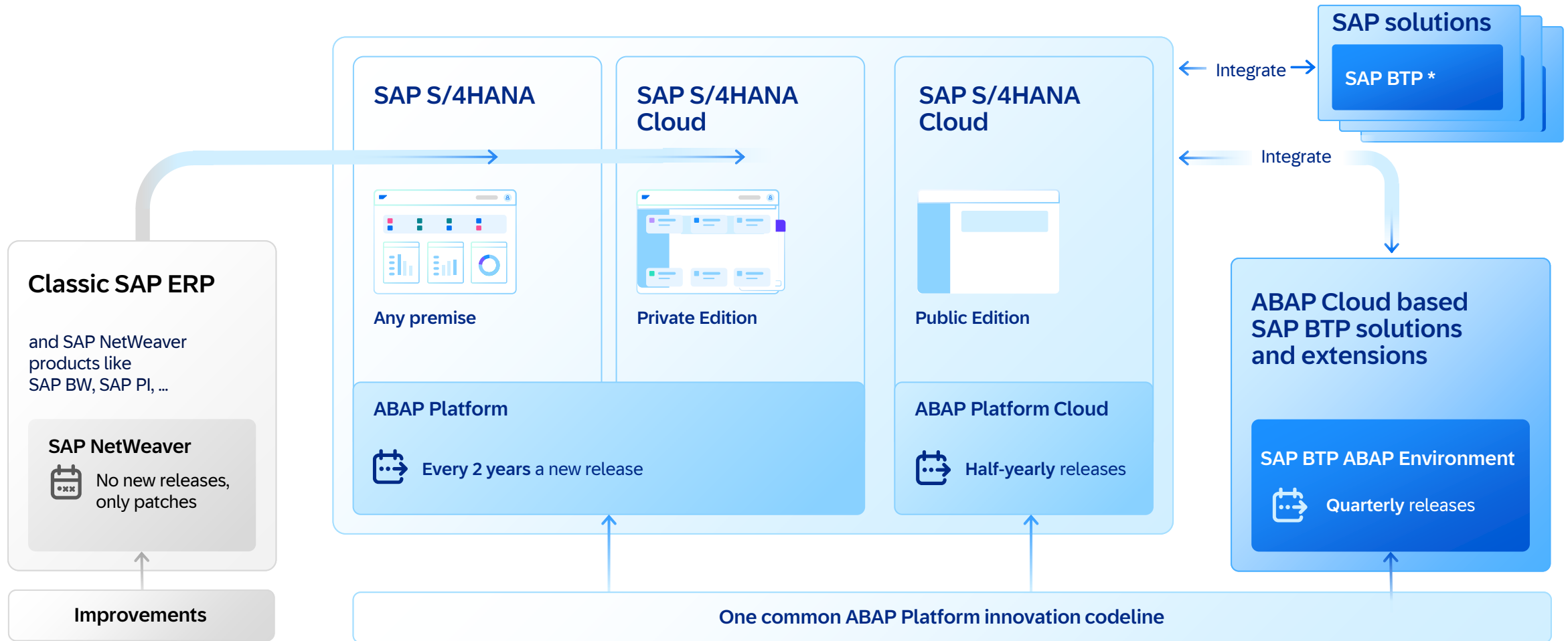




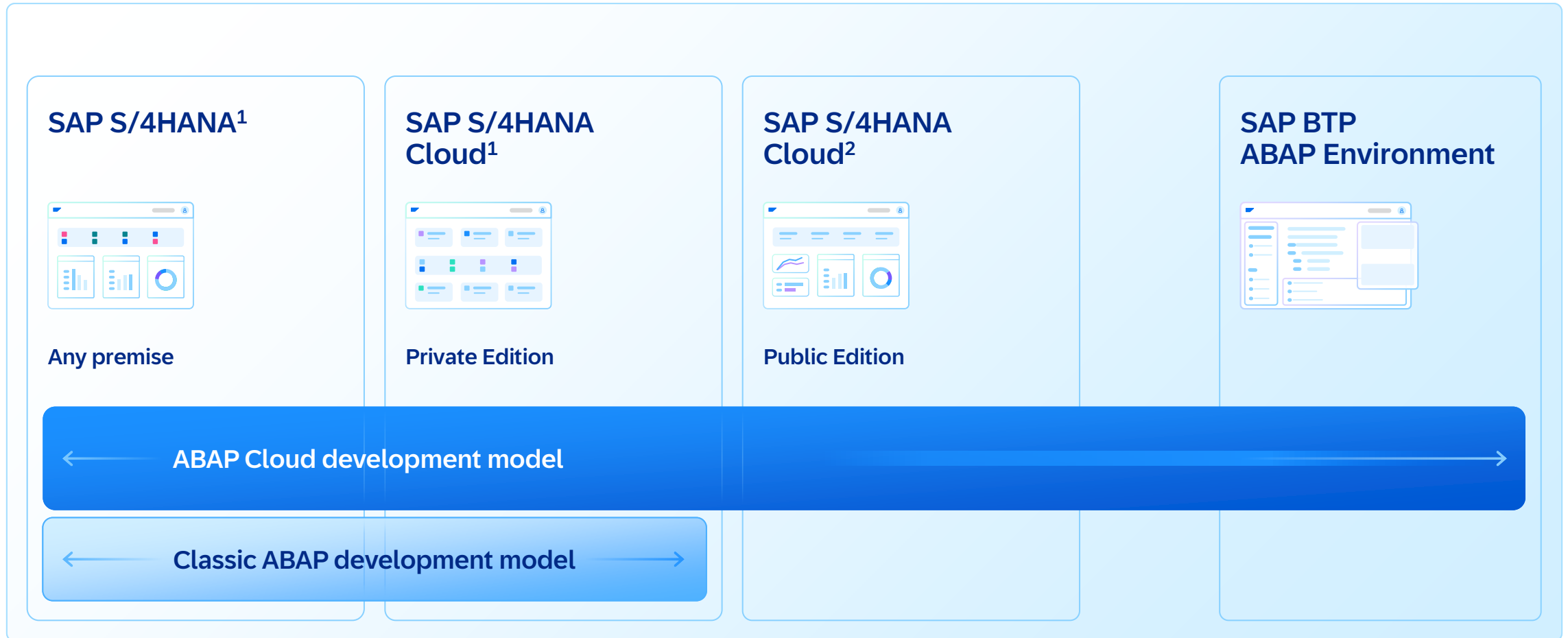
What's new in RAP?

