



Managing Deprecations in SAP S/4HANA Cloud

P U B L I C E D I T I O N

Pascal Renet

I *still* do...because of you.

Table Of Contents

Introduction	6		
Purpose	7		
Deprecate. A definition *	8		
Lifecycle of objects	9		
Update to the lifecycle of CDS Views	10		
What objects are Deprecated in SAP S/4HANA Cloud ?	12		
Deprecation Policies	13		
Where to find information on Deprecated objects	14		
Outside SAP S/4HANA Cloud	15		
The What's New Viewer	15		
Changes in What's New Viewer Category Management	19		
Overview of Changes in Extensibility Objects	20		
In the RASD (Release Assessment and Scope Dependency) App	23		
The CDS Comparator	27		
Color coded changes	31		
Inside SAP S/4HANA Cloud	33		
The View Browser application	33		
The Extensibility Inventory application	37		
		The Dependencies section	41
		The Tasks section	43
		The Custom Fields application	47
		The Custom Logic application	50
		The Display Inbound Services application	54
		The Display Communication Scenarios Application	56
		The Manage Business Role Changes After Upgrade Application	59
		Restriction Types	60
		Deprecated Business Catalogs	63
		Further Reading	67
		The Business Catalogs Application	68
		The Custom CDS application	71
		Further Reading	72
		Inside ADT (ABAP Developer Tools)	73
		CDS Deprecations In Practice	77
		Introduction to the use case 1 - Analytical CDS	80
		Setting the scene	80
		Processing the CDS Deprecation	83
		The deprecation announcement	83
		Understanding the size of the rework	84
		The Process	97

Step 1 - Amending the Custom CDS and adding the successor elements	99
Step 2 - Check the non regression of your changes in the Custom CDS	102
Step 3 - Amending the Custom Analytical Query	103
Step 4 - Amending the Query report - Multi Dimensional Report	111
Step 5 - Amending the Embedded SAC story	112
Step 6 - Amending the custom CDS	116
Step 7 - Better safe than sorry	119
Introduction to the use case 2 - External API	120
Setting the scene	120
Processing the CDS Deprecation	122
The deprecation announcement	122
Understanding the size of the rework	123
The Process	132
Step 1 - Amending the Custom communication scenario	134
Step 2 - Amending the Custom CDS	136
Step 3 - Amending the Custom Communication Scenario	141
Step 4 - Test the Integration flow	142
Amending the order of the field names	147
Wrap-up	151

Introduction

(To) deprecate. Whilst this is something that has been around for probably...ever, it has become more prominent over time as the pace of innovation is relentlessly getting faster and faster and companies wanting to innovate understand that it is not on yesterday's technology that you will engineer tomorrow's Cloud. Whilst the concept of *Deprecating something* is not limited to cloud, as it also applies to on-premise software and everything in between, it is probably more so applicable to cloud solutions (that is SaaS), where the updates and upgrades are that much more frequent, and the timing of these updates and upgrades is imposed on at known dates and times by the service provider, rather than decided on, or requested by customers.

Every software company, irrespective of the mode of delivery of their software, will deprecate - that is not something specific to SAP, but of course what and how something is deprecated will be specific to a company.

Thank you for taking the time to download this document - I hope you get something out of it. If there are specific use cases that you would like to see covered in - maybe - future updates, do not hesitate to get in touch and let me know.

Please note that the content provided in this document is a personal viewpoint.

Purpose

This document has been put together, to try and assemble in one place all the key references that you need to be aware of in the context of deprecations, in the hope that it will help you go through upgrades with more knowledge, confidence and efficiency. The point of this document is not to copy/paste all the information that is available to you on the internet, but simply to point you in the right direction of where to find it.

The aim is to help you be more informed, by knowing where you can find information on deprecated objects, inside and outside of your system, what tools and applications are available to help you digest deprecations and what the individual deprecation policies of deprecated objects are.

This document is essentially in two parts. The first part will be more informational and cover deprecations in general, whereas the second will be more practical and offer a specific focus on the deprecation of Core Data Services (CDS). By looking at a couple of realistic CDS deprecation uses cases (a custom CDS used in an analytical reporting context and a custom CDS exposed as an external API), we will look at the changes that we need to perform in the SAP S/4HANA Cloud system, offering detailed step by step instructions, in order to successfully navigate the deprecation of CDS entities or CDS elements.

Lastly, this document is specifically written for and targeted at users of SAP S/4HANA Cloud, public edition. Whenever a reference is made to SAP S/4HANA Cloud, it is intended to mean the public edition of it.

Deprecate. A definition *

Deprecate /'deɪrɪkeɪt/

Verb

Express disapproval of. (chiefly of a software feature) be usable but regarded as obsolete and best avoided, typically because it has been superseded.
*"this feature is deprecated and will be removed in later versions"**

Before going further in this document, it probably makes sense to start by trying to define what a deprecation actually is. So whilst objects are deprecated with a different cadence and maybe a different apparent lifecycle, the following is a general personal definition that I would put forward to define the term deprecation.

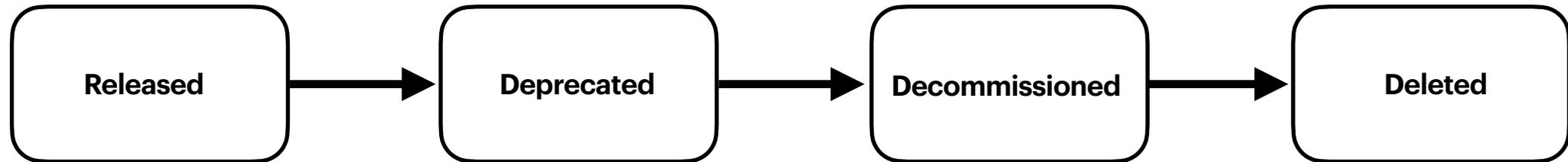
"A Deprecation is when a feature or function that is for now still available, has a more effective successor. From the time of its announcement, adoption of a deprecated object for new uses is strongly discouraged. Furthermore, if the deprecated object is used, every effort to transition to the successor object should be made, before the deprecated object is no longer supported or available, to avoid a disruptive, regressed, or complete loss of functionality. "

You should familiarise yourselves with the object specific lifecycle, deprecation policies and timelines to plan your changes accordingly and execute them serenely.

* Definition from Oxford Languages

Lifecycle of objects

Following the definition of *Deprecate*, we probably also need to clarify where this fits within the lifecycle of an object. Outwardly, to customers objects may use different words to reflect their - technical - lifecycle status (to reflect their availability, usability, supportability...), but generally speaking, the lifecycle of an object will be as depicted below:



- **Released** : With this technical status, the object is released for use by customers. The object is maintained for productive usages and can possibly benefit from enhancements (for example, performance enhancements, or the addition of additional field(s), etc....).
- **Deprecated**: With this technical status, this signals to customers that there is now a need to adapt - to transition to a successor object. Depending on the object, the deprecation clock is now ticking, and there is a risk that if no action is taken - to transition to a successor object - by the end of the deprecation period, that the deprecated object will no longer function as previously or indeed still be available.
- **Decommissioned**: From a user point of view this status is used here to represent a phase where the object might possibly still exist in the system but is no longer supported (in which case it must not be used) and should not be used or cannot be used by design (by controlling visibility or use).
- **Deleted**: Should be self-explanatory, the object is deleted from the system. Any references to it would be errors.

Update to the lifecycle of CDS Views

With the 2408 Release of SAP S/4HANA Cloud, SAP introduced a variation to the 'deprecated' status. This variation or update, applies only to CDS views and CDS elements (fields), and is valid on any premise. This variation was born out of the desire to provide customers with more certainty with respect to the date on which a CDS view or element will be decommissioned, and thus providing customer an additional, key piece of information to better plan the time at which to adapt. To this effect, the deprecated status is further defined as:

- Deprecated - *without a planned decommissioning date*, or
- Deprecated with a planned decommissioning date

SAP may update a CDS view or element from a released status to a 'deprecated' (*without a planned decommissioning date set*) status before further updating it to a 'deprecated with a planned decommissioning date' status. Similarly, SAP may directly transition a CDS view or element from a released status directly to a 'deprecated with a planned decommissioning date' status.

What is noteworthy is:

- From the time a CDS view or element is deprecated with a planned decommissioning date, a minimum period of 12 months will elapse before the object is decommissioned,
- From the time a CDS view or element is deprecated you should plan for adaption, but it is not mandatory. On the other hand, you must plan for adaption once the CDS view or element is deprecated with a planned decommissioning date,
- The planned decommissioning date will be expressed in the format of <MonthName><MonthYear>.

This status update will also affect the criticality of the -adaption- rework tasks that you need to perform, depending on whether a deprecation is set with or without an advertised planned decommissioning date.

The table below recapitulates how the criticality of rework tasks will change, and how a user will be notified, depending on whether the extensibility considered is developer extensibility ADT (ABAP developer Tools) or key user extensibility (e.g. Apps such as Custom CDS Views, Extensibility Inventory, etc...)

	Deprecated, without planned decommissioning date set	Deprecated, with planned decommissioning date set	Decommissioned
Syntax warning in the code editor (ADT for developers or Web Editor for key users)	Warning (no change)	Warning	Warning (no change)
ATC Check (for developers)	Information (today: Warning)	Warning (new)	Error (no change)
Rework task (for key users)	Priority Medium (today: High)	Priority High	Priority Very High (no change)

What objects are Deprecated in SAP S/4HANA Cloud ?

As the technology components that are leveraged to offer SAP S/4HANA Cloud evolve ('code and content'), then so may the type and number of objects that are deprecated. At the time of writing this document the most noteworthy objects that are deprecated are the following:

- **IAM** (Identity Access Management) objects. This typically includes Business Catalogs or Template Business Roles but can also represent Restrictions,
- **Apps** (Short for application), irrespective of the technology powering the app (Fiori, webdynpro..). An app is typically represented as a 'tile' on your Fiori Launchpad,
- **CDS** (Core Data Services). This refers to an entire CDS entity, or element(s) contained within a CDS,
- **API** (Application Programming Interface). In this sense I refer to API as an *external* API, that is used for communications to occur between two connected systems (whether inbound or outbound).

Deprecation Policies

The deprecation policies and deprecation timelines (i.e how much time do you have to adapt) that apply to the individual objects are not identical, and thus it is important for you to know what they are, so that you may organise your transition activities knowing exactly how much time you have before an object reaches the end of its deprecation period. For example the deprecation period of an app, is different to that of a CDS.

The table below will recapitulate for each object where you can find more information regarding their respective deprecation policies.

Object	Link to Policy
Core Data Services (CDS) and CDS elements	Link Lifecycle of a released CDS
Application Programming Interface (API)	Link
Applications (Apps)	Link Related -> Phase-in / Phase-Out
Business Catalogs	Link

Where to find information on Deprecated objects

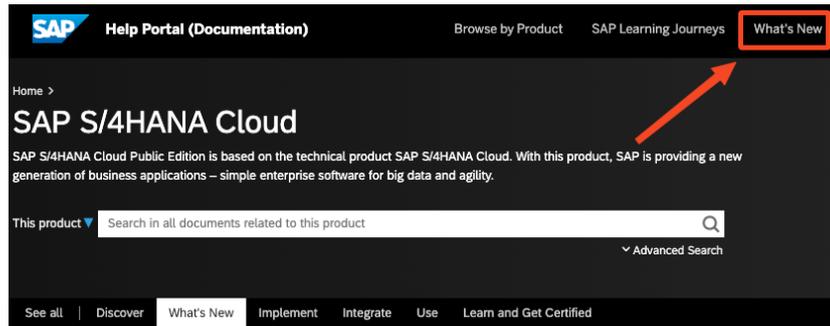
In this section we will present the various information sources that are available to you, to find out more about deprecations.

Some will reside in well known information sources outside you SAP S/4HANA Cloud system such as the [What's New viewer](#), [SAP S/4HANA Cloud Help](#), [Release Assessment and Scope Dependency \(RASD\) tool](#), blogs, etc., some will be inside your SAP S/4HANA Cloud system.

We'll start with the former.

Outside SAP S/4HANA Cloud

The What's New Viewer



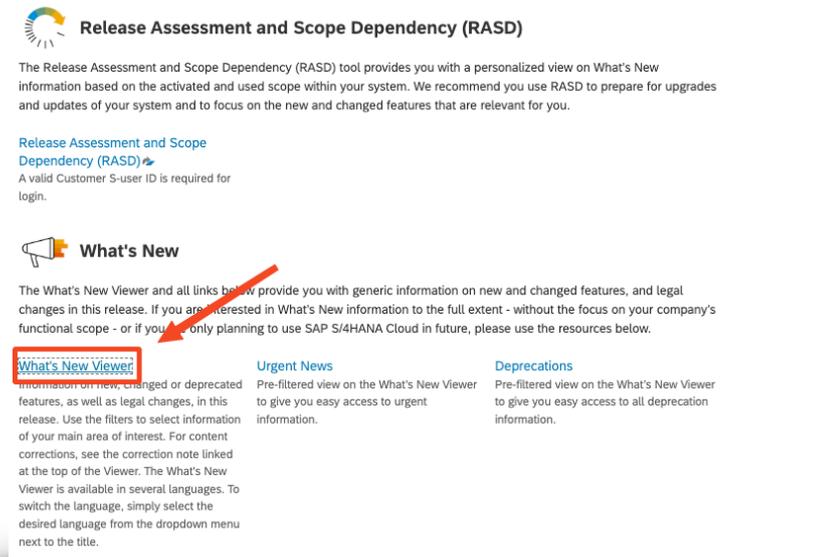
The [What's New Viewer](#) for SAP S/4HANA Cloud provides with you with an exhaustive source of information pertaining to all novelties, changes (including legal changes) and deprecations. However, please note that deprecations only happen during a major release (i.e 2408, 2502..), and not during intermediates updates (2408.2, 2502.1,...) in between major releases.

You can access the What's New viewer from the main [SAP S/4HANA Cloud help](#) page as shown, or directly bookmarking this [link](#).

From the main help landing page, go to the What's new tab, and from there, click anyone of the links shown in the What's New section to access the What's New viewer.

Also note that ahead of a release, about one month prior, a preliminary version of the What's New viewer content will be published. But as the name 'preliminary' indicates, the preliminary content is subject to change and may differ from the final content, when the latest version of SAP S/4HANA Cloud is RTC (Released To Customers).

The What's New Viewer provides *ALL* the information pertaining to an upgrade or update. i.e it is not specific to your usage of the SAP S/4HANA Cloud system.



Once you reach the [What's New Viewer](#) landing page, you may need to adjust the filters in order to be able to limit the information shown to Deprecated items. To do so:

1. Make sure you select a full release, not an in between update,
2. Select the announcement Type value to Deprecated,
3. Fine tune your selection to limit the deprecations output to specific categories, or you can also choose to view them all independent of category.

The screenshot shows the 'What's New Viewer - SAP S/4HANA Cloud' interface. The top navigation bar includes 'Home > SAP S/4HANA Cloud' and 'English'. Below the title is a search bar and utility icons for 'Hide Filters', 'Download CSV', 'Favorite', and 'Download PDF'. The main filter area contains dropdowns for 'Line of Business', 'Solution Area', 'Category', 'Type', 'Scope Item', and 'Valid as Of'. The 'Type' dropdown is set to 'Deprecated'. The 'Valid as Of' dropdown is set to 'SAP S/4HANA Cloud 2408'. The 'Category' dropdown is open, showing a list of categories with 'Deprecated' selected. The 'Valid as Of' dropdown is also open, showing a list of release versions with 'SAP S/4HANA Cloud 2408' selected. A table below the filters shows columns for 'Line of Business', 'Solution Area', 'Solution Capabilities', and 'Short Description'. The table content is partially visible, showing 'Application Platform and Infrastructure' under 'Line of Business' and 'ABAP Platform' under 'Solution Area'. Red circles with numbers 1, 2, and 3 are placed over the 'Valid as Of' dropdown, the 'Type' dropdown, and the 'Category' dropdown respectively, with arrows pointing to the selected items.

This is not shown, but you can further limit the data output to specific lines of business and/or solution areas and/or scope items, which is handy if you are for example responsible, or only interested in a particular area such as Finance or Supply Chain.

Once you have made your selections, a list of matching items corresponding to it will be output, such as the one below. Note that all items have the type 'Deprecated'. For each item that appears on this list, there will be a succinct description giving you a high level overview of the deprecated item, but you can then also click on the *What's New Document* hyperlink, to access more detailed information on it.

Line of Business: [No Selection] Solution Area: [No Selection] Category: Authorization Type: **Deprecated** Scope Item: [No Selection] Valid as Of: SAP S/4HANA Cloud 2408 Preparation Required: [No Selection]

Latest Reference Content Version Required: [No Selection] Reset to Initial Clear All Showing 1 to 6 of 6 entries.

Line of Business	Solution Area	Solution Capability	Title	Short Description	Category	Technical Object Name	Type	Scope Item	Valid as Of
Application Platform and Infrastructure	ABAP Platform		IAM Changes in Information Lifecycle Management	Changes to identity and access management (IAM) objects have been delivered for Information Lifecycle Management. What's New Document	Authorization	See table <i>IAM Objects</i> in the What's New document	Deprecated	1KA	SAP S/4HANA Cloud 2408
Asset Management	Maintenance Management		IAM Objects in Maintenance Management	Changes to identity and access management (IAM) objects have been delivered for Maintenance Management . What's New Document	Authorization	See table <i>IAM Objects</i> in the What's New document	Deprecated	4HI	SAP S/4HANA Cloud 2408
Cross Applications			IAM: Business Catalog for Maintaining Default User Values	Changes to identity and access management (IAM) objects for default user values have been delivered. To be able to use default user values, you must now assign the central business catalog User Default Parameters - General (SAP_CA_BC_USER_PARAM_PC) to the business roles concerned. What's New Document	Authorization	See table <i>IAM Objects</i> in the What's New document	Deprecated	n/a	SAP S/4HANA Cloud 2408
Finance	Accounting and	Corporate Close (SAP	Deprecation of	Changes to identity and access management (IAM)	Authorization	See table <i>IAM Objects</i>	Deprecated	1SG	SAP S/4HANA Cloud

Whilst the information provided in the details will be important, probably the most important pieces of information will be those referencing the names of the deprecated and successor objects. I.e the object that replaces the one that is deprecated. If this item applies to you, you will then be able to jump into your SAP S/4HANA Cloud system and process the deprecation (I.e replace the deprecated object with its successor).

This is highlighted in the example below where an announcement is made, to say that a business catalog is deprecated, and if you want to continue using the apps from that deprecated catalog, then you should perform a system adaption and use the successor business catalog (in other words, revise the business role(s) that use the deprecated catalog, and update them with the new successor business catalog).

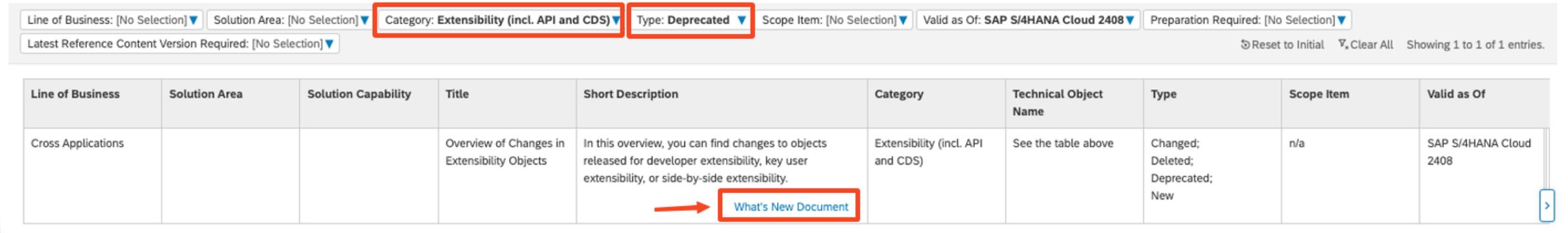
Whilst I have in this case used a business catalog to illustrate my example, the same principle should apply irrespective of the deprecated item category being considered (app, CDS view, etc...).

IAM Objects

IAM Objects				
Hide/Show Columns ▼				Search entire table
Name of IAM Object	Object Type	Changes	Details	Relevant Scope Items
Search column	Search column	Search column	Search column	Search column
Revenue Accounting - Business Rules (SAP_BCR_FIN_FARR_BUSI_RULES_PC)	Business catalog	This business catalog is deprecated and has been replaced.	The business catalog Revenue Accounting - Business Rules (SAP_BCR_FIN_FARR_BUSI_RULES_PC) is deprecated and can no longer be used. The apps that were originally available within this deprecated business catalog are also provided in the business catalog SAP_CA_BC_IC_LND_FIN_FARR0_PC, which is part of business role template SAP_BR_BPC_EXPERT.	3KK (Contract-Based Revenue Recognition - IFRS) 3VS (Contract-Based Revenue Recognition - US GAAP)

Changes in What's New Viewer Category Management

A [change](#) was introduced in 2024 to how some technical categories are maintained in the main What's New Viewer landing page Category drop down selection menu. Where before you had distinct options for the various extensibility objects (CDS Views, API, extensibility...) that you could filter for, this has now been amended and requires you, for ease of access to extensibility information, to navigate to a specific Extensibility page. This new method of analysing extensibility information allows you to be much more precise in narrowing down the information you want to see. To access this new extensibility page, from the main What's new Viewer Landing page, select the Category to be 'Extensibility (incl. API and CDS)' and select the type to be 'Deprecated'. This will restrict the list of What's New items to a single entry, as shown below. To access the What's New information specific to Extensibility objects, simply click the What's New Document link.



Line of Business	Solution Area	Solution Capability	Title	Short Description	Category	Technical Object Name	Type	Scope Item	Valid as Of
Cross Applications			Overview of Changes in Extensibility Objects	In this overview, you can find changes to objects released for developer extensibility, key user extensibility, or side-by-side extensibility. What's New Document	Extensibility (incl. API and CDS)	See the table above	Changed; Deleted; Deprecated; New	n/a	SAP S/4HANA Cloud 2408

You can also bookmark the stable quick access link https://help.sap.com/S4_CE_CHANGES_EXT_OBJ.

I would also encourage you to read the blog by Andreas Doemel titled '[What's New? Overview of Changes In Extensibility Objects in SAP S/4HANA Cloud Public Edition](#)' where he covered these changes.

Overview of Changes in Extensibility Objects

Once you reached the 'Overview of Changes in Extensibility Objects' page, you can maximise the real estate of the page to as to allow the table included in that page to take centre stage. In terms of presentation, this table is identical to the main What's New Viewer landing page table, except that here of course the content, is different and specific to extensibility objects. That said, since we are talking about deprecations here, the first thing you will want to do, is to select the Change Type, as shown below, to be 'Deprecated' only.

Extensibility Type	Object Type	Application Component	Technical Name	Description	Change Type	Change Details	Additional Information
Filter: [No Selection]	Filter: [No Selection]	Filter: [No Selection]	Search column	Search column	Filter: Deprecated <ul style="list-style-type: none"> <input type="checkbox"/> Select All <input type="checkbox"/> Changed <input type="checkbox"/> Deleted <input checked="" type="checkbox"/> Deprecated <input type="checkbox"/> New 	Filter: [No Selection]	Search column
Developer extensibility	Authorization Object	PM (Plant Maintenance)	I_EAM_OM	PM: Output Management	<div style="border: 2px solid red; padding: 2px;"> <input checked="" type="checkbox"/> Deprecated </div>	Use system-internally (C1): Developer extensibility	Use the following successors: <ul style="list-style-type: none"> ▪ I_VORG_ORD ▪ I_VORG_MEL ▪ I_EAM_MJPB
Developer extensibility	BAdI	FI-LOC-LO-IN (Logistics India)	BADI_J_1IG_SUBCON	Badi for SubContracting (India)	Deprecated	Use system-internally (C1): Developer extensibility	Use the following successor: BADI_CE_J_1IG_SUBCON
Developer extensibility	BAdI	LO-RFM-STO-FIO (In-Store Merchandise and Inventory)	BADI_RTST_ARTICLE_IMAGE	Provide Image Data for Articles	Deprecated	Use system-internally (C1): Developer extensibility	Use the following successor: BADI_RTST_DEFINE_ARTICLE_IMAGE More information on SAP Business Accelerator Hub

You can also in the same breath or separately, further fine tune the information you want to see, by using the other available filters available to narrow down your search.

