



Overview of the ABAP RESTful Programming Model in SAP Cloud Platform ABAP Environment

Product Management Technology Platform, SAP
2019

PUBLIC

Legal disclaimer

The information in this presentation is confidential and proprietary to SAP and may not be disclosed without the permission of SAP. This presentation is not subject to your license agreement or any other service or subscription agreement with SAP. SAP has 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's strategy and possible future developments, products, and platforms, directions, and functionality are all subject to change and may be changed by SAP 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. This document is provided without a warranty of any kind, either express or implied, including but not limited to the implied warranties of merchantability, fitness for a particular purpose, or noninfringement. This document is for informational purposes and may not be incorporated into a contract. SAP assumes no responsibility for errors or omissions in this document, except if such damages were caused by SAP's willful misconduct or gross negligence.

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.

For all recent and planned innovations, potential data protection and privacy features include simplified deletion of personal data, reporting of personal data to an identified data subject, restricted access to personal data, masking of personal data, read access logging to special categories of personal data, change logging of personal data, and consent management mechanisms.

Objectives



After completing this learning module, you will be able to:

- ▶ Understand the mission of the ABAP RESTful Programming Model
- ▶ Understand how the new programming model improves developers live
- ▶ Understand the meaning of business objects and business services

Programming Model: The Mission

... for the development of

- ▶ SAP Fiori apps and
- ▶ Cloud APIs / A2X services...

... for customers and partners in **SAP S/4HANA**

... for SAP internal development in **SAP S/4HANA**

... for customers and partners in
SAP Cloud Platform ABAP Environment

... offering an end-to-end experience

- ▶ standardized development flow
- ▶ best practices & development guides
- ▶ high development efficiency
- ▶ focus on business logic, rather than technical aspects

PROVIDE A PROGRAMMING MODEL ...

... supporting the product qualities

- ▶ User experience: SAP Fiori and SAP HANA
- ▶ Cloud: scalability
- ▶ ABAP assets: non-functional requirements

... and being flexible

- ▶ Break-outs for non-standardized implementations
- ▶ Out-of-the-box extensibility & verticalization
- ▶ Reuse components / APIs in “freestyle” development scenarios

Agenda

The ABAP RESTful Programming Model

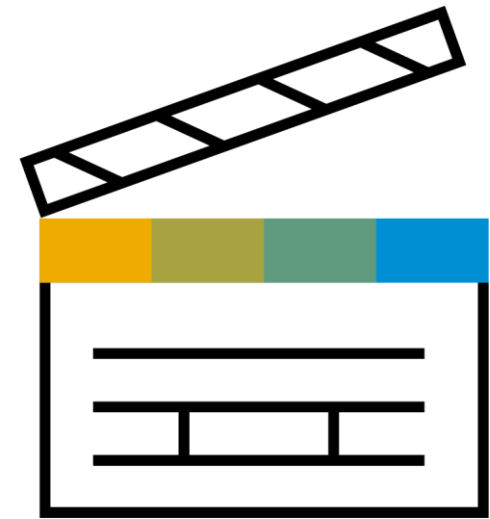
Why and what?

