annotations represent semantic views on business data through the use of specific patterns that are independent of UI technologies. For more information, see Develop UI Specifics | CDS Annotations | SAP Fiori elements Showcase App for RAP And ABAP CDS.

Support for Large Objects (aka OData Streams)

You can enable your RAP application for maintaining large objects (LOBs). By doing so, you provide end users the option to incorporate external binary files or text files when editing entity instances. For more information, see Working with Large Objects

Virtual Elements in CDS Projection Views

Virtual elements represent transient fields in business applications. They are used to define additional CDS elements that are not persisted on the database, but calculated during runtime using ABAP classes that implement the virtual element interface. They are defined at the level of CDS projection views as additional elements within the SELECT list. For more information, see Virtual Elements.

About some of the used RAP features – Compact Glossary (2)

→ BACK TO MAIN PRESENTATION

Late Numbering

Numbering is the process of setting values for primary key fields of entity instances during runtime. Early and late numbering are supported in RAP. In a **late numbering** scenario, the key values are always assigned internally without consumer interaction after the point of no return in the interaction phase has passed, and the SAVE sequence is triggered. For more information, see Numbering.

Validations

Validations are used to ensure the data consistency. Back-end are defined in the BO behavior definitions and implemented in the respective behavior implementation classes. Front-end validations are used to improve the user experience by providing faster feedback and avoiding unnecessary roundtrips. They can be defined in the BO data model using value helps. For more information, see Validations.

Determinations

A determination is an optional part of the business object behavior that modifies instances of business objects based on trigger conditions. A determination is implicitly invoked by the RAP framework if the trigger condition of the determination is fulfilled. Trigger conditions can be modify operations and modified fields. A determination can be triggered on modify or on save. For more information, see Determinations.