These additional selection options, are:

1. Extensibility Type (Developer, Key user, Side by Side)
2. Object Type (to focus on specific technical objects, such as CDS views, or API's, or BAdI's...)
3. Application Component (if you are expert and know the SAP application components)
4. Change Details (to further distinguish the change, e.g: an addition vs a change)

Extensibility Type	Object Type	Application Component	Technical Name	Description	Change Type	Change Details	Additional Information
Filter: [No Selection]	Filter: [No Selection]	Filter: [No Selection]	Search column	Search column	Filter: Deprecated	Filter: [No Selection]	Search column
<input type="checkbox"/> Select All <input type="checkbox"/> Developer extensibility <input type="checkbox"/> Key user extensibility <input type="checkbox"/> Side-by-side extensibility/Integration <input type="checkbox"/> Side-by-side extensibility/Integration	<input type="checkbox"/> Select All <input type="checkbox"/> Authorization Object <input type="checkbox"/> BAdI <input type="checkbox"/> Business Event <input type="checkbox"/> Business Object Interface <input type="checkbox"/> CDS View <input type="checkbox"/> Class <input type="checkbox"/> Interface <input type="checkbox"/> MDI <input type="checkbox"/> OData V2 API <input type="checkbox"/> OData V4 API <input type="checkbox"/> SOAP API	<input type="checkbox"/> Select All <input type="checkbox"/> CO-FIO-PA (Profitability Management) <input type="checkbox"/> CO-OM (Overhead Cost Controlling) <input type="checkbox"/> CO-OM-CCA (Cost Center Accounting) <input type="checkbox"/> CO-PC-OBJ-EBR (Event-Based Revenue Recognition) <input type="checkbox"/> CO-PC-OBJ-ORD (Product Cost by Order) <input type="checkbox"/> CRM-S4-ANA-CDS (S4CRM: Analytics Data Service) <input type="checkbox"/> CRM-S4-BF-CM (S4CRM: Credit)	I_EAM_OM	PM: Output Management	Deprecated	<input type="checkbox"/> Select All <input type="checkbox"/> Extend (C0): Key user extensibility <input type="checkbox"/> Behavior Definition: Use system-internally (C1): Developer extensibility <input type="checkbox"/> Behavior Definition: Use system-internally (C1): Key user extensibility <input type="checkbox"/> CDS View: Extend (C0): Developer extensibility <input type="checkbox"/> CDS View: Extend (C0): Key user extensibility <input type="checkbox"/> CDS View: Use system-internally (C1): Developer extensibility	The following successors: VORG_ORD VORG_MEL EAM_MJPB The following successor: BADI_CE_J_1IG_SUBCON The following successor: BADI_RTST_DEFINE_ARTICLE_IMAGE Information on SAP Business Accelerator Hub
Developer extensibility	BAdI		SUBCON	Badi for SubContracting (India)	Deprecated		
			ARTICLE_IMAGE	Provide Image Data for Articles	Deprecated		
Key user extensibility	BAdI		AMOUNT_CHANCE	Badi for changing base amount	Deprecated		

Furthermore it should be noted, that in this page, the additional information column which is where the details of the change will be provided, will vary in the content offered.

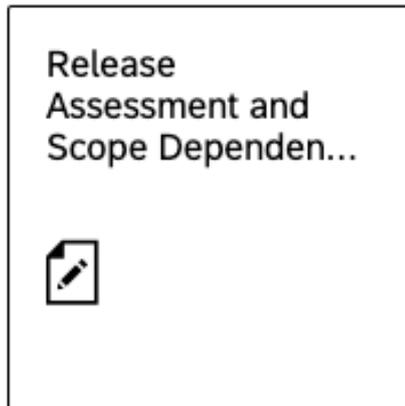
In some cases the information will be condensed to a textual information such as that shown in (1) where you get all the necessary information pertaining to the deprecated object, its successor and in some cases as shown in (2) where you will get a textual information and / or a hyperlink to navigate to for further information.

Change Type	Change Details	Additional Information
Filter: Deprecated ▼	Filter: [No Selection] ▼	Search column
Deprecated	Fields deprecated 1	One or more field is deprecated. Replace with the successor shown in brackets: <ul style="list-style-type: none">▪ Deprecated field: SupplierName (successor: BPSupplierName)
Deprecated	Use system-internally (C1): Key user extensibility 2	Use the following successor: C_MFGORDEREVTBSDWIPVARCQRY More information in the business documentation and on SAP Business Accelerator Hub

In the RASD (Release Assessment and Scope Dependency) App

In the previous section, we introduced the What's New viewer. Being publicly available, the information that is presented there, is going to cast a very wide net and apply to ALL areas of SAP S/4HANA Cloud, irrespective of your usage of SAP S/4HANA Cloud. Whilst this might be just what you need to understand the novelties that are offered with a new release or an update, when it comes to deprecations, you probably want to hone in on those that affect you, because you actually use them and you have to do something about them!

For this, the RASD app becomes very handy because it offers you information that is based on your specific usage of the system! So firstly, how do you access it? Well it is a very easy to remember URL which is <https://www.sap.com/RASD> . In order to log on to it, you will need to have an S-User id, connected to the company to which the SAP S/4HANA Cloud system(s) belong to. At the time of writing this document, RASD only provides information in relation to SAP S/4HANA Cloud, Public Edition.



Once you have access to the launchpad of RASD, you can't go wrong - there is only one tile there!
Click it to open the app.

Once in the app, you will find a certain number of tabs and cards, but in the context of deprecations, we are really only interested in those enclosed by the red rectangle in the picture. What sets them apart, is the information they contain, which is indicated at the top of each card.

For each object type (Apps, Catalogs, etc...) contained within a card you can see how many objects are affected by deprecations. Remember also, that the information displayed here is based on your usage, so the importance of it cannot be stressed enough - this affects YOU, and YOU need to do something.

Of importance also here are the filters at the top of the screen.

1. Just like in the What's New viewer filters the release is important, because deprecations only exist during a major release (not intermediate updates)
2. The tenant type could also be important, depending on where you are in your implementation journey. For example if you are live, you absolutely want to make sure that your production tenant is selected, but you might also wish to select your other tenant types if you are in project mode and implementing functionality that is not yet live (I.e there is no point continuing to work on something that is deprecated).

The screenshot displays the SAP Release Assessment and Scope Dependency tool interface. At the top, there are filters for Customer (50403358), Tenants (Production T...), Regions (India x 7 more), and Release (2408.0). A red rectangle highlights a section titled "Deleted / Deprecated Extensibility Objects" and "Deleted / Deprecated Business Catalogs & Renamed App Tiles". This section contains several cards with the following data:

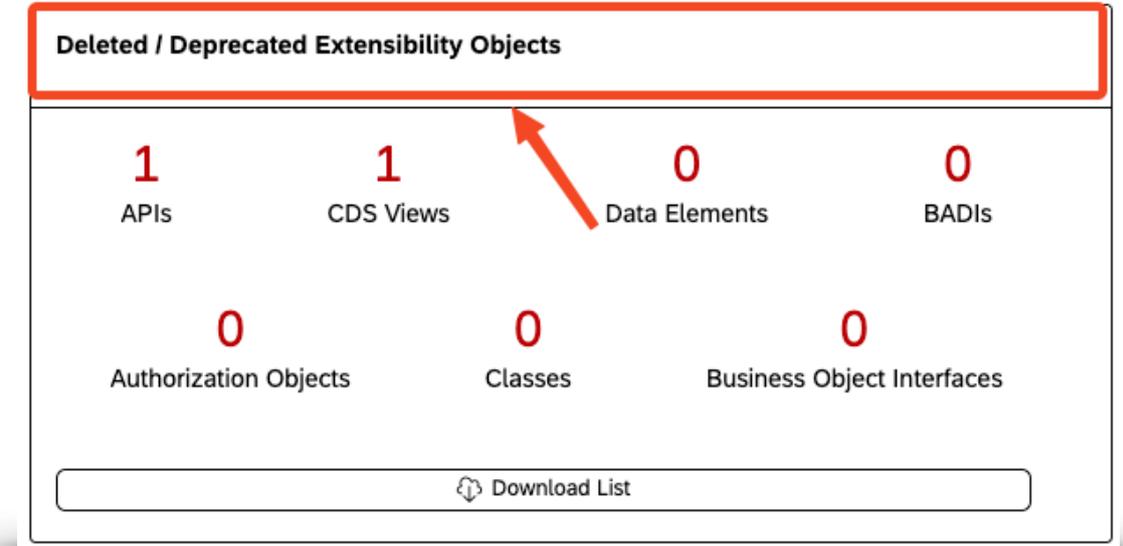
Object Type	Count
APIs	1
CDS Views	0
Data Elements	0
BADIs	0
Authorization Objects	0
Classes	0
Business Object Interfaces	0
Associated Business Catalogs	18
Custom CDS Views	3
Deleted / Deprecated	18
Renamed App Tiles	17

Below the highlighted section, there are two more cards:

- Applications:** 13 Deleted / Deprecated, 14 New, 58 Changed / Must Know.
- Changed Extensibility Objects:** 15 APIs, 3 CDS Views, 0 Data Elements, 0 BADIs, 0 Authorization Objects, 0 Classes, 0 Business Object Interfaces.

To further illustrate the use of RASD in the context of deprecations, we will assume that we are interested in finding out, which CDS views are affected by deprecations, so we will click on the title of the respective card, as shown in the picture to the right, to navigate to a details list output.

Because we only want to view deprecations related to CDS Views, we can in this details list output, use the (1) Types and (2) Change Type filters, to restrict the output to that deprecated object, then click Go to filter the list. You can also click on the (3) link indicated, which will allow you to navigate to the [Overview of Changes in Extensibility Objects](#), which we covered previously.



Search:

Types: Select All (1 of 2)

Change Type: Select All (1 of 1)

Items (2)

- CDS VIEW
- API ODATA V2

Deprecated

↓ SAP Cloud ALM tasks | Export To Excel | ⚙️

What's New Document Title	Type	Name	Description	Change Type	Change Detail
Overview of Changes in Extensibility Objects	API ODATA V2	API_PURCHASEORDER_PROCESS_SRV	Purchase Order	Deprecated	Use as remote API >
3	CDS VIEW	I_BILLINGDOCUM_ENTITEMBASIC	Billing Document Item Basic	Deprecated	Fields deprecated >

For each line that will be output, you can click on that line, to reveal a pane that will provide a large amount of information, to help you understand in depth the deprecation. This pane will provide:

1. The technical name of the SAP object that is affected by this change
2. The extensibility type (helpful, to understand where the deprecation will be processed)
3. The change type (is the entire object, CDS view deprecated or just an element of it)
4. Additional information, similar to what you would see in the What's New Viewer - Overview of Changes in Extensibility Objects, which could be textual information and / or links to further information
5. Where used list. This will not be applicable to all objects, but it is in this case and provides you with information that is specific to your environment.
6. Tenants. This is the system (DEVELOPMENT, TEST or PRODUCTION, depending on whether you are in a 2SL or 3SL landscape), where the deprecation exists
7. Custom CDS Views. This gives you the names of the custom extensibility objects (Custom CDS Views in this case) that you have created and that you need to adapt in order to process the deprecation.

I_BILLINGDOCUMENTITEMBASIC

General Information | Where used list

Line of Business: Cross Applications

Short Description: In this overview, you can find changes to objects released for developer extensibility, key user extensibility, or side-by-side extensibility.

Valid as of: SAP S/4HANA Cloud 2408

Application Component: SD-BIL (Billing)

Extensibility Type: Developer extensibility, Key user extensibility

Technical Name: I_BILLINGDOCUMENTITEMBASIC

Description: Billing Document Item Basic

Change Type: Deprecated

Change Detail: Fields deprecated

Additional Information: One or more fields are deprecated. Replace with the successor shown in brackets:
 Deprecated field: BillingPerformancePeriodStrDte (successor: BillingPeriodOfPerfStartDate)
 Deprecated field: BillingPerformancePeriodEndDte (successor: BillingPeriodOfPerfEndDate)

Where used list

Custom CDS Views

Tenants	Primary Data Source	Custom CDS Views
	I_BILLINGDOCUMENTITEMBASIC	yy1 [blurred]
	I_FINSTATISTICALKEYFIGUREITEM	yy1 [blurred]
	I_PROJECTBILLINGELEMENT	yy1 [blurred]
	I_BILLINGDOCUMENTITEMBASIC	yy1 [blurred]
	I_FINSTATISTICALKEYFIGUREITEM	yy1 [blurred]
	I_PROJECTBILLINGELEMENT	yy1 [blurred]

The CDS Comparator

Recently, an additional analysis tool was introduced to RASD. This tool is named CDS Comparator and aims to further help you in quickly understanding the technical changes related to deprecations in CDS views, by clearly highlighting and colour coding what is new, what is changed or deleted between a deprecated and its successor CDS view. Note that the comparator will only highlight deprecation information for deprecated CDS views provided a successor CDS view exists for it. You can access the CDS comparator in one of two ways:

1. From the list of Deleted / Deprecated Extensibility Objects

In the event where you are using CDS views in that are deprecated, this will be highlighted to you when you access the Deleted / Deprecated Extensibility Objects card (as covered in the previous chapter).

As you can see in the image on the right, a button is available for you to click on, and navigate to the CDS comparator.

Deleted / Deprecated Extensibility Objects

Go Hide Filter Bar Filters

Search: Search Types: Change Type:

Items (2) SAP Cloud ALM tasks Export To Excel

What's New Document Title	Type	Technical Name	Description	Change Type	Change Detail
Overview of Changes in Extensibility Objects	API ODATA V2	API_PURCHASEORDER_PROCESS_SRV	Purchase Order	Deprecated	Use as remote API >
	CDS VIEW	I_BILLINGDOCUMENTITEM	Billing Document Item	Deprecated	Fields deprecated >

2. From the Where used list

By selecting the Custom CDS Views tab, you will find a list of custom CDS view(s) that you have created, and against the associated data sources listed that same familiar icon allowing you to navigate to the CDS Comparator.

2408.0 - Where used list

Go Hide Filter Bar Filters

Search: Search Tenants: * 3 Items Types:

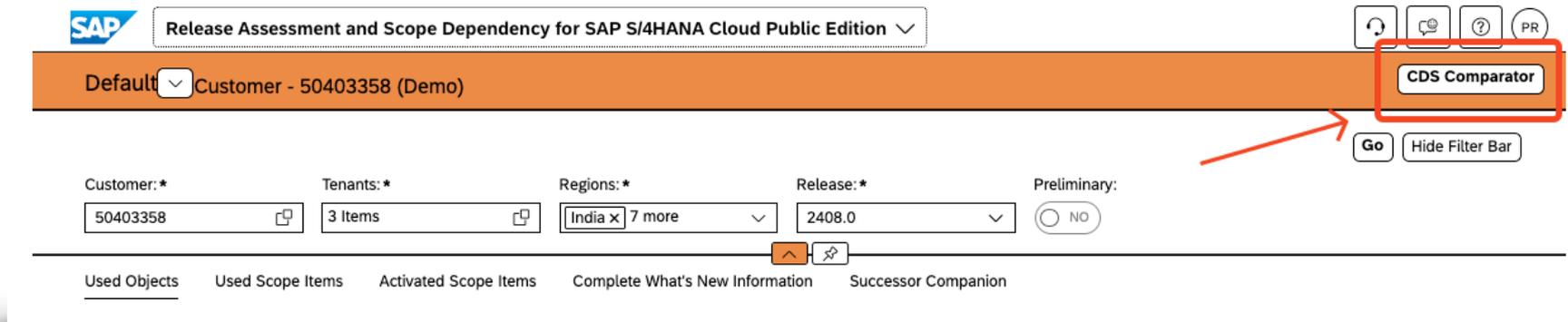
Associated Business Catalogs Custom CDS Views

Export To Excel

Tenants	What's New Title	Primary Data Source	Custom CDS Views
	Overview of Changes in Extensibility Objects	I_ASSETVALUATIONFORLEDGER	yy1_...
		I_PURCHASEORDERAPI0	...
		I_TAXCODE	yy1_...

3. From the Main RASD Landing Page

A new easy access button has been added to the top right of the RASD Landing page, giving you direct access to the tool, as shown in the picture below.



Once you click on the CDS Comparator button, the main CDS Comparator interface will be presented to you, and you can start to use it by entering or selecting the name of the deprecated CDS view, you wish to analyze. Note that the CDS comparator only allows you to select as a starting point, a CDS view that has been deprecated (i.e you cannot search for and analyze a CDS view that is still released).

In our case, we have for the purposes of this analysis selected the CDS view I_EWM_WHSETASKITEMCUBE ([Warehouse Task Item Cube](#)). Let's take a moment to explain this screen.

1. On the left hand side of the screen, you see information pertaining to the deprecated CDS view, that is the basis for the analysis. You can see the name of the CDS view you chose, as well as the elements (fields) that make it up and formatting informations (type of data and fields length).
2. On the right hand side, you can see information pertaining to the successor CDS view. The successor CDS view was automatically determined by the CDS Comparator. Again here, we can see the elements of the successor and their associated formatting information.
3. If changes do exist between the deprecated and successor CDS views, they will be accompanied by a color coding, to make it simple to understand. In the screenshot below, you can see some examples of this color coding, giving you a visual cue that something has changed/is different. In the next section, we will dive a little deeper into what these change type can be.
4. Depending on the number of fields and/or changes that apply, you may wish to filter the list so as to focus only on what has changed, versus what has not changed, versus reverting to the default which is to show everything. This can be achieved by selecting the option that corresponds to your choice.

CDS Comparator

Deprecated
 Changed
 New

Deprecated CDS View:

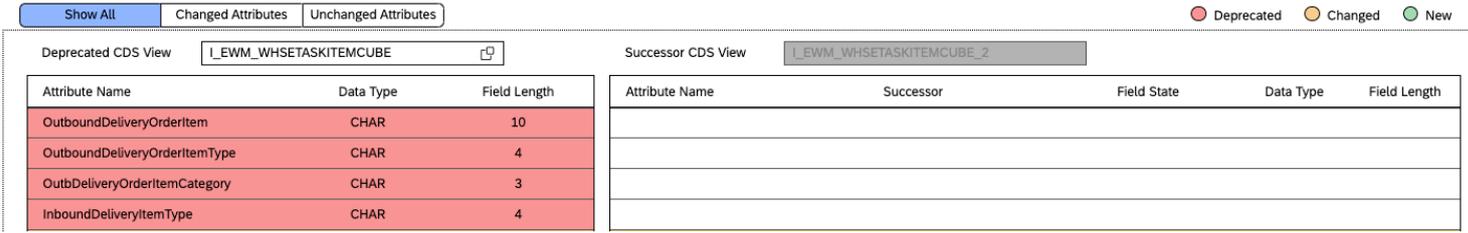
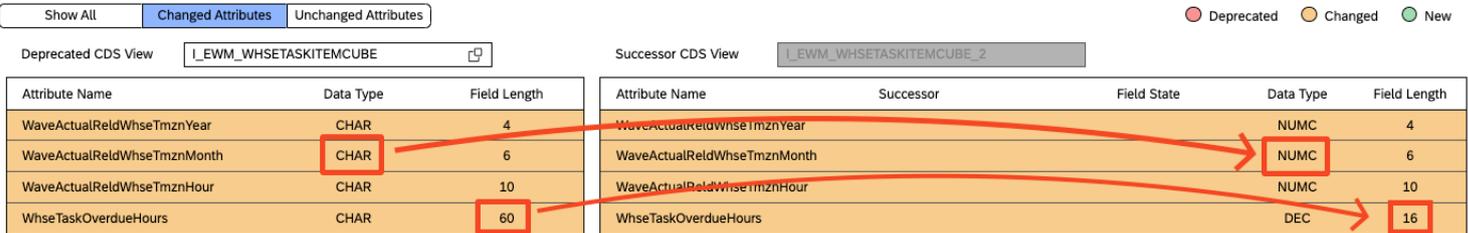
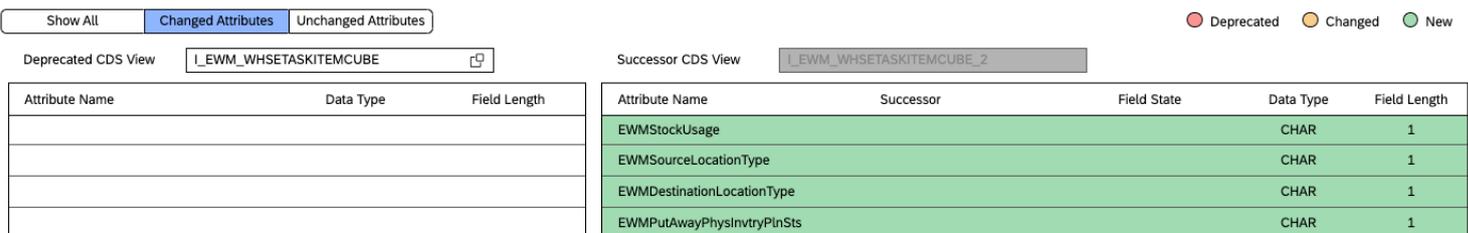
Attribute Name	Data Type	Field Length
DestinationStorageBinStack	CHAR	18
DestinationStorageBinLevel	CHAR	18
DestinationStorageBinAccType	CHAR	4
OutboundDeliveryOrder	CHAR	35
OutboundDeliveryOrderItem	CHAR	10
OutboundDeliveryOrderItemType	CHAR	4
OutbDeliveryOrderItemCategory	CHAR	3
InboundDeliveryItemType	CHAR	4
WaveActualReldWhseTmznYear	CHAR	4
WaveActualReldWhseTmznMonth	CHAR	6
WaveActualReldWhseTmznHour	CHAR	10
WaveTemplateName	CHAR	40

Successor CDS View:

Attribute Name	Successor	Field State	Data Type	Field Length
DestinationStorageBinStack			CHAR	18
DestinationStorageBinLevel			CHAR	18
DestinationStorageBinAccType			CHAR	4
WaveActualReldWhseTmznYear			NUMC	4
WaveActualReldWhseTmznMonth			NUMC	6
WaveActualReldWhseTmznHour			NUMC	10
WaveTemplateName			CHAR	40

Color coded changes

To help you focus on the changes that exist between a deprecated CDS view and its successor, a color coding system was added to help you in this task. The meaning of each color is as below.