Andre Fischer, SAP SE
June, 2026

ABAP as main technology provider for SAP S/4HANA



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 Help Portal

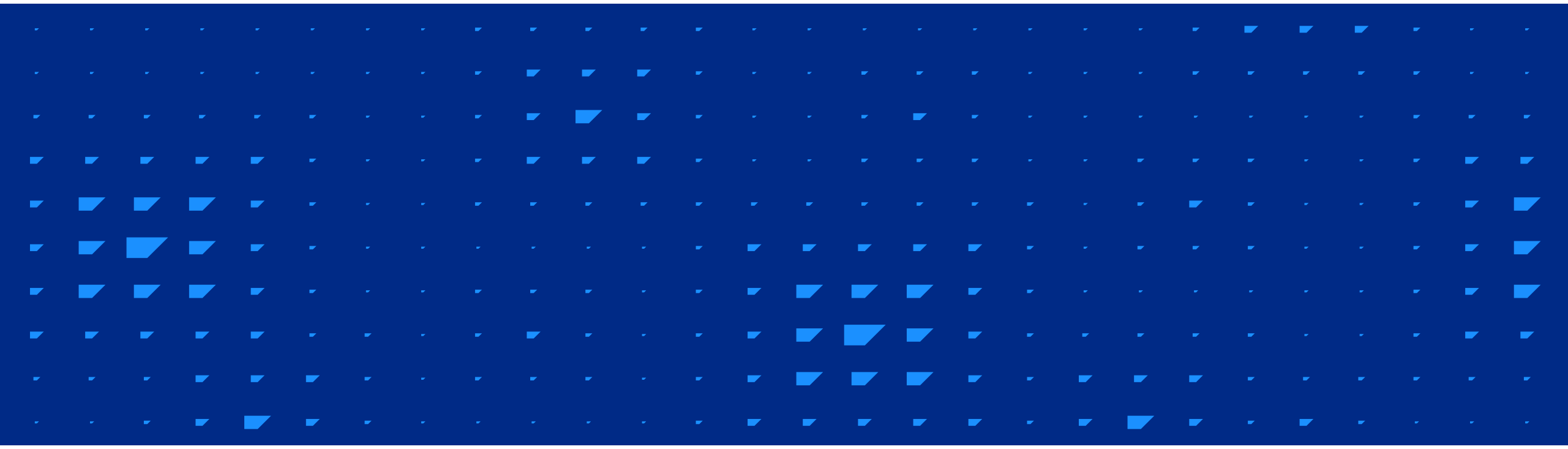
<https://help.sap.com/docs/abap-cloud/abap-rap/what-s-new?q=analytical+table&ai=true&locale=en-US&version=LATEST>

The screenshot shows the SAP Help Portal interface. At the top, there is a navigation bar with the SAP logo, 'Help Portal', and links for 'Products' and 'What's New'. Below this is a breadcrumb trail: 'Home / ABAP Cloud / ABAP RESTful Application Programming Model / What's New'. The main header area displays 'ABAP RESTful Application Programming Model' with dropdown menus for 'Version: ABAP Cross-Product' and 'English'. A navigation menu on the left is titled 'Table of Contents' and lists various topics, with 'What's New' currently selected. The main content area is titled 'What's New' and contains a sub-section 'SAP BTP ABAP environment'. Below this, there is a table with columns for 'Available as of', 'Feature', 'Description', and 'Further Information'. A search bar and a 'Columns: 6 selected' dropdown are also visible.

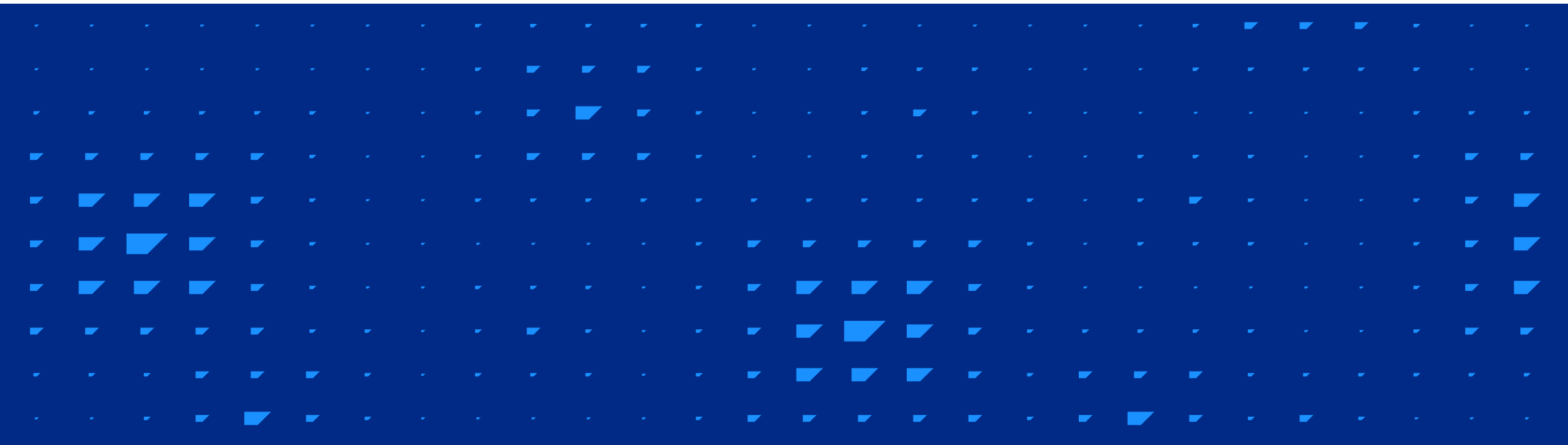


Roadmap

(Last year up to now, 2608, future)



