

ODATA SERVICE DEVELOPMENT OPTIONS

**Be prepared for the
ABAP RESTful Programming Model**

Andre Fischer
July, 2019



Agenda



ODATA V2 DEVELOPMENT OPTIONS



ABAP RESTFUL PROGRAMMING MODEL



ODATA V4



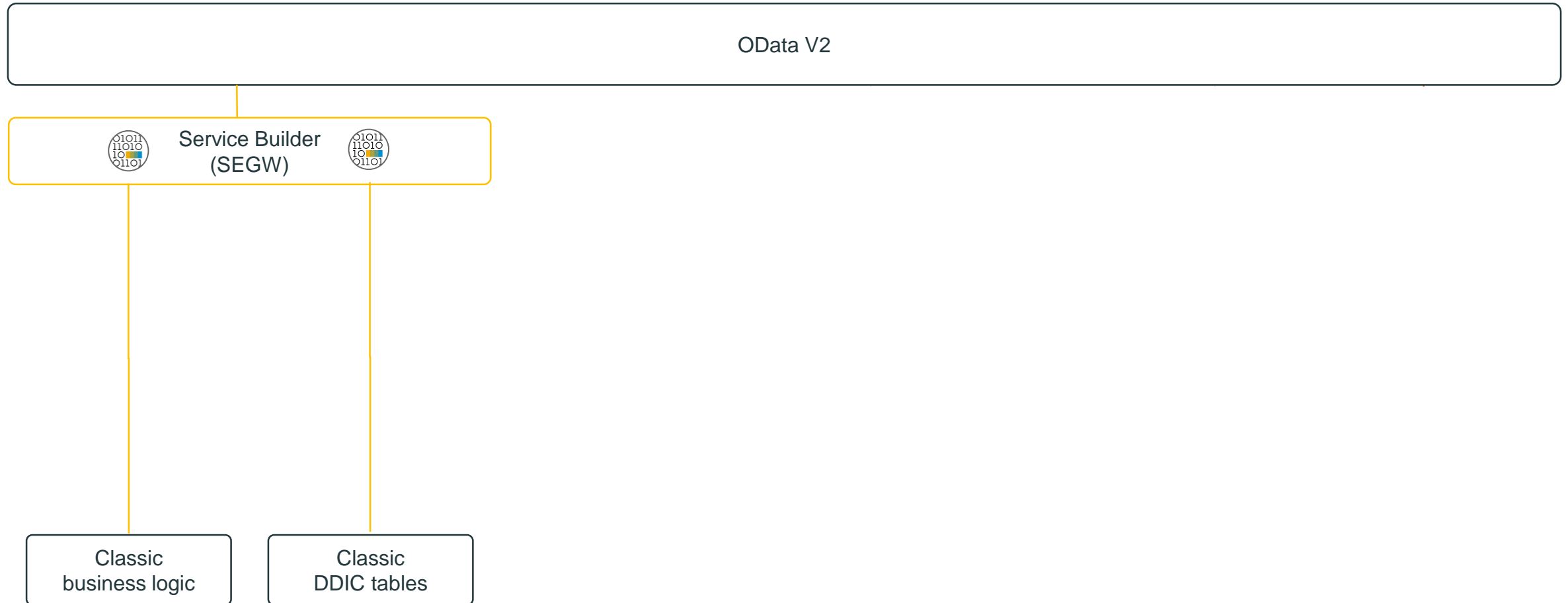
WHERE IT IS SAFE TO INVEST NOW?



OData
development
options

OData V2 service development options

Code based
implementation



SAP Gateway

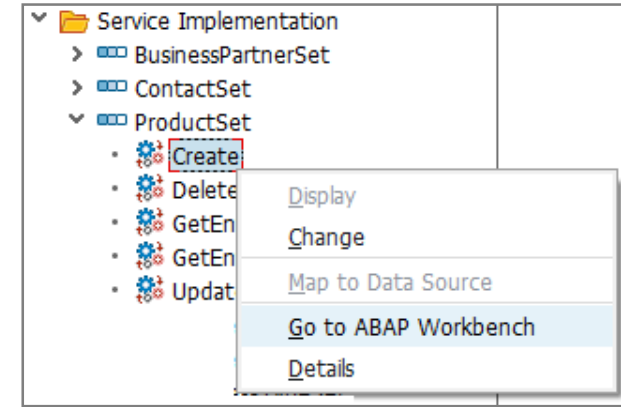
Code-based service development

CODE-BASED IMPLEMENTATION OF CRUD METHODS

- ▶ Create → <Entity_Set>_CREATE_ENTITY
- ▶ Read → <Entity_Set>_GET_ENTITY
- ▶ Query → <Entity_Set>_GET_ENTITYSET
- ▶ Update → <Entity_Set>_UPDATE_ENTITY
- ▶ Delete → <Entity_Set>_DELETE_ENTITY