Color	Meaning	Example																																								
RED	This color is used on the left, in the deprecated CDS view, and is used to signify that this field does not exist in the Successor CDS. The corresponding line in the successor CDS view will be 'blank', devoid of content or color (there is nothing to see).	 <p>Legend: ● Deprecated ● Changed ● New</p> <table border="1"> <thead> <tr> <th>Attribute Name</th> <th>Data Type</th> <th>Field Length</th> </tr> </thead> <tbody> <tr> <td>OutboundDeliveryOrderItem</td> <td>CHAR</td> <td>10</td> </tr> <tr> <td>OutboundDeliveryOrderItemType</td> <td>CHAR</td> <td>4</td> </tr> <tr> <td>OutbDeliveryOrderItemCategory</td> <td>CHAR</td> <td>3</td> </tr> <tr> <td>InboundDeliveryItemType</td> <td>CHAR</td> <td>4</td> </tr> </tbody> </table>	Attribute Name	Data Type	Field Length	OutboundDeliveryOrderItem	CHAR	10	OutboundDeliveryOrderItemType	CHAR	4	OutbDeliveryOrderItemCategory	CHAR	3	InboundDeliveryItemType	CHAR	4																									
Attribute Name	Data Type	Field Length																																								
OutboundDeliveryOrderItem	CHAR	10																																								
OutboundDeliveryOrderItemType	CHAR	4																																								
OutbDeliveryOrderItemCategory	CHAR	3																																								
InboundDeliveryItemType	CHAR	4																																								
ORANGE	This color is used in conjunction in both the deprecated and successor CDS views. It is used to signify that a given field exists in both the deprecated and successor CDS views but the formatting (data type or field length has changed). E.g, a field that was previously of type CHAR is now of type NUMC and/or the field length has changed.	 <p>Legend: ● Deprecated ● Changed ● New</p> <table border="1"> <thead> <tr> <th>Attribute Name</th> <th>Data Type</th> <th>Field Length</th> </tr> </thead> <tbody> <tr> <td>WaveActualReldWhseTmznYear</td> <td>CHAR</td> <td>4</td> </tr> <tr> <td>WaveActualReldWhseTmznMonth</td> <td>CHAR</td> <td>6</td> </tr> <tr> <td>WaveActualReldWhseTmznHour</td> <td>CHAR</td> <td>10</td> </tr> <tr> <td>WhseTaskOverdueHours</td> <td>CHAR</td> <td>60</td> </tr> </tbody> </table> <table border="1"> <thead> <tr> <th>Attribute Name</th> <th>Successor</th> <th>Field State</th> <th>Data Type</th> <th>Field Length</th> </tr> </thead> <tbody> <tr> <td>WaveActualReldWhseTmznYear</td> <td></td> <td></td> <td>NUMC</td> <td>4</td> </tr> <tr> <td>WaveActualReldWhseTmznMonth</td> <td></td> <td></td> <td>NUMC</td> <td>6</td> </tr> <tr> <td>WaveActualReldWhseTmznHour</td> <td></td> <td></td> <td>NUMC</td> <td>10</td> </tr> <tr> <td>WhseTaskOverdueHours</td> <td></td> <td></td> <td>DEC</td> <td>16</td> </tr> </tbody> </table>	Attribute Name	Data Type	Field Length	WaveActualReldWhseTmznYear	CHAR	4	WaveActualReldWhseTmznMonth	CHAR	6	WaveActualReldWhseTmznHour	CHAR	10	WhseTaskOverdueHours	CHAR	60	Attribute Name	Successor	Field State	Data Type	Field Length	WaveActualReldWhseTmznYear			NUMC	4	WaveActualReldWhseTmznMonth			NUMC	6	WaveActualReldWhseTmznHour			NUMC	10	WhseTaskOverdueHours			DEC	16
Attribute Name	Data Type	Field Length																																								
WaveActualReldWhseTmznYear	CHAR	4																																								
WaveActualReldWhseTmznMonth	CHAR	6																																								
WaveActualReldWhseTmznHour	CHAR	10																																								
WhseTaskOverdueHours	CHAR	60																																								
Attribute Name	Successor	Field State	Data Type	Field Length																																						
WaveActualReldWhseTmznYear			NUMC	4																																						
WaveActualReldWhseTmznMonth			NUMC	6																																						
WaveActualReldWhseTmznHour			NUMC	10																																						
WhseTaskOverdueHours			DEC	16																																						
GREEN	This color is used on the right, in the successor CDS view, and is used to signify that this field is -new- in the Successor CDS and does not exist in the deprecated CDS view. The corresponding line in the deprecated CDS view will be 'blank', devoid of content or color (there is nothing to see).	 <p>Legend: ● Deprecated ● Changed ● New</p> <table border="1"> <thead> <tr> <th>Attribute Name</th> <th>Data Type</th> <th>Field Length</th> </tr> </thead> <tbody> <tr> <td>EWMStockUsage</td> <td>CHAR</td> <td>1</td> </tr> <tr> <td>EWMSourceLocationType</td> <td>CHAR</td> <td>1</td> </tr> <tr> <td>EWMDestinationLocationType</td> <td>CHAR</td> <td>1</td> </tr> <tr> <td>EWMPutAwayPhysInvtryPlnSts</td> <td>CHAR</td> <td>1</td> </tr> </tbody> </table>	Attribute Name	Data Type	Field Length	EWMStockUsage	CHAR	1	EWMSourceLocationType	CHAR	1	EWMDestinationLocationType	CHAR	1	EWMPutAwayPhysInvtryPlnSts	CHAR	1																									
Attribute Name	Data Type	Field Length																																								
EWMStockUsage	CHAR	1																																								
EWMSourceLocationType	CHAR	1																																								
EWMDestinationLocationType	CHAR	1																																								
EWMPutAwayPhysInvtryPlnSts	CHAR	1																																								

I would also encourage you to read the blog post by Prasanth Padmanabhan titled '[Release Assessment and Scope Dependency for SAP S/4HANA Cloud Public Edition: What is new in 2024](#)' where he covered the introduction of this feature.

Now that we have covered some of the data sources that are available to you outside your SAP S/4HANA Cloud system, in the next section we will look at the information that is available inside your system.

In your system there will be some information that is generic (irrespective of whether you use it, for example warnings or callouts that an object is deprecated, so that you do not start to use it!), and some that will be specific and relevant, *because* you do make use of it. In this next section, we will look at applications that are available in your SAP S/4HANA Cloud system to either help you investigate deprecation related changes or to clearly advise you that deprecation related rework tasks exist in your system and provide you, via these application, the functionality to process them. This information is not meant to replace the comprehensive information that is already available elsewhere, but merely to make you aware of them, and point you in the direction to find out more on the subject.

Inside SAP S/4HANA Cloud

The View Browser application

The [View Browser app](#) is an app that will show you all the CDSs (Core Data Services) that exist in your system (which does not mean that they are all available for use - some have a status not released), irrespective of whether you use them or not, or indeed irrespective of whether you can actually use them or not. However, this application is extremely handy for you to investigate and identify technically the data sources (SAP Standard as well as custom) that will be available to you when you need a data source for your next extensibility needs. Of course what you do not want, is to base your investigations or worse, build content using a deprecated CDS. This may not be evident, but the view browser app, allows you to filter those CDSs that are deprecated.

In order to access the View Browser app, you will need to have the catalog `SAP_CA_BC_ANA_AQD_PC` (Analytics - Query Design) assigned to a business role, assigned to your business user.

Depending on the extensibility interest that you have, whether Key User and/or Cloud Development, you will be wanting to exclude the data sources that are deprecated from your investigation, because again, the use of deprecated objects should be discouraged. The release state (Rel. State) of a view will be visible in the columns 'Key User Release State' and 'Cloud Development Release State' as shown in the diagram.

In order to filter those objects that are deprecated, then:

1. Click on the Column Title that you are interest to filter
2. Select the Filter Option
3. Call up the Define condition window
4. Choose 'starts with' and enter 'Deprec' (you could of course also exclude them by choosing the option 'Does not start with')

The screenshots illustrate the process of filtering SAP CDS views. The first screenshot shows the 'View Browser' with a table of views. The second screenshot shows the 'Key User Rel. State' column menu with the 'Filter' option selected. The third screenshot shows the 'Filter' dialog box for 'Key User Rel. State'. The fourth screenshot shows the 'Define Conditions' dialog box where 'starts with' is selected and 'Deprec' is entered. The final screenshot shows the filtered results with 191 views, including 'C_BillgDoclrmPrcgElmntBscDEX', 'C_BillingDocumentItemBasi cDEX', and 'C_BusEvtLogEventDEX', all with 'Key User Rel. State' set to 'Deprecated'.

Views (71,011)	Standard	Name	Description	Key User Rel. State	Cloud Dev. Rel. State	Application Component
		/1BS/CDS_EXP_SALESORDER	SalesOrder	Not Released	Not Released	BC-ESI-ESF-BSA
		/1bs/sadl_cds_exp	Test View for CDS Exposure	Not Released	Not Released	BC-ESI-ESF-BSA
		/1BS/SADL_CDS_Param	Test View for CDS Exposure	Not Released	Not Released	BC-ESI-ESF-BSA

Views (191)	Standard	Name	Description	Key User Rel. State	Cloud Dev. Rel. State	Application Component	Data Category	Application ID
		C_BillgDoclrmPrcgElmntBscDEX	Data Extraction for Billing Doc Item Pricing Element Basic	Deprecated	Deprecated	SD-ANA	Fact	>
		C_BillingDocumentItemBasi cDEX	Billing Document Item Basic Extraction Data	Deprecated	Deprecated	SD-ANA	Fact	>
		C_BusEvtLogEventDEX	Business Event Log Data	Deprecated	Not Released	CA-GTF-BEL	Fact	>

Then click on OK, and this will reveal only Deprecated CDS.

Similarly, you could use this method to filter those CDSs that are Released only.

This might be a useful method for you to identify if a CDS entity is completely deprecated. But there are of course cases where it is only element(s) of the CDS entity that is deprecated. In order to access that level of detail, you need to navigate to the detail screen of the CDS in question. For example, let's consider the CDS below `I_SalesOrderItemCube`. For all intents and purposes, we can see that it is released for extensibility scenarios.

Views (1)	Standard*	salesorderitemcube						
1 table filter active: Key User Rel. State								
Favorites	Name	Description	Key User Rel. State	Cloud Dev. Rel. State	Application Component	Data Category	Application ID	
<input type="checkbox"/>	I_SalesOrderItemCube	Sales Order Item - Cube	Released	Released	SD-ANA	Cube		

However, as we delve into the detail view of this CDS, we can see that some fields are flagged as deprecated (Status). But here also you are getting the information pertaining to the successor field. The use of the successor field must be privileged for new developments.

I_SalesOrderItemCube
Sales Order Item - Cube

Application Component: SD-ANA Tags: Supported Capabilities: #ANALYTICAL_PROVIDER [Show All](#)
Description: Sales Analytics [Product Assistance](#) Modeling Pattern: #ANALYTICAL_CUBE

Definition Annotation Cross Reference

Column Name	Data Element	Description	Data Type	Length	Status	Successor
_Material				0	Deprecated	_Product
_SalesOrganization				0		

Coincidentally, whilst in the detail view of a CDS, you will find links or access to more, additional noteworthy information. You can access the SAP help of the given CDS with lots of additional details, you can also view all the supported capabilities of the CDS. This is quite helpful, when processing a CDS entity deprecation, to check that the successor can still be used for the same purpose (e.g: Data Extraction) as the deprecated CDS entity. If you are unsure of the meaning of the codes (the technical names) shown below for the supported capabilities of a CDS, or indeed what those supported capabilities can be, then please check this information here [Supported Capabilities for CDS Views](#).

I_SalesOrderItemCube
Sales Order Item - Cube

Application Component: SD-ANA
Description: Sales Analytics

Tags: [Product Assistance](#)

Supported Capabilities: #ANALYTICAL_PROVIDER [Show All](#)
Modeling Pattern: #ANALYTICAL_CUBE

Supported Capabilities

#ANALYTICAL_PROVIDER, #SQL_DATA_SOURCE, #CDS_MODELING_DATA_SOURCE

[Close](#)

SAP Help Portal (Documentation) Browse by Product SAP Learning Journeys

Home > SAP S/4HANA Cloud > Virtual Data Model and CDS Views > CDS Views > ... > Sales Order > Sales Order Item - Cube

Virtual Data Model and CDS Views 2308.3 English Production This document

Sales Order Item - Cube

Technical Name	I_SalesOrderItemCube
Business Role Template	Sales Manager
Business Catalog	Sales - Sales Analytics
Data Category	Cube
Release Status	Released

Purpose

This CDS view provides the prerequisites for answering the following business questions:

- What are the net values of sales order items?
- What are my top 10 products based on my incoming sales orders?
- How many order items are in the system and how many are still open?
- What are the open values of these items?

Note
Make sure that you use the cubes for analytical purposes only.

Prerequisites

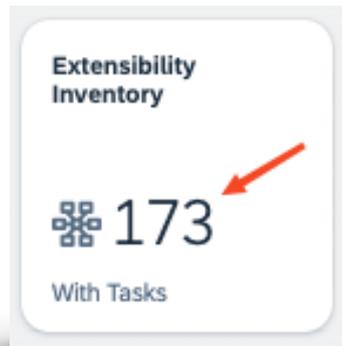
Users who want to run reports using this CDS view must have the read access to:

- Sales order type
- Sales organization

The Extensibility Inventory application

The [Extensibility Inventory app](#) is an app that will, as the name indicates, display for you an exhaustive list of the extensibility objects that you have developed in your system (irrespective of whether a deprecation affects them or not), or included in a software collection. This app has an informational use, in that it will provide for you, in one place, a bird's eye view of all your extensibility objects in great detail, but it similarly also has an instructional use in that it advertises the priority (aka criticality) and type of tasks that you need to perform in order to resolve or attend to a particular situation in your system. Already from the Fiori Launchpad, you will on the tile of the app, have the number of outstanding tasks clearly shown - but not all will necessarily be deprecations related rework tasks .

Once you click on the tile, and unless you have set a specific default display variant, a list screen similar to this one will be presented to you .



< **SAP** Extensibility Inventory ▾

Standard ▾ Open Extensibility Inventory History

Items (2,344) ↑↓ ⚙️ 📄

Name	Type	Uses	Used By	Uses SAP Obj...	Task Priority	Rework Priority Sor...	Task Category
ESH_S_ASSGMTSRCH - Generated Metadata Extension	Enterprise Search - Metadata Extensions					0	
ESH_S_PURCHASE_INFO_REC - Generated Metadata	Enterprise Search - Metadata Extensions					0	
ESH_S_PURCHASE_ORDER - Generated Metadata Extension	Enterprise Search - Metadata Extensions					0	
I_TimeSheetRecord YY1 I_TIMESHEETRECORD	Custom CDS View		1	2	Medium	2	Performance

The first thing that we want to do, is to reduce the list presented to just those items that interest us, namely the items that are the object of a deprecation related rework task. To do so:

1. Adapt the filters if necessary to show the Task Category filter,
2. Adapt the values for the Task Category filter,
3. Select the Deprecation category, and select the Multiple Categories as well, in case it includes deprecations.

The screenshot shows the SAP Extensibility Inventory interface. At the top, there is a navigation bar with the SAP logo, 'Extensibility Inventory', and a search icon. Below this, there is a 'Standard' filter dropdown and a link to 'Open Extensibility Inventory History'. The main filter area includes fields for 'Item Description', 'Last Changed On' (with a date range 'e.g. 22.12.2023-31.12.2...'), 'Software Collection', and 'Task Priority'. A 'Task Category' dropdown menu is open, showing a list of categories: 'Deprecation (DEPRECATE)', 'Error (ERROR)', 'Functionality (FUNCTION)', 'Multiple Categories (MULTIPLE)', 'Performance (PERFORMANC)', 'Security (SECURITY)', and 'Stability (STABILITY)'. The 'Multiple Categories (MULTIPLE)' option is selected. A 'Go' button and 'Adapt Filters (2)' text are visible next to the dropdown. Below the filters, there is a table titled 'Items (2,344)' with columns for Name, Type, Uses, Used By, Uses SAP Obj..., and Task Priority. The table contains two rows of data: 'ESH_S_ASSGMTSRCH - Generated Metadata Extension' and 'ESH_S_PURCHASE_INFO_REC - Generated Metadata'. Red callout boxes with numbers 1, 2, and 3 are placed over the 'Task Category' dropdown, the 'Task Priority' field, and the 'Task Category' dropdown menu respectively.

This will then yield a list focussed on deprecations (1) . Already by looking at the 'Type' column, we see we have a variety of objects (2) that are affected by deprecations. You can further tweak the list to your liking by applying sorts or filters, and you can also export this list in spreadsheet format. This could be very useful for example, if you are going to incorporate some of this information in an upcoming project plan or do some offline planning.

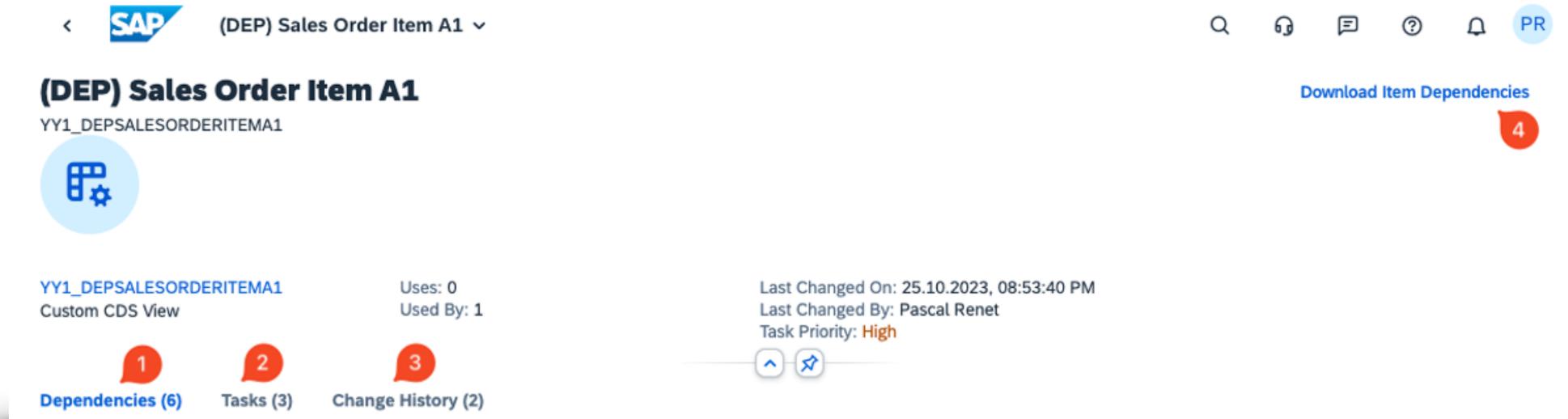
The screenshot shows the SAP Extensibility Inventory interface. At the top, there's a navigation bar with the SAP logo and 'Extensibility Inventory' dropdown. Below that, a 'Standard' dropdown is visible. The main area contains filter fields for 'Item Description', 'Last Changed On' (with a date range 'e.g. 22.12.2023-31.12.2...'), 'Software Collection', 'Task Priority', and 'Task Category' (set to '2 Items'). A 'Go' button and 'Adapt Filters (2)' are also present. Below the filters, there's a table titled 'Items (41)' with a search bar and a list of items. The table has columns: Name, Type, Uses, Used By, Uses SAP Obje..., Task Priority, Rework Priority Sor..., and Task Category. The 'Task Category' column is highlighted with a red box and a '1' in a red circle. The 'Type' column is also highlighted with a red box and a '2' in a red circle. A green arrow points from the 'Adapt Filters (2)' button to a spreadsheet export icon in the top right corner of the table area.

Name	Type	Uses	Used By	Uses SAP Obje...	Task Priority	Rework Priority Sor...	Task Category
Billing_test YY1 BILLING TEST	Custom CDS View			3	Medium	2	Deprecation
CustomerProject YY1_CUSTOMERPROJECT	Custom CDS View		1	2	Medium	2	Multiple Categories
CustomWBSField YY1_CUSTQMWSFIELD	Custom Field				High	3	Deprecation
LMANTICIPO YY1 LMANTICIPO	Custom Logic	1		29	Medium	2	Deprecation

A small thumbnail image showing a spreadsheet export of the data from the SAP Extensibility Inventory. The spreadsheet contains multiple columns and rows of data, including item names, types, and task categories, matching the data shown in the main screenshot.

Once you have performed your sorts and filters, you can then click on each line to reach further details, spread across two or three sections depending on the object considered, named as shown below:

1. Dependencies
2. Tasks
3. Change History
4. Coincidentally, as we will see in a moment, you can also download this information in a spreadsheet ready format



The screenshot shows the SAP Fiori interface for the object '(DEP) Sales Order Item A1'. The breadcrumb navigation at the top left includes the SAP logo and '(DEP) Sales Order Item A1'. The main title is '(DEP) Sales Order Item A1' with the technical name 'YY1_DEPSALESORDERITEMA1' below it. A blue circular icon with a grid and gear is positioned to the left of the title. On the right side, there is a search bar, a help icon, a chat icon, a question mark icon, a notification bell icon, and a user profile icon labeled 'PR'. Below the title, there is a blue link 'Download Item Dependencies' with a red notification bubble containing the number '4'. The interface is divided into three main sections: 'YY1_DEPSALESORDERITEMA1 Custom CDS View' with 'Uses: 0' and 'Used By: 1'; 'Dependencies (6)' with a red notification bubble containing '1'; 'Tasks (3)' with a red notification bubble containing '2'; and 'Change History (2)' with a red notification bubble containing '3'. On the right side of these sections, there is a 'Last Changed On: 25.10.2023, 08:53:40 PM', 'Last Changed By: Pascal Renet', and 'Task Priority: High'. At the bottom right, there are two small icons: a blue arrow pointing up and a blue star.

Let's look at this custom CDS that has been flagged as the object of a deprecation in more detail.

The Dependencies section

This is a rich graphical representation of the object being considered (here, our deprecation relevant custom CDS), with a clear representation of its lineage, i.e where it is used. In the example below we see that the custom CDS, is used in an Analytical Query, used in an app for Analytical Queries, used in a business catalog. This is quite powerful in visual representation to understand the impact of a change and where you will need to intervene to make sure that the switch from a deprecated to successor object still produces the desired end result.

In concert with the Dependencies section, you can also, as seen earlier, download these dependencies in a spreadsheet format. Selecting the Download option will download a zip file containing 2 csv files. A 'Used_by...' and a 'Depending_on...' file. The 'Used_by...' file will give you a tabular representation of the graphic shown in the dependencies section. Having the information in this format would be very helpful as you are planning your deprecations, maybe to analyse and group changes logically, maybe to assign responsible people to tasks, etc...

The hierarchy levels columns, provide you with that 'used-by' relationship information.

Item ID	Item Type	Item Description	Rework Priority	Rework Category	Hierarchy Lev	API State	Used By Item ID	Used By Item Type	Used By Item Description
YY1_DEPSALESORDERITEMA1	Custom CDS View	(DEP) Sales Order Item A1	High	Deprecation	1		YY1_DEPSALESORDERITEMCQ1	Custom Analytical Query	YY1_DEPSALESORDERITEMCQ1
YY1_DEPSALESORDERITEMCQ1	Custom Analytical Query	YY1_DEPSALESORDERITEMCQ1			2		YY1_1698230020161_AQWG	App for Analytical Querie	Sales Orders
YY1_DEPSALESORDERITEMCQ1	Custom Analytical Query	YY1_DEPSALESORDERITEMCQ1			2		YY1_K.1698230271114	Smart Business Content	Sales Analysis
YY1_K.1698230271114	Smart Business Content	Sales Analysis			3		YY1_F42UG4C4I4PN5HHEVJ46DZN PCU	Custom Catalog Extension	App YY1_1698231044889_SSB is used in business catalog SAP_SD_BC_ANLY_OPERATIONAL_PC.
YY1_K.1698230271114	Smart Business Content	Sales Analysis			3		YY1_F42UG4C4I4PN5HHFMKLCGC NPCU	Custom Catalog Extension	App YY1_1698230995253_SSB is used in business catalog SAP_SD_BC_ANLY_OPERATIONAL_PC.
YY1_1698230020161_AQWG	App for Analytical Queries	Sales Orders			3		YY1_F42UG4C4I4PN5HHGBNXL6JP PCU	Custom Catalog Extension	App YY1_1698230020161_AQWG is used in business catalog SAP_SD_BC_ANLY_OPERATIONAL_PC.