Last year up to now

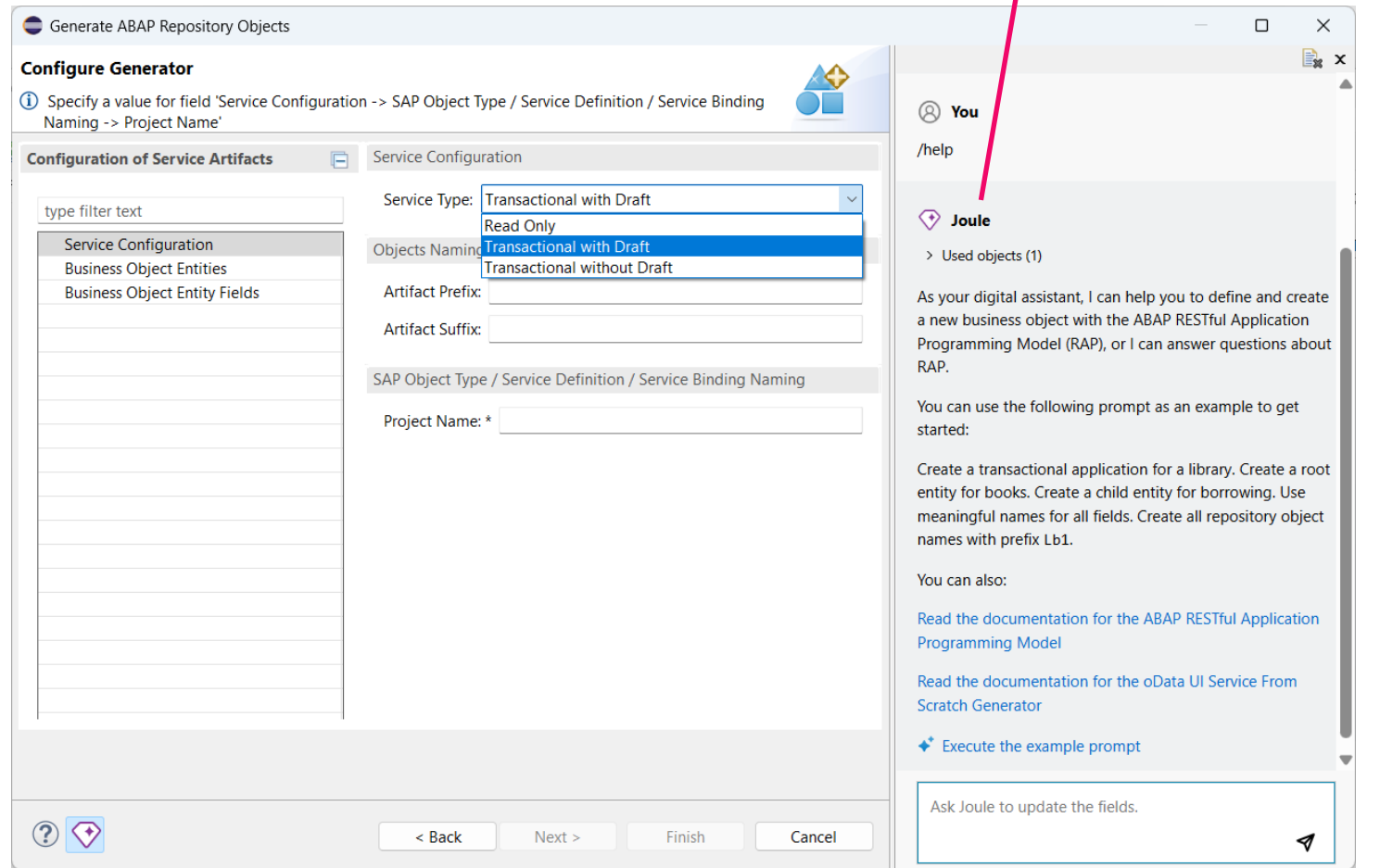


From scratch generator (2025 + Cloud)



Scope

- Create complete RAP BO including persistency (tables)
- Transactional with and without draft
- Readonly
- Support for UUID based key layout
- AI Support
- “DDIC Import”
→ Existing objects can be used as a template
- Tables
- Structures
- CDS views



The screenshot shows the 'Generate ABAP Repository Objects' configuration window. The main area is titled 'Configure Generator' and contains a 'Configuration of Service Artifacts' table and a 'Service Configuration' section. The 'Service Configuration' section includes a 'Service Type' dropdown menu with options: 'Transactional with Draft', 'Read Only', 'Transactional with Draft', and 'Transactional without Draft'. The 'Transactional with Draft' option is selected. Below this, there are fields for 'Artifact Prefix', 'Artifact Suffix', and 'Project Name: *'. The 'Project Name' field is empty. At the bottom of the window, there are navigation buttons: '< Back', 'Next >', 'Finish', and 'Cancel'. On the right side, there is an AI assistant sidebar for 'Joule'. The sidebar shows the user 'You' and the assistant 'Joule'. The assistant's message reads: 'As your digital assistant, I can help you to define and create a new business object with the ABAP RESTful Application Programming Model (RAP), or I can answer questions about RAP. You can use the following prompt as an example to get started: Create a transactional application for a library. Create a root entity for books. Create a child entity for borrowing. Use meaningful names for all fields. Create all repository object names with prefix Lb1. You can also: Read the documentation for the ABAP RESTful Application Programming Model Read the documentation for the oData UI Service From Scratch Generator Execute the example prompt'. Below the message is a text input field with the text 'Ask Joule to update the fields.' and a send button.

AI Support

AI Support

- Add entities based on prompt
- Tables with UUID based keys and administrative fields will be generated

The screenshot shows the 'Generate ABAP Repository Objects' tool. The 'Configure Generator' window is open, showing the 'Business Object Entities' configuration table. The table has the following data:

Entity Alias	Parent Na...	Cardinality	Template Name
Travel		None	/DMO/TRAVEL...
Booking	Travel	To Many	/DMO/BOOKL...

The 'Joule' AI assistant is visible on the right, displaying a prompt and its response. The prompt is: "Create a transaction application for travel management. Create the entity Travel based on the structure /DMO/TRAVEL_DATA. The second entity is Booking based on /DMO/BOOKING_DATA. The generated objects should end with suffix '001'". The response is: "I propose the following: Service Configuration: Objects Naming: Artifacts Suffix will be changed to 001, Service Binding Naming: Project Name will be changed to Trave1. Business Object Entities: Entity Booking will be created, Entity Trave1 will be created. Business Object Entities Fields: Entity Trave1: Fields BookingFeeCurr, Status, Description, CurrencyCode, TotalPrice, BookingFee, EndDate, BeginDate, CustomerId, AgencyId and TravelId will be created, Entity Booking: Fields FlightPriceCurr, CurrencyCode, FlightPrice, FlightDate, ConnectionId, CarrierId, CustomerId, BookingDate and BookingId will be created".

Add Business Entity manually

- Add entities based on templates (tables, structures, cds views, ...)

(aka DDIC Import in SEGW 😊)

- Tables with UUID based keys and administrative fields will be generated

The screenshot shows the 'Generate ABAP Repository Objects' configuration tool. The main window is titled 'Configure Generator' and specifies values for the generator, 'OData UI Service from Scratch'. On the left, there is a 'Configuration of Service Artifacts' pane with a search filter and a tree view containing 'Service Configuration', 'Business Object Entities', and 'Business Object Entity Fields' (with sub-items 'Cart' and 'Item'). The main area displays a table of 'Business Object Entities' with columns for Entity Alias, Parent Name, Cardinality, and Template Name. The table contains two rows: 'Cart' (None) and 'Item' (To Many). An 'Add Business Entity' dialog box is open in the foreground, showing fields for Entity Alias (* SubItem), Parent Name (Item), Cardinality (* To Many), and Template Name. The dialog has 'Add' and 'Cancel' buttons at the bottom.

Entity Alias	Parent Name	Cardinality	Template Name
Cart		None	ZDMO_CART_02
Item	Cart	To Many	ZDMO_ITEM_02

Add Business Entity

Entity Alias: * SubItem

Parent Name: Item

Cardinality: * To Many

Template Name:

Maintain fields

- Maintain fields on entity level
- E.g. annotate a field as Semantic Key

Generate ABAP Repository Objects

Configure Generator

Specify values for the generator, 'OData UI Service from Scratch'.

