Build modern apps with the ABAP RESTful Application Programming Model (RAP) – Overview and Outlook

Carine Tchoutouo Djomo, André Fischer, SAP
January 25, 2022

PUBLIC

THE BEST RUN SAP

Speakers

Carine Tchoutouo Djomo
Product Manager for ABAP Platform

André Fischer
Product Manager for ABAP Platform and SAP Gateway

Disclaimer

The information in this presentation is confidential and proprietary to SAP and may not be disclosed without the permission of SAP. Except for your obligation to protect confidential information, 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 presentation or any related document, or to develop or release any functionality mentioned therein.

This presentation, or any related document and SAP's strategy and possible future developments, products and or 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 presentation is not a commitment, promise or legal obligation to deliver any material, code or functionality. This presentation 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 non-infringement. This presentation is for informational purposes and may not be incorporated into a contract. SAP assumes no responsibility for errors or omissions in this presentation, except if such damages were caused by SAP’s intentional 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.

Agenda

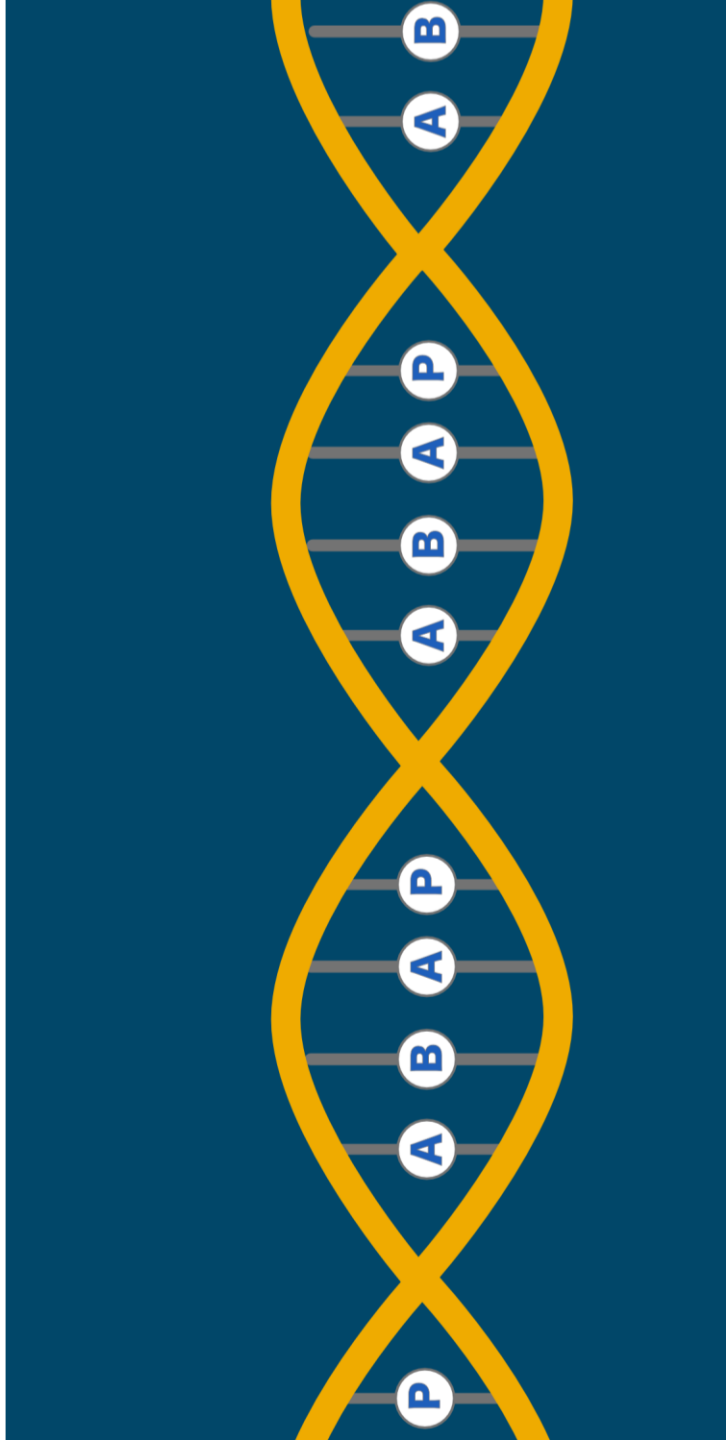
INTRODUCTION

BIG PICTURE

DEMO

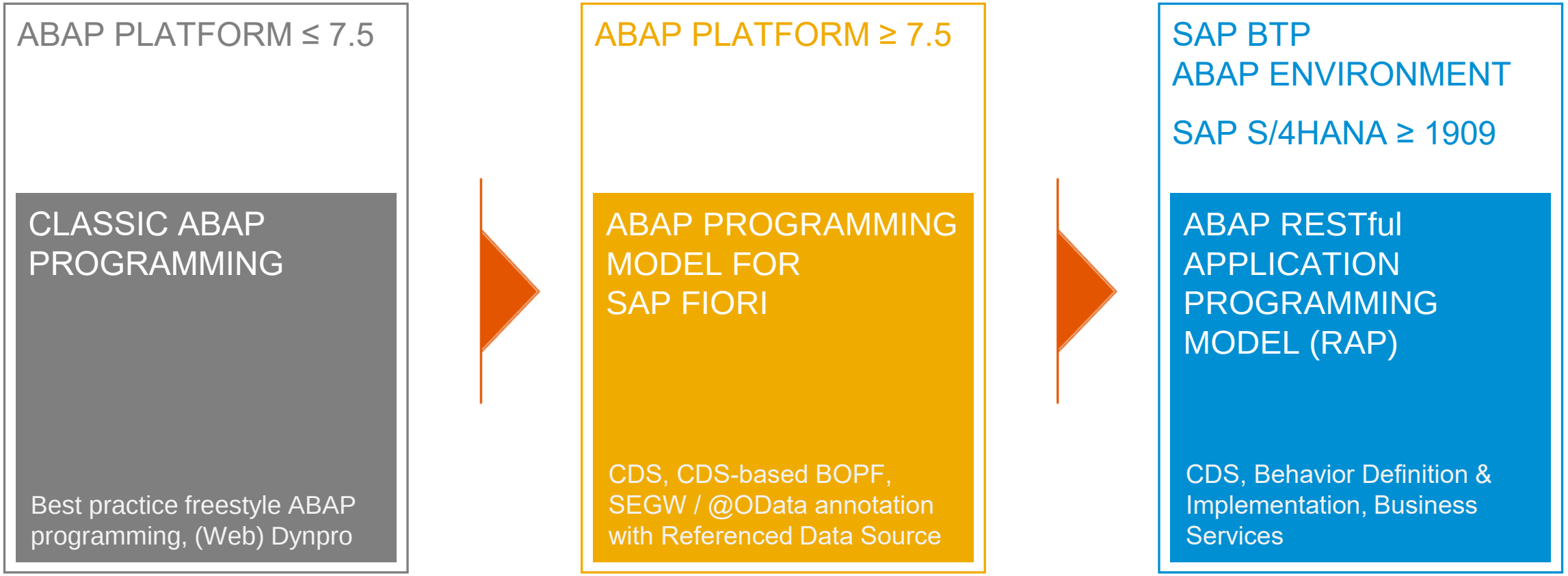
WHAT'S NEW

SUMMARY & OUTLOOK



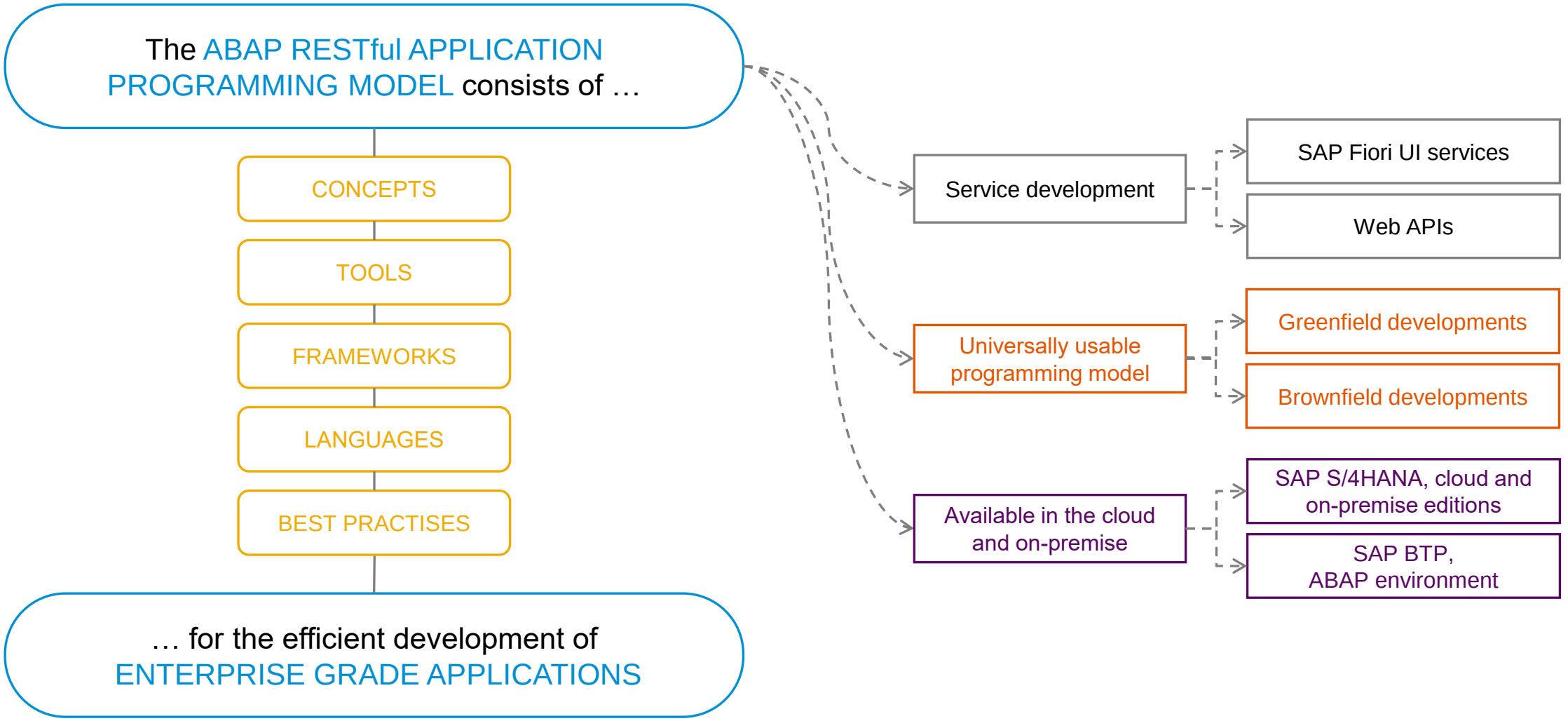
INTRODUCTION

Evolution of the ABAP programming model



Safe investments!