The Tasks section

This section will provide you with more technical details, that will uniquely and precisely identify the reason of the deprecation (i.e what is the object that is deprecated) and the successor object (the object that you must switch to).

As we can see in the example below, our custom CDS is impacted by three depreciations. To be more specific we are given on each line, the name of the deprecated element and that of the SAP standard CDS from which we are sourcing it. More importantly, we are also provide with the name of the successor element.

The first line, in plain English is saying: The field `Material`, from the Standard SAP CDS named `I_SalesOrderItemCube`, that your custom CDS is using has been deprecated, and thus you should no longer use this field, and instead use the successor field named `PRODUCT`.

Dependencies (6) **Tasks (3)** Change History (2)

Standard ▾ 

Task Priority	Task Category	Description	Registered On	
 High	Deprecation	Element <code>Material</code> of CDS view <code>I_SalesOrderItemCube</code> is deprecated. Use element <code>PRODUCT</code> instead.	25.10.2023, 12:49:12 PM	>
 High	Deprecation	Element <code>MaterialGroup</code> of CDS view <code>I_SalesOrderItemCube</code> is deprecated. Use element <code>PRODUCTGROUP</code> instead.	25.10.2023, 12:49:12 PM	>
 High	Deprecation	Element <code>NetAmount</code> of CDS view <code>I_SalesOrderItemCube</code> is deprecated. Use element <code>NETAMOUNT_2</code> instead.	25.10.2023, 12:49:12 PM	>

Let's now look at a Custom field that has been flagged as the object of a deprecation.

Name	Type	Uses	Used...	Uses SAP Ob...	Task Priority	Task Category
CustomerProject YY1_CUSTOMERPROJECT	Custom CDS View		1	2	Medium	Multiple Categories
CustomWBSField YY1_CUSTOMWBSFIELD	Custom Field				High	Deprecation
DB_SLS_ORD_DTI						Multiple

From the list of deprecated objects, we again click on the selected line to navigate to the object details.

Once we reach the detail screen, the first thing we can notice is that for this object there is no Dependencies section. However we are once again provided (1) with the useful information advising us of a deprecation, the object deprecated and the successor object.

At this point, you might be asking yourself, why am I getting a notification about a custom field deprecation in a SAP Standard CDS? Well actually, the notification is telling you that the object of the deprecation is the SAP Standard CDS, and this deprecation affects your custom field! So to try and get more details and better understand the relationship between our custom field and this SAP Standard CDS let's go to the [Custom Fields](#) app. Before we do this, let's make a note of the custom field name (2). If you wish to pursue this use case, then please go to the chapter [The Custom Field App](#).

CustomWBSField

YY1_CUSTOMWBSFIELD

YY1_CUSTOMWBSFIELD Custom Field

Uses: 0
Used By: 0

Last Changed On: 27.08.2022, 02:50:24
Last Changed By: [User]
Task Priority: High

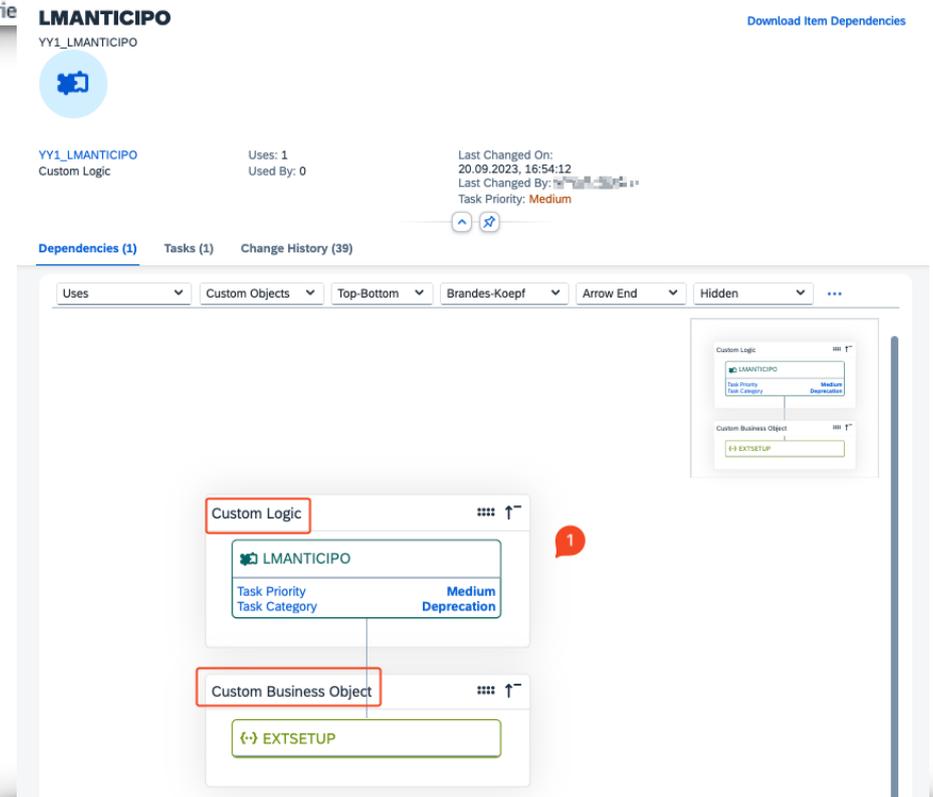
Tasks (1) Change History (3)

Task Priority	Task Category	Description	Registered On
High	Deprecation	Field usage C_WORKPACKAGEDETAILS is deprecated. Please use its successors C_WORKPACKAGE.	13.02.2023, 17:32:28

Let's now look at a Custom Logic that has been flagged as the object of a deprecation.

Items (42)						
Name	Type	Uses	Used By	Uses SAP Obj...	Task Priority	Task Category
LMANTICIPO YY1_LMANTICIPO	Custom Logic	1		29	Medium	Deprecation
LMANTICIPOQRY YY1_LMANTICIPOQRY	Custom CDS View	1		6	Medium	Multiple Categories

From the list of deprecated objects, we again click on the selected line to navigate to the object details.



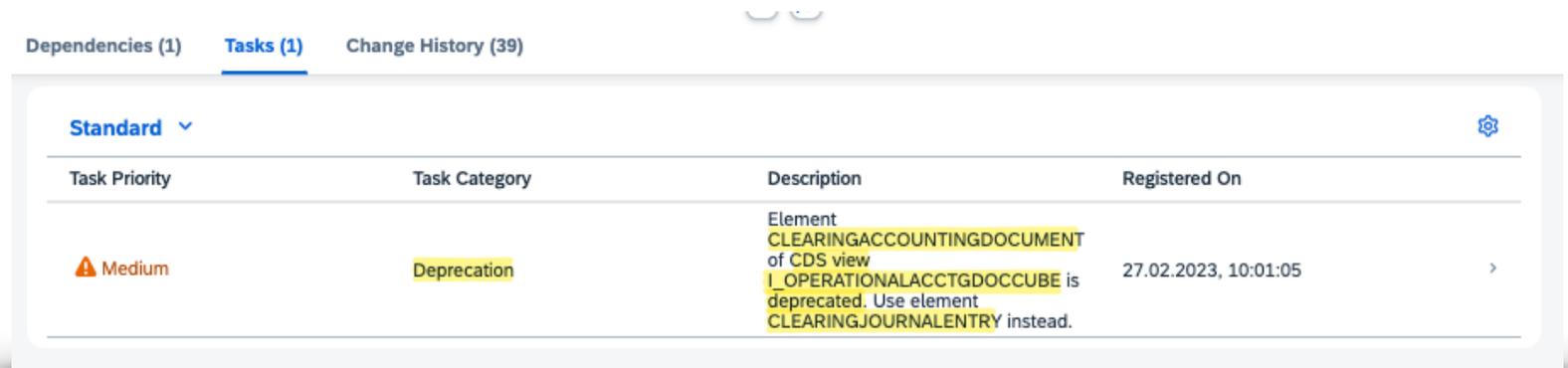
Once we reach the detail screen, the first thing we can notice is that for this object we again find that familiar graphical Dependencies section, which in this case establishes a link between the Custom Logic and a Custom Business Object.

We can then also move on to the Tasks tab to see what technical details we have there, because again at this point you might be asking yourself why are my Custom Objects being flagged for deprecation ?

In the Tasks tab, we are getting closer to our answer, as we are provided with information pertaining to a deprecated field, that is part of a CDS we seemingly use and of course the name of the successor field is also provided here.

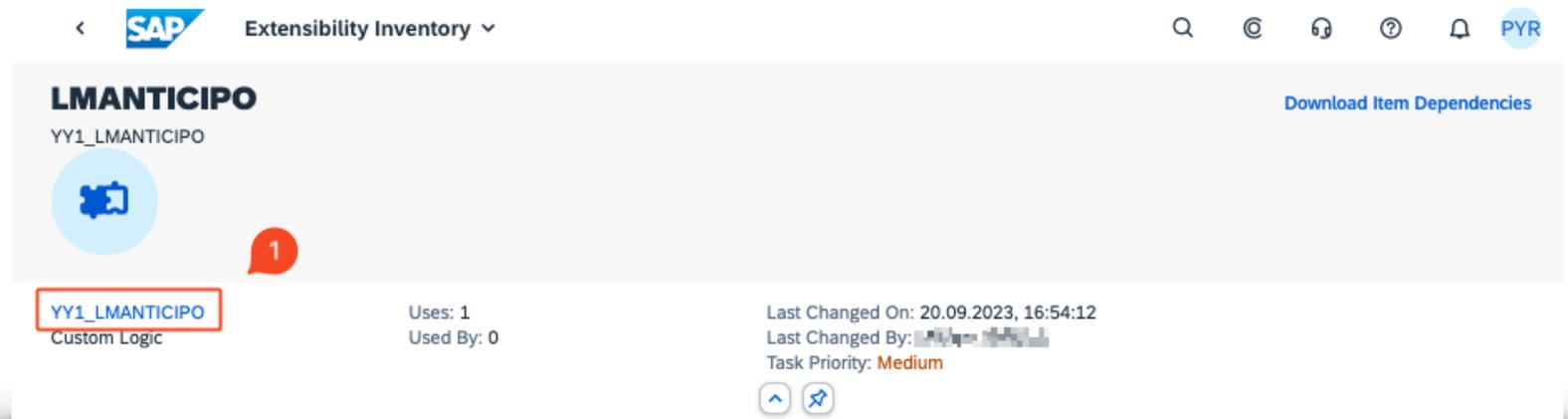
To proceed with the next step of our investigation, we need to dig into this custom logic, so that we can see exactly where and how in our custom logic we are using this deprecated field.

In order to reach the custom logic, you can directly from the header of the extensibility inventory item header, click the link (1) shown, or you can access it using the [Custom Logic app](#). Before we do this, let's make a note of the custom logic name (1) . If you wish to pursue this use case, then please go to the chapter [The Custom Logic App](#).



The screenshot shows the 'Tasks' tab in the SAP Extensibility Inventory. It displays a table with the following data:

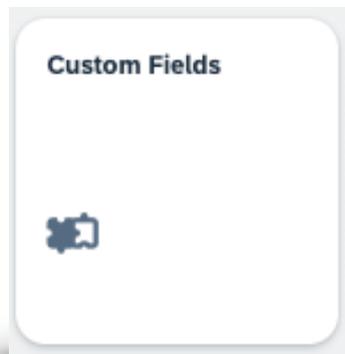
Task Priority	Task Category	Description	Registered On
Medium	Deprecation	Element CLEARINGACCOUNTINGDOCUMENT of CDS view I_OPERATIONALACCTGDOCCUBE is deprecated. Use element CLEARINGJOURNAENTRY instead.	27.02.2023, 10:01:05



The screenshot shows the header for the 'LMANTICIPO' extensibility inventory item. The item name 'YY1_LMANTICIPO' is highlighted with a red box and labeled as 'Custom Logic'. A red notification bubble with the number '1' is next to it. Other details include 'Uses: 1', 'Used By: 0', and 'Last Changed On: 20.09.2023, 16:54:12'. The task priority is 'Medium'.

The Custom Fields application

To be able to access this app, you will need to have the Business Catalog `SAP_CORE_BC_EXT_FLD` assigned to a Business Role, assigned to your business user. The initial screen of this app also shows a Task Category column advising you of possible issues with a given custom fields. Deprecation relevant items will be flagged in this column. If you do not see it, then click the personalisation (1) icon and bring the category column to your table. Alternatively, due to the limited filtering and sorting option afforded in this screen, you may wish to directly use the search box to find the custom field that you may be concerned with.



The screenshot shows the SAP Custom Fields and Logic application interface. At the top, there is a navigation bar with the SAP logo, the title "Custom Fields and Logic", and several utility icons (search, home, help, etc.). Below the navigation bar are three tabs: "Custom Fields" (selected), "Data Source Extensions", and "Custom Logic". The main content area displays a table titled "Custom Fields (403)". The table has columns for "Identifier", "Business Context", "Type", "Status", and "Task Category". There are three rows of data. The first row, "YY1_mr3211", has a status of "Not Published" and a red arrow pointing to the "Task Category" column. The second row, "YY1_taxindicator", has a status of "Published". The third row, "YY1_Indicator", has a status of "Published". There are two red notification bubbles: one with the number "2" above the search bar and one with the number "1" above the personalisation icon (a gear with a plus sign).

In our case, and picking up on our example that we had identified in the [Extensibility Inventory app](#), we opted to search for a custom field, and we also added the Task Priority column. As usual with such a list display, we can click on the line to access more details. Remember, here we are trying to make sense of this deprecation message, since this is a custom field after all, not an SAP managed field.

Custom Fields Data Source Extensions Custom Logic

Custom Fields (14) YY1_CUSTOMWBSFIELD X Q [Menu] [Settings] +

<input type="checkbox"/>	Identifier	Business Context	Type	Status	Task Priority	Task Category
<input type="checkbox"/>	YY1_MTECustomfields	Master Data: Business Partner	Text	Published		>
<input type="checkbox"/>	YY1_CustomWBSField	Work Package	Text	Published	High	Deprecation >
<input type="checkbox"/>	YY1_customfieldSK	Service Header	Text	Published		>

After clicking on the line of interest, already when we reach the first screen, we see there is a warning message notification in the status bar. Selecting it reveals more details pertaining to the deprecation.

< x

Field usage C_WORKPACKAGEDETAILS is deprecated. Please use its successors C_WORKPACKAGE.

Field usage C_WORKPACKAGEDETAILS is deprecated. Please use its successors C_WORKPACKAGE.

1

A custom field can be enabled for use in a number of objects (an API, a CDS, a UI, a form,...) so it is a case of cycling through the various tabs of this screen to see where our custom field is used and finding out where the deprecation is applicable.

In our case we can see that this information is available in the User Interfaces section / CDS views tab. This screen also now gives us clarity in the meaning of the deprecation message (it is not our custom field that is deprecated !).

The message was advising us that our custom field (1) was used in a CDS (2) that is deprecated (3). With the notification of the successor CDS, the takeaway is that we should use the successor CDS and enable our custom field in conjunction with the successor CDS

CustomWBSField
Field Type: Text (4) Published
Identifier: YY1_CustomWBSField
Business Context: Work Package (/CPD/CUST_PROJ_WORKPACKAGE)

General Information **User Interfaces (8)** Analytics (2) APIs (6) BO Operations Email Templates More

All (8) Services (2) **CDS Views (6)** SAP GUI (0) Search

Data Source	Description	Type	Status	Action
> /CPD/SC_PLAN_INT_PROJ_SRV	Create and Plan Internal Project	OData Service	Disabled	Enable Usage
> /CPD/SC_PROJ_ENGAGMNT_MAINT_SRV	Create and Plan Customer Project	OData Service	Disabled	Enable Usage
C_WORKPACKAGE	Package	CDS View	Enabled	Disable Usage
C_WORKPACKAGEDETAILS	Work Package Details	CDS View	Deprecated	Disable Usage
L_CUSTOMPROJCTWO		CDS View	Disabled	Enable Usage
		CDS View (Analytical Dimension)	Disabled	Enable Usage
		CDS View (BO Interface)	Disabled	Enable Usage
		CDS View (Analytical Dimension)	Enabled	Disable Usage

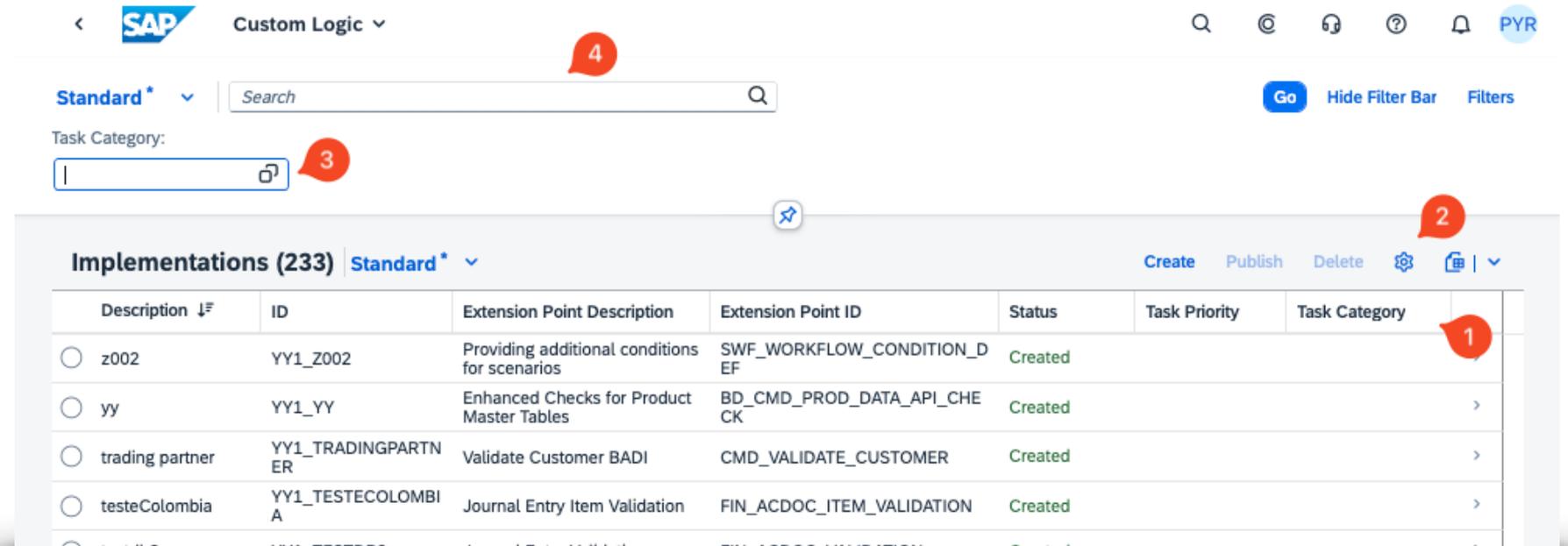
Field usage C_WORKPACKAGEDETAILS is deprecated. Please use its successors C_WORKPACKAGE.

Field usage C_WORKPACKAGEDETAILS is deprecated. Please use its successors C_WORKPACKAGE.

Save **Publish** Discard Changes Delete Cancel

The Custom Logic application

To be able to access this app, you will need to have the Business Catalog `SAP_CORE_BC_EXT_BLE` assigned to a Business Role, assigned to your business user. The initial screen of this app also shows a Task Category column advising you of possible issues with a given custom logic. Deprecation relevant items will be flagged in this column (1). If you do not see it, then click the personalisation (2) icon and bring the category column to your table. You can then enter the name of the custom logic you are looking for in the search (4) box and/or also look for items that are the object of a deprecation by entering `Depre*` in the Task Category (3) filter.



The screenshot shows the SAP Custom Logic application interface. At the top, there is a navigation bar with the SAP logo and the title 'Custom Logic'. Below this, there is a search bar (4) and a 'Task Category' filter (3). The main content area displays a table of 'Implementations (233)' with columns for Description, ID, Extension Point Description, Extension Point ID, Status, Task Priority, and Task Category. A red callout (1) points to the 'Task Category' column header. A red callout (2) points to the personalisation icon in the top right corner of the table area. A red callout (3) points to the 'Task Category' filter input field. A red callout (4) points to the search bar.

Description	ID	Extension Point Description	Extension Point ID	Status	Task Priority	Task Category
z002	YY1_Z002	Providing additional conditions for scenarios	SWF_WORKFLOW_CONDITION_DEF	Created		
yy	YY1_YY	Enhanced Checks for Product Master Tables	BD_CMD_PROD_DATA_API_CHECK	Created		
trading partner	YY1_TRADINGPARTNER	Validate Customer BADI	CMD_VALIDATE_CUSTOMER	Created		
testeColombia	YY1_TESTECOLOMBIA	Journal Entry Item Validation	FIN_ACDOC_ITEM_VALIDATION	Created		

Once the list is updated with a list corresponding to our search results, we can as usual click on a line in the list to access the details.

The screenshot shows the SAP Custom Logic search interface. At the top, there is a search bar containing 'YY1_LMANTICIPO' and a 'Go' button. Below the search bar, there is a 'Task Category' filter set to 'Depre*'. The main area displays a table of search results under the heading 'Implementations (1)'. The table has columns for Description, ID, Extension Point Description, Extension Point ID, Status, Task Priority, and Task Category. The first row is highlighted, showing 'LMANTICIPO' with ID 'YY1_LMANTICIPO', Extension Point Description 'Journal Entry Item Validation', and Extension Point ID 'FIN_ACDOC_ITEM_VALIDATION'. The status is 'Published' and the task priority is 'Medium'. A red box highlights the 'Task Category' column, which contains the text 'Deprecation'.

Description	ID	Extension Point Description	Extension Point ID	Status	Task Priority	Task Category
LMANTICIPO	YY1_LMANTICIPO	Journal Entry Item Validation	FIN_ACDOC_ITEM_VALIDATION	Published	Medium	Deprecation

As soon as you enter the detail screen, you will be greeted with a message (1), re-iterating the technical details of the deprecation.

You can also click on the Messages (2) hyperlink, to provide you with further details, including a more prescriptive message.

The screenshot shows the detail screen for the 'LMANTICIPO' implementation. The title is 'LMANTICIPO' and the ID is 'YY1_LMANTICIPO'. The status is 'Published'. There is a 'Messages' section with a red notification bubble containing the number '1'. Below this, there is a 'Messages' modal window with a red notification bubble containing the number '2'. The modal window displays a warning message: 'Element CLEARINGACCOUNTINGDOCUMENT of CDS view I_OPERATIONALACCTGDOCCUBE is deprecated. Use element CLEARINGJOURNALENTY instead.' The modal window also shows the same message again: 'Element CLEARINGACCOUNTINGDOCUMENT of CDS view I_OPERATIONALACCTGDOCCUBE is deprecated. Use element CLEARINGJOURNALENTY instead.' There is a 'Close' button at the bottom right of the modal window.

As you can see here, it is clearly telling you what you should do: Replace the deprecated element with the successor element. To proceed with the change, we need to delve into the code editor (2).

The screenshot shows the SAP Custom Logic editor interface. At the top, there is a navigation bar with the SAP logo, a back arrow, the text "Custom Logic", and a search icon. Below this, the main header area displays "LMANTICIPO" in large bold letters, with "YY1_LMANTICIPO" underneath. To the right of the header is a blue button labeled "Open Code Editor" with a red speech bubble containing the number "2".