Business Objects

Definition, Demo, Roadmap

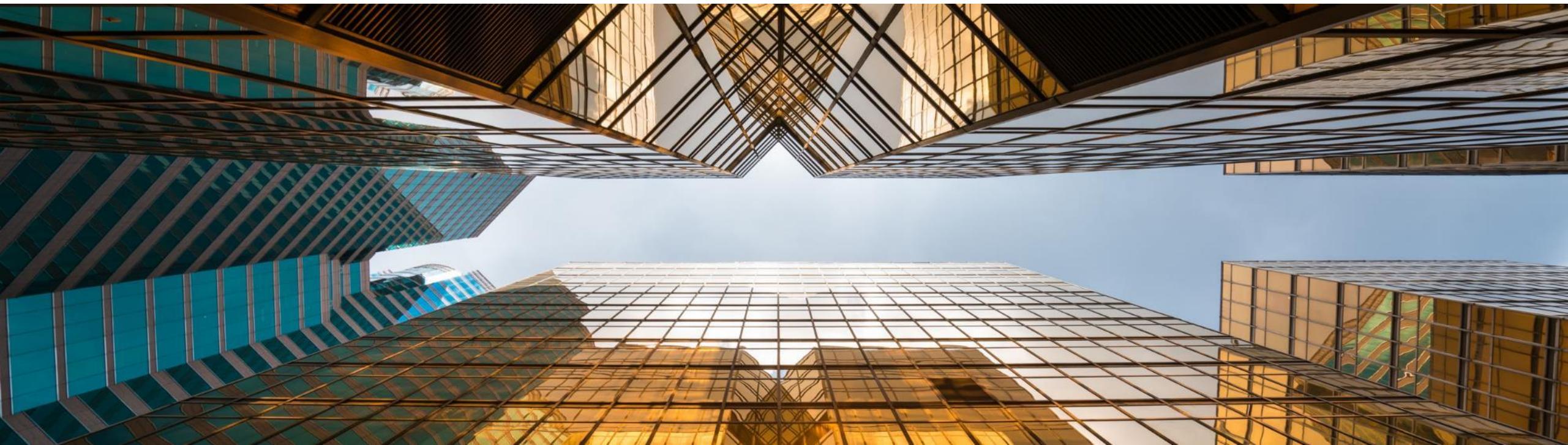
Business Services

Definition, Demo, Roadmap



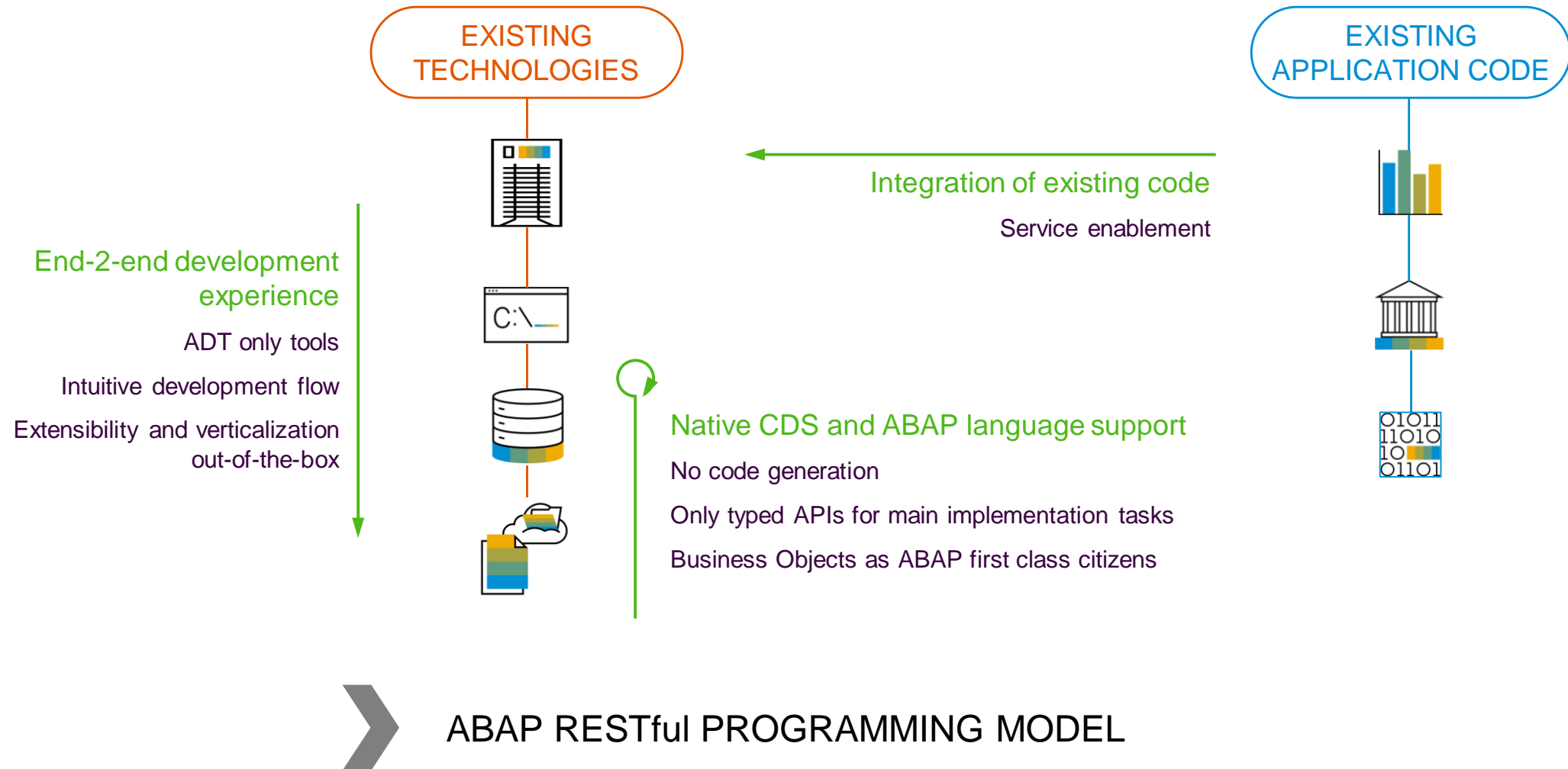
The ABAP RESTful Programming Model

Why and what?



Improvement of the ABAP Programming Model

Integration of existing code and Development Efficiency



Guiding Principles

ABAP RESTful Programming Model

All standard development tasks...

... are based on ABAP Development Tools in Eclipse

... with an end-to-end development experience

... with native CDS / ABAP language support (for 80% use cases)

- ▶ Typed signatures
- ▶ Static code checks
- ▶ Auto-completion, Element Info...

There is

... **NO** use of generic framework APIs

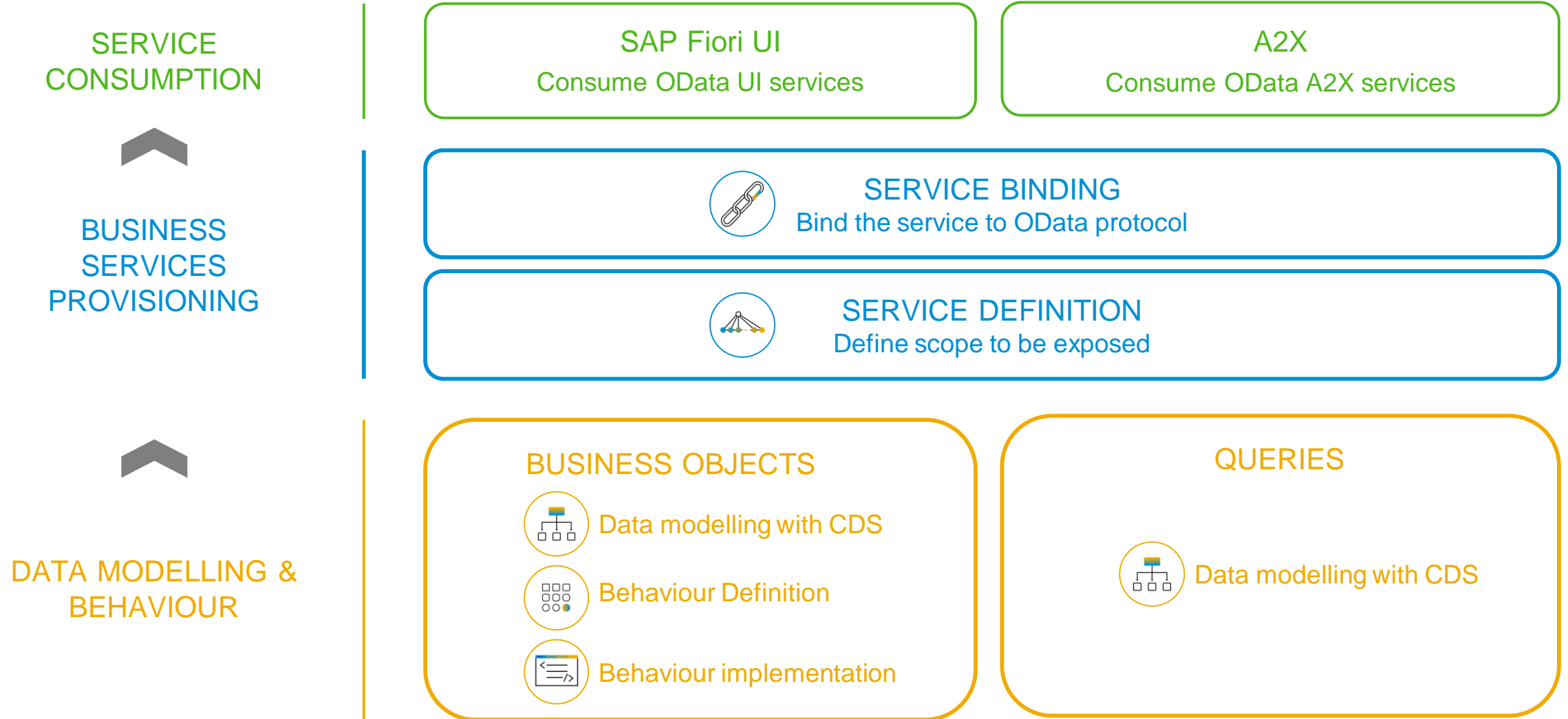
... **NO** code generation

... **NO** redundancy in transport-objects

- ▶ Improved lifecycle (transport, patches,...)