ABAP RESTful Application Programming Model (RAP) – At a glance



The key players

ABAP Development Tools in Eclipse for all development tasks

Easy developer onboarding
End-to-end development flow

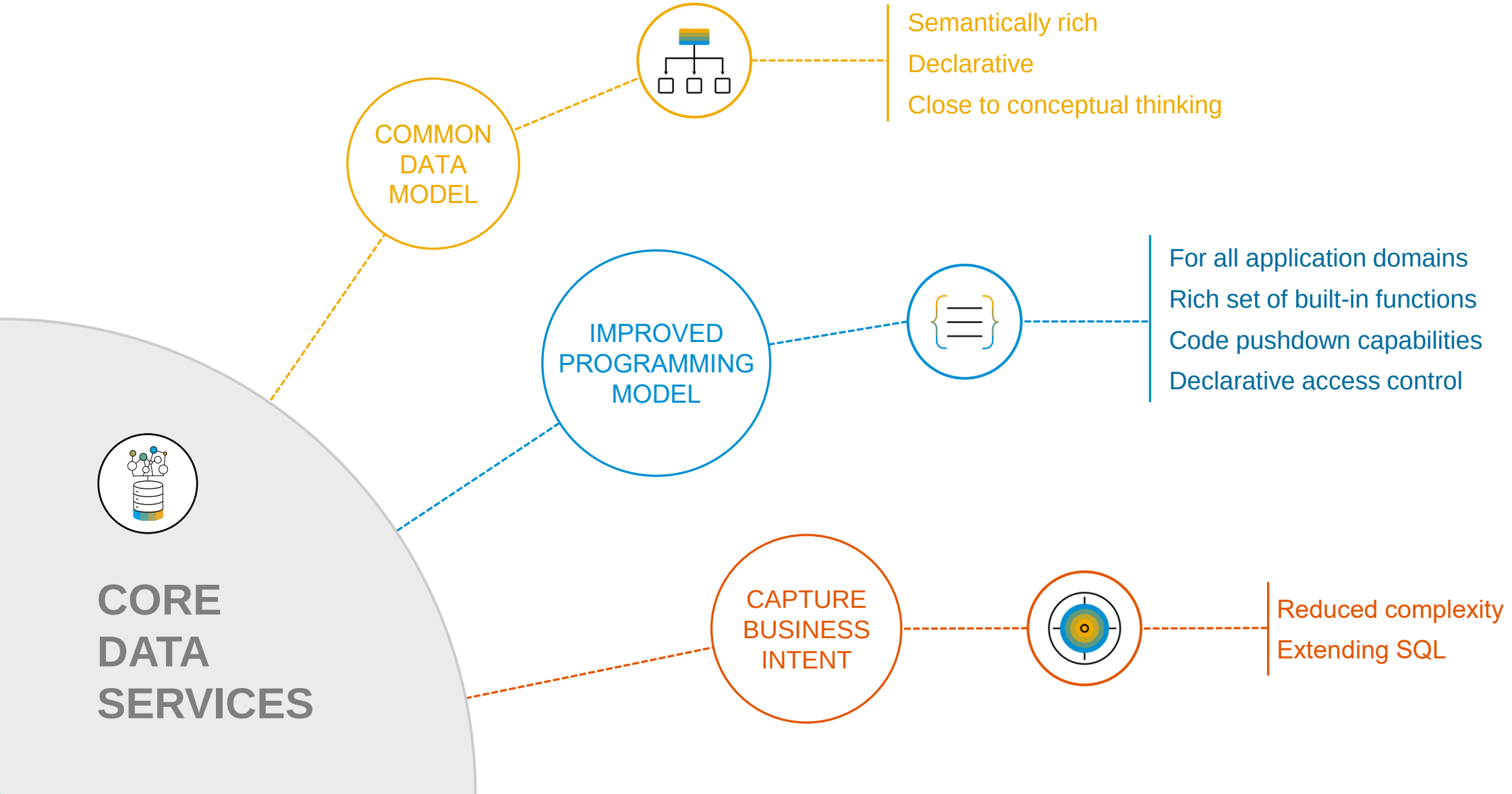
Languages: ABAP and Core Data Services (CDS)

Standard implementation tasks via typed APIs supporting
static code checks, auto-completion, element info

Powerful frameworks

Take over technical implementation tasks
Business logic added in code exits on protocol-agnostic layers