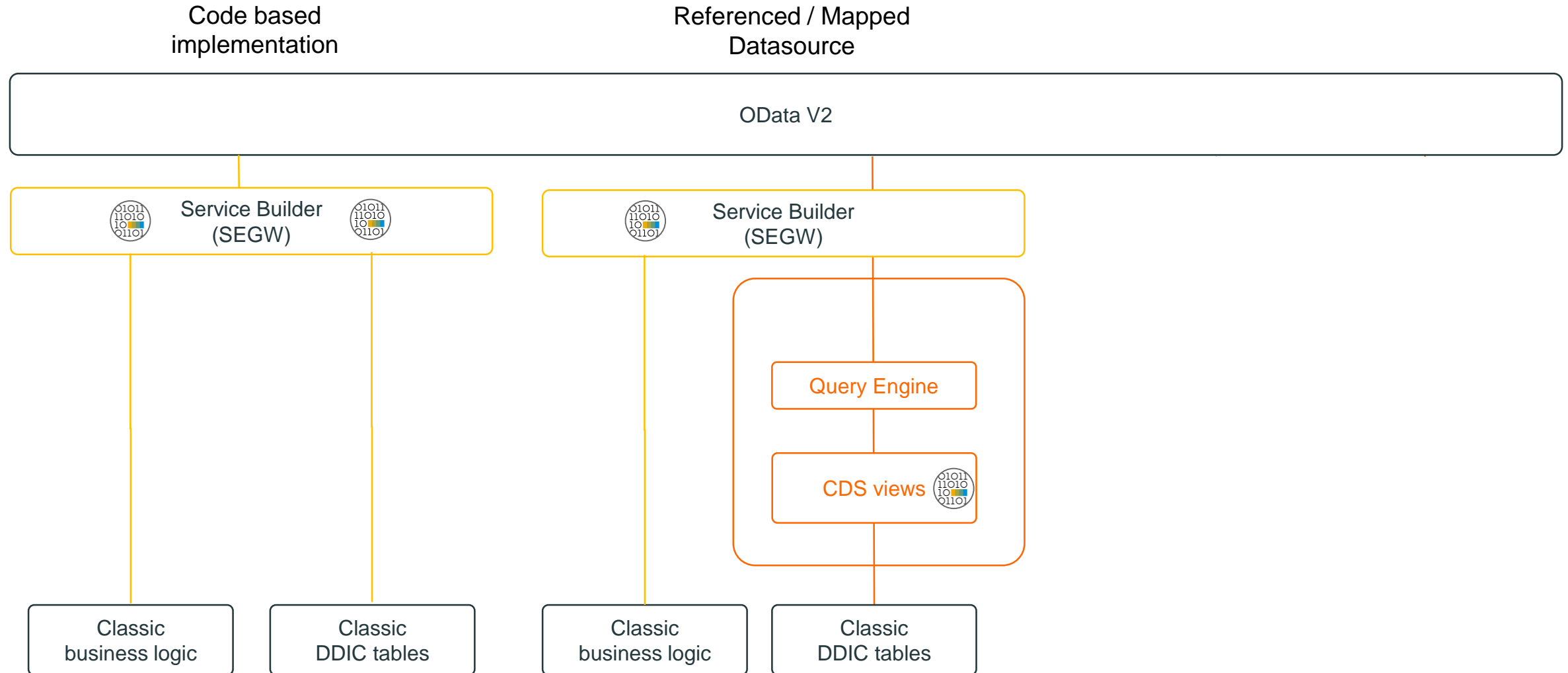
CODE BASED IMPLEMENTATION OF ADVANCED ODATA FEATURES

- ▶ Offline scenarios (\$deltatoken, \$skiptoken)
- ▶ Complex transactional behavior
(SoftState, DeepInsert, ChangeSets in \$batch)



```
METHOD productset_get_entityset.  
  
DATA: lv_source_entity_set_name TYPE /iwbp/mgw_tech_name.  
  
lv_source_entity_set_name = io_tech_request_context->get_source_entity_set_name( ).  
  
IF lv_source_entity_set_name IS INITIAL.  
  prod_get_entityset(  
    EXPORTING  
      io_tech_request_context = io_tech_request_context  
    IMPORTING  
      et_entityset           = et_entityset  
      es_response_context   = es_response_context ).  
ENDIF.
```

OData V2 service development options



Mapped Data Source

The screenshot displays the SAP Gateway Service Builder interface. On the left, a tree view shows the project structure for 'ZCDS_MDS', including 'Data Model', 'Service Implementation', and 'Runtime Artifacts'. The 'Service Implementation' folder is expanded to show 'BusinessPartnerSet' with a 'Mapping' sub-item. The main workspace is titled 'Map Modeled Data Source Elements to Properties' and contains a table with the following data:

Property	ABAP Field Name	Key	Element
Businesspar...	BUSINESSPARTNERUUID	<input checked="" type="checkbox"/>	BUSINESSPARTNERUUID
Businesspar...	BUSINESSPARTNERID	<input type="checkbox"/>	BUSINESSPARTNERID
Companyname...	COMPANYNAME	<input type="checkbox"/>	COMPANYNAME

On the right side, a 'Modeled Data Source Name' tree shows 'ZCDS_BP_EXAMPLE' with a 'Primary Key' property. The interface also includes various toolbars and navigation options.

- OData Model has to be modelled in SEGW
- Manual mapping of READ and QUERY methods
- Available as of 7.40
- New fields are not automatically included
- Code based implementation of CREATE, UPDATE, and DELETE methods