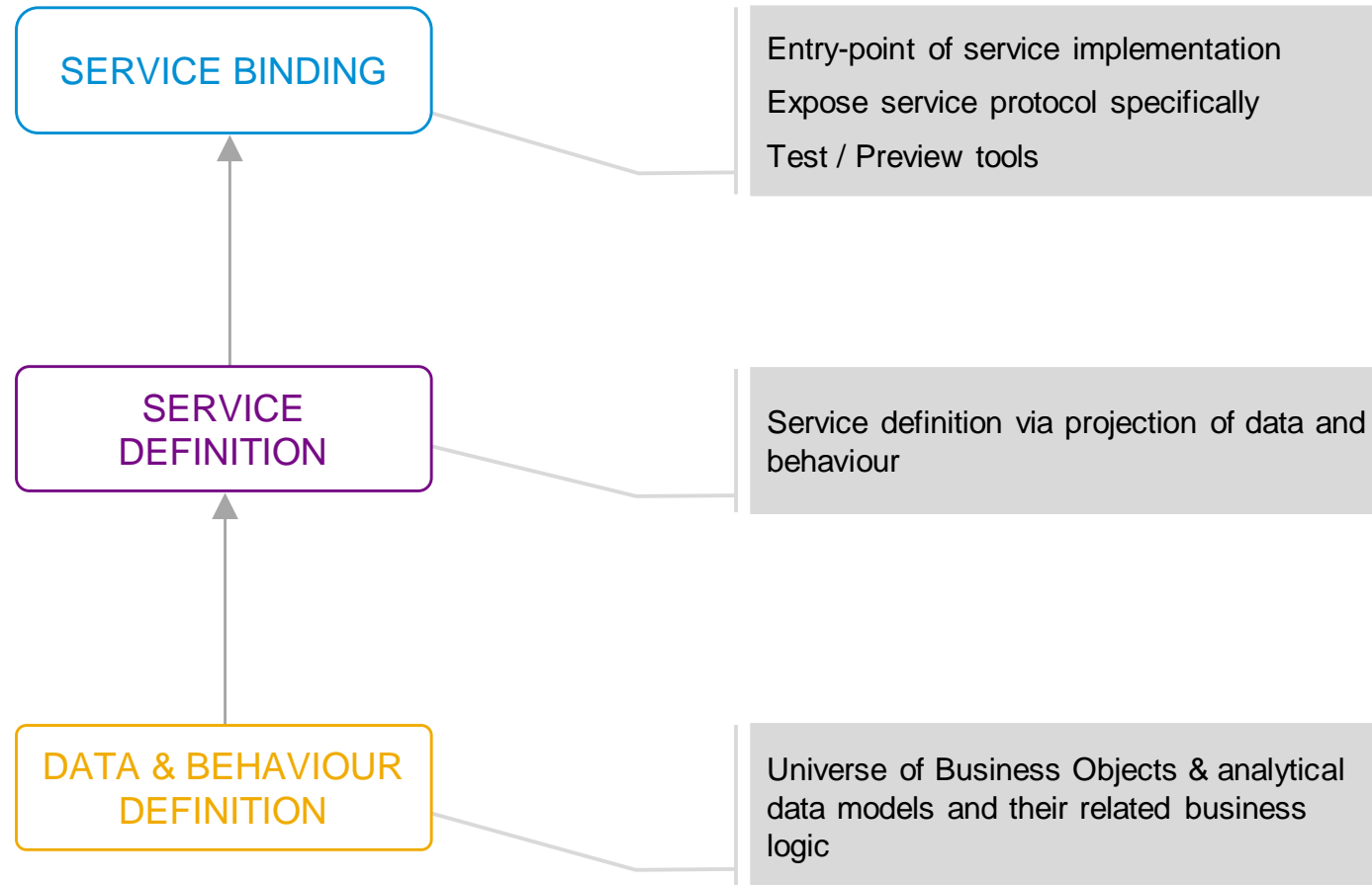


ABAP RESTful Programming Model

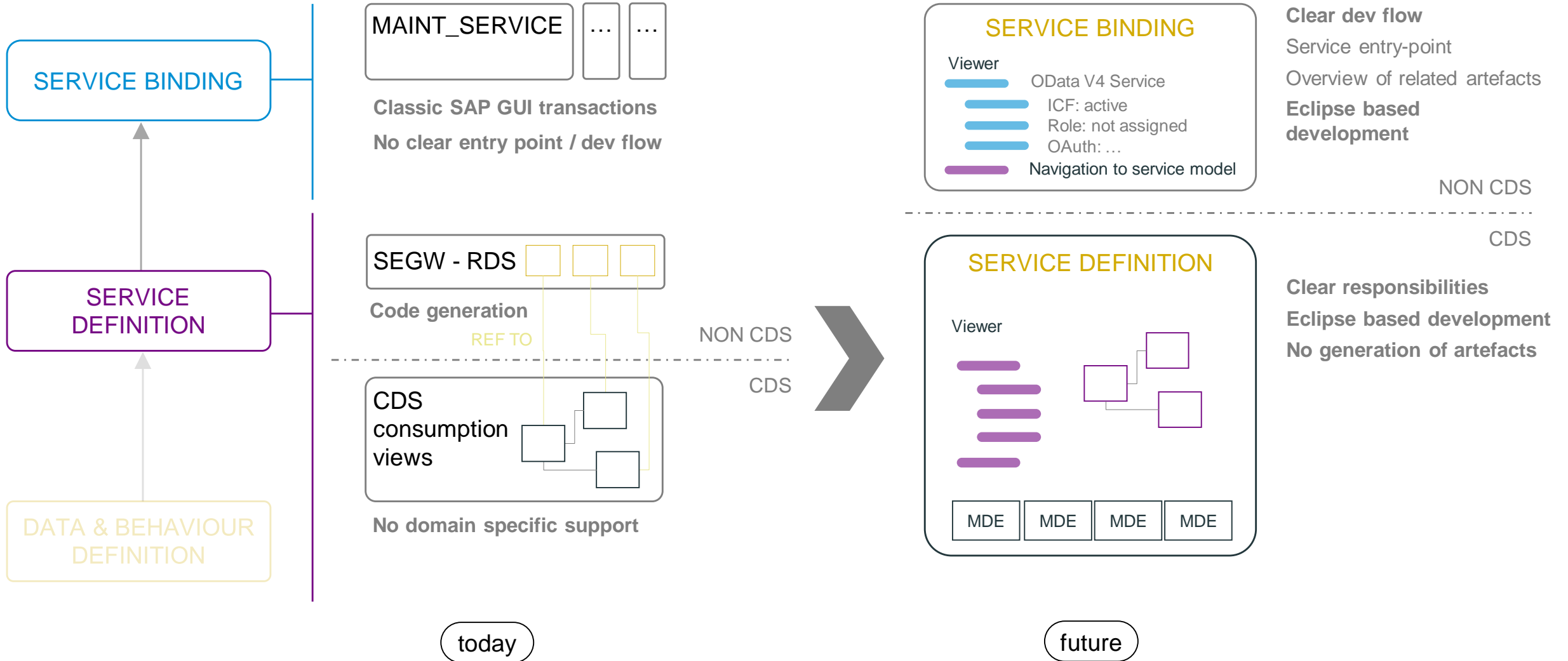


ABAP RESTful Programming Model

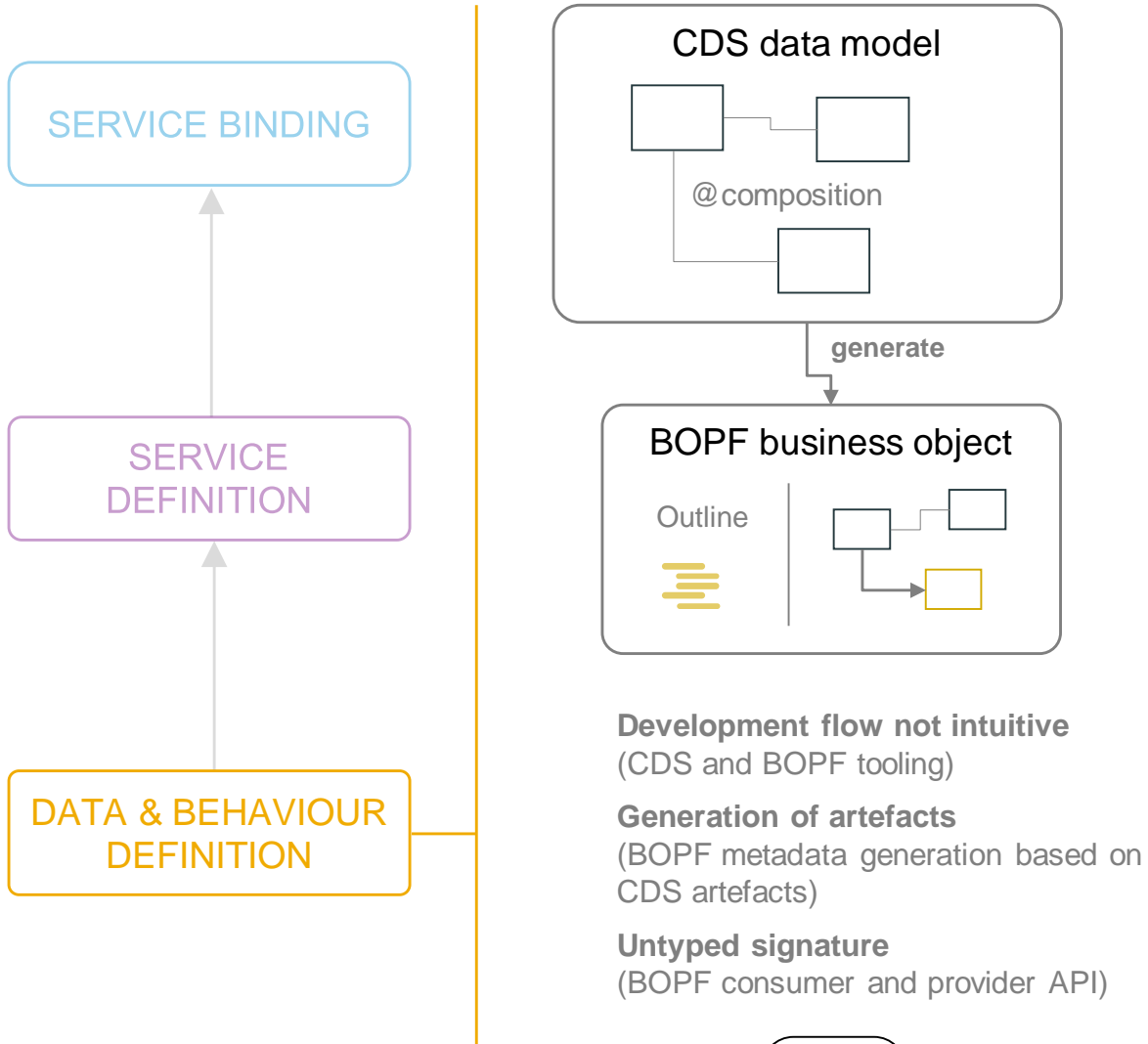
Business Services



ABAP RESTful Programming Model Business Services



ABAP RESTful Programming Model Business Objects

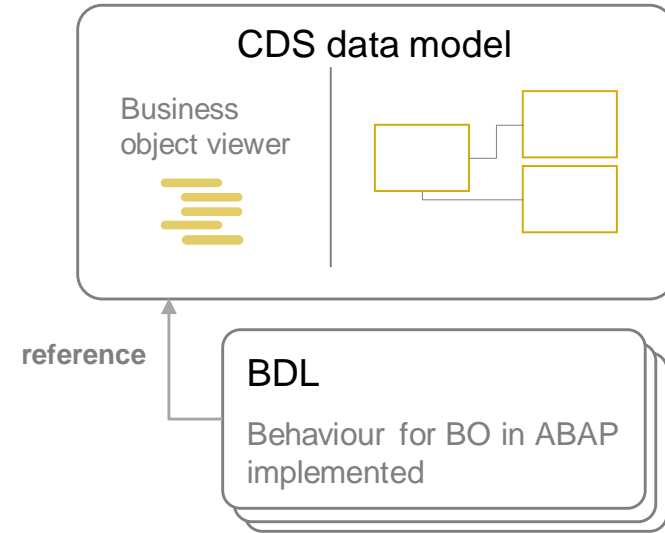


Development flow not intuitive
(CDS and BOPF tooling)

Generation of artefacts
(BOPF metadata generation based on CDS artefacts)