Configuration of Service Artifacts

type filter text

- Service Configuration
- Business Object Entities
- Business Object Entity Fields
 - Cart
 - Item

Business Object Entity Fields

Entity Alias: Item

Fields

Field Alias	Annotate as Semantic Key	Semantic Type	D...	Dat...	Length	De...	Currenc...	U...	Tex...
ItemId	Yes	ID		Char	10	0			
OrderedItem	No	Custom		Char	40	0			
Quantity	No	Custom		Nu...	4	0			
Currency	No	Currency Code		Cur...	0	0			
ItemPrice	No	Custom		De...	11	2	Currency		
ItemUnitPrice	No	Custom		De...	11	2	Currency		

Buttons: Add..., Edit..., Remove, Up, Down

Navigation: < Back, Next >, Finish, Cancel

Collaborative Draft (2025 + Cloud)



Collaborative Draft

- Multiple Users can work on the same draft instance in parallel
- Like in Teams where several users can work on the same file

The image displays two parallel instances of the SAP S/4HANA Collaborative Draft interface. Both screens show the 'General Information' section with fields for Notes, RequestedDeliveryDate (e.g., Dec 31, 2025), Changed By (CB9980000000), OrderID (11), Created By (CB9980000000), Created On (Sep 13, 2025, 3:58:05 PM), TotalPrice (0.00), and Created On (Sep 13, 2025, 3:44:19 PM). The 'Item Items' section is empty, displaying a 'No items available' message. The bottom of each screen shows 'Draft updated', 'Create', and 'Discard Draft' buttons. The left instance is associated with user 'aa' and the right with user 'AF'.

Requirements / Implementation details

- No need to change the existing ABAP code of your behavior implementation
- Changed Repository Objects
 - Behavior Definition
 - Behavior Projection
- Additional Repository Objects
 - Draft query view for the root entity
 - DCL for this query view

The screenshot displays the SAP ABAP Repository Explorer for the project 'ZWHATS_NEW_IN_RAP_2025_DEMO_02 (29) Collaborative Draft'. The tree structure is as follows:

- ✖ ZWHATS_NEW_IN_RAP_2025_DEMO_02 (29) Collaborative Draft
 - > 📁 Authorizations (1)
 - ✖ 📁 Business Services (2)
 - ✖ 📁 Service Bindings (1)
 - > 📄 ZDMOUI_CART_04_02 Service Binding for ZDMOUI_CART_04_02
 - > 📁 Service Definitions (1)
 - > 📁 Cloud Communication Management (1)
 - ✖ 📁 Core Data Services (14)
 - ✖ 📁 Access Controls (5)
 - > 📄 ZDMOC_CART_02 Access Control for ZDMOC_CART_02
 - > 📄 ZDMOC_ITEM_02 Access Control for ZDMOC_ITEM_02
 - > 📄 ZDMOR_CART_02 Access Control for ZDMOR_CART_02
 - > 📄 ZDMOR_CART_QUERY_02 Access Control for ZDMOR_CART_QUERY_02
 - > 📄 ZDMOR_ITEM_02 Access Control for ZDMOR_ITEM_02
 - ✖ 📁 Behavior Definitions (2)
 - > 📄 ZDMOC_CART_02 Projection Behavior Definition for ZDMOC_CART_02
 - > 📄 ZDMOR_CART_02 Behavior Definition for ZDMOR_CART_02
 - ✖ 📁 Data Definitions (5)
 - > 📄 ZDMOC_CART_02 Projection View for ZDMOR_CART_02
 - > 📄 ZDMOC_ITEM_02 Projection View for ZDMOR_ITEM_02
 - > 📄 ZDMOR_CART_02 View Entity for ZDMOCART_02
 - > 📄 ZDMOR_CART_QUERY_02 Draft query view
 - > 📄 ZDMOR_ITEM_02 View Entity for ZDMOITEM_02
 - > 📁 Metadata Extensions (2)
 - > 📁 Dictionary (6)
 - > 📁 Identity and Access Management (1)
 - > 📁 SAP Object Type Management (2)

Additional repository objects

Changed repository objects

Changes to BDEF

- with collaborative draft;
- Draft table
<draft_table> query
<draft_query_view>
- draft action Share;

```
1 managed implementation in class ZDMOBP_R_CART_02 unique;
2 strict ( 2 );
3 //changed for collaborative draft
4 //with draft;
5 with collaborative draft;
6
7 extensible;
8⊕define behavior for ZDMOR_CART_02 alias Cart
9 persistent table ZDMOCART_02
10 extensible
11 draft table ZDMOCART_D_02 query ZDMOR_CART_QUERY_02
12 etag master LocalLastChangedAt
13 lock master total etag LastChangedAt
14 authorization master( global )
15 {
16   field ( readonly )
17     UUID,
18     LocalCreatedBy,
19     LocalCreatedAt,
20     LocalLastChangedBy,
21     LocalLastChangedAt,
22     LastChangedAt;
23
24   field ( numbering : managed )
25     UUID;
26
27   create;
28   update;
29   delete;
30
31   draft action Activate optimized;
32   draft action Discard;
33   draft action Edit;
34   draft action Resume;
35   draft determine action Prepare;
36
37   //added for collaborative draft
38   draft action Share;|
39
40⊕ mapping for ZDMOCART_02 corresponding extensible
41 {
42   UUID = uuid;
43   Currency = currency;
```

Changes to BDEF projection

- use collaborative draft;
- use action Share;

```
1 projection implementation in class ZDMOBP_C_CART_02 unique;  
2 strict (AF2.0);  
3 extensible;  
4  
5 //use draft;  
6 use collaborative draft;  
7  
8 use side effects;  
9 define behavior for ZDMOC_CART_02 alias Cart  
10 extensible  
11 use etag  
12 {  
13     use create;  
14     use update;  
15     use delete;  
16  
17     use action Edit;  
18     use action Activate;  
19     use action Discard;  
20     use action Resume;  
21     use action Prepare;  
22  
23     //added for collaborative draft  
24     use action Share;  
25  
26     use association _Item { create; with draft; }  
27  
28 }  
29  
30 define behavior for ZDMOC_ITEM_02 alias Item  
31 extensible  
32 use etag
```

5 //use draft;
6 use collaborative draft;

23 //added for collaborative draft
24 use action Share;

23 //added for collaborative draft
24 use action Share;