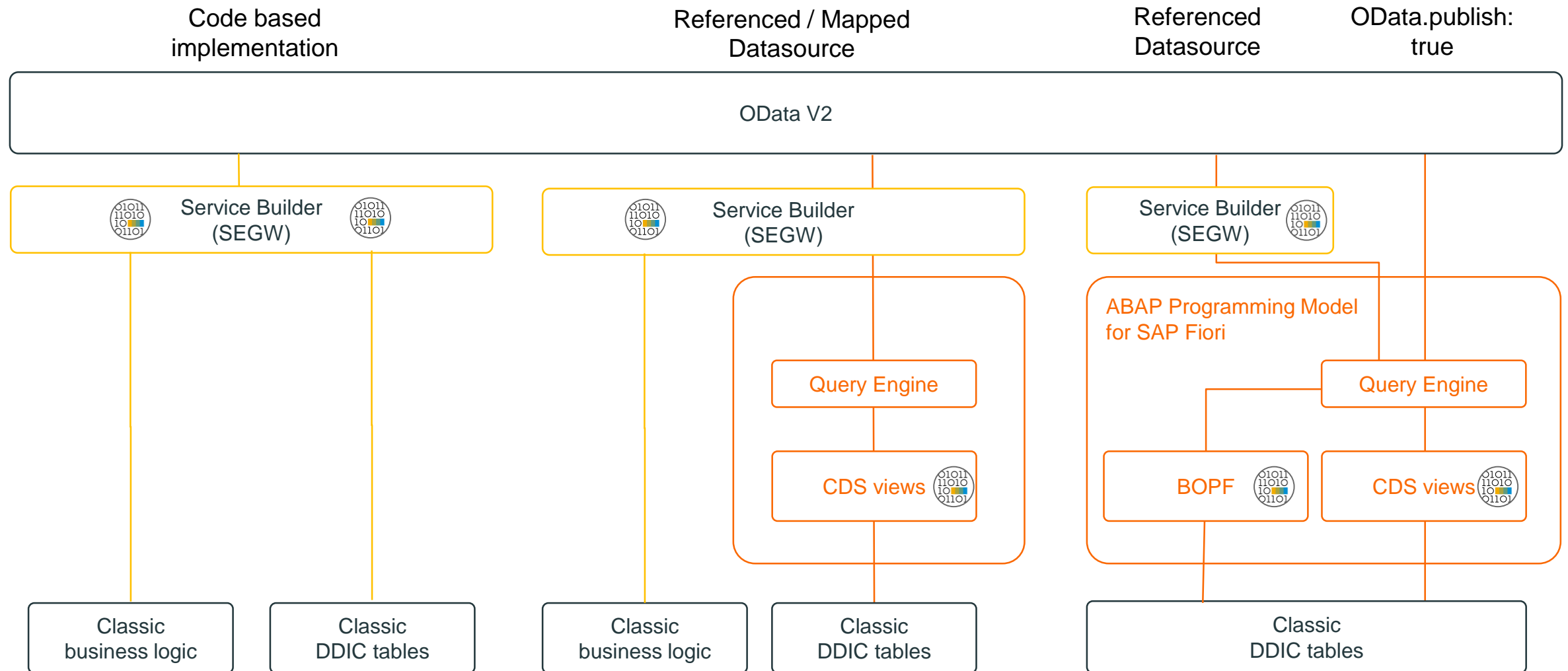
Referenced Data Source

The screenshot displays the SAP Gateway Service Builder interface. The left pane shows a tree view of the data model for 'Z_BE2UI_SOL'. The right pane shows a list of CDS-Entity Exposures with checkboxes for selection and a 'Value Help' column.

CDS-Entity Exposures	Selected	Value Help
SEPMRA_I_ProductMainCategory	<input checked="" type="checkbox"/>	✓
_Category	<input type="checkbox"/>	
SEPMRA_I_ProductCategory		
ZC_Be2Ui_Product_Ex4	<input checked="" type="checkbox"/>	
_Currency	<input checked="" type="checkbox"/>	
I_Currency	<input checked="" type="checkbox"/>	
_ProductCategory	<input checked="" type="checkbox"/>	
SEPMRA_I_ProductCategory	<input checked="" type="checkbox"/>	
_ProductReview	<input checked="" type="checkbox"/>	
ZC_Be2Ui_ProductReview_Ex4	<input checked="" type="checkbox"/>	
_Supplier	<input checked="" type="checkbox"/>	
SEPMRA_I_Supplier	<input checked="" type="checkbox"/>	

- OData Model references selected CDS views and associations
- Available as of 7.50
- New fields are automatically included
- Code based implementation of CREATE, UPDATE, and DELETE methods

OData V2 service development options



ABAP Programming Model for SAP Fiori – CDS based BOPF Objects

The screenshot shows the 'Determination Overview' for the node 'CREATE_SO_ID'. The 'General Information' section includes the name 'CREATE_SO_ID', description 'Create Sales Order ID', and implementation class 'ZCL_S4H_D_CREATE_SO_ID_999'. The 'Triggers' section shows a table with columns for Node, Association, Create, Update, Delete, Load, and Determine. The 'Create' checkbox for the node 'ZS4H_I_SLSORDER_TP_999' is checked and highlighted with a red box.

Node	Association	Create	Update	Delete	Load	Determine
ZS4H_I_SLSORDER_TP_999		<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>



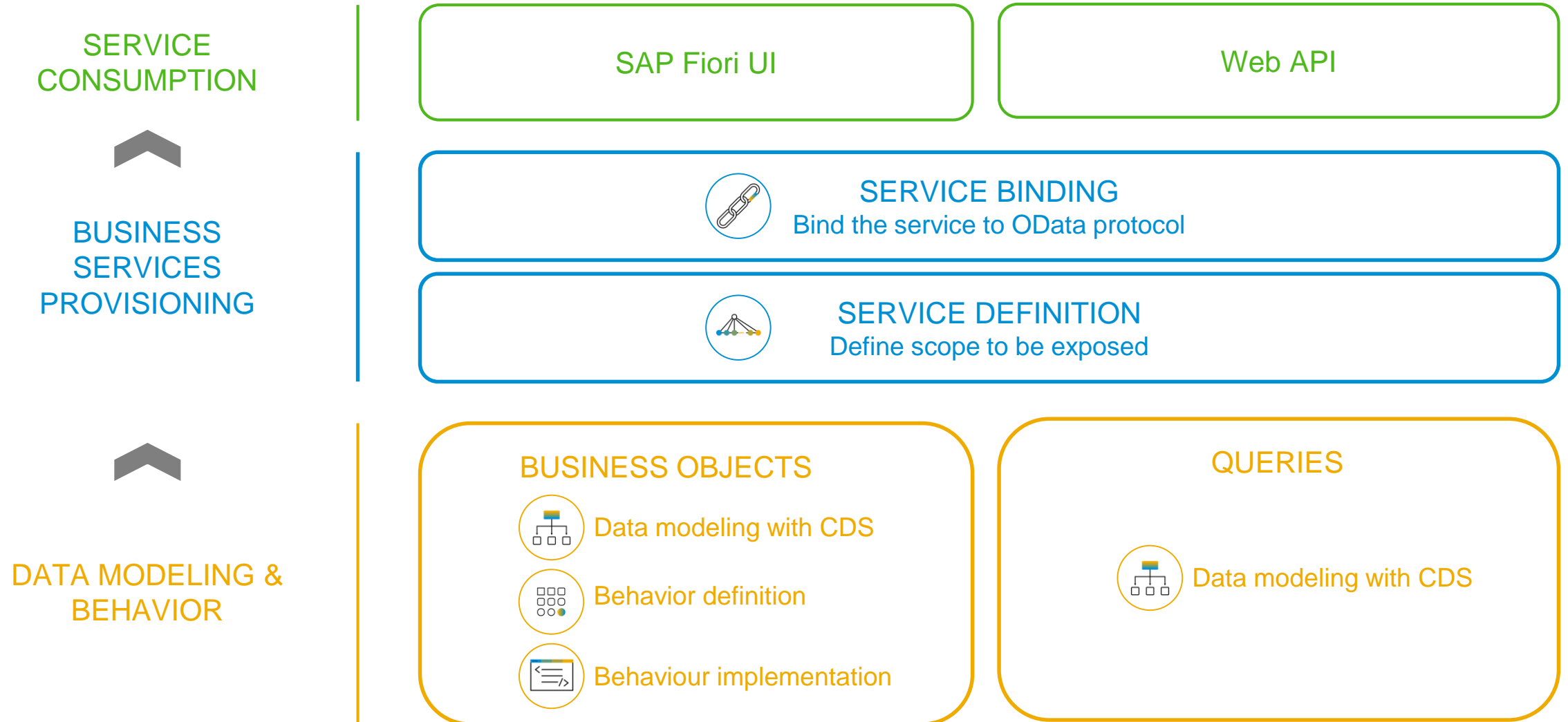
```
[A4H] ZS4H_I_SLSORDER_TP_999 [A4H] ZCL_S4H_D_CREATE_SO_ID_999
ZCL_S4H_D_CREATE_SO_ID_999 /BOBF/IF_FRW_DETERMINATION~EXECUTE
1 CLASS zcl_s4h_d_create_so_id_999 DEFINITION
2 PUBLIC
3 INHERITING FROM /bobf/cl_lib_d_supercl_simple
4 FINAL
5 CREATE PUBLIC .
6
7 PUBLIC SECTION.
8
9     METHODS /bobf/if_frw_determination~execute
10    REDEFINITION .
11 PROTECTED SECTION.
12 PRIVATE SECTION.
13 ENDClass.
14
15
16
17 CLASS zcl_s4h_d_create_so_id_999 IMPLEMENTATION.
18
19
20 METHOD /bobf/if_frw_determination~execute.
21 ENDMETHOD.
22 ENDClass.
```