Untyped signature
(BOPF consumer and provider API)

today



Clear development flow

(CDS for data modelling with ABAP language extensions for behaviour)

No redundancy and no generation of artefacts

(clear references on language level)

Typed APIs

(For BO consumer and provider)

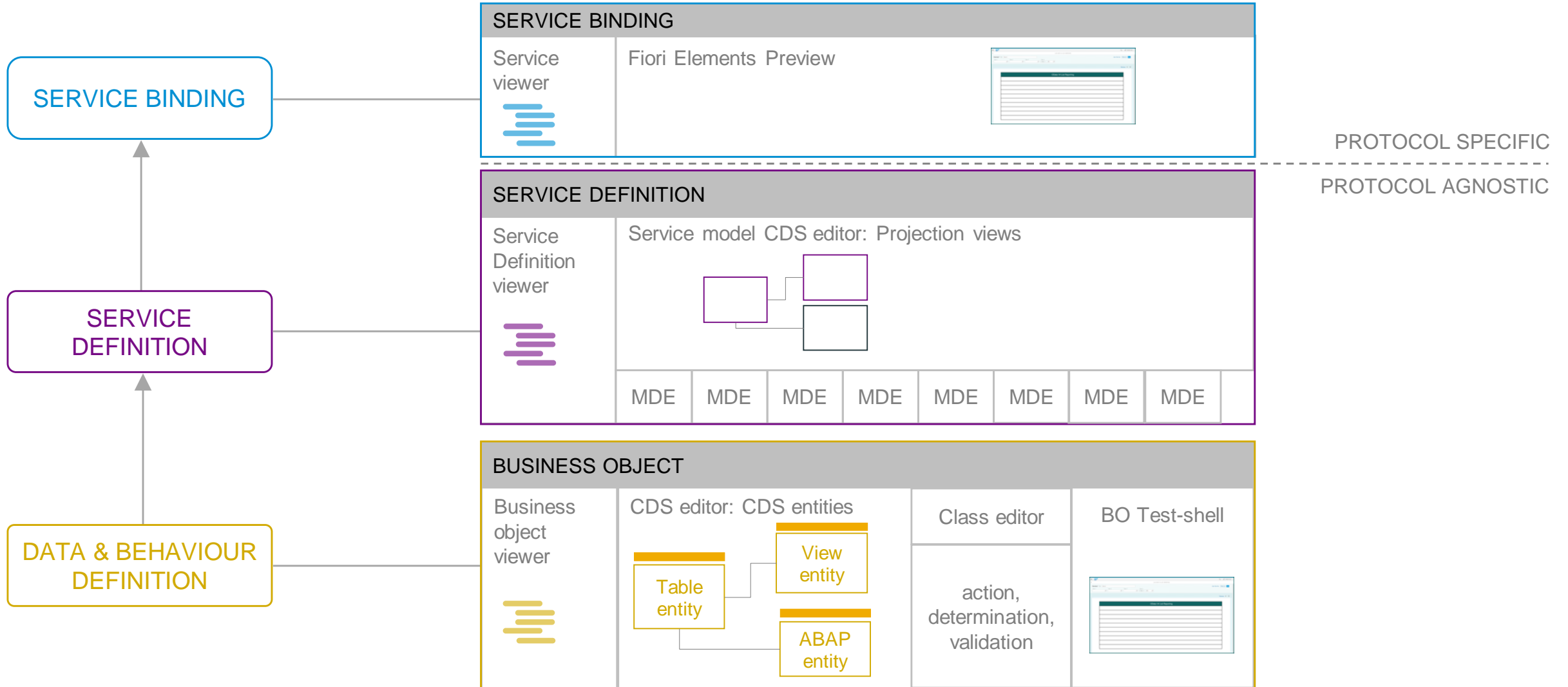
→ **The Business Object becomes an ABAP first class citizen**

(Known by ABAP compiler, language and tools including static checks)