New objects – Draft query view and DCL

```
1 @AbapCatalog.viewEnhancementCategory: [#NONE]
2 @AccessControl.authorizationCheck: #NOT_REQUIRED
3 @EndUserText.label: 'Draft query view'
4 //@Metadata.ignorePropagatedAnnotations: true
5 @ObjectModel.usageType:{
6     serviceQuality: #X,
7     sizeCategory: #S,
8     dataClass: #MIXED
9 }
10 define view entity ZDMOR_CART_QUERY_02 as select from zdmocart_d_02
11 {
12     key uuid as Uuid,
13     currency as Currency,
14     notes as Notes,
15     orderid as Orderid,
16     totalprice as Totalprice,
17     requesteddeliverydate as Requesteddeliverydate,
18     localcreatedby as Localcreatedby,
19     localcreatedat as Localcreatedat,
20     locallastchangedby as Locallastchangedby,
21     locallastchangedat as Locallastchangedat,
22     lastchangedat as Lastchangedat,
23     draftentitycreationdatetime as Draftentitycreationdatetime,
24     draftentitylastchangeddatetime as Draftentitylastchangeddatetime,
25     draftadministrativeatauid as Draftadministrativeatauid,
26     draftentityoperationcode as Draftentityoperationcode,
27     hasactiveentity as Hasactiveentity,
28     draftfieldchanges as Draftfieldchanges
29 }
30
```

```
1 @EndUserText.label: 'Access Control for ZDMOR_CART_QUERY_02'
2 @MappingRole: true
3 define role ZDMOR_CART_QUERY_02 {
4     grant
5         select
6         on
7             ZDMOR_CART_QUERY_02
8         where
9             inheriting conditions from entity ZDMOR_CART_02;
10 }
11
```

- Future releases:
 - Support for table entities as draft persistence
➔ No draft query view
 - Table entity can be secured by DCL itself

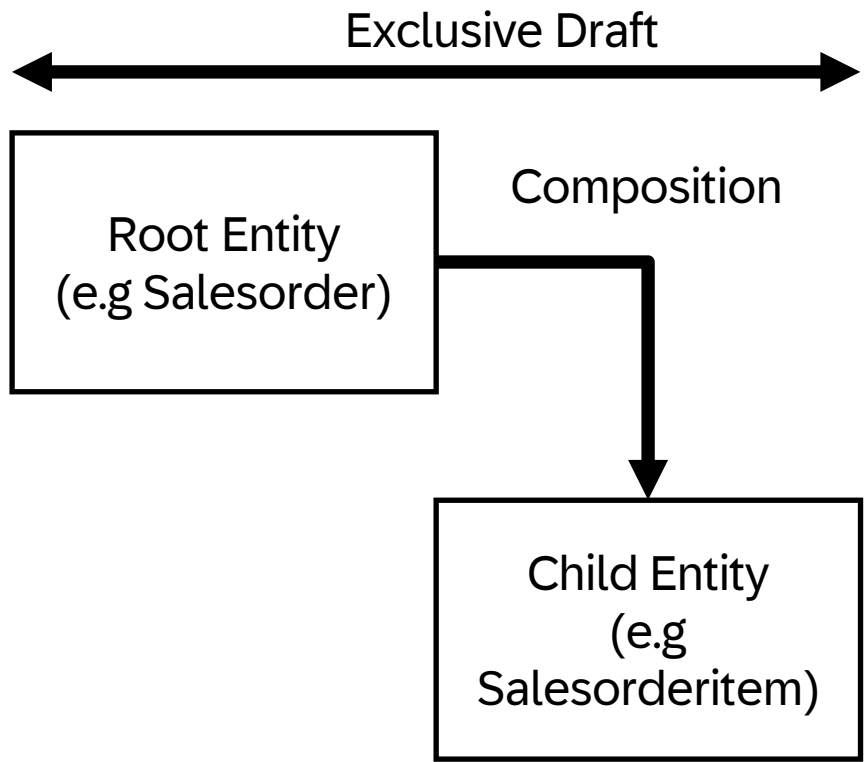
Collaborative Draft - Demo

adt://XYZ/sap/bc/adt/businessservices/bindings/%2fdmo%2fui_travel_o4_cd

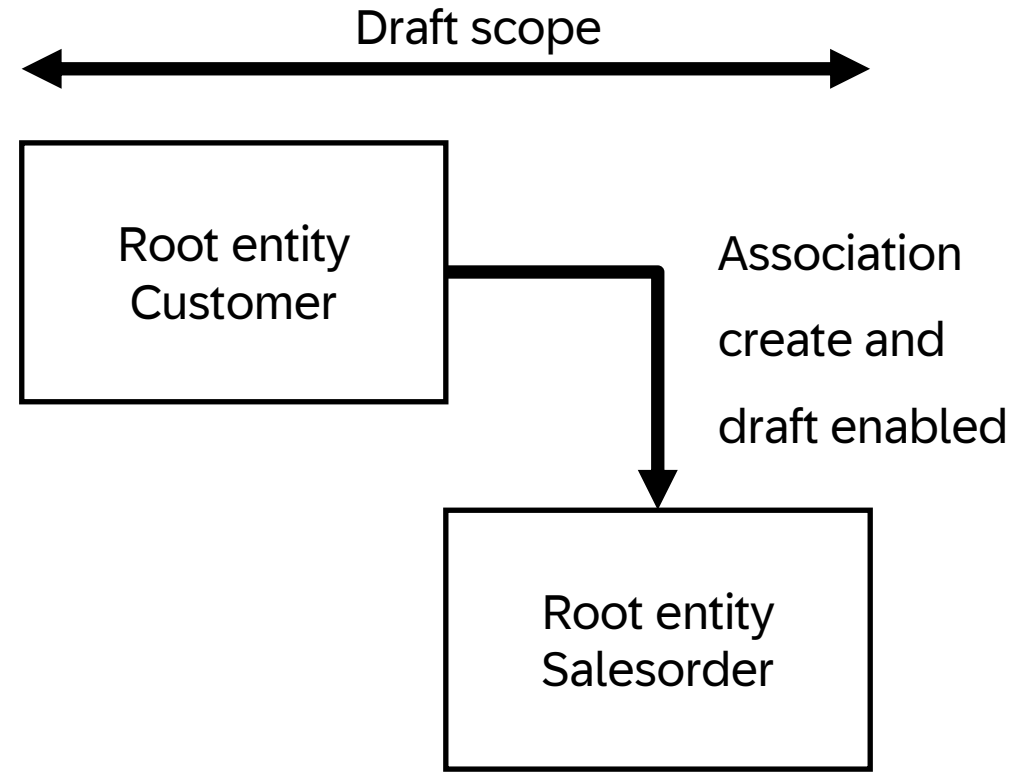
Draft scope 2602



Exclusive Draft vs. Draft Scope



- If root entity is locked, all Items are locked as well



If a customer is edited all salesorders that belong to this customer are locked as well (dependend object)
If a customer is created but in draft alongside with salesorders all objects are deleted when the customer is deleted

Draft scope - demo

adt://AI5/sap/bc/adt/bo/behaviordefinitions/%2fdmo%2fc_traveltp_xbo_draftdp

projection;

strict (2);

extensible;

use side effects;