- OData Model definition via CDS views
- Available as of 7.50 (Draft as of 7.51)
- Code based implementation of Determinations, Validations and Actions
- Managed scenario: Query engine (SADL) orchestrates all CRUD-Q calls to the SAP Gateway (OData) framework



ABAP
RESTful
Programming
Model

ABAP RESTful Programming Model - big picture



CDS views are used to define business objects

```
define root view /DMO/I_Travel_U
  as select from /dmo/travel as Travel
  -- the travel table is the data source for this view

composition [0..*] of /DMO/I_Booking_U as _Booking

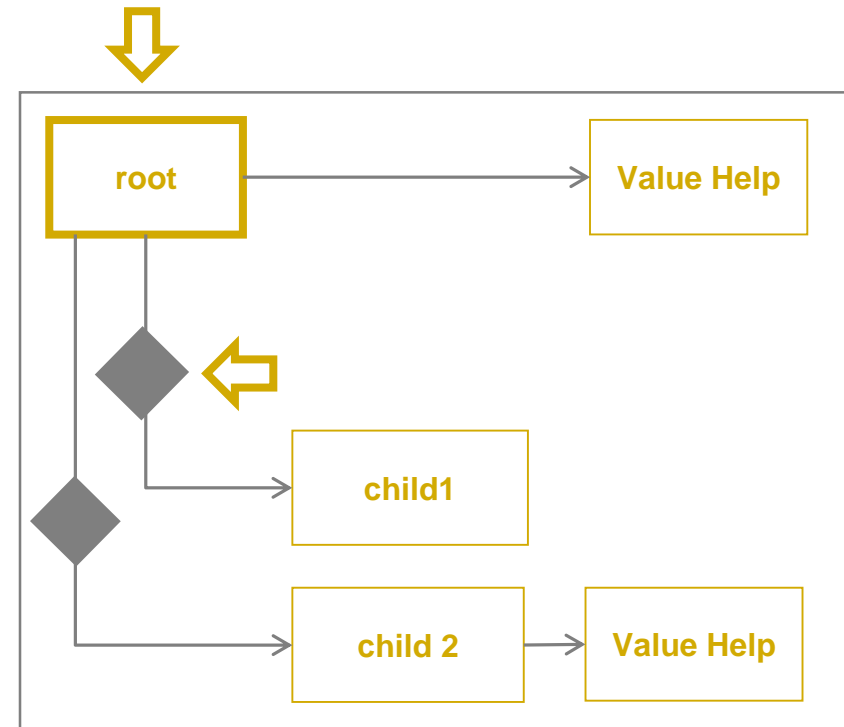
  association [0..1] to /DMO/I_Agency as _Agency
    on $projection.AgencyID = _Agency.AgencyID

  association [0..1] to /DMO/I_Customer as _Customer
    on $projection.CustomerID = _Customer.CustomerID

  association [0..1] to I_Currency as _Currency
    on $projection.CurrencyCode = _Currency.Currency
```

```
define view /DMO/I_Booking_U
  as select from /dmo/booking as Booking

  association to parent /DMO/I_Travel_U as _Travel
    on $projection.TravelID = _Travel.TravelID
```



Behaviour definition language

```
implementation unmanaged;
```

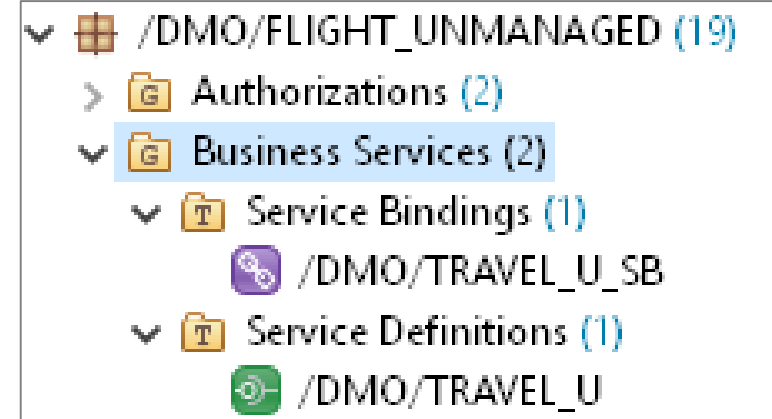
```
// behavior definition for the TRAVEL root entity  
define behavior for /DMO/I_Travel_U alias travel  
implementation in class /DMO/BP_TRAVEL_U unique  
etag LastChangedAt
```

```
{  
  field (read only) TravelID;  
  field (mandatory) AgencyID, CustomerID,  
  BeginDate, EndDate;  
  
  create;  
  update;  
  delete;  
  
  action set_status_booked result [1] $self;  
  
  association _Booking { create; }  
}
```

The screenshot displays a SAP Fiori application interface. At the top, there is a search bar and filter fields for 'Travel ID', 'Agency ID', and 'Customer ID'. Below this is a table titled 'Travels (4,136)' with columns 'Travel ID' and 'Agency ID'. The table contains two rows: one with '1' and 'Maxitrip (7004)' and another with '2' and 'Hot Socks Tra'. To the right of the table are buttons for 'Set to Booked', 'Delete', and a plus sign. A modal window is open over the table, showing a form for a 'Travel' object. The form has a title '<Unnamed Object>' and a 'Travel' section. The 'Travel ID' field is highlighted with a yellow box. Other fields include '*Agency ID', '*Customer ID', '*Starting Date', '*End Date', 'Booking Fee', 'Total Price', and 'Comment'.