Next generation data modeling and access with ABAP CDS



Next generation data modeling and access with ABAP CDS – Example

View annotations

View definition

Associations

selection

Data source

Element annotations

```
/DMO/I_TRAVEL_D
1  @EndUserText.label: 'Travel View Entity for Draft RefScen'
2  @AccessControl.authorizationCheck: #NOT_REQUIRED
3
4  define root view entity /DMO/I_Travel_D
5    as select from /dmo/a_travel_d
6
7    composition [0..*] of /DMO/I_Booking_D        as _Booking
8
9    association [0..1] to /DMO/I_Agency           as _Agency        on $projection.AgencyID = _Agency.AgencyID
10   association [0..1] to /DMO/I_Customer         as _Customer      on $projection.CustomerID = _Customer.CustomerID
11   association [1..1] to /DMO/I_Overall_Status_VH as _OverallStatus on $projection.OverallStatus = _OverallStatus.OverallStatus
12   association [0..1] to I_Currency              as _Currency      on $projection.CurrencyCode = _Currency.Currency
13
14 { ///dmo/a_travel_d
15   key travel_uuid           as TravelUUID,
16
17       travel_id             as TravelID,
18       agency_id             as AgencyID,
19       customer_id           as CustomerID,
20       begin_date            as BeginDate,
21       end_date              as EndDate,
22       @Semantics.amount.currencyCode: 'CurrencyCode'
23       booking_fee           as BookingFee,
24       @Semantics.amount.currencyCode: 'CurrencyCode'
25       total_price           as TotalPrice,
26       currency_code         as CurrencyCode,
27       description           as Description,
28       overall_status        as OverallStatus,
29
40       //total etag field
41       @Semantics.systemDateTime.lastChangedAt: true
42       last_changed_at       as LastChangedAt,
43
44       //Associations
45       _Booking,
46
47       _Agency,
48       _Customer,
49       _OverallStatus,
50       _Currency
51 }
```

ABAP Flight Reference Scenario -
Example available in package /DMO/FLIGHT_DRAFT

Entity Manipulation Language (EML) – Extension of the ABAP Language

Extension of the ABAP Language with SQL-like syntax

Used to control the transactional BO behavior

Direct API-based access to RAP BOs

Data consistency ensured by database LUW

Available in the ABAP keyword documentation

EXAMPLES

```
"Modify travel instance
MODIFY ENTITIES OF /DMO/I_Travel_D IN LOCAL MODE
  ENTITY Travel
    UPDATE FIELDS (  OverallStatus )
    WITH VALUE #( FOR key IN keys ( %tky          = key-%tky
                                    OverallStatus = travel_status-accepted ) ).
```

```
"Read changed data for action result
READ ENTITIES OF /DMO/I_Travel_D IN LOCAL MODE
  ENTITY Travel
    ALL FIELDS WITH
    CORRESPONDING #( keys )
  RESULT DATA(travels).

result = VALUE #( FOR travel IN travels ( %tky   = travel-%tky
                                          %param = travel ) ).
```