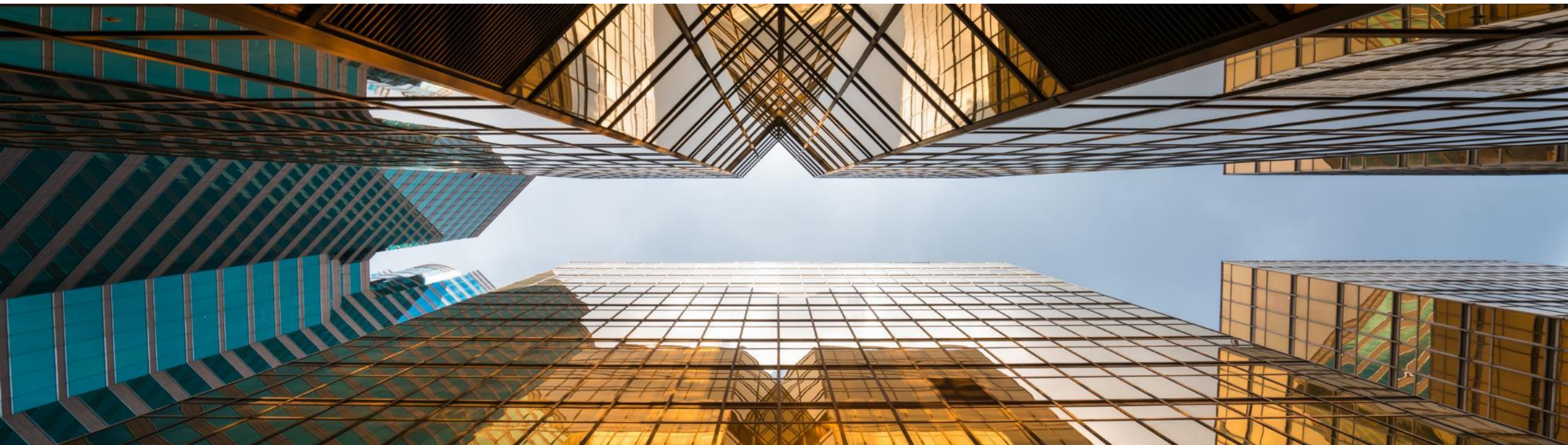
future

ABAP RESTful Programming Model

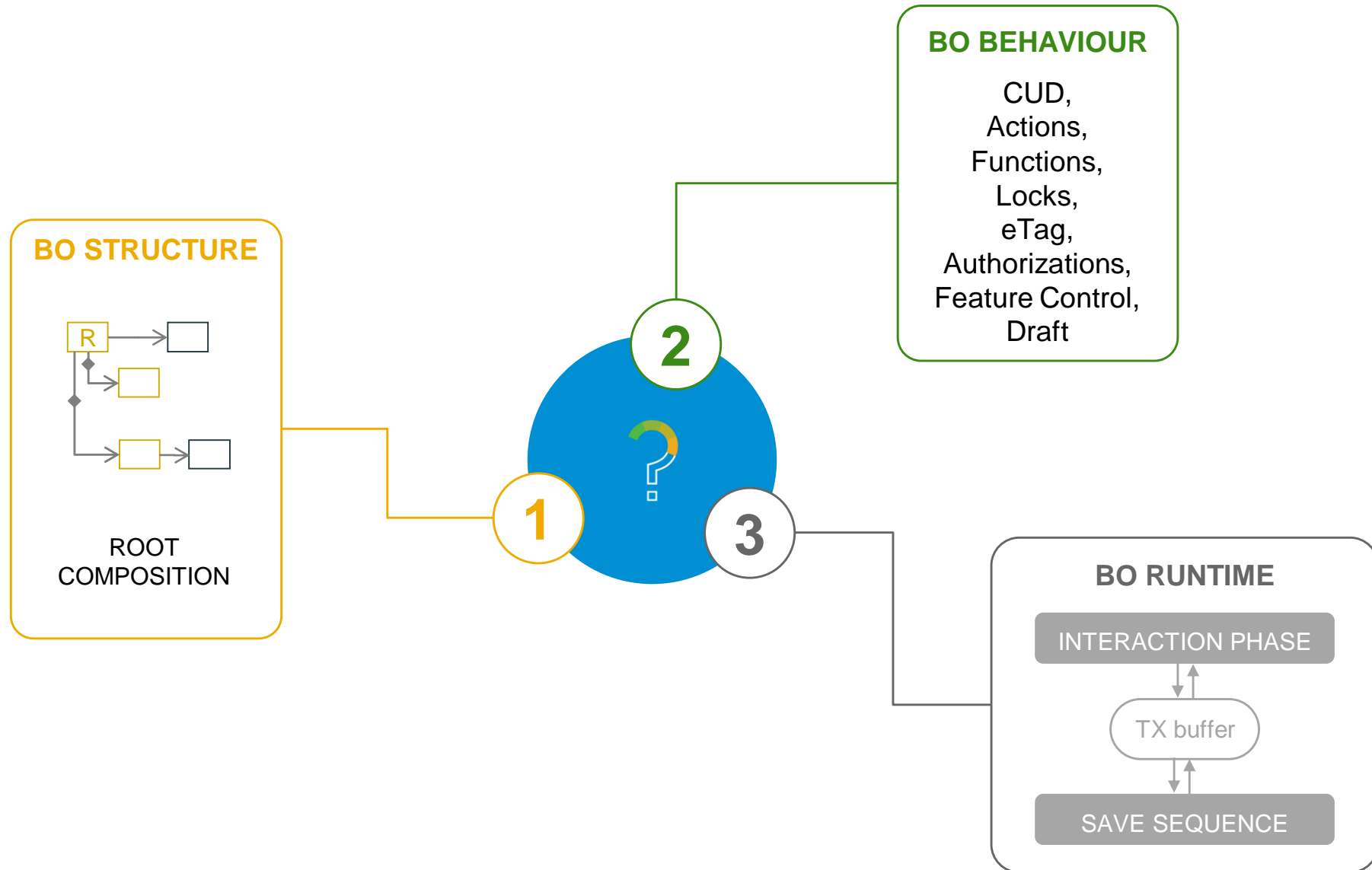
Big Picture



BUSINESS OBJECTS



What is a Business Object (BO)?

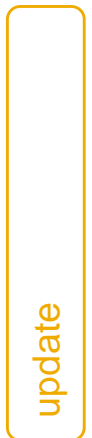


Business Object Runtime

**BUSINESS
OBJECT
RUNTIME**

INTERACTION PHASE

MODIFY



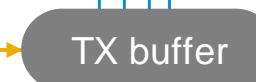
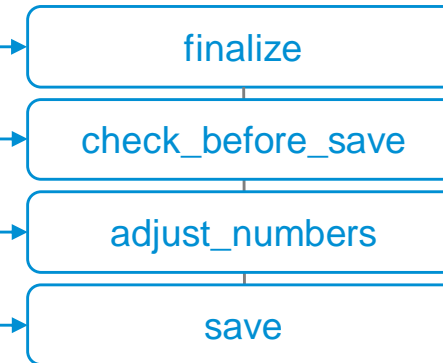
READ



LOCK

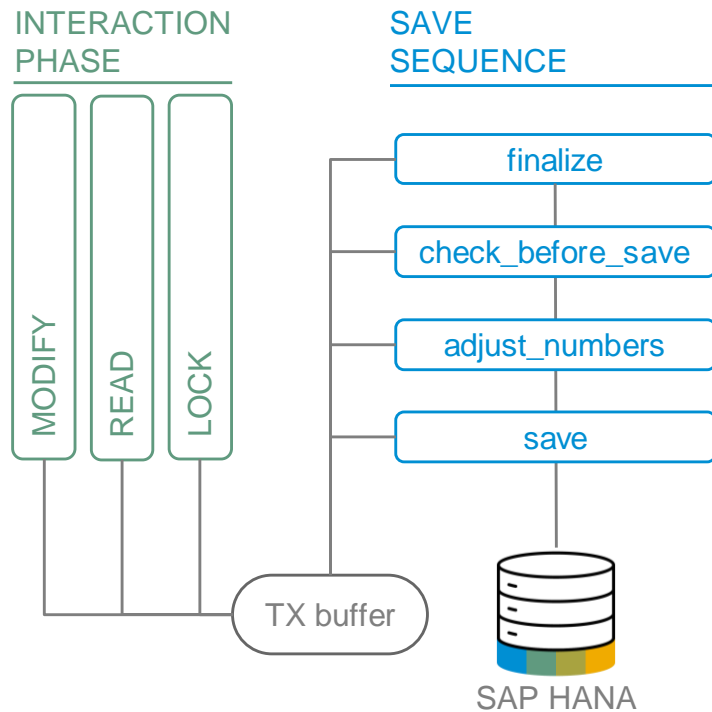


SAVE SEQUENCE



SAP HANA

Business Objects – Status Overview



1

UNMANAGED

Application coding available
(e.g. Purchase Order, Sales Order,...)