Below the header, there are two tabs: "Status" (with "Published" below it) and "Messages" (with a warning icon and the number "1" below it). A modal dialog box is open over the "Messages" tab. The dialog has a close button in the top right corner and a back arrow in the top left. The text inside the dialog reads:

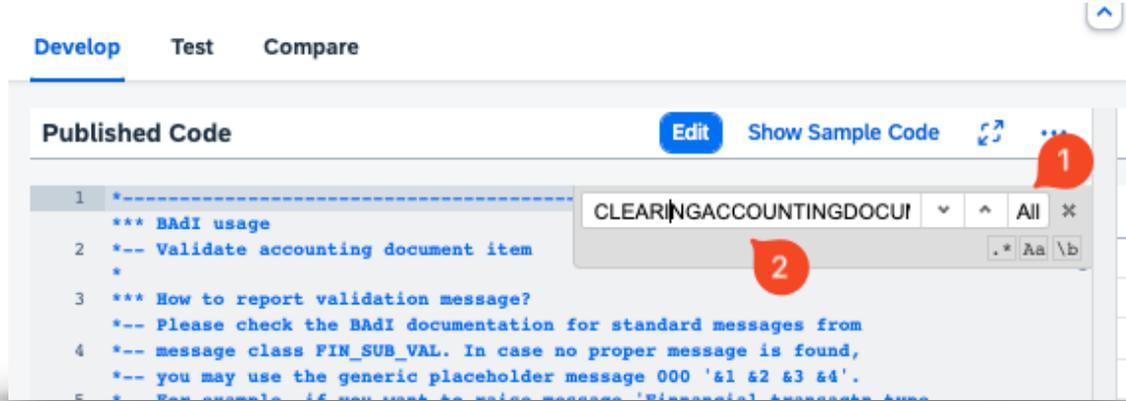
If the edited object uses a formerly released but now deprecated CDS view element, this results in a syntax warning. This CDS view element can either be a field or a public association.

Procedure 1

Replace the use of deprecated CDS view element CLEARINGACCOUNTINGDOCUMENT by its successor CLEARINGJOURNALENTRY.

Message no. LA007

Once you are in the code editor, you can switch to Edit mode, if you are planning on actually making the change here and now, but you can also stay in display mode, if at this time you only want to identify where you will need to make changes. If your code is quite lengthy or complex, you can use the keyboard keys Ctrl + F, to open a search box (1) and enter in it, the name of the Deprecated CDS element (2) that you need to replace.



You can then cycle through all the occurrences of the deprecated CDS element in your program and replace them with the successor CDS element.

Note, that in this case, we only needed to change a CDS element, it could also be that you need to change the CDS itself.

```
101 Select distinct AccountingDocument, AmountInTransactionCurrency
102 from I_OperationalAcctgDocCube into corresponding fields of table @xs_www
103 Where FiscalYear = @accountingdocheader-fiscalyear and
104 CompanyCode = @accountingdocheader-companycode and
105 GLAccount = '0021190000' and
106 SpecialGLCode = 'A' and
107 ClearingAccountingDocument = '' and
108 AccountingDocumentType = 'ZD' and
109 DebitCreditCode = 'H' and
110 Customer = @accountingdocitem-customer and
111 InvoiceReference = ''
```

The Display Inbound Services application

The [Display Inbound Services](#) application is an application that you will probably want to use if you have previously ascertained that an API you are using is deprecated, or maybe as part of your regular upgrade workflow, you will want to check if an API you are using is deprecated. But of course, once you know that an API is deprecated, you need to know in what communication arrangement(s) you are using it!

When you launch the application, unless you have already setup a display variant, it will likely display a large number of services, so you will probably first want to limit this list to just the deprecated APIs. To do so, adapt the filters and add the Status Filter (1) and restrict the filter with the value

Deprecated. This is optional, but depending on your context, you can further limit the selection with a release (2) information. Once you click Go, the resulting list will display Deprecated services (3), the release at which it was deprecated (4). The older the release, if you are still using it, the greater the urgency might be to process the deprecation! Also note that an inbound service (API) could be used by more than one Communication Scenario (meaning that it may be used in more than one Communication Arrangement. We'll assume that we are interested in

The screenshot shows the SAP Display Inbound Services application interface. At the top, there is a search bar and several filter fields: Name, Inbound Service ID, Service Type, Type, Status, and Deprecated with Release. The Status filter is set to 'Deprecated' (marked with a red '1'). The Deprecated with Release field is empty (marked with a red '2'). Below the filters is a table titled 'Inbound Services (34)'. The table has columns for Name, Inbound Service ID, Service Type, Type, Status, Deprecated with Rel..., Changed On, and Communication Scenarios. The Status column is highlighted with a blue box and a red '3'. The Deprecated with Rel... column is highlighted with a blue box and a red '4'. The Communication Scenarios column for the 'Purchase Contracts' row is highlighted with a blue box and a red '5'. A 'Go' button and 'Adapt Filters (1)' text are visible at the bottom right of the filter section.

Name	Inbound Service ID	Service Type	Type	Status	Deprecated with Rel...	Changed On	Communication Scenarios
Remote API for Bank Account Signatory	API_BANKACCOUNTSIGNATORY_SRV_0001_IWSG	OData V2	Managed by SAP	Deprecated	2308	05.11.2018	1 >
Maintenance Order	API_MAINTENANCEORDER_0001_IWSG	OData V2	Managed by SAP	Deprecated	2308	21.05.2021	1 >
MDO_ORCHESTRATIONADMIN_0001 (Deprecated)	MDO_ORCHESTRATIONADMIN_0001_IWSG	OData V2	Managed by SAP	Deprecated	2308	08.04.2020	1 >
MDO_REPLICATION_0001 (Deprecated)	MDO_REPLICATION_0001_IWSG	OData V2	Managed by SAP	Deprecated	2308	28.02.2020	1 >
Defect	API_DEFECT_SRV_0001_IWSG	OData V2	Managed by SAP	Deprecated	2302	17.11.2017	1 >
Business Place - Read (Deprecated)	API_LOGBR_BUSINESS_PLACE_SRV_0001_IWSG	OData V2	Managed by SAP	Deprecated	2208	24.07.2019	1 >
Purchase Contracts	API_PURCHASECONTRACT_PROC_SRV_0001_IWSG	OData V2	Managed by SAP	Deprecated	2202	05.07.2017	2 >
Service Quotation (A2X)	API_SERVICE_QUOTATION_SRV_0001_IWSG	OData V2	Managed by SAP	Deprecated	2202	05.07.2017	2 >

finding more information on one such API, which is going to be `API_BILL_OF_MATERIAL_SRV_0001_IWSG`. To get further details on our chosen service, we click on the line (1) corresponding to it. A new pane will open to the right of your window, revealing more details on this service. An important piece of information that you may have already inventoried, is the (2) successor service (API) that you should transition to. More importantly, we can now also identify the SAP communication scenario(s), that make use of this (3) inbound service. This will help us to identify which communication arrangement(s) we have setup, that make use of this deprecated service! To find that information we need to navigate to the [Display Communication Scenarios](#) application which we will cover in the next chapter. To do so, we can click on the communication scenario ID (4), or open the application that goes with the same name, the [Display Communication Scenarios Application](#).

Standard ▾

1 filter active: Status

Inbound Services (34) [Filter] [Settings] [Refresh] [More]

Name
Communication Scenarios: 1
Bills of Material >
Inbound Service ID: API_BILL_OF_MATERIAL_SRV_0001_IWSG
Service Type: OData V2
Type: Managed by SAP
Status: Deprecated
Deprecated with Release: 2105
Changed On: 29.01.2018
Communication Scenarios: 1
Handling Unit - Read (A2X) >
Inbound Service ID: API_HANDLING_UNIT_0001_IWSG
Service Type: OData V2
Type: Managed by SAP
Status: Deprecated
Deprecated with Release: 2105
Changed On: 07.08.2019
Communication Scenarios: 2
Enterprise Project (Deprecated) (Obsolete) >
Inbound Service ID: API_ENTERPRISE_PROJECT_0001_IWSG
Service Type: OData V2
Type: Managed by SAP
Status: Deprecated
Deprecated with Release: 2102
Changed On: 02.09.2017
Communication Scenarios: 1
Process Order >
Inbound Service ID: API_PROCESS_ORDERS_0001_IWSG
Service Type: OData V2

Bills of Material
API_BILL_OF_MATERIAL_SRV_0001_IWSG

Service Type: OData V2
Type: Managed by SAP
Release State:
Deprecated with Release: 2105
Changed On: 29.01.2018
Changed By: SAP

Documentation Successors Used by Communication Scenarios

Maintain bills of material using this asynchronous inbound service.
The service enables you to read, create, and update bills of material with and without version. The service is based on the OData protocol and can be consumed in SAP Fiori apps and on other user interfaces.
The service contains either 2 header, one header, or multiple headers for the specified material and none or more items for each header.

Successors

Standard ▾ [Search] [Filter] [More]

Name	Successor Service ID
Bills of Material	API_BILL_OF_MATERIAL_SRV_0002_IWSG

Used by Communication Scenarios

Standard ▾ [Search] [Filter] [More]

Name	Communication Scenario ID	Deprecated within Scenario
Product Lifecycle Management - Master Data Integration	SAP_COM_0105	

The Display Communication Scenarios Application

This application will be helpful to understand if you are using a deprecated API and if you are, in which communication arrangement it is being used. In SAP S/4HANA Cloud, API's are implemented through communication arrangements. That is, you need to create a communication arrangement, by referencing a SAP Standard Scenario ID (they usually have a technical name such as **SAP_COM_NNNN**, where NNNN is a four digit numerical value). A SAP Scenario, will usually logically group together APIs (i.e they are related due to their functions, such as Integration to Success Factors or line of business, such as Asset Management). Compared to previously seen apps, the Display Communication Scenarios application does not have a rework task or priority indicator. However it has a Deprecated (Yes or No) indicator. That said a subtle distinction must be made, in that a Communication Scenario can be deprecated, which is not the same as Communication Scenario that is not deprecated, that includes one or more deprecated APIs! To only see those communication scenarios that are deprecated, then restrict the Status filter by choosing the Deprecated value. You can further restrict the result by opting to only see those scenarios that you actually use (i.e are used in a communication arrangement), by restricting the Used By Communication Arrangement filter to Yes.

Standard* ▾

Search Scenario Name: Scenario ID: Type: Status: Release: Used By Communication Arran...:

Adapt Filters (2)

Scenario Name	Scenario ID	Type	Status	Release	Contains Inbound	Contains Outbound	Changed On	Communication Arrangements
SAP Web IDE Integration (Obsolete)	SAP_COM_0013	Managed by SAP	Deprecated	7.84	✓		14.04.2022	1 >
SAP Cloud Platform - View Replication Integration (Deprecated)	SAP_COM_0273	Managed by SAP	Deprecated	2308		✓	22.05.2023	1 >
Application Monitoring	SAP_COM_0454	Managed by SAP	Deprecated	7.87	✓		20.10.2021	1 >

Another way, if you already know the ID of the communication scenario you are interested in, or if you clicked on the communication scenario ID in the Display Inbound Services application, you can use the Scenario ID (1) filter. As usual, you can click on the resulting line, which will reveal a details pane on the right. The information we are seeking here, is the Used by Communication Arrangements tab (2) - as the name indicates, this will allow you to identify your communication arrangement(s) that use this API, as well as the communication system that is connected to your SAP S/4HANA Cloud system. From here you can further drill down to the Communication Arrangement or the Communication System, by clicking on the hyperlink that corresponds to them.

The screenshot shows the SAP 'Display Communication Scenarios' interface. On the left, there are filter fields for Scenario Name, Scenario ID (containing '=SAP_COM_0105'), Type, Status, and Release. A 'Go' button and 'Adapt Filters (1)' link are also visible. Below the filters is a table of 'Communication Scenarios (1)' with one entry: 'Product Lifecycle Management - Master Data Integration' with ID 'SAP_COM_0105'. On the right, the details pane for this scenario is shown, with tabs for 'General', 'Inbound', 'Outbound', and 'Used by Communication Arrangements'. The 'Used by Communication Arrangements' tab is active and contains a table with 4 entries.

Arrangement Name	Communication System
SAP_COM_0105	EC2
ZPRE_COM_0105	PRE_MEL
EXCEL_SAP_COM_0105	EXCEL_BOM
TEST_SAP_COM_0105	TEST_0105

For the sake of completeness, we will click on the hyperlink that corresponds to one of those identified Communication Arrangements. This will allow us to navigate to the [Communication Arrangements application](#). Once we reach the application and the details of the communication arrangement we selected are displayed, we can indeed see our communication arrangement (1), the SAP Communication Scenario ID (2) and finally, the service that we identified as used and deprecated (3) originally.

The screenshot shows the SAP Communication Arrangements application interface. At the top, there is a navigation bar with the SAP logo, a back arrow, and the text 'Communication Arrangements'. On the right side of the navigation bar, there are icons for search, copyright, help, and a user profile labeled 'PYR'. Below the navigation bar, the communication arrangement ID 'ZPRE_COM_0105' is highlighted with a red box and a red circle containing the number '1'. Below this, the product lifecycle management details 'Product Lifecycle Management - Master Data Integration' and the SAP Communication Scenario ID 'SAP_COM_0105' are also highlighted with red boxes and a red circle containing the number '2'. On the right side of the main content area, there are three buttons: 'Edit', 'Display Changes', and 'Delete'. Below the navigation bar, the section 'Inbound Services' is displayed. It contains a table with the following data:

Service	Application Protocol	Service URL / Service Interface	WSDL/Service Metadata	Additional Properties
Bills of Material <small>Deprecated with release 2105</small>	OData V2	https://api.s4hana.ondemand.com/sap/opu/odata/sap/API_BILL_OF_MATERIAL_SRV	↓	
Change Master - Read <small>Deprecated with release 1908</small>	OData V2	https://api.s4hana.ondemand.com/sap/opu/odata/sap/API_CHANGEMASTER	↓	
Attachments	OData V2	https://api.s4hana.ondemand.com/sap/opu/odata/sap/API_CV_ATTACHMENT_SRV	↓	

The Manage Business Role Changes After Upgrade Application

This [Manage Business Role Changes After Upgrade](#) application really is your workhorse application if you are responsible for analysing and/or processing changes and deprecations to what I would broadly refer to as IAM (Identity Access Management) objects. That is, restrictions, business catalogs and business roles, as the key ones but not only. Whilst this app goes beyond this, we will for the purpose of this document only focus on its use in the context of Deprecations. In order to utilise this application you will need to have an administrator type catalog (such as SAP_CORE_BC_IAM_UPGRADE, SAP_CORE_BC_IAM_UM, SAP_CORE_BC_IAM_RA, etc...) assigned to a business role assigned to your user.

We will here provide a brief introduction to this application, but not focus greatly on it as a wealth of blogs and tutorials already exist for it on the internet, some of which we will suggest in the further reading section. This application allows for what I would call a bi-directional analysis. I.e you could for example start by looking at Restrictions and find out which of your roles are affected by this change, but you could also start by looking at one of your roles and drill down to all the changes that affect it.

Upon first entering the application, you will see a screen and list, similar to the one below, offering a number of tabs to segregate the object you want to look at. The filters that are available for every tab are contextual, and will change depending on the tab you are on.

Restriction Type	Restriction Type ID	Change	Details	Affected Business Catalogs
Access restriction for EHS applications	LOCTYP_STS_AGRP_PLNT_CSTCT R_CC	Added	Write Added, Read Added	SAP_EHS_BC_ENV_MWST_TRAREQ_PC
Access restriction for EHS applications	LOCTYP_STS_AGRP_PLNT_CSTCT R_CC	Added	Read Added	SAP_EHS_BC_ENV_WST_ANLTICS_PC
Access Restrictions to EHS Compliance Requirement	EHFND_REQ	Access Categories Changed	Write Added	SAP_EHS_BC_ENV_MNG_CHEM_PC
Access Restrictions to EHS Compliance Requirement	EHFND_REQ	Added	Read Added	SAP_PSS_BC_PC_PROD_CRR_MON_PC
Access Restrictions to EHS Compliance Scenario	LOCTYPE_LOCSTATUS_COUNTRY_ REGI	Access Categories Changed	Write Added	SAP_EHS_BC_ENV_MY_EMIS_DEC_PC
Access Restrictions to EHS Compliance Scenario	LOCTYPE_LOCSTATUS_COUNTRY_ REGI	Added	Read Added	SAP_EHS_BC_IM_TASK_CAL_PC

Restriction Types

Since you are interested in Deprecations, the first thing you will want, is to change the selection filters, such that the change type shown is [Phase-Out](#). This status denotes that the restriction will be removed in the next release.

Manage Business Role Changes After Upgrade

Search Restriction Type: Restriction Type ID: Change: Affected Business Catalogs:

[Restriction Types \(167\)](#) [Business Catalog Dependencies \(466\)](#) [Deprecated Business Catalogs \(41\)](#) [Business Role Templates \(111\)](#) [Affected Business Roles \(220\)](#)

Note that the resulting output might lead to a restriction being present multiple times. This multiplication in appearances could be due to the restriction being used in several catalogs and/or with varying uses (Read, Write, Value Help).

Asset Class/Company Code (Deprecated)	ANLKL_BUKRS	Phase-Out	Phase-Out (Read)	SAP_FIN_BC_AA_GEN_REP_PC
Asset Class/Company Code (Deprecated)	ANLKL_BUKRS	Phase-Out	Phase-Out (Write, Read)	SAP_FIN_BC_AA_MDAT_LDT_PC
Asset Class/Company Code (Deprecated)	ANLKL_BUKRS	Phase-Out	Phase-Out (Write, Read)	SAP_FIN_BC_AA_MDAT_REG_PC
Asset Class/Company Code (Deprecated)	ANLKL_BUKRS	Phase-Out	Phase-Out (Write, Read)	SAP_FIN_BC_AA_PER_ACT_PC
Asset Class/Company Code (Deprecated)	ANLKL_BUKRS	Phase-Out	Phase-Out (Write, Read)	SAP_FIN_BC_AA_POST_REG_PC
Asset Class/Company Code (Deprecated)	ANLKL_BUKRS	Phase-Out	Phase-Out (Write, Read)	SAP_FIN_BC_AA_REVAL_PC
Asset Class/Company Code (Deprecated)	ANLKL_BUKRS	Phase-Out	Phase-Out (Read)	SAP_FIN_BC_ACCR_REP_PC

Clicking on one line, will provide us with more details, but this is subject to the Restrictions actually being restricted (indeed if a restriction is Unrestricted or No access, then concern for this phase out does not really exist) in the catalog and role where it is used. If we click on a line, the detail presented will be similar to this.

Manage Business Role Changes After Upgrade

Type: **Restriction Type**
 Name: [Asset Class/Company Code \(Deprecated\)](#)
 Change: Phase-Out
 Details: Phase-Out (Write, Read)

Affected Business Roles (6)

Business Role ID	Business Role Description	Write Restricted	Read Restricted	Value Help Restricted
ZBR_AA_ACCOUNTANT_JIN	Asset Accountant			
ZBR_AA_ACCOUNTANT	Asset Accountant			
BR_AA_ACCOUNTANT	Asset Accountant			
FH_BR_AA_ACCOUNTANT	Asset Accountant	✓	✓	
ZZBR_TEST	TEST			

This detail is essentially advising us, that the Restriction Type Asset Class/Company Code, is used in the business catalog SAP_FIN_BC_AA_PER_ACT_PC (selection from the previous screen), and that this business catalog is used in 6 business roles. It is also telling us that the restriction is Write Restricted (1) in one role, and Read Restricted in another role (2). The business roles that appear in the detail screen, are hyperlinked to the business role in question. Let's click on the hyperlink of the role FH_BR_AA_ACCOUNTANT (the one write restricted)(1) to navigate to the business role itself.

As soon as we enter the business role, we can indeed confirm that it is Write Restricted (1).

SAP Maintain Business Roles

Asset Accountant

FH_BR_AA_ACCOUNTANT

Write Access: Restricted
Read Access: Unrestricted
Value Help Access: Unrestricted

Changed By: [User]
Changed On: 05.08.2023, 01:54:13

Editing Status: Active
Exported: No

Access Categories

Write, Read, Value Help:	Restricted
Read, Value Help:	Unrestricted
Value Help:	Unrestricted

Others

Price Category: Advanced
Business Role Template ID: SAP_BR_AA_ACCOUNTANT
Leading Business Role ID:
Is Leading Business Role:
Exposed to SAP BTP:

Let's click on the navigation option (2) Display Restrictions (Deprecated).

Because the Write Access is restricted, we can see the various restrictions that are available. In this case we can see that for this phase out restriction, we had not maintained any values.

Note that you also have a separate application named [Display Restriction Types](#).

SAP Maintain Business Roles

Asset Accountant

FH_BR_AA_ACCOUNTANT

Write Access: Restricted
Read Access: Unrestricted
Value Help Access: Unrestricted

Changed By: Martin Ransheim
Changed On: 05.08.2023, 01:54:13

Editing Status: Active
Exported: No

Asset Class/Company Code (Deprecated) (Phase-Out)

<input type="checkbox"/>	Asset Class		
	Company Code		

Asset Class/Asset Transaction Type

<input type="checkbox"/>	Asset Class		
	Asset Transaction Type		

Deprecated Business Catalogs

This tab title has the merit of being clear! It will show you business catalogs in use that are deprecated, and only that. If you have been staying on top of your business catalog deprecations, then you really should only be seeing the deprecations that were introduced during the last upgrade, but you can also use the filter **Deprecated with Release (1)** , if you want to focus on the deprecations of that specific release. Note that in the output list, we have that all familiar column named **Successors (2)** , which presents the Successor Business catalog(s).

Manage Business Role Changes After Upgrade

Business Catalog: Business Catalog ID: **Deprecated with Release:** Successors: **Go**

Restriction Types (167) Business Catalog Dependencies (-) **Deprecated Business Catalogs (21)** Business Role Templates (-) Affected Business Roles (-)

Deprecated Business Catalogs (21) [Download](#)

Business Catalog	Business Catalog ID	Deprecated with Release	Successors
Extensibility - Situation Handling (Deprecated)	SAP_CA_BC_EXT_SIT_PC	2308	SAP_CORE_BC_EXT_SIT_PC
End to End Implementation Experience - Feature Management (Deprecated)	SAP_CA_BC_FM_DAD_PC	2308	SAP_CA_BC_IC_LND_FTG_PC
Advanced Financial Closing - Configuration (Deprecated)	SAP_CA_BC_IC_LND_FIN_AFC_PC	2308	
Whats New - Whats New in Your System (Deprecated)	SAP_CA_BC_WHATS_NEW_PC	2308	
Master Data - Location (Deprecated)	SAP_CMD_BC_LOCATION_PC	2308	SAP_CMD_BC_LOCN_PC, SAP_CMD_BC_LOC_CRSP_PC

As usual, clicking on a particular line, will bring up the details of that business catalog. To illustrate our example, we will use the one shown below, which we can see has two successors.

Deprecated Business Catalogs (1) Download ↓↑			
Business Catalog	Business Catalog ID	Deprecated with Release	Successors
Master Data - Location (Deprecated)	SAP_CMD_BC_LOCATION_PC	2308	SAP_CMD_BC_LOCN_PC, SAP_CMD_BC_LOC_CRSP_PC

Once we reach the details screen, we are provided with the information that the deprecated catalog we chose, is used by two business roles. We also see an option to 'Adopt Changes' - meaning that we can perform the switch from the old deprecated catalog by removing it from the selected role(s) and replace it by adding the successor Business catalog(s) to it. So let's go ahead and try that.