BIG PICTURE

The big picture

CONSUMPTION

DATA INTEGRATION
Consume SQL based services

WEB APIs
Consume OData based services

SAP FIORI UIs
Consume OData based UI services

SAP ANALYTICS CLOUD
Consume InA based UI services for live data access

BUSINESS SERVICE PROVISIONING

SERVICE BINDING – Bind to protocol version and scenario

SERVICE DEFINITION – Define scope to be exposed

BUSINESS OBJECT PROJECTION

CDS: BO projection views


BDEF: Behavior projection


ABAP: Behavior implementation

ANALYTICAL PROJECTION

CDS: Analytical projection views

DATA MODELING & BEHAVIOR

CDS ENTITIES

CDS: Data modeling

BUSINESS OBJECTS

CDS: Data modeling


BDEF: Behavior definition


ABAP: Behavior implementation

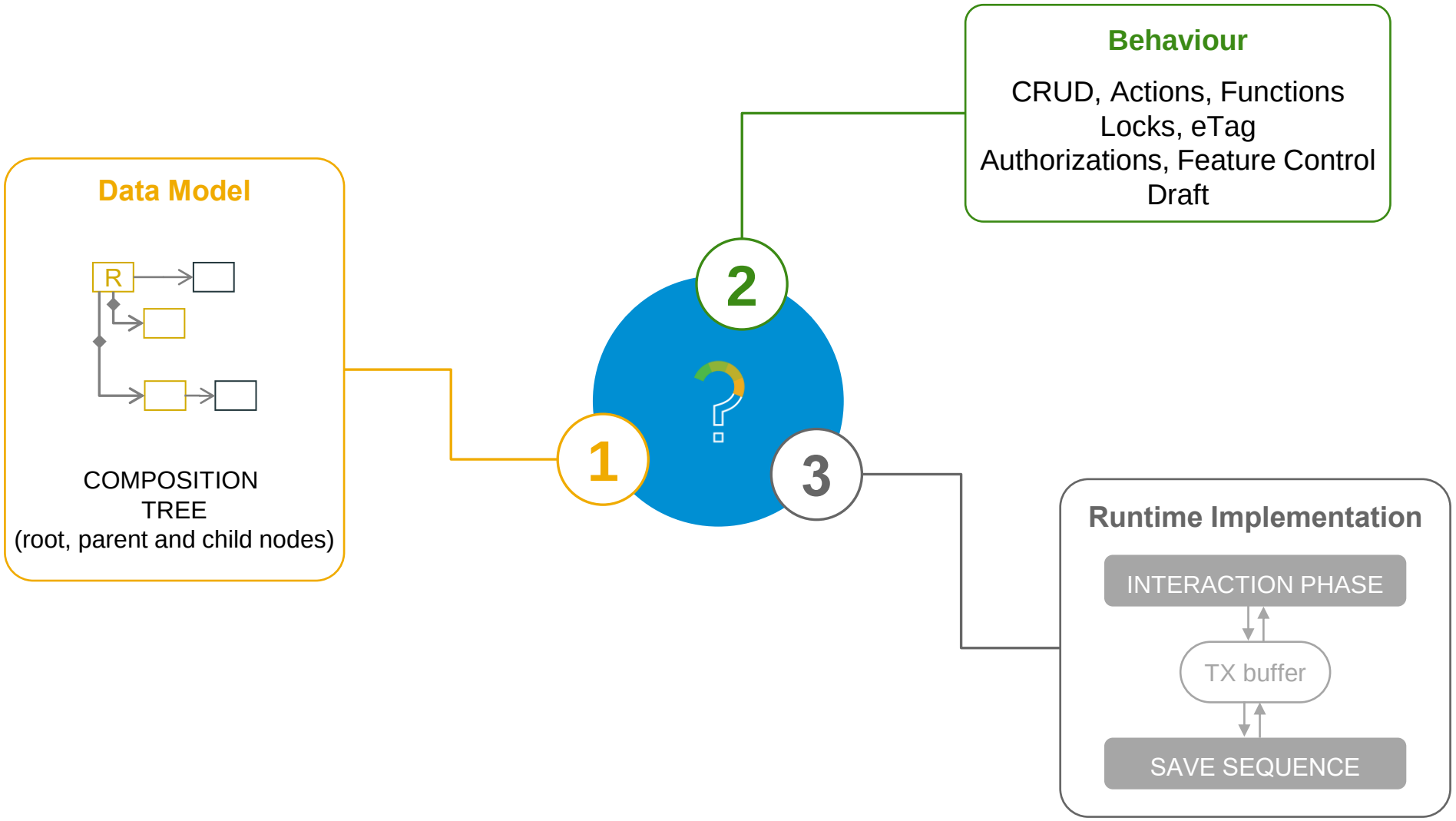
ANALYTICAL MODEL

CDS: Analytical cubes

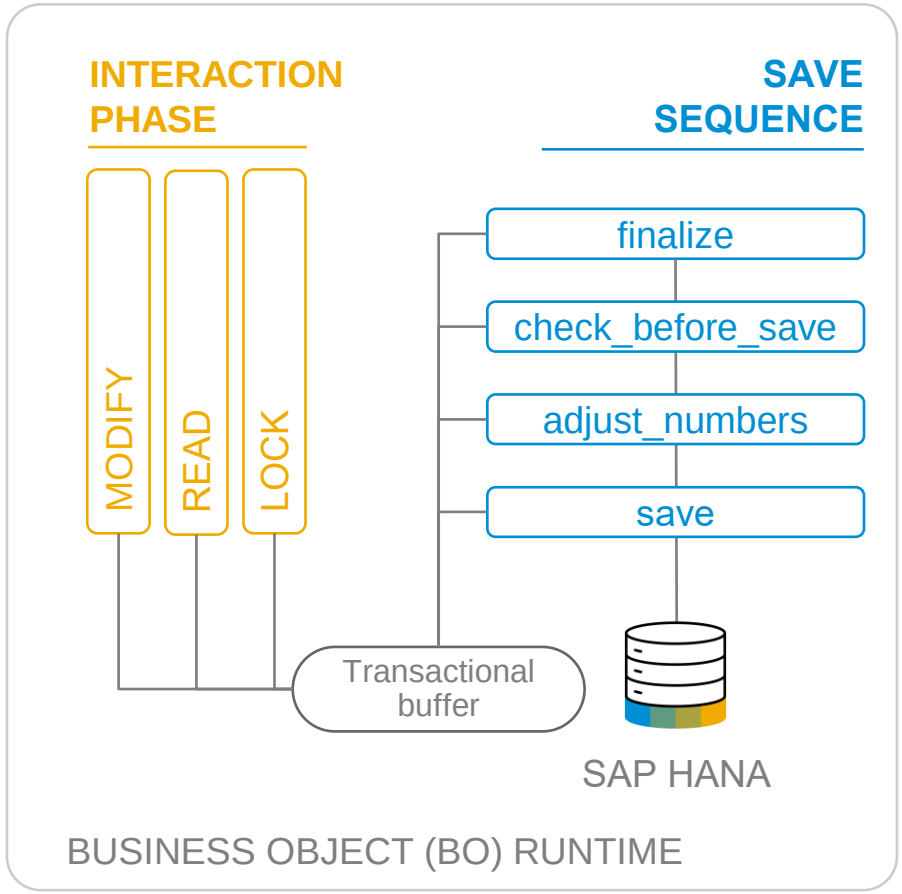

CDS: Analytical dimensions

CDS ABAP/EML

What is a business object (BO) ?