2

MANAGED

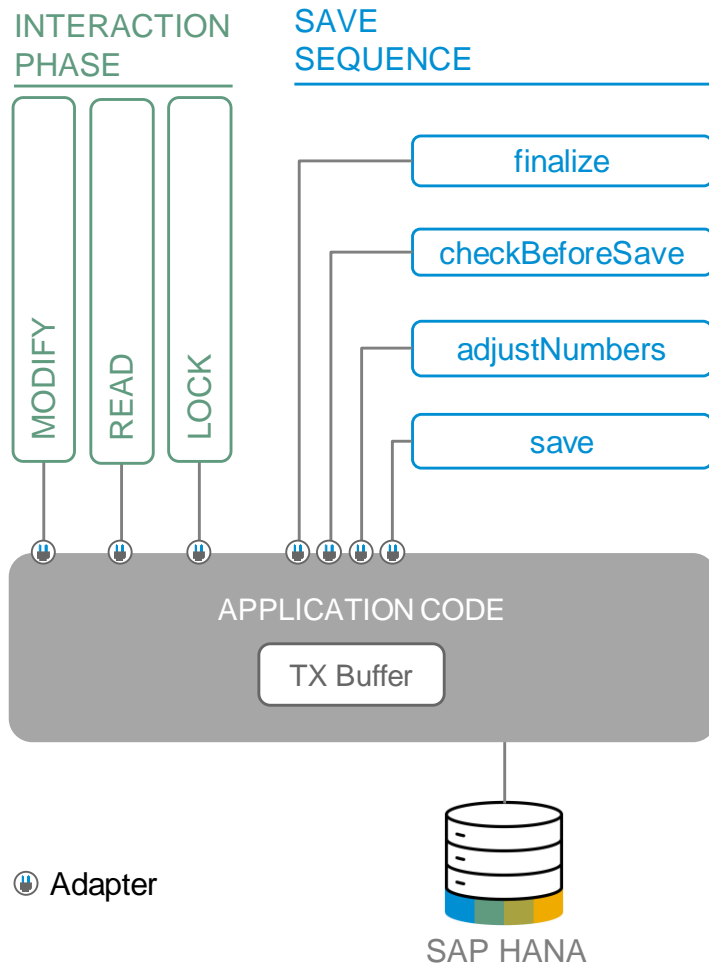
Green field development

3

(MANAGED WITH) SAVE SELF-IMPLEMENTED

Update task function module available
(e.g. Business Partner, Product,..)

Business Objects - Unmanaged



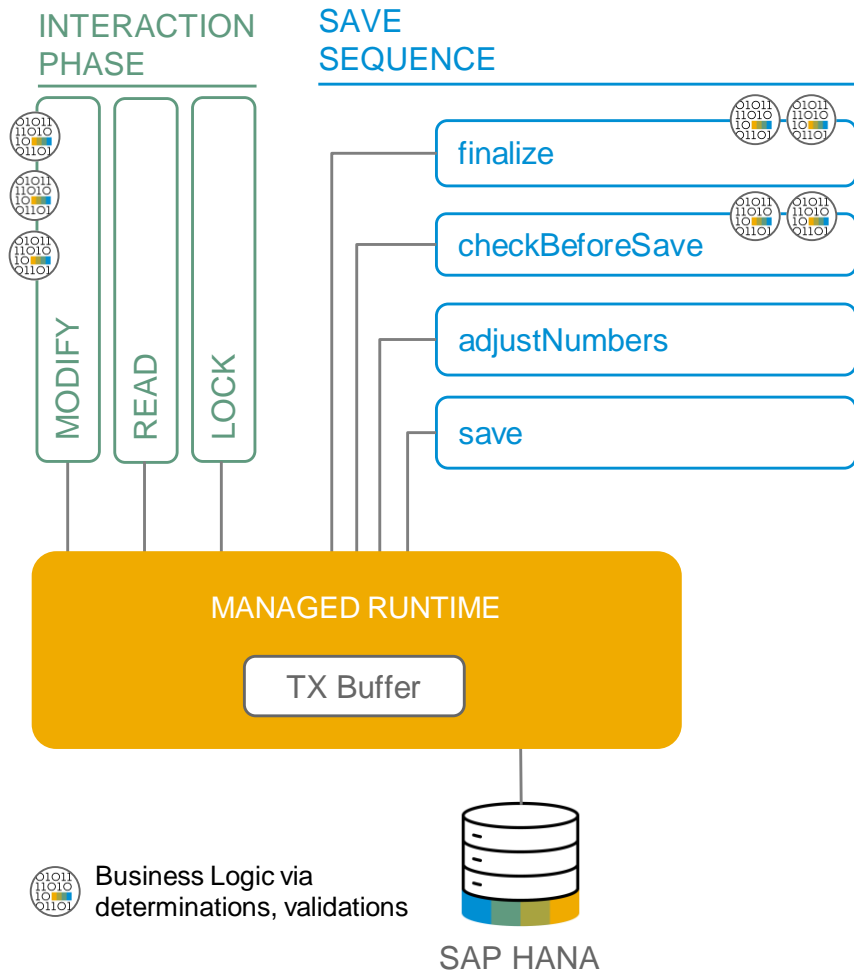
Application coding

- ▶ already available
- ▶ for interaction phase, transactional buffer and save sequence
- ▶ decoupled from UI technology

Examples

- ▶ Sales Order, Purchase Order

Business Objects – Managed



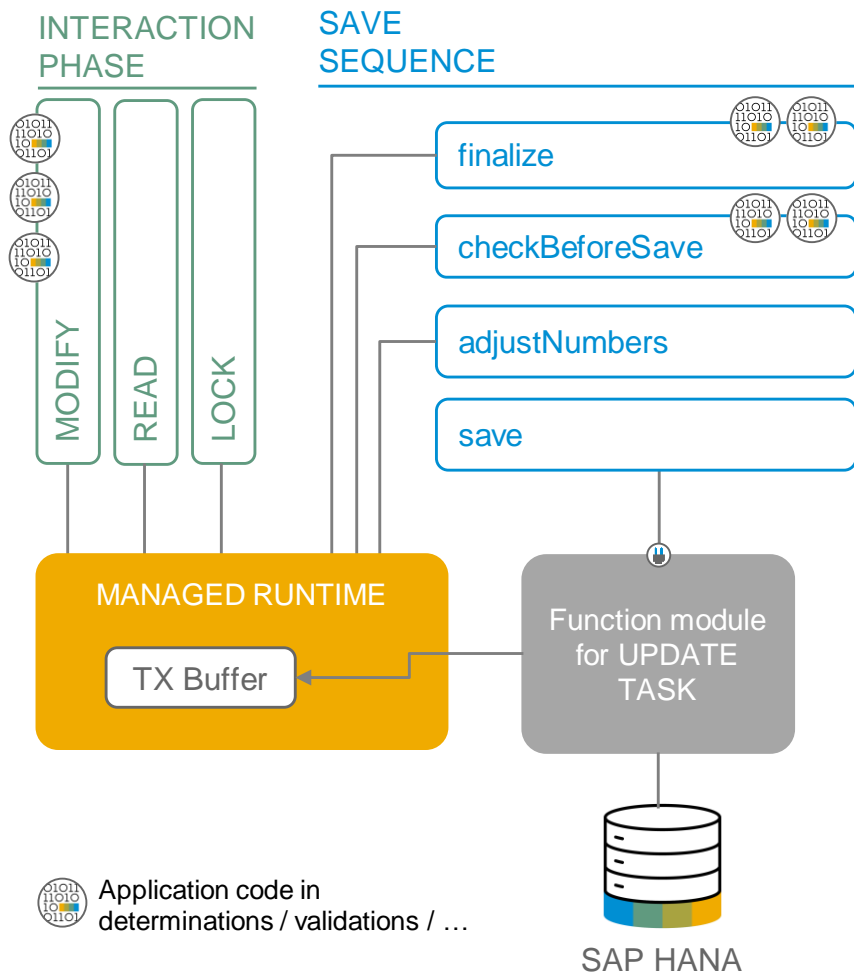
Application coding

- ▶ not yet available or fine granular reusable code available
- ▶ technical implementation aspects taken over by BO infrastructure
- ▶ developer focus on business logic, implemented via determinations, validation, actions,...