<  Manage Business Role Changes After Upgrade ▾ 🔍 © 🗣️ ? 🔔 PYR

Manage Business Role Changes After Upgrade

Type: Business Catalog Name: Master Data - Location (Deprecated)
Change: 2308

Affected Business Roles (2) Adopt Changes ↻ ↓↑					
<input type="checkbox"/>	Business Role ID	Business Role Description	Write Restricted	Read Restricted	Value Help Restricted
<input type="checkbox"/>	 BR_TRANSP_MDSPEC	Master Data Specialist - Transportation Management			
<input type="checkbox"/>	 ZBR_INTERNAL_SALES_REP	Internal Sales Representative			

In terms of actions,

1. We selected the role(s) we wanted to adapt (1),
2. We clicked the Adopt Changes button (2),
3. A popup was displayed asking us to confirm our wish to perform the adoption during which we were proposed to also add the dependent business catalogs,
4. We clicked Ok (4) to validate the action to perform the change,
5. A progress window was displayed showing progress,
6. An end of adoption window was presented confirming the end of the adoption.

The screenshot shows the SAP 'Manage Business Role Changes After Upgrade' interface. At the top, the title is 'Manage Business Role Changes After Upgrade'. Below the title, it shows 'Type: Business Catalog' and 'Name: Master Data - Location (Deprecated) Change: 2308'. A table titled 'Affected Business Roles (2)' lists two roles: 'BR_TRANSP_MDSPEC' (selected with a checkmark) and 'ZBR_INTERNAL_SALES_REP'. A red box highlights the 'Adopt Changes' button. A dialog box is open, asking 'Do you want to adopt the changes?' with a checkbox for 'Add the dependent business catalogs of the successor business catalogs'. A progress bar at the bottom shows 1% completion.

The screenshot shows the 'Adopting Change Results' dialog box. It contains a table with the following data:

Business Role ID	Result
BR_TRANSP_MDSPEC	Changes were adopted successfully for this business role.

A red circle highlights the 'BR_TRANSP_MDSPEC' entry in the table. A 'Close' button is visible at the bottom right.

If we also perform a Before and After adoption comparison of the business catalog(s) that were assigned to the role we just adapted, we can on the left see the presence of a deprecated catalog and after adoption on the right, we see not only that the deprecated catalog is no longer there, but that it has been replaced with two successor catalogs.

Assigned Business Catalogs (4)						
Business Catalog	Business Catalog ID	Price Category	Status	Read Only	Dependencies	
<input type="checkbox"/> Master Data - Business Partner Display	SAP_CMD_BC_BP_DISP_PC	Self Service		✓	0	>
<input type="checkbox"/> Master Data - Location (Deprecated)	SAP_CMD_BC_LOCATION_PC	Core	Deprecated with 2308		0	>
<input type="checkbox"/> Transportation - Master Data	SAP_TM_BC_MAST_PC	Self Service			1	>
<input type="checkbox"/> Transportation - Service Product Management	SAP_TM_BC_SPMG_PC	Self Service			0	>

Before

Assigned Business Catalogs (5)						
Business Catalog	Business Catalog ID	Price Category	Status	Read Only	Dependencies	
<input type="checkbox"/> Master Data - Business Partner Display	SAP_CMD_BC_BP_DISP_PC	Self Service		✓	0	>
<input type="checkbox"/> Master Data - Location	SAP_CMD_BC_LOCN_PC	Core			0	>
<input type="checkbox"/> Master Data - Location Creation from Shipping Points and BP	SAP_CMD_BC_LOC_CRSP_PC	Core			1	>
<input type="checkbox"/> Transportation - Master Data	SAP_TM_BC_MAST_PC	Self Service			1	>
<input type="checkbox"/> Transportation - Service Product Management	SAP_TM_BC_SPMG_PC	Self Service			0	>

After



Please note, that whilst this process might seem expedient, please make sure that you have done your diligence with respect to dependent business catalogs, persistence of restriction values if applicable and assignment to Fiori pages.

Further Reading

SAP Learning | [Managing Business Role Changes after Upgrade](#)

SAP Help | [Best Practices for Managing Business Role Changes After an Upgrade](#)

SAP Help | [How to Manage Business Role Changes After Upgrade](#)

SAP Help | [Phase-In / Phase-Out Status](#)

SAP Support | [Note 2975653 Identity and Access Management \(IAM\): Change Overview for SAP S/4HANA Cloud](#)

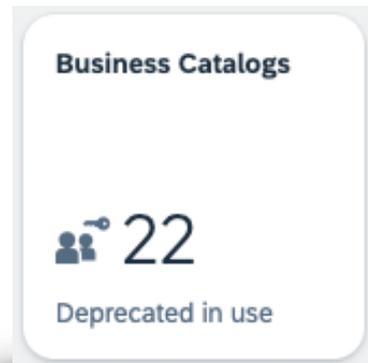
SAP Support | [Note 3093696 - How to manage business role changes after system upgrade](#)

Microlearning | [Video How to Adapt Changes in Business Role in SAP S/4HANA Cloud, Public Edition](#)

Microlearning | [Video How to Manage Deprecated Business Catalogs in SAP S/4HANA Cloud, Public Edition](#)

The Business Catalogs Application

In the previous chapter we showed how you could process deprecated business catalogs via the [Manage Business Role Changes after Upgrade](#) application. You can however also use the [Business Catalogs](#) application directly to process a business catalog deprecation. In order to launch the application you need a business catalog such as `SAP_A4C_BC_IAM_DISP_PC`, `SAP_CORE_BC_IAM_RM...` assigned to a business role, assigned to your user. Already in the Fiori Launchpad, you can see the corresponding application tile, giving you an indication of the number of deprecated business catalogs used in your system. Launching the application will produce a list, such as the one below, for which you can use the filter Status (1) and the value Deprecated to limit the output to Deprecated business catalogs only. Of importance as well here, is the number of business roles that you have in your system that are using this deprecated business catalog (2).



The screenshot shows the SAP Business Catalogs application interface. At the top, there's a navigation bar with the SAP logo and "Business Catalogs" dropdown. Below it, a filter bar includes fields for "Business Catalog:", "Business Catalog ID:", "Status:" (set to "Deprecated"), "Release:", and "Application:". There are also search and "Go" buttons. A table below lists 23 business catalogs, with columns for "Business Catalog", "Business Catalog ID", "Read Only", "Status", "Release", "Price Category", and "Used In Business Roles". Red boxes highlight the "Status" column and the "Used In Business Roles" column. A red arrow points from the "Status" filter to the "Status" column, and another red arrow points from the "Used In Business Roles" column to a red notification bubble with the number "2".

Business Catalog	Business Catalog ID	Read Only	Status	Release	Price Category	Used In Business Roles
Extensibility - Situation Handling (Deprecated)	SAP_CA_BC_EXT_SIT_PC		Deprecated	2308	Advanced	3 >
End to End Implementation Experience - Feature Management (Deprecated)	SAP_CA_BC_FM_DAD_PC		Deprecated	2308	Advanced	2 >
Advanced Financial Closing - Configuration (Deprecated)	SAP_CA_BC_IC_LND_FIN_AFC_PC		Deprecated	2308	Advanced	2 >
SAP Cloud Platform - View Replication Configuration (Deprecated)	SAP_CA_BC_VIEW_REPLICATI ON_PC		Deprecated	2208	Advanced	1 >
SAP Cloud Platform - View Replication Blacklist Fields (Deprecated)	SAP_CA_BC_VR_BLACKLIST_P C		Deprecated	2208	Advanced	1 >
Whats New - Whats New in Your System (Deprecated)	SAP_CA_BC_WHATS_NEW_PC		Deprecated	2308	Self Service	2 >

As usual, clicking on a line will reveal details for the selection (1). We have in the right details pane a number of tabs (2) we can navigate to, and interestingly in this app you also have information pertaining to the deprecation policy of business catalogs. In the Successors tab (4), we can get information on the (5) Successor business catalog(s)

The screenshot displays the SAP Business Catalogs interface. On the left, a list of business catalogs is shown, with the first entry, "Extensibility - Situation Handling (Deprecated)", highlighted by a red box and labeled with a red '1'. The main pane on the right shows the details for this catalog, including its ID (SAP_CA_BC_EXT_SIT_PC) and a deprecation notice. A red box labeled '2' highlights the navigation tabs: General, Applications (2), Restriction Types (0), Dependencies (0), and Successors (1). A dropdown menu is open over the Successors tab, listing options like Custom Catalog Extensions (0), Scope Items (1), Used In Business Roles (3), and Used In Business Role Templates (0). A red box labeled '3' highlights the deprecation description: "This Business Catalog is deprecated. Due to ongoing development in SAP S/4HANA Cloud, including the development of new features and new applications, we are deprecating some business catalogs periodically. This means that some business catalogs will be deprecated and replaced by new ones. You will need to assign roles and users to these new catalogs. Rather than disappearing, such business catalogs are marked as deprecated, which allows you to identify them at a glance. You can also check how many deprecated business catalogs you still have in use with the Business Catalogs app. This app lets you change assignments from the old, deprecated business catalogs to the new, active catalogs quickly and easily." A red box labeled '4' highlights the Successors (1) tab, and a red box labeled '5' highlights the successor catalog ID, SAP_CORE_BC_EXT_SIT_PC, in the Successors table.

Business Catalog	Read Only	Status
Extensibility - Situation Handling (Deprecated)		Deprecated
End to End Implementation Experience - Feature Management (Deprecated)		Deprecated
Advanced Financial Closing - Configuration (Deprecated)		Deprecated
SAP Cloud Platform - View Replication Configuration (Deprecated)		Deprecated
SAP Cloud Platform - View Replication Blacklist Fields (Deprecated)		Deprecated

Extensibility - Situation Handling (Deprecated)
SAP_CA_BC_EXT_SIT_PC
Changed On: 14.12.2022, 04:26:35 PM
Price Category: Advanced
Deprecated with Release: 2308
Read Only: No
Component: CA-SIT

General Applications (2) Restriction Types (0) Dependencies (0) Successors (1) More

Description

This Business Catalog is deprecated

Due to ongoing development in SAP S/4HANA Cloud, including the development of new features and new applications, we are deprecating some business catalogs periodically. This means that some business catalogs will be deprecated and replaced by new ones. You will need to assign roles and users to these new catalogs. Rather than disappearing, such business catalogs are marked as deprecated, which allows you to identify them at a glance. You can also check how many deprecated business catalogs you still have in use with the Business Catalogs app. This app lets you change assignments from the old, deprecated business catalogs to the new, active catalogs quickly and easily.

Once the deprecation of a business catalog is announced with the Business Catalogs app, the catalog stays in the system for at least 6 months before being deleted. During these at least 6 months, you can use the old or the new business catalogs. Within this timeframe, you can replace them when it suits you best. In the Business Catalogs app, you can see the release in which the deprecation of a business catalog was announced.

Successors (1)

Business Catalog	Business Catalog ID
Extensibility - Situation Handling	SAP_CORE_BC_EXT_SIT_PC

To proceed and process the deprecation, i.e the removal of the deprecated business catalog(s), and the addition of the successor business catalog(s) to the business role(s) that make use of the deprecated business catalog(s), we need to go to the Used in Business Roles tab (6). Once there, as we did in the [Manage Business Role changes after Upgrade](#) application, we select the (7) business Role(s), that we want to adapt and then click the Adopt Changes (8) button. Just like in the Manage Business Role changes after Upgrade application, we will have the ability to add dependent business catalogs. Once the adaption is processed, if all went well, we will get a corresponding notification, and the list of Business roles adapted.

Extensibility - Situation Handling (Deprecated)

SAP_CA_BC_EXT_SIT_PC

Changed On: 14.12.2022, 04:26:35 PM

Price Category: Advanced

Deprecated with Release: 2308

Read Only: No

Component: CA-SIT

<input type="checkbox"/>	Business Role	Business Role ID	Price Category
<input checked="" type="checkbox"/>	Administrator	BR_ADMINISTRATOR	Advanced
<input type="checkbox"/>	Extensibility Specialist	BR_EXTENSIBILITY_SPEC	Advanced
<input type="checkbox"/>	All Roles	Z_SAP_ALL1	Advanced

Business Role ID BR_ADMINISTRATOR has been adapted

Cancel

The Custom CDS application

The [Custom CDS](#) app, is the key-user extensibility application to use to create Custom CDSs. In order to be able to use, you will need to have a business catalog such as `SAP_CORE_BC_EXT` or `SAP_CORE_BC_EXT_CCV` assigned to a business role, assigned to your user. We will in this section not cover it in great detail, as we will during the exploration of our use cases, make ample use of it.

Already on the Fiori Launchpad, you can see an indication of the number of urgent rework tasks that you need take care of. Note that this number represents only the urgent (High Priority) tasks, not ALL the tasks that need to be taken care of! Once the app opens, you can use the (1) filters and limit the result, so as to only see those CDSs that are the object of a deprecation (2 - Task Category), as well as the the priority (3) synonymous of the urgency.

Custom CDS Views

Custom CDS Views

Filters active: Editing Status, Task Category

Custom CDS Views (7)

Name	Label	Scenario	Protection	Exported	Status	Task Priority	Rework	Task Category
YY1_BillingDocumentItemCub	Analytics - Sales Volume Cube	Analytical Cube	Protected	No	Changed	Medium	✓	Deprecation
YY1_E3U	E3U	External API	Protected	No	Published	Medium	✓	Multiple Categories
YY1_JOURNALENTYITEM_CDS	JOURNALENTYITEM	External API	Protected	No	Published	High	✓	Deprecation
YY1_PRE_DEPR_01	Deprecation Custom CDS - Sales 01	Analytical Cube	Protected	No	Published	High	✓	Deprecation
YY1_Purchasing	Purchase	External API	Protected	No	Published	High	✓	Deprecation
YY1_SalesOrderDemoReport	Sales Order Demo Report	Analytical Cube	Protected	No	Published	High	✓	Deprecation
YY1_SALESORDERITEMCUBE	SALESORDERITEMCUBE	External API	Protected	No	Published	Medium	✓	Multiple Categories

Urgent Tasks

Create Copy Delete

Delete Non-Migrated Views Migrate Views

Further Reading

As said, there will be ample opportunities to further explore the Custom CDS application when we delve into the deprecation uses cases. However, I would be remiss to not link here the excellent blogs written by [Andreas Riehl](#) available on SAP blogs. I would recommend you subscribe to his feed and be on the lookout for the blog that he produces at the time of each SAP S/4HANA Cloud upgrade.

SAP Blogs | [Custom CDS View - What's new in 2105](#) (an oldie but Goldie!)

SAP Blogs | [Custom CDS Views - FAQ](#)

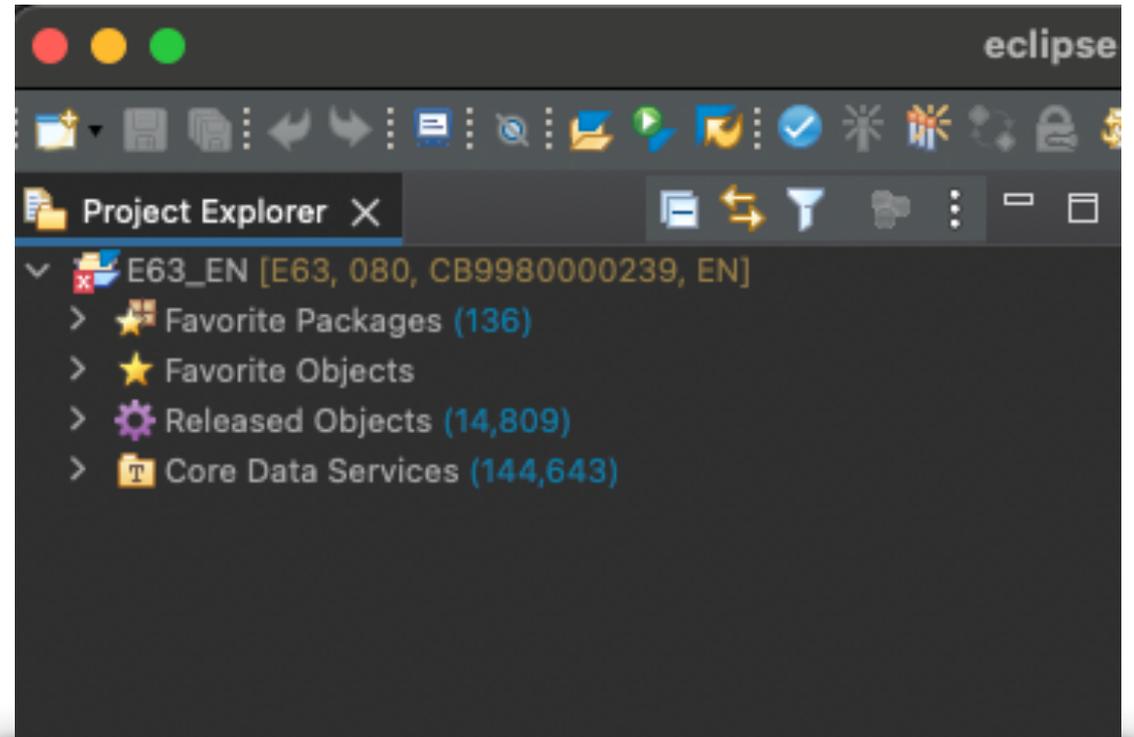
SAP Blogs | [The all new Custom CDS Views App](#)

SAP Tutorial | [Create and Expose Custom CDS Views](#)

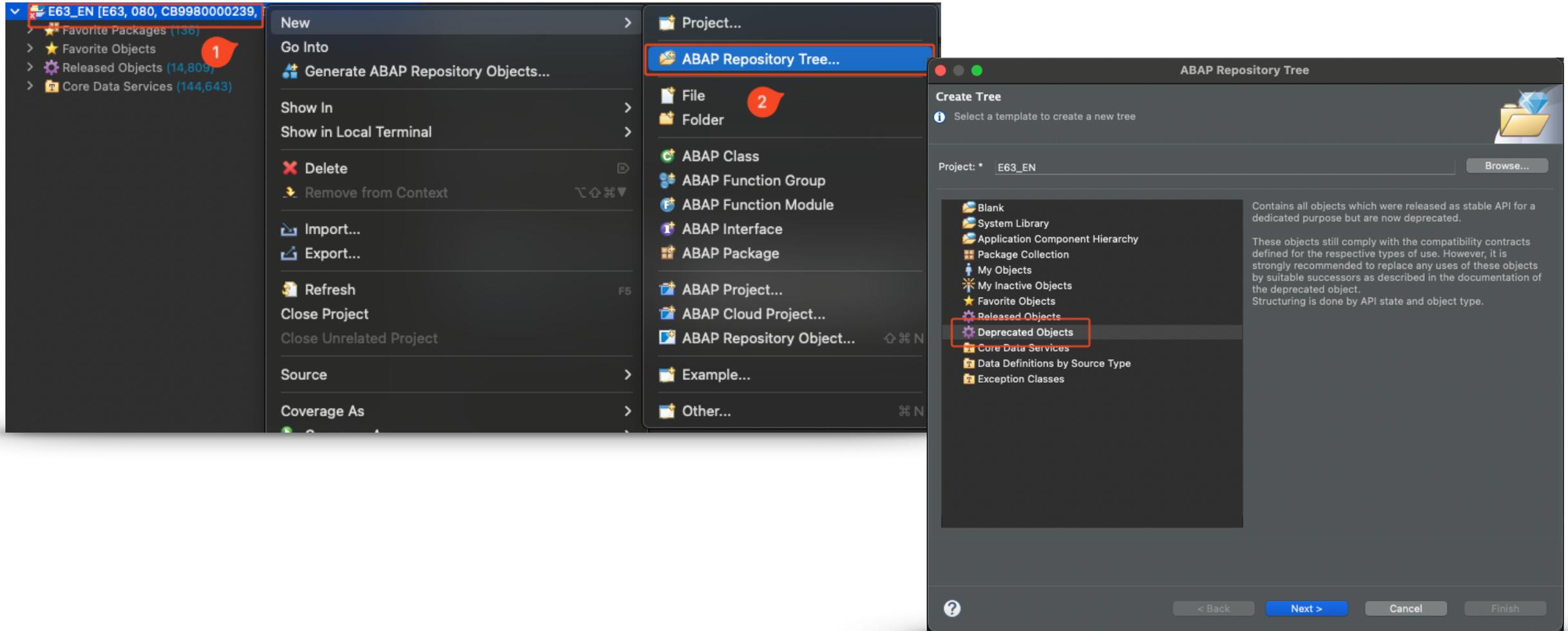
Inside ADT (ABAP Developer Tools)

So far, when it came to in-system information we looked at what was available inside SAP S/4HANA Cloud and this is of course very valid in the context of Key User Extensibility. However with the advent of the 3 system landscape (3SL), came the introduction of [Developer Extensibility](#). I.e with the addition of a Development client (as the name indicates dedicated to Development) added to your Development system, you now have access to ABAP Cloud and a fully fledged IDE (Integrated Development Environment). You might also have seen references to 'Embedded Steampunk'.

In this IDE, it is also possible to view deprecated objects, relevant to developer extensibility. It is assumed that you have created an ABAP Cloud Project, thus setting up the connection to the Development client of your development system, and you can thus in the Project Explorer pane view a hierarchy of objects that might look like the below. At the very top of the hierarchy, we see the 3 letter identifier of your development system, the client number (always 080), your CB* user and language.



To view all deprecated objects, you can call a dedicated menu tree. To do so, right click (1) the top most node of the hierarchy (the one that corresponds to your system), then select New and select the ABAP Repository Tree menu (2). This will open a pop-up from which you will select the Deprecated Objects entry.



Lastly, note that independent of this menu tree, you can check the state of an object, by looking at into the API State tab in the Properties view. You can find more information on the stability contracts (C0, C1, C2) in the [SAP help here](#).

The screenshot displays the SAP IDE interface. On the left, the Project Explorer shows a tree of objects, with `I_MAINTENANCEPLANDATA` and `I_BUSINESSUSER` highlighted. The main area shows the Properties view for these objects, with the 'API State' tab selected. The 'API State' tab is highlighted with a red box in the Project Explorer and the Properties view. The Properties view for `I_MAINTENANCEPLANDATA` shows the following details:

- Extend (Contract C0):** Contract C0 cannot be set.
- Use System-Internally (Contract C1):** Release State: **Deprecated** (highlighted with a red box).
- Use as Remote API (Contract C2):** Contract C2 cannot be set.
- API Catalog Assignments:** The object is not assigned to any API catalog.

The Properties view for `I_BUSINESSUSER` shows the following details:

- Extend (Contract C0):** Contract C0 cannot be set.
- Use System-Internally (Contract C1):** Release State: **Deprecated** (highlighted with a red box); Use in Cloud Development: **Yes** (highlighted with a red box); Use in Key User Apps: **Yes** (highlighted with a red box).
- Use as Remote API (Contract C2):** Contract C2 cannot be set.
- API Catalog Assignments:** The object is not assigned to any API catalog.

CDS Deprecations In Practice

Up until now, we have glossed over deprecations in general. In this section here, we will focus in detail on CDS deprecations.

When dealing with -CDS- deprecations, the kind and extent of changes that you will need to perform will depend on the kind of deprecation you are looking at (i.e one or more deprecated CDS fields within a still released CDS entity or/and an entire deprecated CDS entity) and the extent to which you have made use of the deprecated object(s) (do you have a complex stack of CDSs, was your custom CDS exposed as an API, was it used in a custom analytical query, in an embedded SAC story, etc...).

Whilst it would be impossible to cover each and every kind of scenario affected by a CDS related deprecation, we will, via two uses cases, propose a workflow to process the deprecation of CDS entities and CDS elements, when exposed as external APIs or used within a business user reporting capabilities (query report and eSAC story).

Whatever the case may be, planning is key! Below are a few bullet points that you may want to consider, before diving head first into changing things!