BO runtime implementation types



UNMANAGED

For brownfield developments with available application code for interaction phase, transactional buffer, and save sequence

➔ Developers in charge of BO runtime: CRUD operations

➔ Adapters needed to integrate the existing code

MANAGED

For greenfield developments with standard implementation
(opt. unmanaged appl. components: DB tables, lock/PFCG object, update task FM)

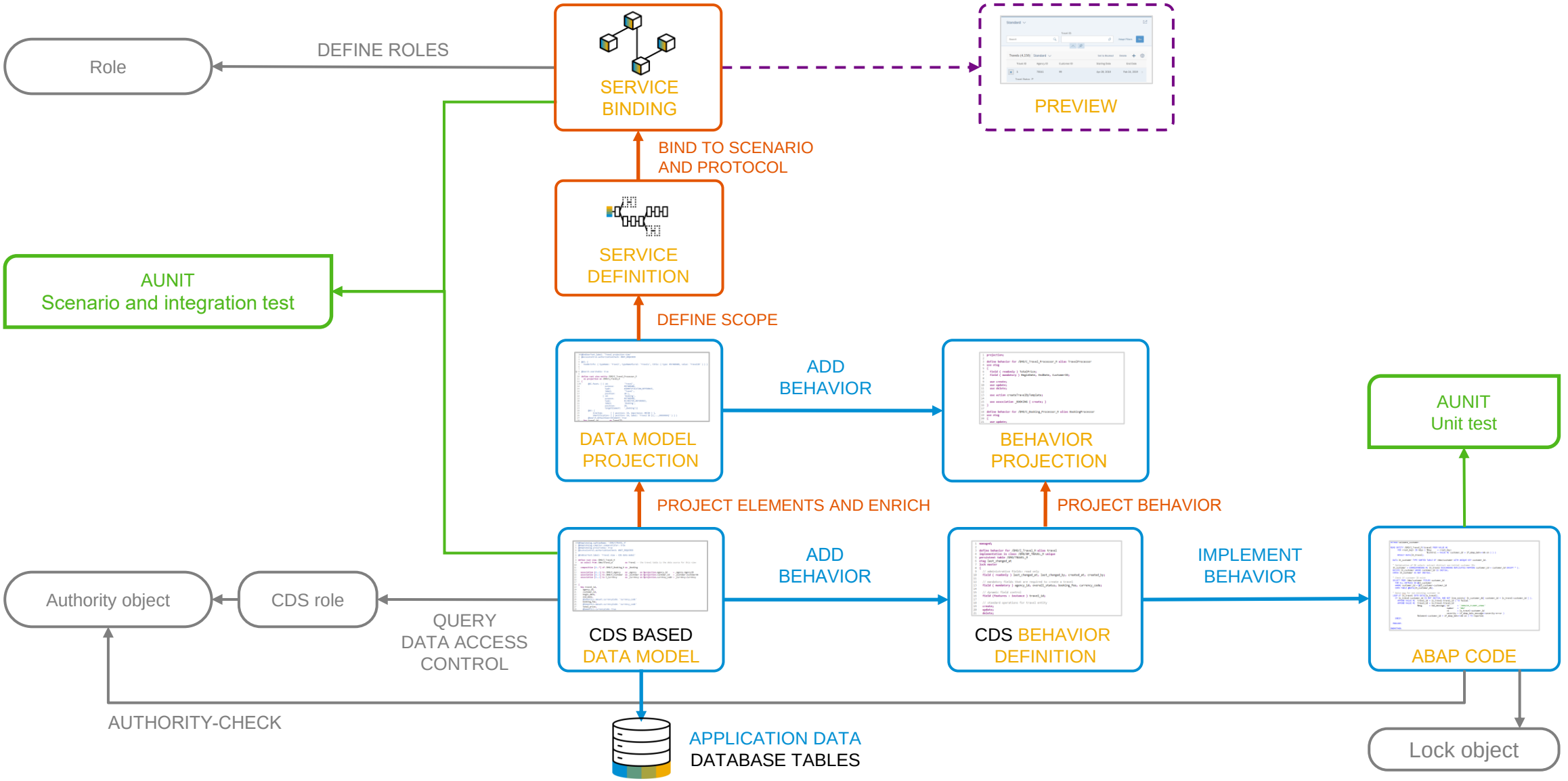
➔ Standard CRUD operations work out of the box

➔ Developers add BO-specific business logic

DEMO

GET THE IDEA

Typical development flow



WHAT'S NEW

What's new? – Latest[1] RAP enhancements (1)

SUPPORT FOR ADDITIONAL PROTOCOLS IN THE SERVICE BINDING EDITOR

SAP Information Access (InA) based services for exposing CDS analytical queries with live data connection

SQL-based services for accessing ABAP-managed data (CDS view entities) from external ODBC based clients

SUPPORT FOR MULTI-INLINE-EDITING ON AN SAP FIORI UI

Implementation of RAP BOs with singleton mechanism to support multi-inline-editing on a Fiori UI

UNMANAGED EARLY NUMBERING

To automatically draw keys during the `CREATE` operation

Available for unmanaged BOs with draft and managed BOs

ENHANCEMENTS FOR THE PROJECTION LAYER

Specification of provider contract in the data model projection to enable scenario-specific features