Service Definition

```
@EndUserText.label: 'Service Definition for Managing Travels'  
define service /DMO/TRAVEL_U {  
  expose /DMO/I_Travel_U as Travel;  
  expose /DMO/I_Booking_U as Booking;  
  expose /DMO/I_BookingSupplement_U as BookingSupplement;  
  expose /DMO/I_Supplement as Supplement;  
  expose /DMO/I_SupplementText as SupplementText;  
  expose /DMO/I_Customer as Passenger;  
  expose /DMO/I_Agency as TravelAgency;  
  expose I_Currency as Currency;  
  expose I_Country as Country;  
  expose /DMO/I_Carrier as Airline;  
  expose /DMO/I_Connection as FlightConnection;  
  expose /DMO/I_Flight as Flight;  
}
```



Service Binding

Service Binding: /DMO/TRAVEL_U_SB

General Information
This section describes general information about this service binding

Binding Type: ODATA V2 (UI - User Interface: Consumed in SAPUI5 Apps)

Service Versions
View information on the service versions associated with the Service Binding

type filter text

Version	Service Definition
0001	/DMO/TRAVEL_U

Add...
Remove

Service Version Details
View information on selected service version

Local service endpoint: Active Deactivate

Local Service Endpoint Information

Service URL: /sap/opu/odata/DMO/TRAVEL_U_SB

Preview

Type filter text

Entity Set and Association

- TravelAgency
- BookingSupplement
- Booking
- Airline
- FlightConnection
- Passenger
- Flight

Travels (4,136)

Travel ID	Agency ID	Customer ID	Starting Date	End Date	Travel Status
1	Maxitrip (70041)	Ryan (594)	Jan 26, 2019	Nov 24, 2019	P
2	Hot Socks Travel (70007)	Prinz (608)	Jan 26, 2019	Jan 26, 2019	N



MANAGED **VERSUS** UN-MANAGED SCENARIOS

Managed vs. Un-Managed Scenarios

SCENARIO	USE-CASE	ABAP PROGRAMMING MODEL FOR SAP FIORI	ABAP RESTFUL PROGRAMMING MODEL
un-managed	Brownfield implementation	— *	✓
managed	Greenfield implementation	✓	(planned)
managed with self-implemented save	Brownfield implementation	—	(planned)

MANAGED SCENARIO

Query engine (SADL) orchestrates all CRUD-Q calls to the SAP Gateway (OData) framework

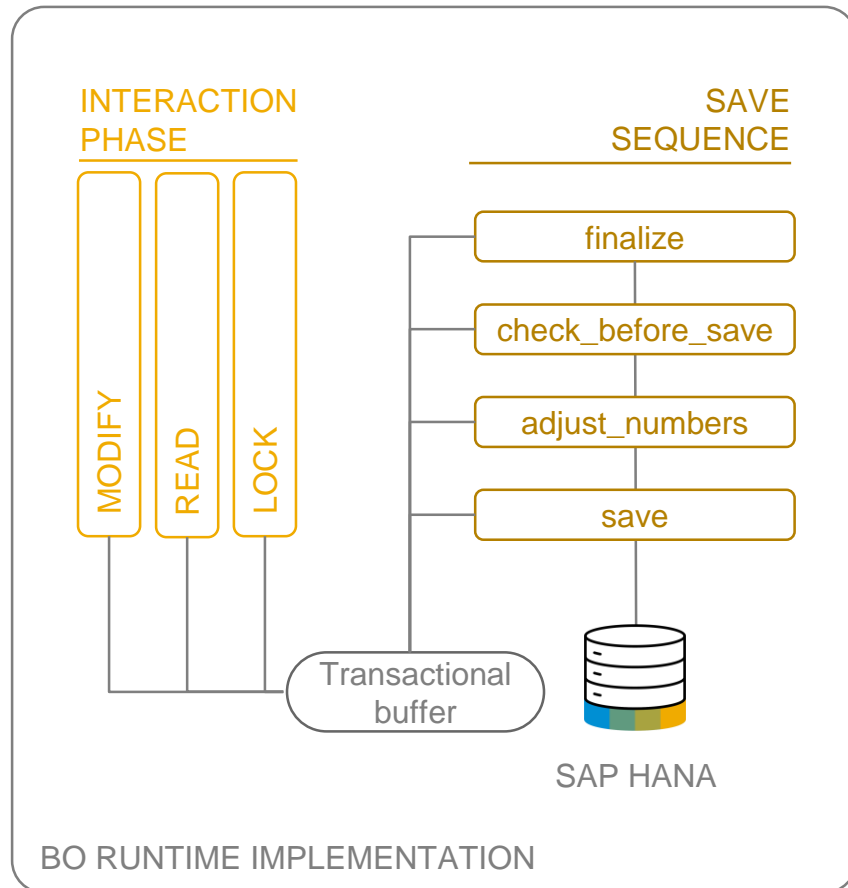
— = not supported

✓ = supported

—* = not supported via BOPF but via SEGW (RDS + DPC_EXT)