- Is your object used productively?
 - The more objects you have to change, the longer the change and testing process will be. You have to know that once you start to change existing objects in the development client, you can no longer cater to changes for production to happen in parallel (e.g a business user suddenly realises that a calculated field is providing incorrect results and wants it changed),
 - If the object is already available in the production client and actively used, then extra care must be taken to ensure the changes will not cause regressions and careful before/after change comparisons made at every step of the change process,

- If you are not an expert in the data or area (finance, supply chain..) being manipulated/affected by the change, ask an informed business user to validate that at the end of the changes the data presented is still correct/as expected (especially if you are for example using things such as calculation elements or calculated measures),
- Plan with the impacted people / systems the time at which changes will be pushed into production to monitor for potential errors that 'slipped through the cracks',
- To avoid impacting business users more than necessary, can you plan these changes in concert with other changes (e.g. incorporate a backlog of requested changes, rolling out a new functionality, re-design of business roles...) ,
- How extensively is your object used?
 - Make sure you have 100% visibility and knowledge of how the objects you are about to change are used (as we will see later on, there is room for omissions to occur, that will leave you with objects that do not function - correctly or at all - anymore. For example you could change a Custom CDS, but forget to change the associated eSAC story that uses that custom CDS),
 - Is it easier to change your existing objects, or is it simpler to create brand new objects?
- Is your Custom CDS exposed as an external API?
 - Are the changes you are going to make, going to affect the 3rd party application(s) that call your API? For example, are you going to remove or add or amend (change data format) or rename fields? If, yes, then clearly you need to plan and execute these changes not just in your SAP S/4HANA Cloud system, but also in the 3rd party applications so that they can parse incoming payloads,
 - If you have several 3rd party applications, will you able to execute the changes in all 3rd party systems at the same time? If no, what does that mean? Do you need to delay the roll out of the changes until such a time when all 3rd party applications can be changed, or does that mean that you need to create a new API so as to have some 3rd party application on one API, and some remain on the still unchanged API a little longer? But if you create a new API, then the URL endpoint will also change and thus also bring about additional changes,
- Is your change going to require some change management or re-training of business users?

- Are your changes susceptible to causing issues or confusion with your business users (for example, will the field names shown on the UI change? How will the users react to this? Are previously saved report variants not going to work anymore, will you need to update user manuals...). Be ahead of this and plan on mitigating any impact to the business users - at a minimum, communicate,
- How often is your object used?
 - You have to know and plan for the day / time when you will roll out the changes to your production client. I.e do the business users use this report multiple times during a day, every day vs maybe some report that is only required 2-3 days at the end of the month? Or is the custom CDS exposed as an external API used every 5 minutes vs used once at the end of the day?
 - Inform the business users / technical integration users of incoming changes.

Introduction to the use case 1 - Analytical CDS

Setting the scene

To work through a deprecation process, we have devised the following hypothetical storyline:

In our demo organisation we are using Sales and Distribution sales orders to sell products, but we have also been using Customer Material info records to store the Customer Material ID to facilitate the customer ordering process, as well as order data such as minimum deliver quantities, delivery tolerances, etc.. Having received complaints and refund requests due to incorrect product deliveries, we were able to identify that the root cause of the problem was that the Customer Material info records we had created incorrectly linked the customer material ID to our products. We then decided to create a reporting capability that would allow us, after correction of our Customer Material info records, to identify these erroneous orders including order information. Sensing this was not a one time need, we decided to provide business users with a self-serve reporting capability they could use as and when they needed it.

The screenshot shows the SAP Customer Material info record for Hypercom AG. The product is 'Trad.Good 12,Reorder Point,Reg.Trad (TG12)' and the customer material number is 'HANACT1'. A red callout box points to the product name with the text 'The correct product should have been TG11!'. Below the header, a table lists the items:

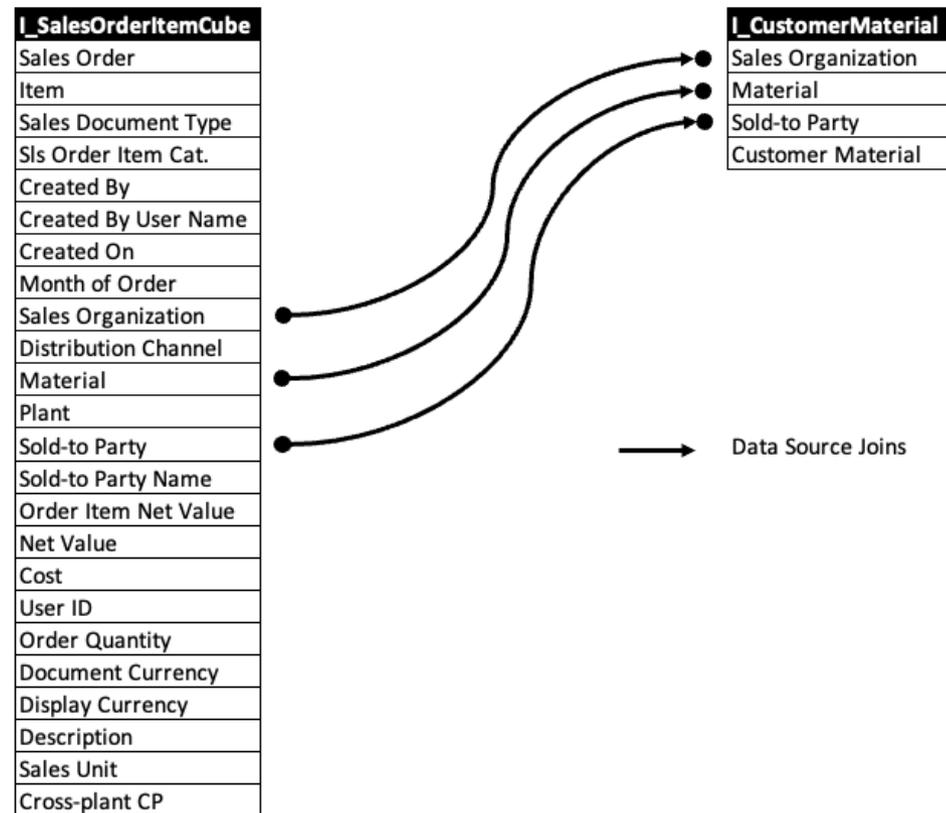
Item	Material	Order Quantity	Un	S	Item Description	Customer Material Number
10	TG12		1	PC	HAWA 12, Bestellpunkt, normaler ...	HANACT1

Orders were incorrectly created as a result of wrong Product to Customer Material associations.

At the time of the requirement, based on the intended end-use of this reporting (analytical query and embedded SAC story) a custom CDS, with the scenario Analytical Cube was created, by joining together two CDSs that provided all the required data. These CDSs were:

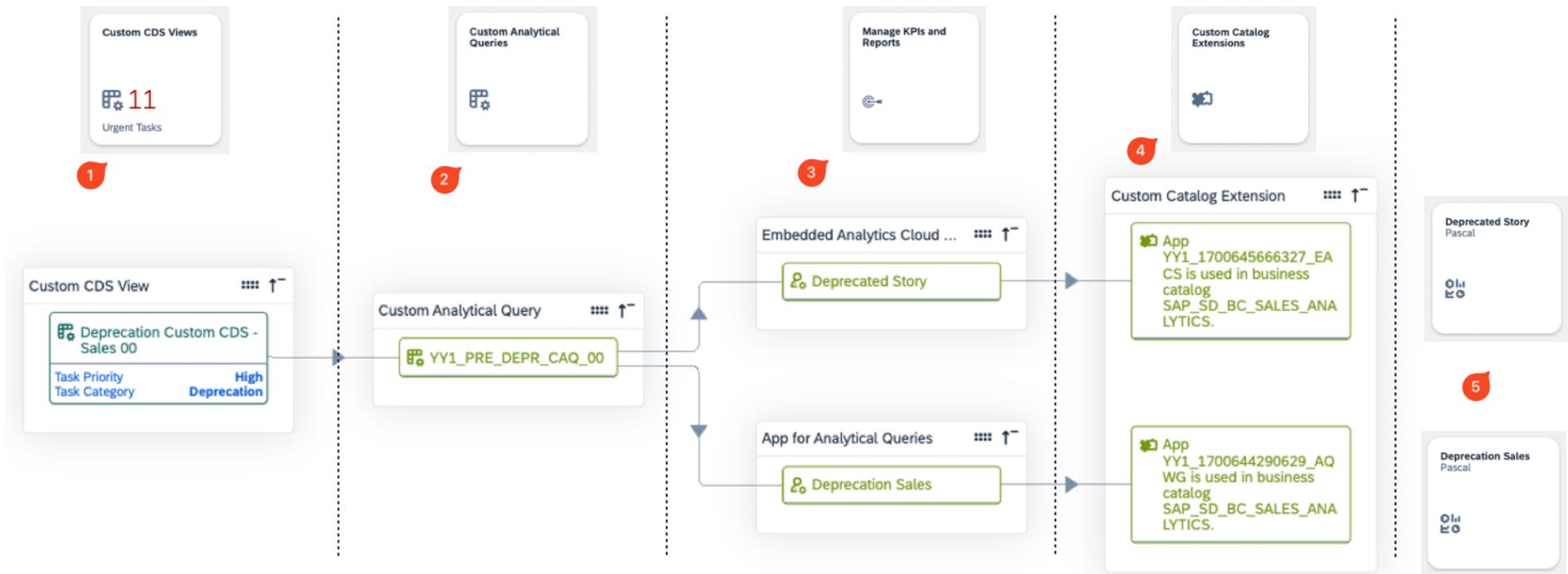
- **I_SalesOrderItemCube** - Sales Order Item - Cube, which provided us with sales order item data, and
- **I_CustomerMaterial** - Customer Material Information, which provided us with the customer material ID.

The diagram below shows which fields were selected as Elements to be on the reporting cube and/or used to join the two CDSs together.



The diagram below, shows the workflow we adopted to produce the required reporting capability:

1. We used the Custom CDS views app to create the base of our reporting pyramid
2. We used the Custom Analytical Queries app to create a query
3. We used the Managed KPIs and Reports app to, based on our previously create analytical query, create a multi dimensional report and an eSAC story
4. Lastly we published the report and eSAC story to a business catalogs
5. The report and eSAC story were exposed as tiles on the users' launchpads.



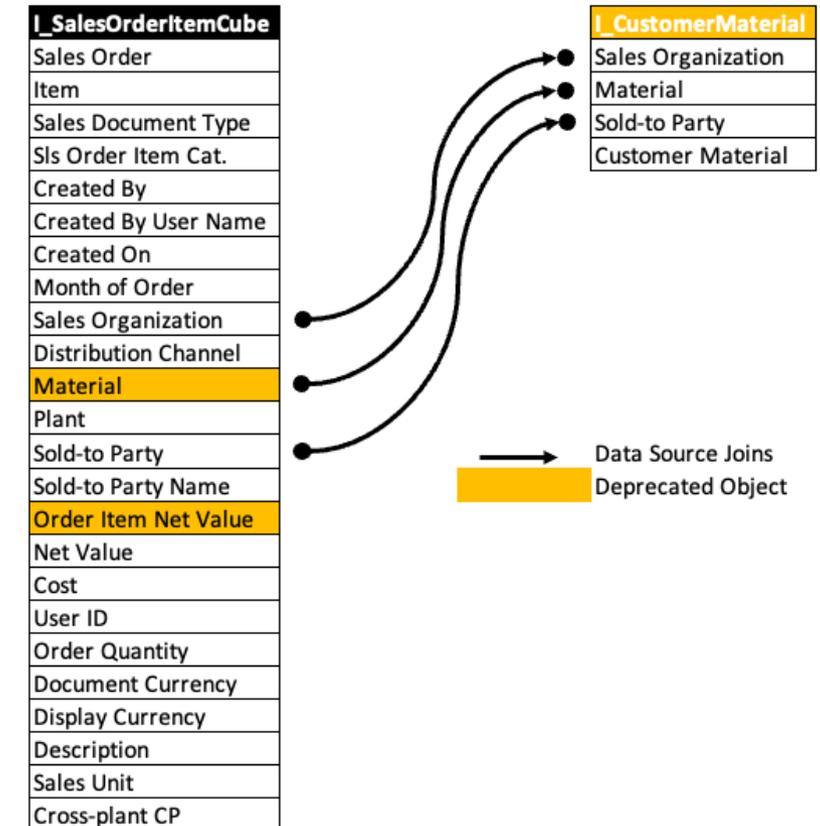
Processing the CDS Deprecation

The deprecation announcement

Having in the previous chapter set the scene for our scenario, we now fast forward to a hypothetical upgrade of SAP S/4HANA Cloud, public edition and we have taken the time to take stock of the various announcements that go with such an event, and in particular the advertised depreciations. Being responsible for 'Analytics and Reporting' we have with interest looked at the depreciations of CDS entities and CDS fields. We have done our initial due diligence:

- We have looked at the What's New viewer and filtered the data to look at depreciations of CDSs only,
- We have looked at the RASD tool, to get a usage specific view of how the announced depreciations directly affect us, based on our actual usage of the deprecated CDS announcements, and it has advised us that our custom CDS View `YY1_PRE_DEPR_00` is impacted by a deprecation
- We have an understanding of the Deprecation Policy of CDSs, and we know that the Deprecated CDS will still reside in our system for at least the next 12 months - this gives us time to plan and execute this deprecation.
- We have decided that we will change the existing objects, rather than create new ones.

The diagram on the right, for the sake of setting the scene, shows how the deprecation announcements affect our Custom CDS. In one CDS entity, we have two deprecated fields, and the other CDS entity we use is completely deprecated.



Understanding the size of the rework

We know the name of the custom CDS that is impacted by the deprecation announcements, but as we have quite a substantial reporting stack, just knowing the name of the CDS does not allow us to judge how much rework this will require, so to understand this we need to get on the system, plus we also want to double check that we have not missed anything.

We launch the Custom CDS view application and we amend the filters, so as to output only those custom CDS entities that are affected by a deprecation. We quickly identify the Custom CDS(s) in our area of responsibility, and in particular the one that caught our attention when looking at the RASD tool.

Label: Name: Scenario: Editing Status: Protection: Last Changed By: Last Changed On:

Task Priority: Exported: Rework: Task Category: Created By:

Custom CDS Views (3) Create Copy Delete ⚙️

Name	Label	Scenario	Status	Task Priority	Rework	Task Category
<input type="radio"/> YY1_PRE_API_00	Deprecation API Sales CDS - 00	External API	Published	⊗ High	☑️	Deprecation
<input type="radio"/> YY1_PRE_DEPR_00	Deprecation Custom CDS - Sales 00	Analytical Cube	Published	⊗ High	☑️	Deprecation
<input type="radio"/> YY1_PRE_DEPR_01	Deprecation Custom CDS - Sales 01	Analytical Cube	Published	⊗ High	☑️	Deprecation

We click on the line, that corresponds to our Custom CDS, to get more details about it.

As soon as we get into the detail screen of the custom CDS, there are already a number of call-outs that attract our attention.

We have in the top left an indication that Dependencies exist and Compatible changes are allowed. This means that this custom CDS is used by another object and we can only make changes that will not affect or break the object that is using our custom CDS. For example, you would not want to be able to just delete a field from your custom CDS, if it was used by an analytical query!

Clicking on the hyperlink titled (Compatible Changes Allowed), will open a pop-up window showing you the actual name and object type of the object(s) that use the custom CDS

Further down, in the list of Data Sources, we can see that the SAP Standard CDS `I_CustomerMaterial` has the status of Deprecated. We also note that since this CDS is used in an association, we will also need to re-visit the join conditions.

Dependent Objects			
Dependencies (1)			
View Name	Dependent Object	Draft Indicator	Publish Indicator
YY1_PRE_DEPR_CAQ_00	Custom Analytical Query	No	Yes

Deprecation Custom CDS - Sales 00

Dependencies: 1 (Compatible Changes Allowed)

Publishing Successful

Name: YY1_PRE_DEPR_00
 Representative Key: Not required
 Scenario: Analytical Cube

Published By: [User]
 Published On: 11/22/2023, 07:40:40 PM
 Protection: Protected

Created By: [User]
 Created On: 11/21/2023, 11:22:47 PM
 Status: Published

Data Sources (2)

Name	Alias	Type	Cardinality	Tasks	Status	Parameters	Join Condition
I_SalesOrderItemCube	<code>I_SalesOrderItemCube</code>	Primary Datasource			Released	<input checked="" type="checkbox"/>	
I_CustomerMaterial	<code>I_CustomerMaterial</code>	Associated Datasource	Zero or One [0..1]		Deprecated		↗

Buttons: Edit, Preview, Data Browser, Inventory

We can also click on the join (chain) icon to see how our two data sources are joined together. We see that three fields are used to join the data sources, and one of those fields is the field `Material`, which is deprecated but note that this is not indicated. We make a note that we will need to correct this as well.

Define Join Conditions

Name: I_CustomerMaterial Type: Associated Datasource

Alias: _I_CustomerMaterial Cardinality: Zero or One [0..1]

Join Conditions (3)

<input type="checkbox"/>	D	Type	Operator	Value Type	Value
<input type="checkbox"/>	Customer	CHAR (10)	Equal	Field	SoldToParty
<input type="checkbox"/>	Material	CHAR (40)	Equal	Field	Material
<input type="checkbox"/>	SalesOrganization	CHAR (4)	Equal	Field	SalesOrga...

Deprecated Field

Join Condition

Join icon

We then move on to the Tasks tab, and in the Rework Tasks section, get more prescriptive information, clearly telling us what is deprecated as well as the successor object(s). Note that in this case we have rework tasks that pertain to CDS elements, as well as a complete CDS entity.

Another thing that is important, is that whilst we know what has been deprecated, the message is not telling us where the deprecated object(s) are used! I.e are we using deprecated objects as a join condition and/or filter condition and/or a displayed element and/or a calculation! As we shall see later on this is important, as not all checks are automated!

Deprecation Custom CDS - Sales 00
 Dependencies: 1 (Compatible Changes Allowed) Publishing Successful

Name: YY1_PRE_DEPR_00
 Representative Key: Not required
 Scenario: Analytical Cube

Published By: [User]
 Published On: 11/22/2023, 07:40:40 PM
 Protection: Protected

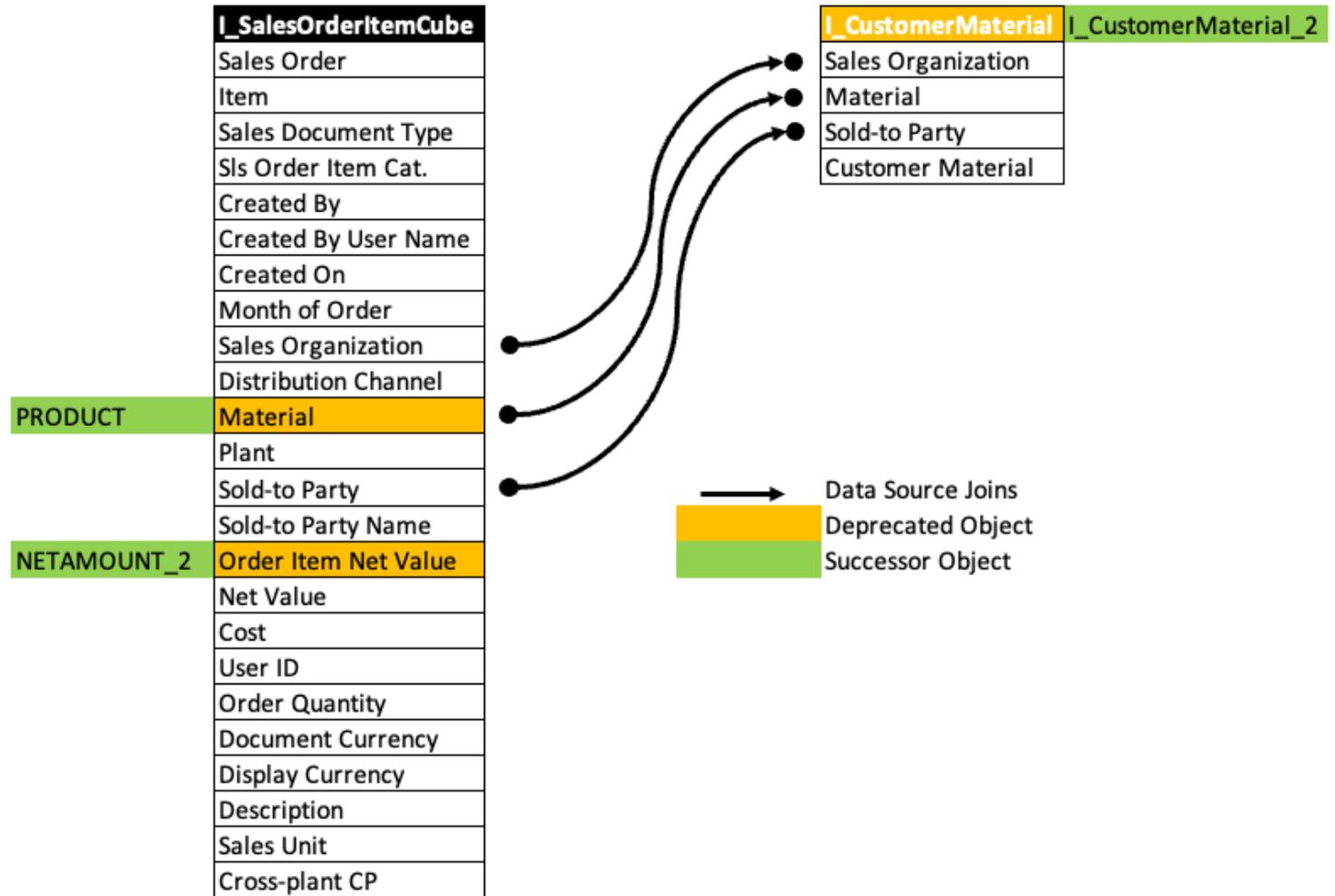
Created By: [User]
 Created On: 11/21/2023, 11:22:47 PM
 Status: Published

Rework Tasks (3)

Task Priority	Task Category	Description	Registered On
High	Deprecation	Element Material of CDS view I_SalesOrderItemCube is deprecated. Use element PRODUCT instead.	11/21/2023, 11:23:57 PM
High	Deprecation	Element NetAmount of CDS view I_SalesOrderItemCube is deprecated. Use element NETAMOUNT_2 instead.	11/21/2023, 11:23:57 PM
High	Deprecation	CDS Entity I_CUSTOMERMATERIAL is deprecated. Use CDS Entity I_CUSTOMERMATERIAL_2 instead.	11/22/2023, 04:59:02 PM

Buttons: Edit, Preview, Data Browser, Inventory

On the basis of the information found in the Tasks tab, we can update our Custom CDS data schema, and add to it the Successor information we have now confirmed. Next to the deprecated objects, we have added the successor information.



With respect to the deprecated CDS entity, it could be that the successor CDS entity is different, either in terms of capabilities or in terms of fields (there could be more or less fields and whilst rare the format of fields could have changed), so it is important that you also analyse the differences between these two CDS entities. As we covered earlier, you could do this by checking the View Browser app in your system, or you could check the [SAP Business Accelerator Hub](#), or you could use the Excel tool we have purposefully created for this purpose. You can [download it here](#). We used this tool to compare the (CustomerMaterial) CDS entities. We can see that the successor CDS (on the right) does not support the Data Extraction scenario, but this does not affect us, and we can also see that the field we were using - `MaterialByCustomer` - is still present in the successor CDS.