Examples

- ▶ New applications in SAP Cloud Platform ABAP Environment

Business Objects – Save Unmanaged



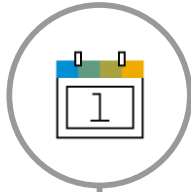
Application coding

- ▶ “update-task function module” available
- ▶ coding for interaction phase not available (e.g. highly coupled in older UI technology: DYNP - PBO / PAI)
- ▶ technical implementation aspects to be taken over by BO infrastructure

Examples

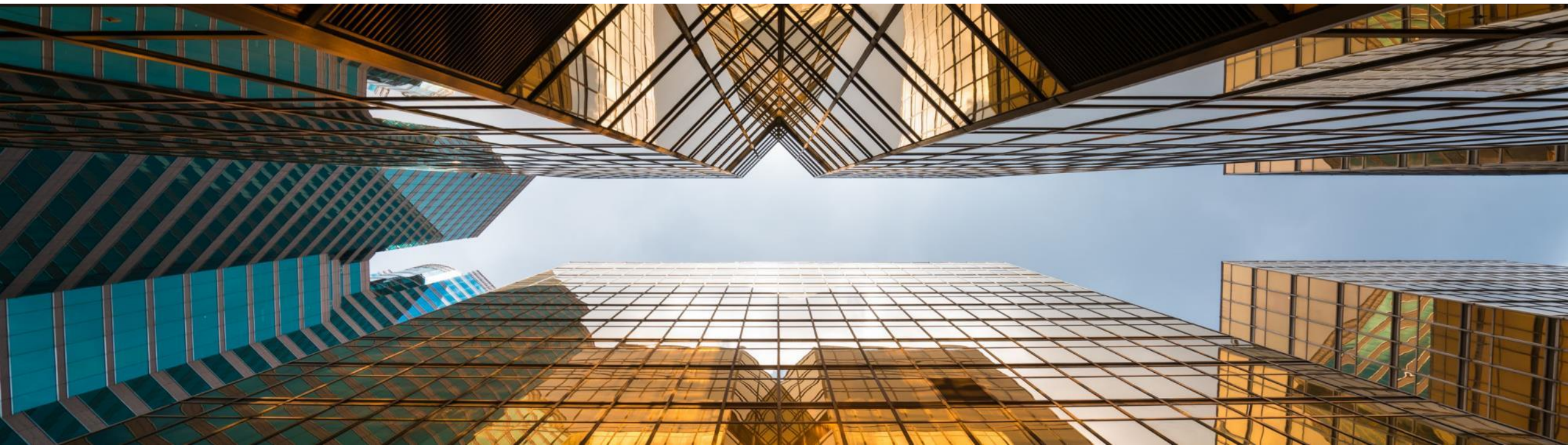
- ▶ Business Partner, Product

Business Objects - Roadmap

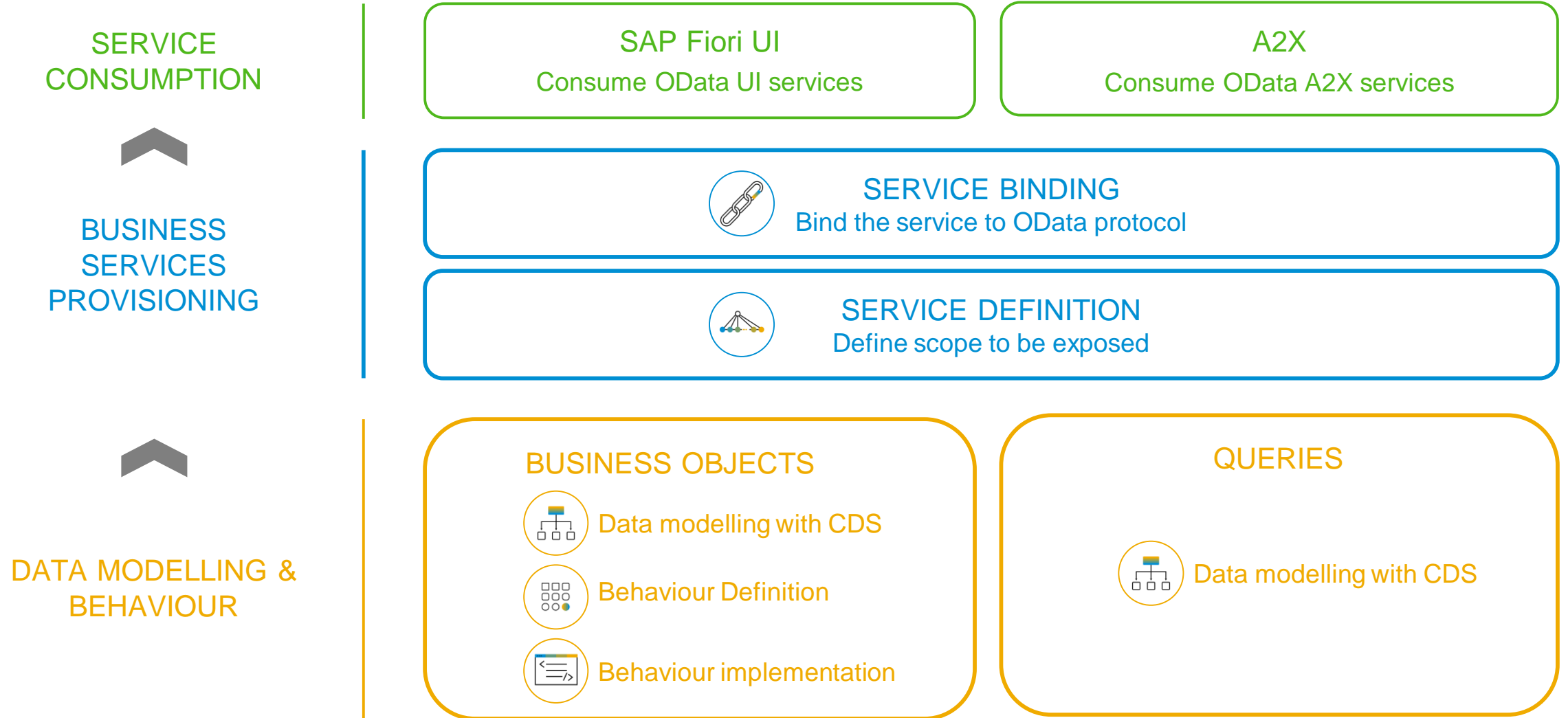


- 1 ADT development experience: Business Object *onboarding*
- 2 Support for managed BO runtime (with unmanaged save)
- 3 Business Object Access API
- 4 Draft, feature control (incl. authorizations)
- 5 Role-specific service projections
- 6 Support for managed BO runtime: determinations and validations