The ABAP Programming Model for SAP Fiori

ABAP RESTful Programming Model – Implementation types



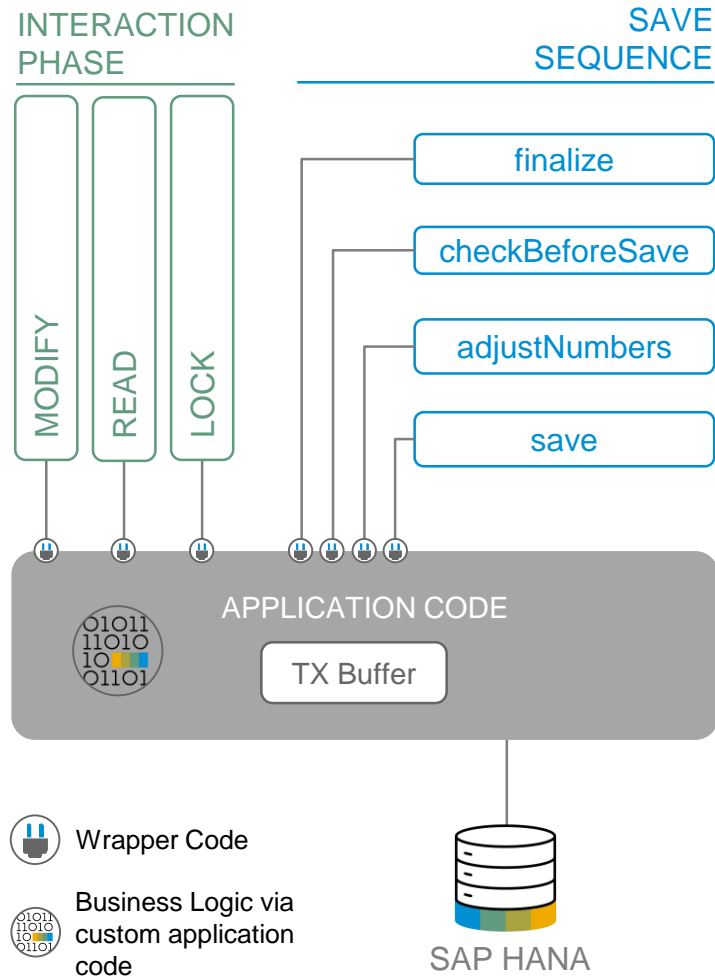
UNMANAGED

Brownfield development with application coding fully available:
Interaction phase + Transactional buffer + Save sequence

MANAGED

Greenfield development with standard implementation
(opt. appl. components: DB tables, lock objects, PFCG objects, update task FM)