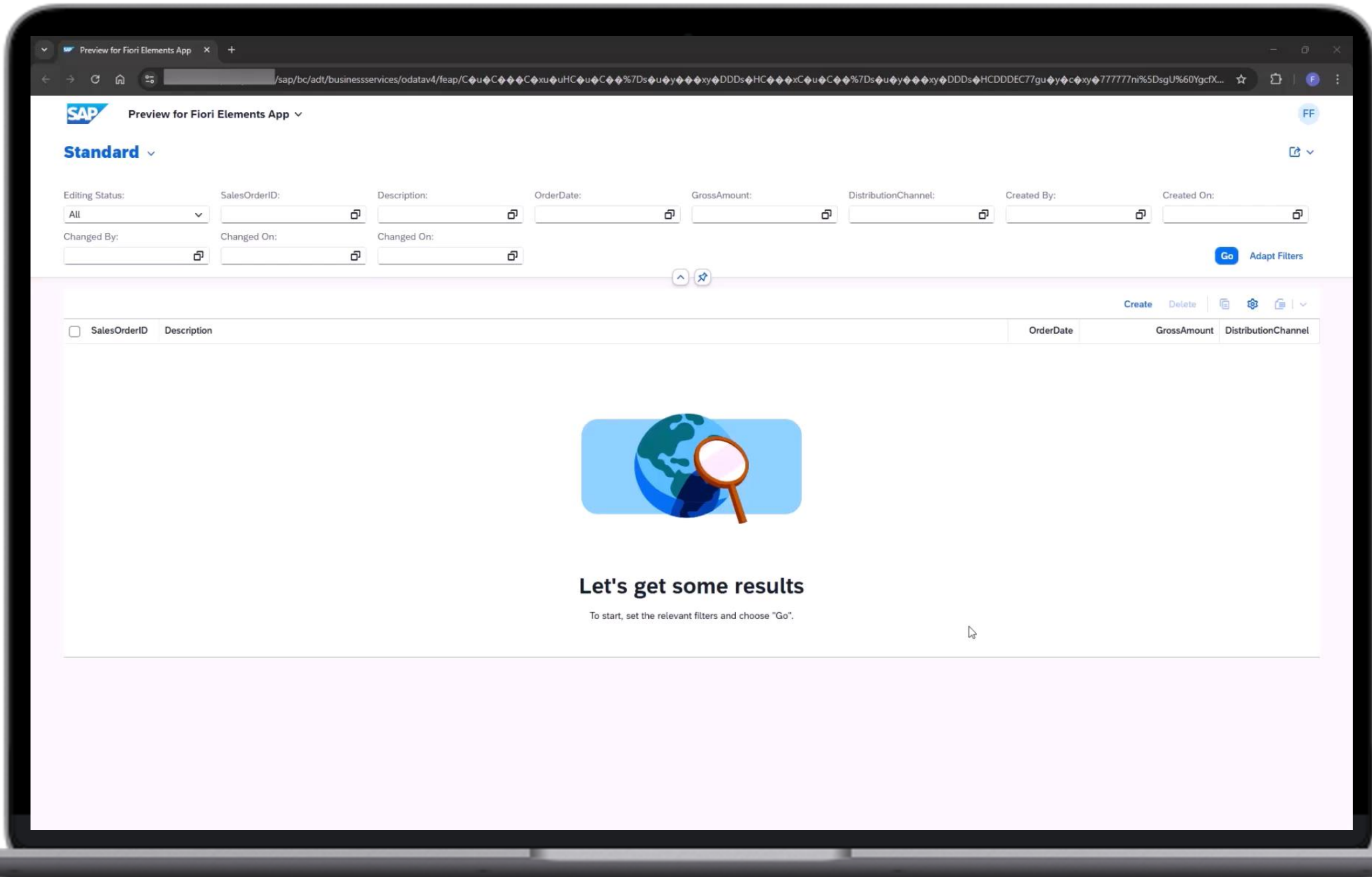
use draft as dependent;

define behavior for /DMO/C_TRAVELTP_XBO_DRAFTDP alias Travel

Analytical table – read only (2511)

Analytical Table

Aggregation and grouping capabilities



Recommendations (2602)



Accelerating Business AI *with Input Assistance*

Supplier:

Code Group:

- 1 | Inconsistent Data Entry**
Slightly different terms of different business users may lead to **data inconsistencies**
- 2 | Time-Consuming**
Manually typing in values, especially for long or complex terms is **inefficient**
- 3 | Knowledge Enablement**
Business users which are not yet familiar in their domain are **struggling to search and find** the right dependent values

Accelerating Business AI *with Input Assistance*

Supplier:

Code Group:

i Recommendation based on your business data.

Recommendations
Orlando, FL
Other
Bremerhaven Harbor, HB (DE BRV)
Amsterdam Harbor (NL AMD)

- 1 | Standardized Data Entry**
The system encourages the use of **standardized terminology**, leading to more consistent and accurate data
- 2 | Increased Efficiency**
A business user can **quickly select recommended values** from the dropdown list, significantly speeding up the data entry process
- 3 | Knowledge Enablement**
Machine learning algorithms are trained to **predict the most likely values** for each input field based on the context of the data

Accelerating Business AI with Input Assistance



Supplier:

Code Group:

i Recommendation based on your business data.

Recommendations

- Orlando, FL

Other

- Bremerhaven Harbor, HB (DE BRV)
- Amsterdam Harbor (NL AMD)

behavior definition

```
{
  field ( recommendations ) Supplier, CodeGroup;
  recommendation function getRecommendedValuesFunction;
}
```

behavior implementation

```
method getRecommendedValuesFunction.
  ...
endmethod.
```