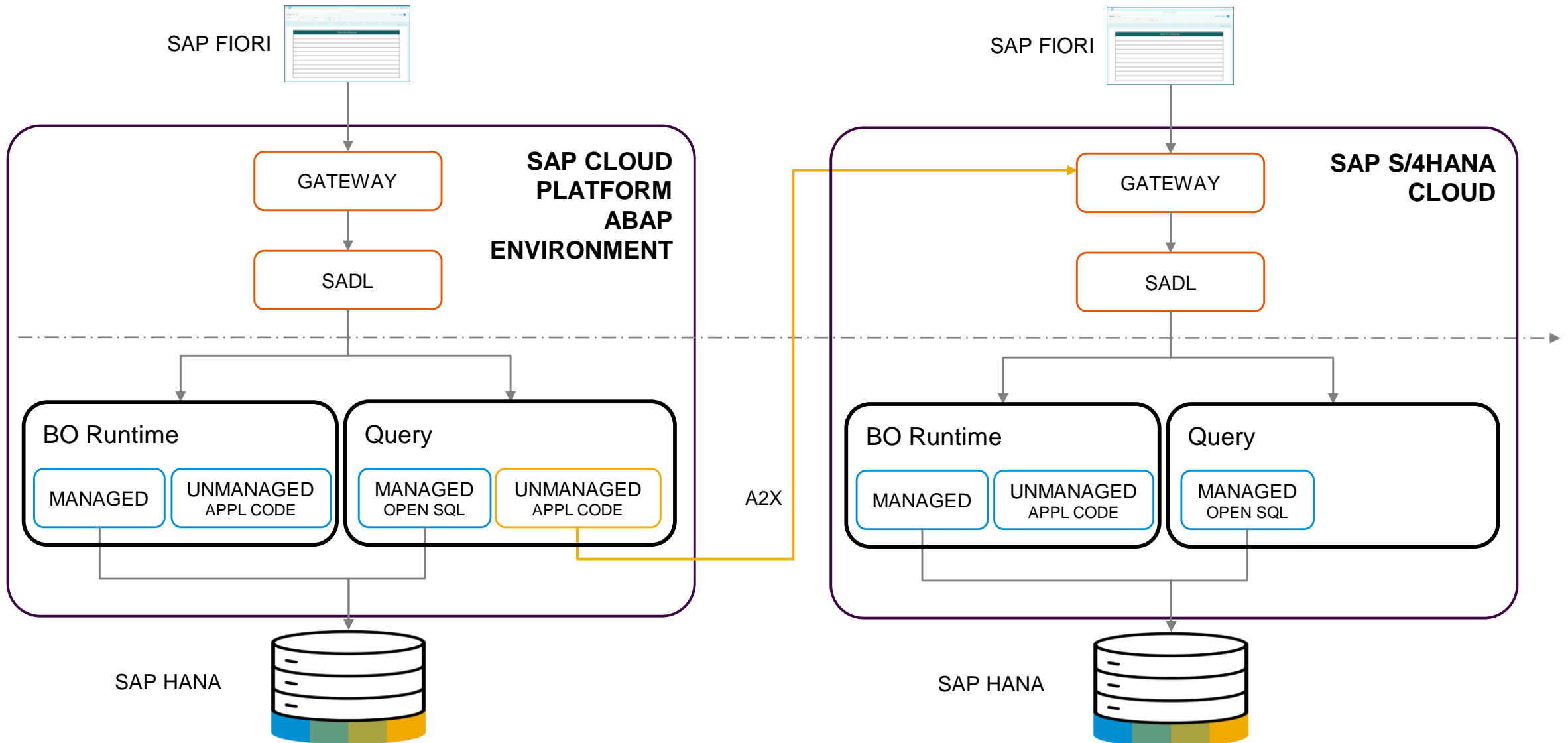
BUSINESS SERVICES



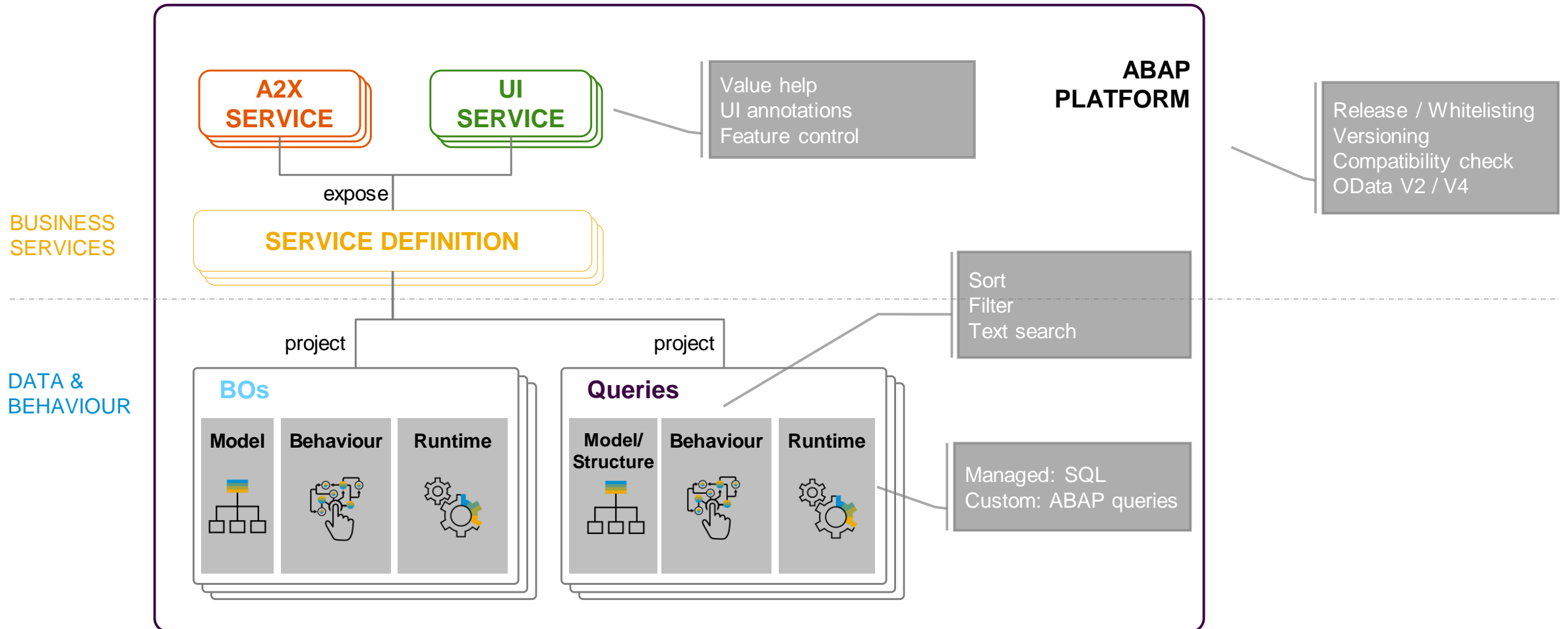
ABAP RESTful Programming Model



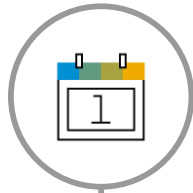
Business Services Big Picture– Runtime stack



Business Services Big Picture– Design Time stack



Business Services - Roadmap



- 1 A2X OData V4 outbound services
- 2 Versioning, release and compatibility check for A2X OData V2 and OData V4
- 3 Inbound services for A2X consumption of OData V2 and OData V4
- 4 SAP Fiori UI OData V4 services

Thank you.

Contact information:

Product Management Technology Platform, SAP