Business objects - unmanaged



Application coding

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

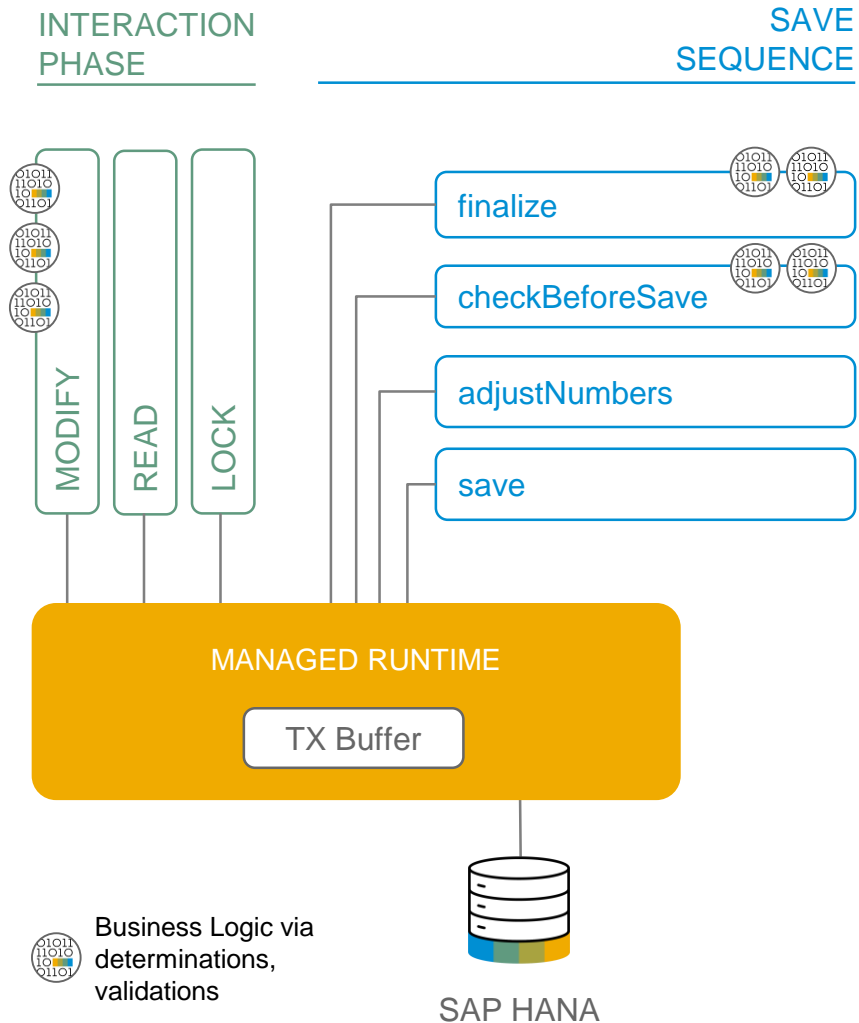
Examples

- ▶ Sales Order, Purchase Order

Business objects - managed



PLANNED



Application coding

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

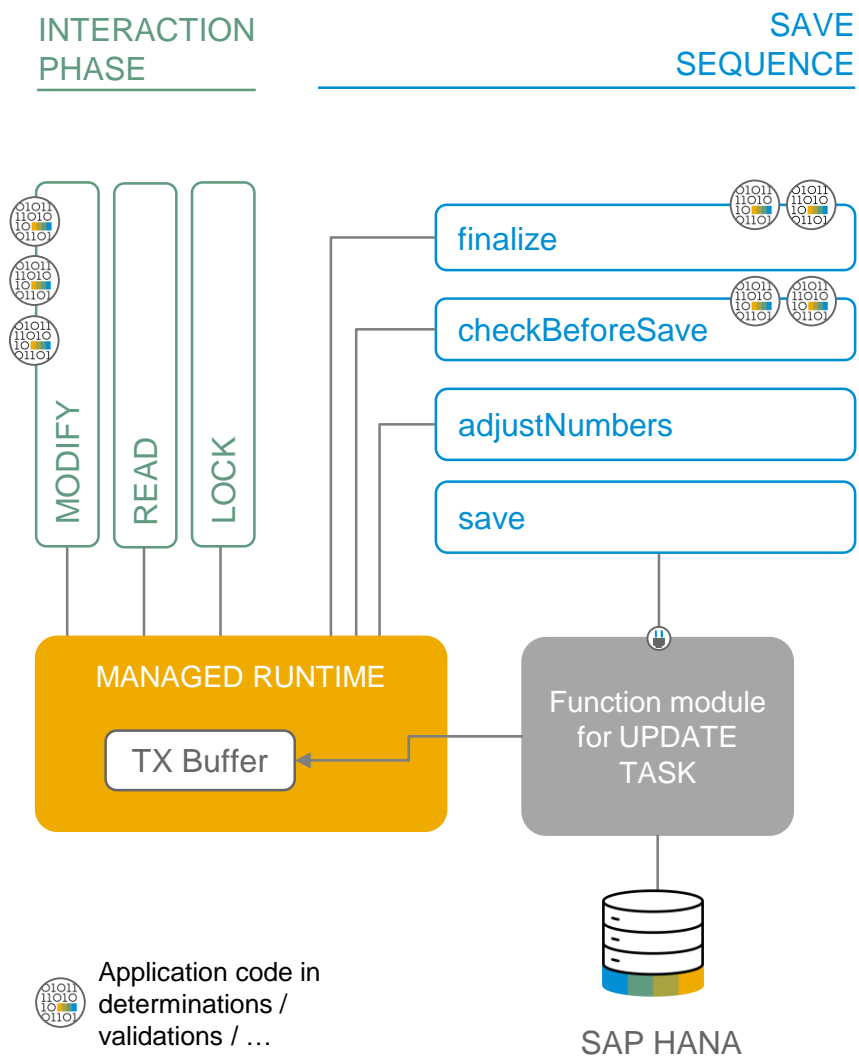
Examples

- ▶ New applications in SAP Cloud Platform ABAP Environment

Business objects – managed (with self-implemented save)