Adding actions and functions in a behavior definition projection

Suppression of fields in transactional handling

Flexible association modeling

[1] Delivered in Q3-Q4 2021 in the SAP BTP and SAP S/4HANA 2021

What's new? – Latest[1] RAP enhancements (2)

VARIOUS ENHANCEMENTS FOR DETERMINATIONS, ACTIONS, AND VALIDATIONS

Nested Determinations on modify
Flag always for determinations/validations on sub-entities
Always flag in determine actions
Mapping for action and function parameters
Additional implementation for draft actions
Save actions for managed BOs

ENHANCED FEATURE CONTROL

Global feature control
Field control for modify-enabled fields

ENHANCED AUTHORIZATION CONTROL

Global authorization control
Delegation of authorization checks
Authorization and feature control checks in unmanaged business objects

SPECIFICATION OF STRICT MODE IN BEHAVIOR DEFINITIONS

For the implementation lifecycle-stable and upgrade-safe RAP BOs

[1] Delivered in Q3-Q4 2021 in the SAP BTP and SAP S/4HANA 2021

What's new? – Latest[1] RAP enhancements (3)

UNIT AND CURRENCY TYPES AS BUILT-IN ENTITY SETS

Both built-in entity sets automatically added to service metadata sets of OData V2 services

Unit and currency values are then always displayed correctly and validated regarding decimals on UI

RAP REUSE DATA ELEMENTS

For the administrative fields

ENHANCEMENTS AROUND EML

Harmonization of `COMMIT WORK` and `COMMIT ENTITIES`

`COMMIT ENTITIES` in simulation mode

Using `AUTO FILL CID` option

Runtime changes for cleanup execution

IDEMPOTENCY FOR ODATA V4 SERVICES

Ensuring that synchronous messages are delivered exactly once

[1] Delivered in Q3-Q4 2021 in the SAP BTP and SAP S/4HANA 2021

What's new? – Latest[2] RAP enhancements (4)

ENHANCEMENTS FOR MANAGED IMPLEMENTATION TYPE

Late numbering for a late value assignment of the primary key fields

Use of client-independent tables as data sources for managed BOs with draft capabilities

ENHANCEMENTS FOR DETERMINATIONS, ACTIONS, AND VALIDATIONS

Instance factory actions to model actions that create new instances based on an input instance

Providing operations in reported response parameter; using derived type component `%op`

ENHANCEMENTS FOR FIORI UIS

Support for multiple message targets in OData V4 RAP services using the derived type component `%element` in the `reported` structure of RAP handler methods

Use of `UI.hidden` annotation with action input parameters fields to prevent the input fields from being displayed on the UI, while still available for a Web API client

FURTHER ENHANCEMENTS

ADT wizard for generating transactional RAP services based on single database tables

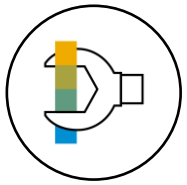
RAP services related errors are now displayed in the service binding and can also be retrieved by running ATC checks

Use of Read Access Logging (RAL) for RAP services to monitor database read accesses

[2] Delivered in Q4 2021 in the SAP BTP; not yet available in SAP S/4HANA

SUMMARY & OUTLOOK

Key takeaways



The **ABAP RESTful Application Programming Model (RAP)** helps to efficiently and rapidly build enterprise-grade services with intrinsic built-in cloud qualities. RAP best support **SAP HANA** and **SAP Fiori elements**.

RAP is **available** on **SAP BTP ABAP Environment** and **SAP S/4HANA**, cloud and on-premise editions**.**

RAP is available SAP S/4HANA on-premise starting with edition 1909.

The **feature scope** of RAP is enhanced on a **quarterly** basis in SAP BTP ABAP Environment and SAP S/4HANA Cloud, and on a **yearly** basis in SAP S/4HANA on-premise.

Check out the SAP Road Map Explorer for updates.

Guidance for RAP usage in SAP S/4HANA on-premise releases

SAP S/4HANA 1909

FIRST RAP DELIVERY WITH LIMITED FEATURE SET

managed scenario for greenfield development not supported,
no draft support for SAP Fiori development

GUIDANCE

Delivery with SAP S/4HANA 1909

✎ Use the *ABAP Programming Model for SAP Fiori*

SAP S/4HANA 2020

BASIC RAP DELIVERY

incl. **managed** scenario (*with major limitations regarding key layout and numbering*),
draft support for managed or unmanaged BOs to build transactional SAP Fiori apps,
OData V4 support for Web APIs and SAP Fiori UIs with FPS01