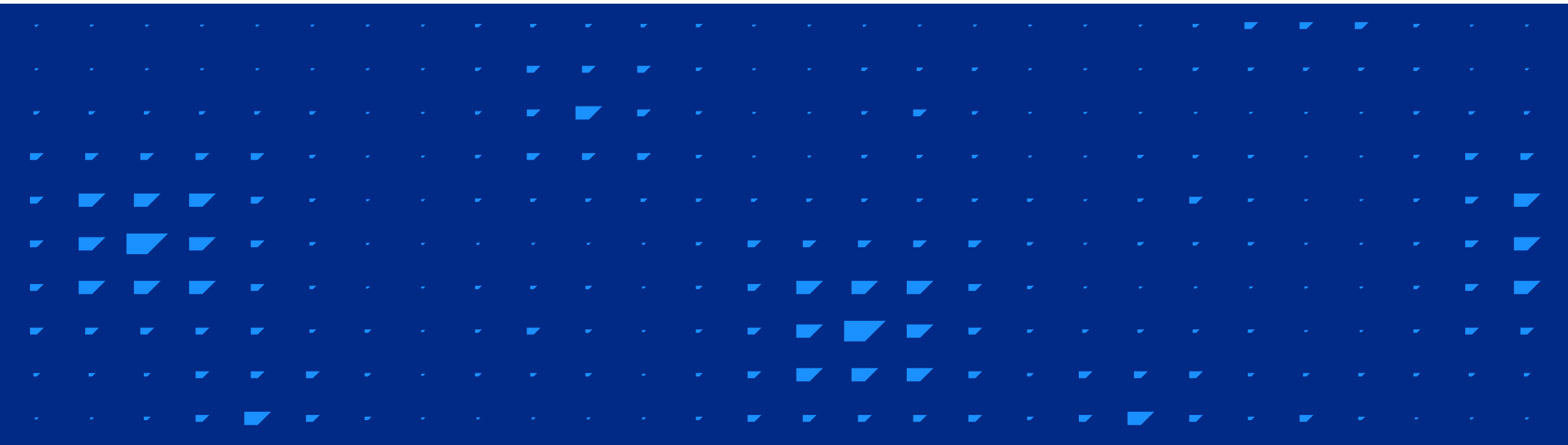
asynchronous communication

Event-Driven Side Effect

recommendation providers



2605



ADT for VSCode – First version

Available on the [VS Code marketplace](#)

- Supports backends from SAP NetWeaver 7.3 EHP1 SP04 or higher
- First version focuses on RAP UI service development. Includes MCP server supporting ABAP development via AI agents
- See [announcement blog post](#) and [roadmap](#) for more information on the next releases

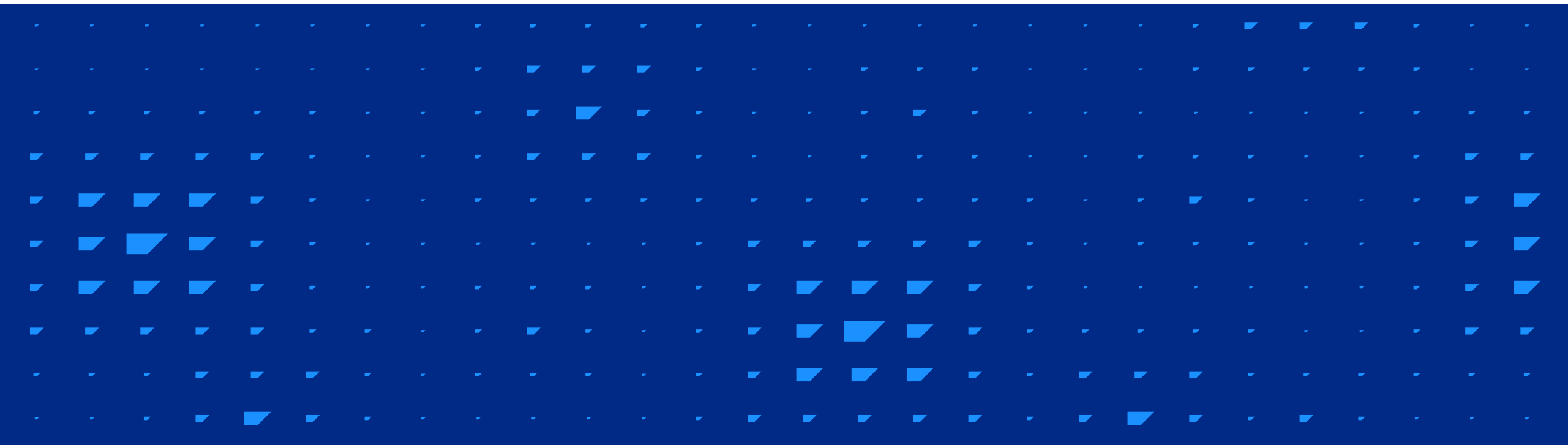
CDS Table Entities as Draft Persistency

Support of CDS table entities as basis for the transactional draft

The screenshot shows a SAP Fiori draft form for a table entity. The form is divided into several sections, each containing a grid of input fields. The top section has a header bar with the SAP logo. The form is currently in a draft state, as indicated by the 'Draft Table Entity' label in the code. At the bottom right, there are 'Save' and 'Cancel' buttons.

```
1 @ClientHandling.type: #CLIENT_DEPENDENT
2 @AbapCatalog.deliveryClass: #APPLICATION_DATA
3 @AccessControl.authorizationCheck: #NOT_REQUIRED
4 @EndUserText.label: 'Draft Table Entity'
5 define table entity ZDRAFT_TABLE_ENTITY
6 {
7   key uuid          : sysuuid_x16;
8   salesorderid      : abap.char(10);
9   description       : abap.char(200) null;
10  orderdate         : abap.dats null;
11  @Semantics.amount.currencyCode : 'zsalesorder003_d.grossamountcurr'
12  grossamount       : abap.curr(15,2) null;
13  distributionchannel : abap.char(10) null;
14  grossamountcurr   : abap.cuky null;
15  localcreatedby    : abp_creation_user null;
16  localcreatedat    : abp_creation_tstmpl null;
17  locallastchangedby : abp_locinst_lastchange_user null;
18  locallastchangedat : abp_locinst_lastchange_tstmpl null;
19  lastchangedat     : abp_lastchange_tstmpl null;
20  include sych_bdl_draft_admin_inc;
21
22 }
```

2608, 2611

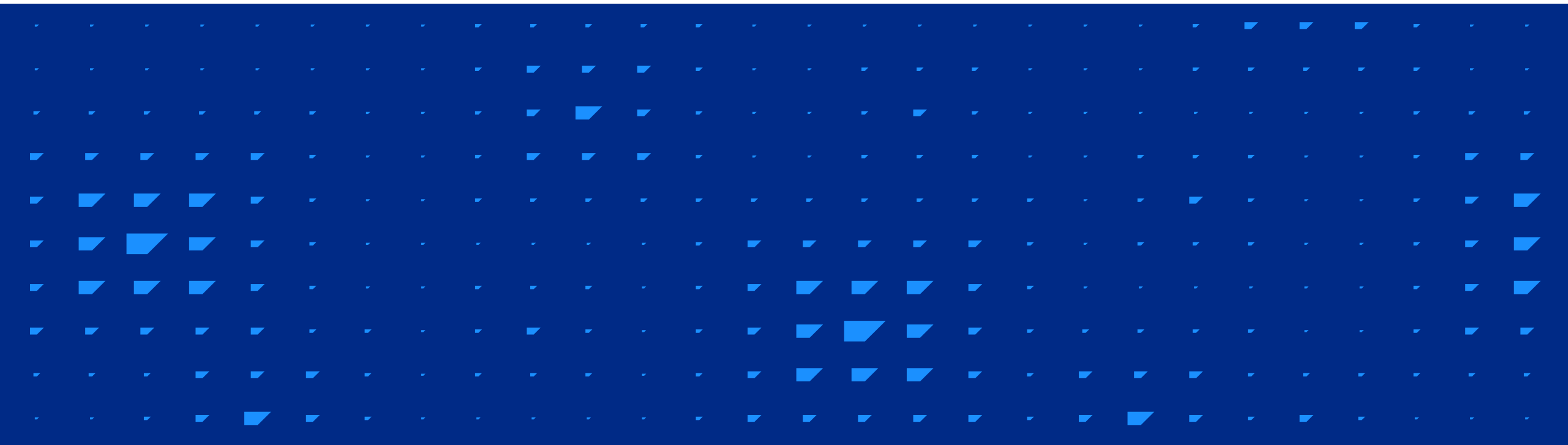


ADT for VSCode – adding most wanted features

- Support for basic authentication
- Creation of DDIC objects and packages
- ...