PLANNED



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



OData V4

OData V4 service development options

V4-API Code based implementation

OData V4

ADT Tools

Available as of 7.50, but not recommended

Classic
business logic

Classic
DDIC tables



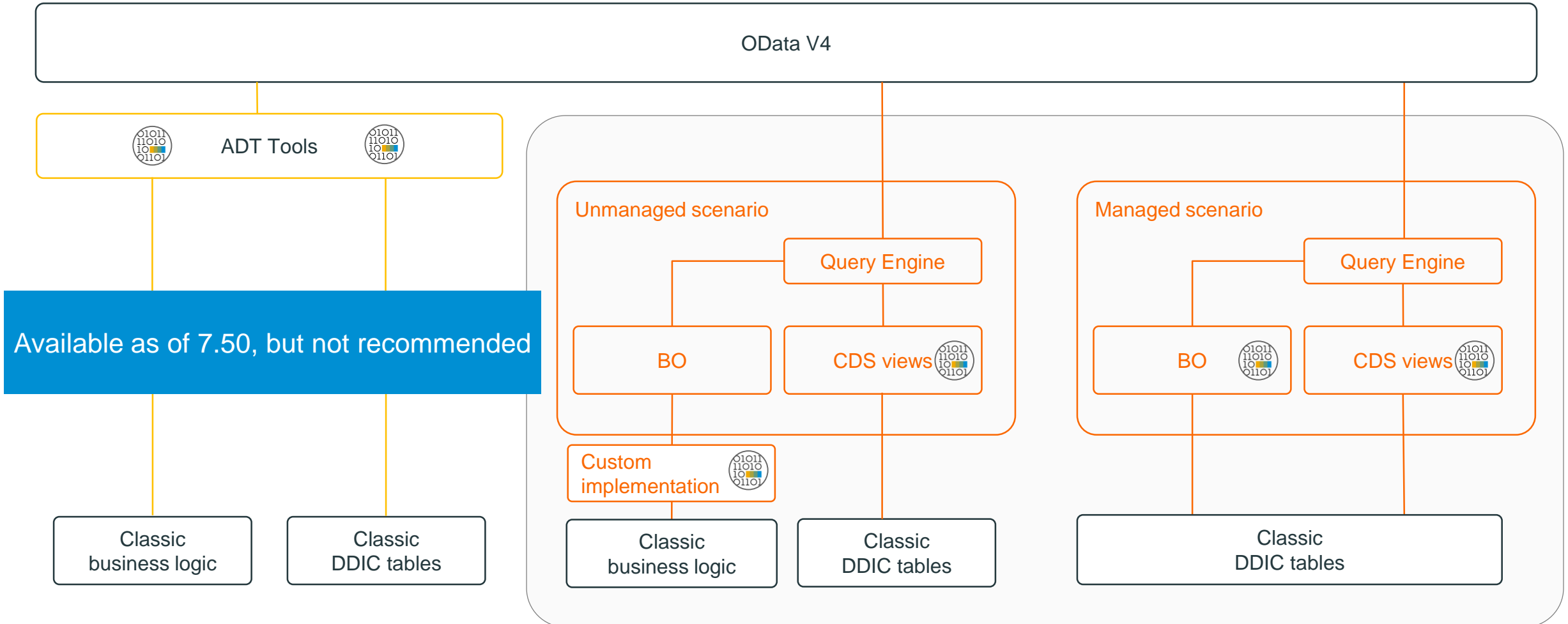
OData V4 service development options



PLANNED

V4-API Code based implementation

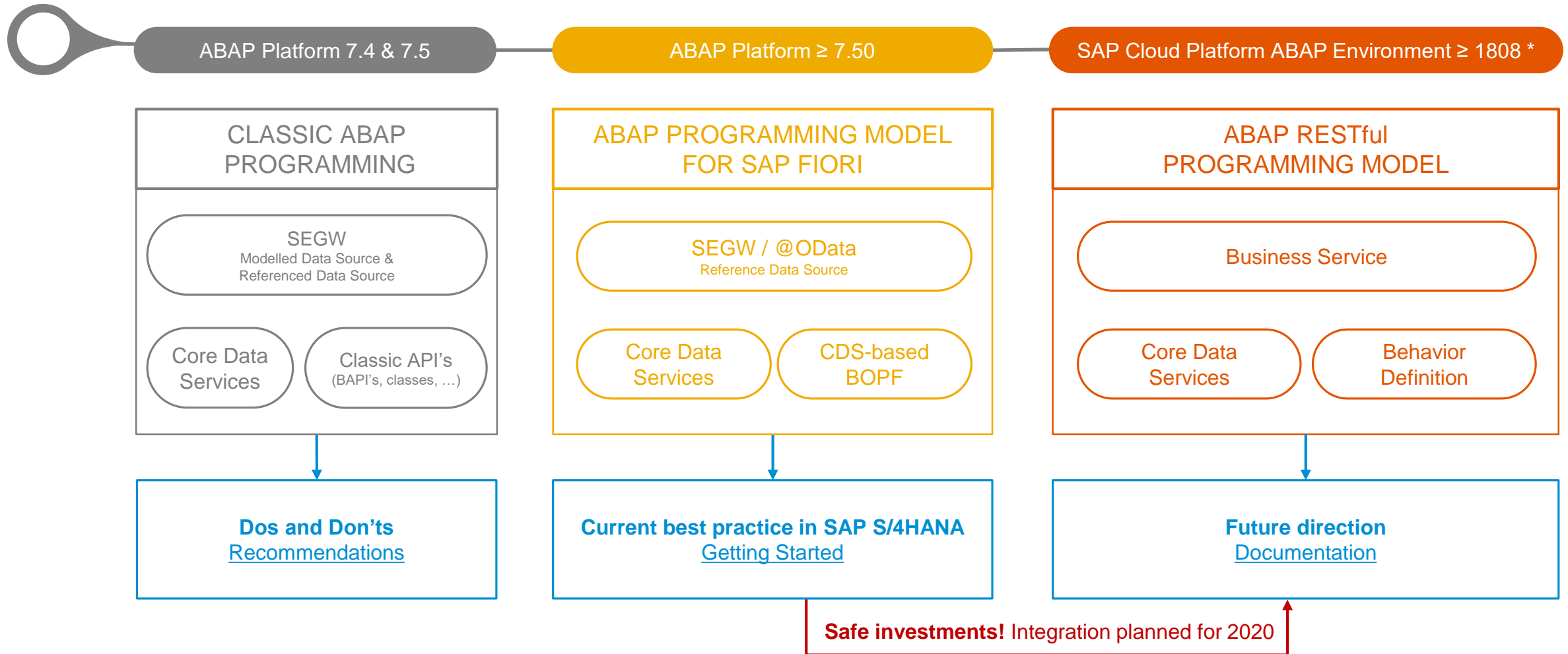
ABAP RESTful Programming Model





Where it is safe to
invest **now** ?

Evolution of OData Service Development



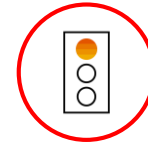
Where to invest now to be prepared for the ABAP RESTful Programming Model?



YOU SHOULD

Follow the programming model and [best practices](#) and use...

- ▶ Core Data Services (CDS) for database artefacts *
(≥ 7.40, [Documentation](#))
- ▶ CDS Metadata Extensions for UI Annotations *
(≥ 7.51 SP2, [Documentation](#))
- ▶ DCL for read/query instance-based authority checks *
(≥ 7.50, [Documentation](#))
- ▶ BOPF stand-alone
(≤ 7.40, [Documentation](#), newer releases: CDS/BOPF integration)
- ▶ BOPF and CDS integration including draft *
(≥ 7.51 SP2, [Documentation](#))
- ▶ Gateway integration of CDS or BOPF
(= 7.40, [Documentation](#), newer releases: OData Exposure)
- ▶ **OData Exposure of CDS / BOPF for SAP Fiori and future development ***
(≥ 7.50 SP5, [Documentation](#))
- ▶ Floorplan-Manager integration of CDS and BOPF
(≥ 7.40, [Documentation](#))



DO NOT

Implement things already solved

- ▶ Manual implementation of **read-only calls** to DB
- ▶ Business logic mixed with **technical aspects**
(e.g. locks, authority-check, LUW handling, persistency)
- ▶ Business logic mixed with **protocol specific APIs**
(e.g. PBO/PAI, Gateway classes: DPC_EXT)



BENEFITS

Reuse / prepare your skillset and coding for the future

- ▶ Reuse CDS and DCL in SAP S/4HANA
- ▶ Lower TCD for the future: Minimal investment on technical protocol level

Key Take Aways



Where to safely invest now? ...

USE CDS AS THE ONE AND ONLY DATA MODELLING LANGUAGE

Use CDS / BOPF integration and get familiar with BOPF concepts like determinations, validations and actions

Use OData exposure: @OData.publish:true or SEGW / Referenced Data Source for OData V2

Avoid DPC / MPC specific coding

MAKE USE OF THE ABAP PROGRAMMING MODEL FOR SAP FIORI

The ABAP Programming Model for SAP Fiori is available as of AS ABAP 7.5 and is the current best practice in SAP S/4HANA

BE PREPARED FOR THE ABAP RESTFUL PROGRAMMING MODEL

The ABAP RESTful Programming Model is already available in the SAP Cloud Platform ABAP Environment and is planned to be released on premise with upcoming SAP S/4HANA releases

AVOID ODATA V4 IF NOT MANDATORY FOR YOUR BUSINESS SCENARIO

Odata V4 is planned to be supported with the ABAP RESTful programming model soon

Code based implementation of Odata V4 API's cannot easily be reused.



Further information

SAP DEVELOPER NETWORK - COMMUNITIES

<https://www.sap.com/community/topic/gateway.html>

SAP DEVELOPER NETWORK – BLOG POSTS

<https://blogs.sap.com/2017/12/12/odata-service-development-options/>

<https://blogs.sap.com/2019/02/08/evolution-of-the-abap-programming-model/>

SAP SOURCES

<https://cloudplatform.sap.com/business-platform/abap.html>

BOOKS AND COURSES

SAP Press Book - SAP Gateway and OData (English)

https://www.sap-press.com/sap-gateway-and-odata_4724/

GW100 - SAP Gateway - Building OData Services

<https://training.sap.com/course/gw100-sap-gateway---building-odata-services-classroom-016-us-en/>

Thank you.

Contact information:

Andre Fischer

andre.fischer@sap.com

Engage the community.sap.com

You are invited to network, collaborate and learn while enjoying these community offerings:



Questions and answers



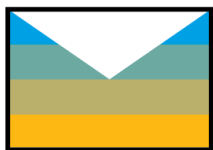
Blogging



Recognition Program



Personalization



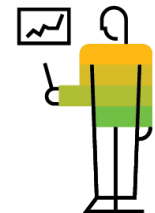
Direct Messaging



Coffee Corner



Events



Expert Pages