GUIDANCE

If limitations are crucial

✎ Use the *ABAP Programming Model for SAP Fiori*

If limitations are not critical

✎ Use RAP

SAP S/4HANA 2021

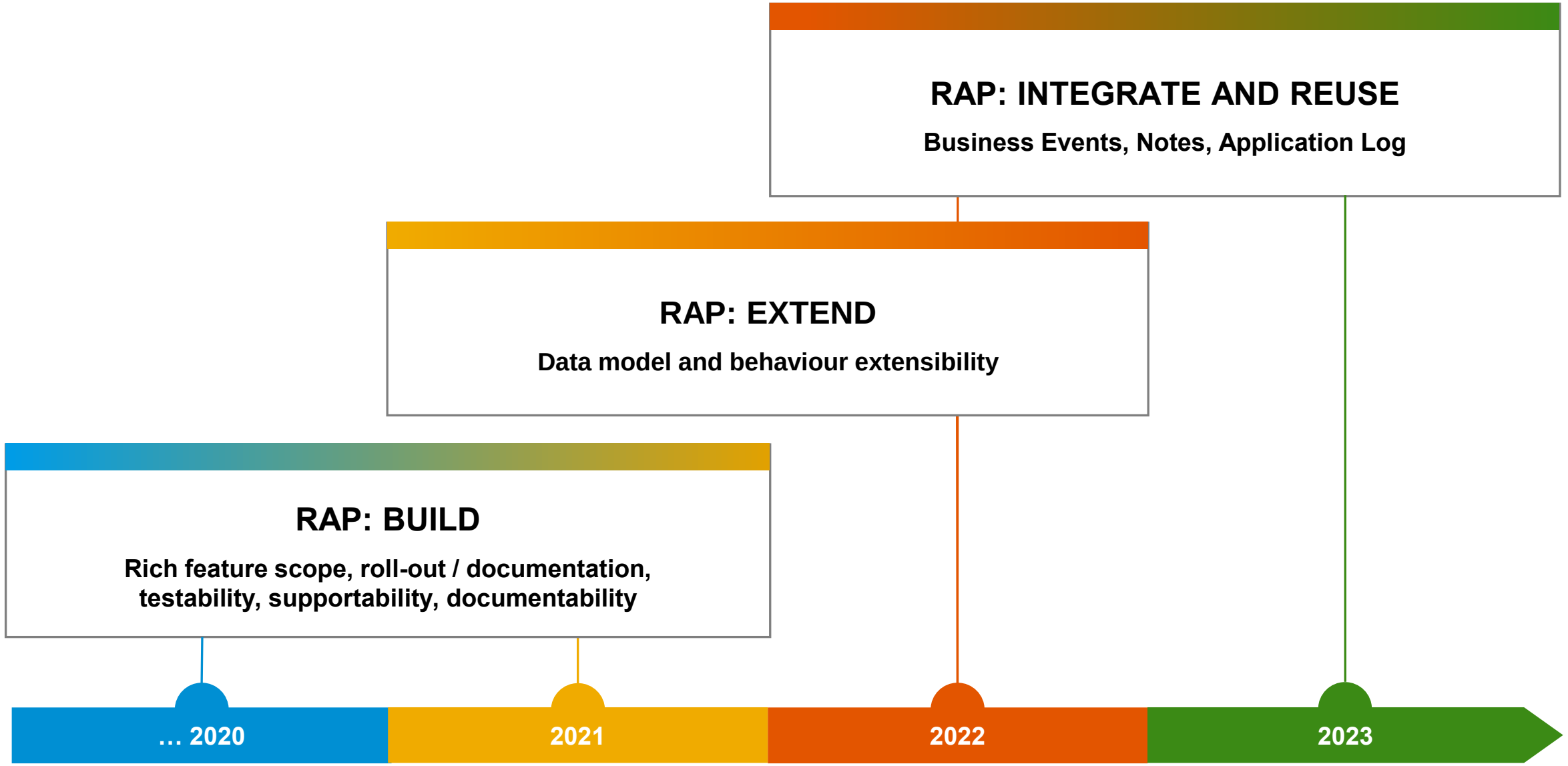
MASS ADOPTION READY RAP DELIVERY

managed scenario now supports nearly all key layouts,
late numbering for managed implementation type planned with **FPS01**,
Enhanced testability, documentability and supportability support

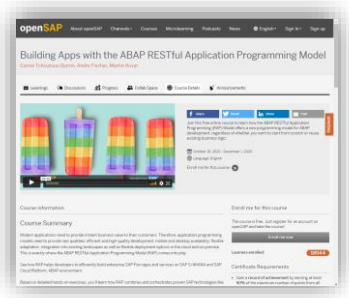
GUIDANCE

✎ Exclusively use RAP

Outlook – Next steps



FREE openSAP COURSE



Building Apps with the ABAP RESTful Application Programming Model (RAP)

Self-paced mode

Week 1: Introduction
Week 2: Developing a Read-Only List Report App
Week 3: Enabling the Transactional Behavior of an App
Week 4: Dealing with Existing Code
Week 5: Service Consumption and Web APIs
Week 6: Final Exam

ENROLL NOW!
https://open.sap.com/courses/cp13

More information

Further information

Getting started with the ABAP RESTful Application Programming Model (RAP)

Building Apps with the ABAP RESTful Application Programing Model (*openSAP course*)

Modernization with RAP

RAP – What's New on SAP BTP ABAP Environment

RAP – What's New on SAP S/4HANA

Public SAP Web sites

ABAP Development community: www.sap.com/community/topic/abap.html

SAP BTP ABAP Environment community: https://community.sap.com/topics/cloud-platform-abap-environment

ABAP Testing and Analysis Community: https://community.sap.com/topics/abap-testing-analysis

SAP products: www.sap.com/products

SAP training and certification opportunities

www.sap.com/education

Thank you.

Contact information:

Carine Tchoutouo Djomo
Product Manager for ABAP Platform
carine.tchoutouo.djomo@sap.com

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

Follow us

www.sap.com/contactsap

THE BEST RUN SAP