The screenshot displays the Visual Studio Code interface with the ADT (ABAP Development Tools) extension. The Explorer view on the left shows a project structure with folders like 'ZLOCAL [PMD_AF]', 'ZABAP_CLOUD_IN_ACTION', and 'ZR_ACIA_DEMO_01'. The main editor shows an ABAP code snippet for a managed implementation in class ZBP_R_ACIA_DEMO_01. The code includes annotations for draft, extensible, and persistent table, and defines a table ZACIA_DEMO_01 with fields like OrderUuid, LastChangedAt, CreatedBy, CreatedAt, LocalLastChangedBy, LastChangedBy, and LocalLastChangedAt. On the right, a chat window displays a prompt to 'GENERATE SAP FIORI TRAVE...' and a response providing a specification for a transactional SAP Fiori app for travel management, including package, entity, and naming conventions.

Future



Change Documents API

Access change documents via publicly released APIs

The screenshot displays the SAP IDE interface. On the left, the ABAP editor shows the definition of the change document `I_CHANGEDOCUMENTFORCHGDOCOBJ`. The code includes metadata, a root view entity, parameters, and a list of keys. On the right, the Properties window shows the configuration for this object, including contract settings and API catalog assignments.

```
1 @AccessControl.authorizationCheck: #NOT_REQUIRED
2 @EndUserText.label: 'Change Documents Parameter View'
3 @Metadata.ignorePropagatedAnnotations: true
4
5 define root view entity I_ChangeDocumentForChgDocObj
6   provider contract transactional_interface
7
8   with parameters
9     @Semantics.developmentObject.instanceOf: 'CHDO'
10    p_ChangeDocObject : changedocobject
11    as projection on R_ChangeDocumentForChgDocObj
12
13
14   key ChangeDocObjectClass,
15   key ChangeDocObject,
16   key ChangeDocument,
17   key ChangeDocDatabaseTable,
18   key ChangeDocShortTableKey,
19   key ChangeDocDatabaseTableField,
20   key ChangeDocChangeType,
21   key ChangeDocTableKey,
22
23   CreatedByUser,
24   // username -> Text
25   ChangeDocCreatedByUser,
26
27   @Semantics.systemDate.createdAt: true
28   CreationDate,
29   @Semantics.systemTime.createdAt: true
30   CreationTime,
31
32   // Created on
33   CreationDateTimeUTC,
34   CreationDateTimeSystem as CreationDateTime,
35
36   ChangeTransactionCode,
37   ChangeTypeText,
38   @Semantics.language: true
39   ChangeDocLanguage,
40
41   ChangeDocNewFieldValue, // with CURR or UNIT
42   ChangeDocPreviousFieldValue, // with CURR or UNIT
43
44   ChangeDocTextIsChanged,
45
46   // string values
47   ChangeDocNewString,
48   ChangeDocPreviousString,
49   ChangeDocNewShortString,
50   ChangeDocPreviousShortString,
51   ChangeDocNewRawString,
52   ChangeDocPreviousRawString
53
54
55 where ChangeDocObjectClass = $parameters.p_ChangeDocObject
56
```

Properties Window: [Y13] I_CHANGEDOCUMENTFORCHGDOCOBJ

- General**
- Specific**
- API State**
 - Contract C0 not yet set.
 - Use System-Internally (Contract C1)**
 - Release State: **Released**
 - Local Comment:
 - Use in Cloud Development: Yes
 - Use in Key User Apps: Yes
 - Last Changed by: VAHL (Britta Vahl)
 - Last Changed on: April 23, 2026
 - Use as Remote API (Contract C2)**
 - Contract C2 is not set.
 - API Catalog Assignments**
 - The object is not assigned to any API catalog.
- Transport**

RAP Editable Analytical Table

Aggregation and grouping capabilities with draft support

<input type="checkbox"/>	Document Number	Description	Posting Key	Document Type	Posting Date	Document Amount	Currency	Offset Account	Assignment
<input checked="" type="checkbox"/> Assignment 2938481									
<input type="checkbox"/>	90000201	Laptops	40	Invoice	11.21.2020	3,100.90	EUR	660200	2938481 >
<input type="checkbox"/>	90000213	Servers	40	Invoice	11.21.2020	11,932.71	EUR	700000	2938481 >
<input type="checkbox"/>	90000183	Tables	40	Invoice	11.21.2020	39,392.74	EUR	70000	2938481 >
<input type="checkbox"/>	90000623	Printers	40	Invoice	11.21.2020	9,349.17	EUR	660200	2938481 >
<input type="checkbox"/>	90000871	3D Printers	40	Invoice	11.21.2020	1,283.57	EUR	550000	2938481 >
						65,056.09	EUR		
<input checked="" type="checkbox"/> Assignment 29392182									
<input type="checkbox"/>	90042121	Laptops	50	Invoice	11.21.2020	4,935.71	EUR	660200	2938481 >
<input type="checkbox"/>	90003421	3D Printers	50	Invoice	11.21.2020	6,192.00	EUR	700000	2938481 >
<input type="checkbox"/>	90002125	Servers	40	Invoice	11.21.2020	12,342.00	EUR	70000	2938481 >
						23,469.71	EUR		
<input checked="" type="checkbox"/> Assignment 29392182									
<input type="checkbox"/>	90000201	Phone	40	Invoice	01.14.2021	11,193.70	EUR	660200	2938481 >
<input type="checkbox"/>	90000213	Servers	40	Invoice	01.14.2021	2,932.71	EUR	700000	2938481 >
<input type="checkbox"/>	90000183	Chair	40	Invoice	01.14.2021	11,193.70	EUR	70000	2938481 >
<input type="checkbox"/>	90000623	Printers	40	Invoice	01.14.2021	5,349.17	EUR	660200	2938481 >
<input type="checkbox"/>	90000871	3D Printers	40	Invoice	01.14.2021	2,283.57	EUR	550000	2938481 >
						33,151.89	EUR		
						148,989.99	EUR		

CABAP

**SURVEY
IS NOW
LIVE.**



Scan to participate



Thank you.

Name
Job title
Address
City
Email





CABAP

SURVEY
IS NOW
LIVE.



Scan to participate