Supported Capabilities

End User Capability Name

Data Source in SQL Select	SQL_DATA_SOURCE	● SQL_DATA_SOURCE
Data Source for Defining CDS Entities	CDS_MODELING_DATA_SOURCE	● CDS_MODELING_DATA_SOURCE
Association Target for Defining CDS Entities	CDS_MODELING_ASSOCIATION_TARGET	● CDS_MODELING_ASSOCIATION_TARGET
Analytical Dimension	ANALYTICAL_DIMENSION	● ANALYTICAL_DIMENSION
Data Source for Data Extraction	EXTRACTION_DATA_SOURCE	●

Deprecated CDS					Successor CDS					
Field Name	Description	Field State	Successor	Data Type	Field Length	Description	Field State	Successor	Data Type	Field Length
SalesOrganization	Sales Organization			CHAR	4	● Sales Organization			CHAR	4
DistributionChannel	Distribution Channel			CHAR	2	● Distribution Channel			CHAR	2
Customer	Customer number			CHAR	10	● Customer number			CHAR	10
Material	Material Number			CHAR	40	●				
Product	Product Number			CHAR	40	● Product Number			CHAR	40
SortField	Sort Field			CHAR	10	●				
Etag	Sort Field			CHAR	10	●				
MaterialByCustomer	Material Number Used by Customer			CHAR	35	● Material Number Used by Customer			CHAR	35
MaterialDescriptionByCustomer	Customer Description of Material			CHAR	40	● Customer Description of Material			CHAR	40
Plant	Plant (Own or External)			CHAR	4	● Plant (Own or External)			CHAR	4

We can also look at the Elements tab, which provides us with further essential information. We can see:

1. A clear indication of those fields that are deprecated, as well as an indication of the successor field that we should be using
2. We also see a field that is part of the deprecated CDS entity, but the field itself is not seemingly deprecated and thus there is no special indication for it here
3. We also see that a calculated field is present (this is a custom data element that we have added), but we see no indication of deprecation against it, even though it uses a deprecated field.

Custom CDS View Details

Deprecation Custom CDS - Sales 00

Dependencies: 1 (Compatible Changes Allowed)

Information

There is additional information for this element

Label: Material

Alias: Material

Deprecated Element

The field is deprecated and should not be used. Please use the successor instead.

Successor: Product

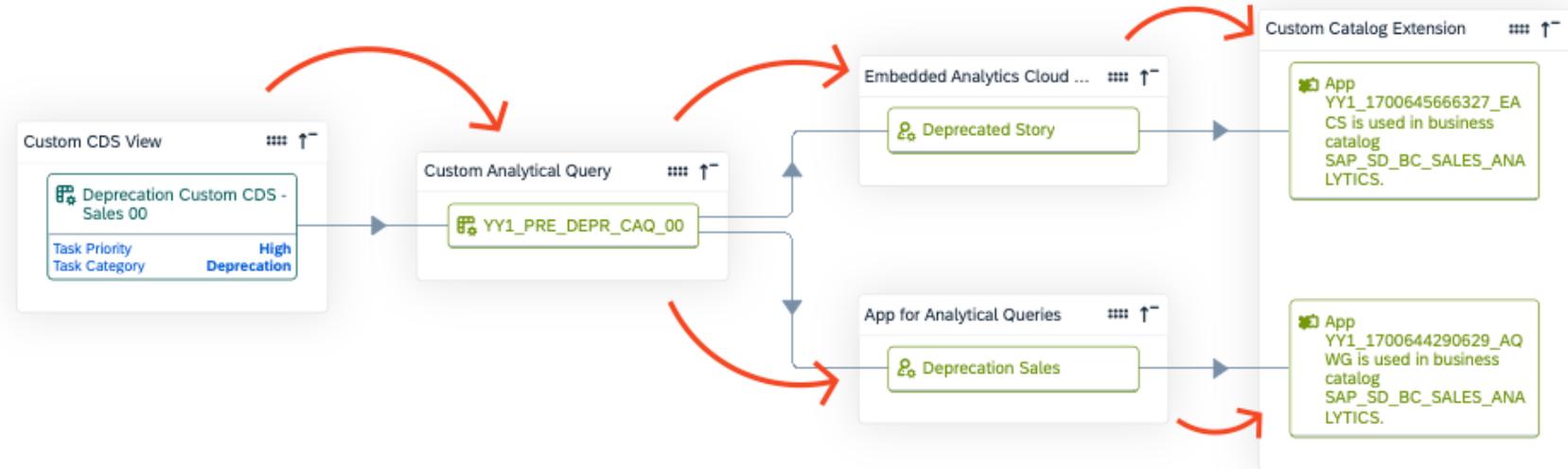
Key	Alias	Type	Path	Successor
OFF	Material	CHAR (40)	I_SalesOrderItemCube.Material	Product
OFF	Plant	CHAR (4)	I_SalesOrderItemCube.Plant	
OFF	SoldToParty	CHAR (10)	I_SalesOrderItemCube.SoldToParty	
OFF	SoldToPartyName	CHAR (80)	I_SalesOrderItemCube.SoldToPartyName	
OFF	NetAmount	CURR (15,2)	I_SalesOrderItemCube.NetAmount	NetAmount_2
OFF	NetAmountInDisplayCurrency	CURR (19,2)	I_SalesOrderItemCube.NetAmountInDisplayCurrency	Net Value
OFF	CostAmount	CURR (13,2)	I_SalesOrderItemCube.CostAmount	Cost
OFF	UserID	CHAR (12)	I_SalesOrderItemCube._CreatedByUser.UserID	User ID
OFF	OrderQuantity	QUAN (15,3)	I_SalesOrderItemCube.OrderQuantity	Order Quantity
OFF	TransactionCurrency	CUKY (5)	I_SalesOrderItemCube.TransactionCurrency	Document Currency
OFF	DisplayCurrency	CUKY (5)	I_SalesOrderItemCube.DisplayCurrency	Display Currency
OFF	UserDescription	CHAR (80)	I_SalesOrderItemCube._CreatedByUser.UserDescription	Description
OFF	OrderQuantityUnit	UNIT (3)	I_SalesOrderItemCube.OrderQuantityUnit	Sales Unit
OFF	MaterialByCustomer	CHAR (35)	_I_CustomerMaterial.MaterialByCustomer	Customer Material
OFF	CrossPlantConfigurableProduct	CHAR (40)	I_SalesOrderItemCube._Product.CrossPlantConfigurableProduct	Cross-plant CP
OFF	Y_Difference			Amount Difference

This last point is important, because if we click on the Calculated field icon we can reveal the formula used to calculate this field and notice that the formula uses one field (NetAmount), that is deprecated ! This use will not be flagged by any check, so make sure you check. For this field to continue to function correctly, we will need to replace it with the successor also.

The screenshot displays a software interface with a navigation bar at the top containing icons for Data Sources, Parameters, Elements, Element Properties, Filter, Log, Tasks, and Next Steps. Below the navigation bar is a section titled "Elements (38)" with a search bar and a table of elements. The table has columns for Key, Alias, Type, Path, Label, Calculation, Task, and Successor. The "NetAmount" row is highlighted, and a "Display Calculation" dialog box is open over it. The dialog shows the formula: $1 \text{ I_SalesOrderItemCube.NetAmount} - \text{I_SalesOrderItemCube.CostAmount}$. The "NetAmount" field in the formula is highlighted with a red box. A red arrow points from this box to a warning icon in the "Task" column of the "NetAmount" row. Another red arrow points from the warning icon to a "NetAmount_2" entry in the "Successor" column.

Key	Alias	Type	Path	Label	Calculation	Task	Successor
<input type="checkbox"/>	<input type="checkbox"/>	SoldToPartyName	CHAR (80)	I_SalesOrderItemCube.SoldToPartyName	Sold-to Party Name		
<input type="checkbox"/>	<input type="checkbox"/>	NetAmount	CURR (15,2)	I_SalesOrderItemCube.NetAmount	Order Item Net Value		NetAmount_2
<input type="checkbox"/>	<input type="checkbox"/>	NetAmountInDisplayC...	CURR (19,2)				
<input type="checkbox"/>	<input type="checkbox"/>	CostAmount	CURR (13,2)				
<input type="checkbox"/>	<input type="checkbox"/>	UserID	CHAR (12)				
<input type="checkbox"/>	<input type="checkbox"/>	OrderQuantity	QUAN (15,3)				
<input type="checkbox"/>	<input type="checkbox"/>	TransactionCurrency	CUKY (5)				

In the bottom right of the Custom CDS detail screen, we also have some buttons that offer navigation options. Let's first look at the one name Inventory. Clicking it, will open a new browser window and open the Extensibility Inventory application, but specifically as it applies to our custom CDS. This will show us a hierarchy, depicting how and where our custom CDS is used. In our case, it shows us a representation such as the one below (your rendition of it may differ depending on the display options you have set for yourself). In our case we have child → parent representation, showing that our custom CDS is used by a Custom Analytical Query, that is used by a Query report and eSAC story, and each are published to a business catalog (custom catalog extension).



We also have in the top right a button named Download Item Dependencies that allows us to download these dependencies in tabular format.

Download Item Dependencies

Once your download is completed, you will have a zip archive containing two files. One starting with the word

'depending' and one starting with the words 'used_by'. If we look at the latter file that pertains to our scenario we have, the below information, which really provides us with a roadmap of where we need to make changes.

Item ID	Item Type	Item Description	Rework Priority	Rework Category	API State	Used By Item ID	Used By Item Type	Used By Item Description
YY1_PRE_DEPR_00	Custom CDS View	Deprecation Custom CDS - Sales 00	High	Deprecation		YY1_PRE_DEPR_CAQ_00	Custom Analytical Query	YY1_PRE_DEPR_CAQ_00
YY1_PRE_DEPR_CAQ_00	Custom Analytical Query	YY1_PRE_DEPR_CAQ_00				YY1_1700644290629_AQWG	App for Analytical Queries	Deprecation Sales
YY1_PRE_DEPR_CAQ_00	Custom Analytical Query	YY1_PRE_DEPR_CAQ_00				YY1_B48050026295D5AE94D6472BFAAAC90A	Embedded Analytics Cloud Story	Deprecated Story
YY1_1700644290629_AQWG	App for Analytical Queries	Deprecation Sales				YY1_UXBV4TEPIYPN5IVC43GPNMCMWE	Custom Catalog Extension	App YY1_1700644290629_AQWG is used in business catalog SAP_SD_BC_SALES_ANALYTICS.
YY1_B48050026295D5AE94D6472BFAAAC90A	Embedded Analytics Cloud Story	Deprecated Story				YY1_UXBV4TEPIYPN5IVDZOGIABMWI	Custom Catalog Extension	App YY1_1700645666327_EACS is used in business catalog SAP_SD_BC_SALES_ANALYTICS.

Custom CDS Views

11

Urgent Tasks

Custom Analytical Queries

Manage KPIs and Reports

SAP Manage KPIs and Reports

274 KPI 222 Reports 4 Multidimensional Reports 1 Stories

Standard Filtered By: None

Items (1)

Title	Subtitle	Data Source	Application ID
Deprecation Sales	Pascal	YY1_PRE_DEPR_CAQ_00	YY1_1700644290629_AQWG

SAP Manage KPIs and Reports

274 KPI 222 Reports 4 Multidimensional Reports 1 Stories

Standard

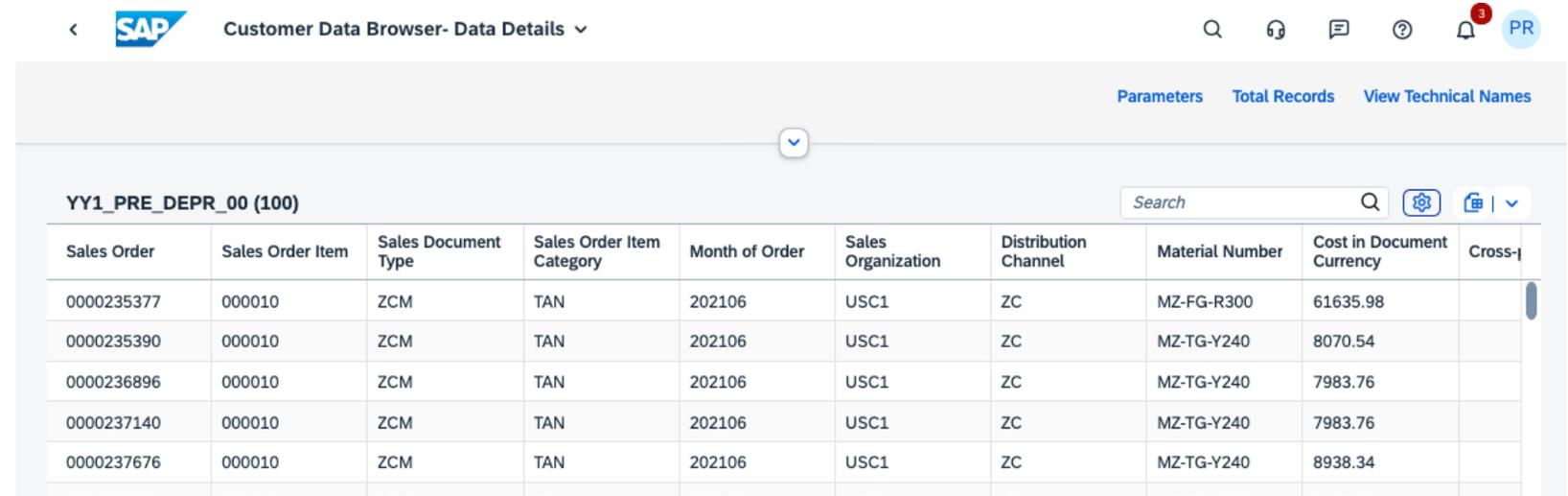
Source: YY1_PRE_ Tags: All CDS View:

Embedded Stories (1) Stand-alone Stories

Source	Story Name	Description	Data Source	Tags	Job Status	Applications
	Deprecated Story		YY1_PRE_DEPR_CAQ_00		Export Successful	1

If we now go back to the custom CDS detail screen, we can try the former button named Data Browser. Clicking this button will also open a new browser window, in which the app Customer Data Browser will be launched. This app will show in an tabular like presentation layer the data content of our custom CDS, in tabular format.

If you are like us in this scenario making changes to the existing objects and depending on the amount of data provided by your custom CDS, I would recommend that before you start making changes, you make a complete download of this data (elements and content), or at least download a good sample of data that is later going to allow you to verify that you have processed the depreciation correctly.



The screenshot displays the SAP Customer Data Browser interface. At the top, there is a navigation bar with the SAP logo, the title "Customer Data Browser- Data Details", and several utility icons (search, help, chat, question mark, notification, and profile). Below the navigation bar, there are tabs for "Parameters", "Total Records", and "View Technical Names". A search bar is located above the table, and a table with 10 columns is displayed below. The table contains data for sales order items, including sales order numbers, item numbers, document types, categories, months, organizations, channels, material numbers, and costs.

Sales Order	Sales Order Item	Sales Document Type	Sales Order Item Category	Month of Order	Sales Organization	Distribution Channel	Material Number	Cost in Document Currency	Cross-I
0000235377	000010	ZCM	TAN	202106	USC1	ZC	MZ-FG-R300	61635.98	
0000235390	000010	ZCM	TAN	202106	USC1	ZC	MZ-TG-Y240	8070.54	
0000236896	000010	ZCM	TAN	202106	USC1	ZC	MZ-TG-Y240	7983.76	
0000237140	000010	ZCM	TAN	202106	USC1	ZC	MZ-TG-Y240	7983.76	
0000237676	000010	ZCM	TAN	202106	USC1	ZC	MZ-TG-Y240	8938.34	

Your download should at the very least include ALL the deprecated fields and any other field(s) that makes use of the deprecated fields (for example a calculated element). You will need this because once you have removed the deprecated fields (remember that is why we are doing all this!) You will need to be able to verify that your custom CDS is still providing all the data that it, as a data source needs to provide to the rest of the reporting objects (query report, eSAC, etc...). In our case, we have on the right highlighted the fields that are deprecated and that we will ultimately remove from our custom CDS. So once the deprecation is processed, we need to be able to verify that the successor objects, and the use we make of them do not present a regression.

The screenshot shows the SAP Customer Data Browser interface. The table title is 'YY1_PRE_DEPR_00 (100)'. The table has the following columns: SALESORDER, SALESORDERITEM, SALESORDERTYPE, SALESORDERITEMCATEGORY, SALESORDERDATEYEARMONTH, MATERIAL, NETAMOUNT, MATERIALBYCUSTOMER, and SALESORGANIZATION. The 'MATERIAL' and 'NETAMOUNT' columns are highlighted with a red box and labeled with a red callout '_SalesOrderItemCube'. The 'MATERIALBYCUSTOMER' and 'SALESORGANIZATION' columns are highlighted with a green box and labeled with a green callout '_CustomerMaterial'. The table contains five rows of data.

SALESORDER	SALESORDERITEM	SALESORDERTYPE	SALESORDERITEMCATEGORY	SALESORDERDATEYEARMONTH	MATERIAL	NETAMOUNT	MATERIALBYCUSTOMER	SALESORGANIZATION
0000386440	000130	TA	TAN	202311	MR0011	916.30	CUST-MR0011	1710
0000385825	000010	TA	TAN	202311	MR0011	2200.00	CUST-MR0011	1710
0000385826	000010	TA	TAN	202311	MR0011	1100.00	CUST-MR0011	1710
0000385512	000010	TA	TAN	202311	MR0011	2200.00	CUST-MR0011	1710
0000385526	000010	TA	TAN	202311	MR0011	2200.00	CUST-MR0011	1710

In a similar vain, we would recommend that you also make downloads and/or take screenshots of the query report, eSAC story and any other ways you are making use of the custom CDS to make sure that any filters, columns, calculations you are doing still function correctly after the changes are made. In the event where your objects have already been transported to the Test and/or Production system, you will be able to have a reference with respect to filters and layouts, but the data will be specific to each system, and you do not want to be in a position where your business users, in production let you know that something does not look right !

As an example, we have below added screenshots of both the query report, as well as the eSAC story and highlighted where use of the deprecated elements is made. We will skip showing this step, but we have made copies of the data served and layout where applicable, from the customer data browser, the query report and the eSAC story.

SAP Deprecation CAQ - Sales 00

Standard *

Exchange Rate Type: M x Display Currency: EUR (Eur... x) Sales Organization: 1010 (Dom. Sales ... x) Sold-to Party: Material: Customer Material:

Adapt Filters (3) Go

SAP Depreciated Story

Standard *

File Edit Tools Display { } Page 1 < 1/1 >

Depreciated Sales Demo

YY: PRE_DEPR_CAQ_00

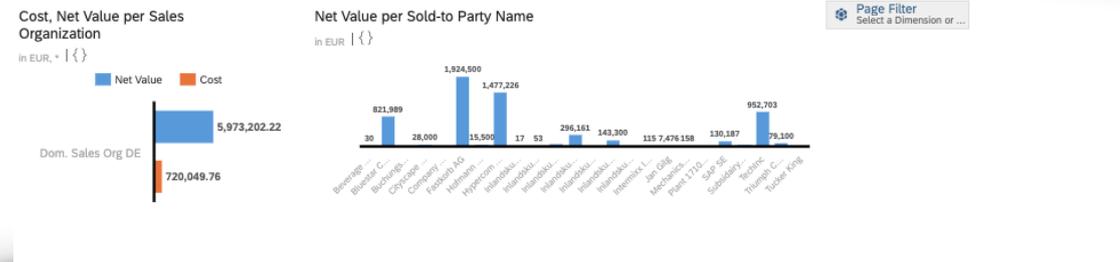
Order Item Net Value

Sales Document Type	Sales Order	Sold-to Party Name	Item	Material	Plant	Customer Material	Order Item Net Value	Cost	Amount Difference	Net Value
SO01	217596	Fastkorb AG	1	P002	1010	#	EUR1,924,500.00	EUR0.00	EUR1,924,500.00	EUR1,924,500.00
OR	385517	Hypercom AG	10	AVC_RBT_ROBOT2	1010	#	EUR995,000.00	EUR40,000.00	EUR955,000.00	EUR995,000.00
OR	386583	Inlandskunde DE 4	10	SM100	1010	#	EUR200,839.24	EUR0.00	EUR200,839.24	EUR200,839.24
SO01	36733	Techinc	2	P002	1010	#	EUR180,000.00	EUR0.00	EUR180,000.00	EUR180,000.00
OR	386456	Hypercom AG	20	FG130	1010	#	EUR109,989.00	EUR3,168.00	EUR106,821.00	EUR109,989.00
SO01	241953	Techinc	1	P002	1010	#	EUR104,300.00	EUR0.00	EUR104,300.00	EUR104,300.00

Data Analysis Graphical Display Query Information

<No Bookmark Loaded>

Sal...	Dist...	Month of...	Sold-to Party	Sales ...	Sales Order	Sold-to Party	Amount Difference	Net Value	Order Quantity	Margin
		12/2016	10100001	SO01	23212	Hypercom AC	23,400.00 EUR	23,400.00 EUR	0 AU	100.00
					32283	Hypercom AC	81.00 EUR	351.00 EUR	20 PC	23.08
		04/2017	10100001	OR	32284	Hypercom AC	162.00 EUR	702.00 EUR	40 PC	23.08
					32285	Hypercom AC	162.00 EUR	702.00 EUR	40 PC	23.08
							50,000.00 EUR	50,000.00 EUR	0 AU	100.00
		06/2017	10100002	SO01	36733	Techinc	180,000.00 EUR	180,000.00 EUR	0 AU	100.00
							20,000.00 EUR	20,000.00 EUR	0 AU	100.00
							20,000.00 EUR	20,000.00 EUR	0 PC	100.00
		07/2017	10100001	OR	39505	Hypercom AC	1.75 EUR	17.55 EUR	1 PC	9.97
					39509	Hypercom AC	1.75 EUR	17.55 EUR	1 PC	9.97
		08/2017	10100001	OR	41853	Hypercom AC	1.75 EUR	17.55 EUR	1 PC	9.97
					41860	Techinc	1.75 EUR	17.55 EUR	1 PC	9.97
					42526	Techinc	1,200.00 EUR	1,200.00 EUR	0 AU	100.00
		09/2017	10100002	SO01	43232	Techinc	42,000.00 EUR	42,000.00 EUR	0 AU	100.00
					44730	Techinc	42,000.00 EUR	42,000.00 EUR	0 AU	100.00
			10401010	SO03	44728	Buchungskre	0.00 EUR	1,000.00 EUR	0 AU	0.00
1010	10									



The Process

Now that we have done our analysis, we know what we need to change and where we need to change things, and we have also made backups of our data so we can make before change and after change comparisons at every step of the way.

Logically, one might think that we will start at the bottom and make our way up the stack. By that I mean that, one might think that it would be easy enough to remove the deprecated fields from the custom CDS, add the successor fields in the custom CDS, save and publish, then move on to the Custom Analytical Query, redo the wiring there, then move on to the eSAC story and so on. Unfortunately, in practice it does not work quite like this. I would also stress that some objects are hard coupled and some are loosely coupled. By that I mean that some CDS changes are checked/validated against the custom analytical query (hard coupled), but on the other hand you can remove a field in the custom analytical query irrespective of whether it is used in the eSAC story or not, without any error or validation message (hence also why we suggested making those data downloads for before/after comparisons!).

To illustrate this, let's for example go to our custom CDS into edit mode and try to remove one of the deprecated fields —> **Material**. As you can see, the Delete button remains invariably inaccessible - because deleting this field would be an incompatible change (as the field is used in the Custom Analytical Query). However, we can see that the Add option is available, so we can add - successor fields.

Deprecation Custom CDS - Sales 00
Dependencies: 1 (Compatible Changes Allowed) Draft

Navigation: Data Sources >> Parameters >> **Elements** >> Element Properties >> Filter >> Log >> Tasks More ▾

Elements (38) Add Delete ⚙️ ↺

<input type="checkbox"/>	Key	Alias	Type	Path	Label	Calculation	Task
<input type="checkbox"/>	OFF	SalesOrganization	CHAR (4)	I_SalesOrderItemCube.SalesOrganization	Sales Organization		
<input type="checkbox"/>	OFF	DistributionChannel	CHAR (2)	I_SalesOrderItemCube.DistributionChannel	Distribution Chan...		
<input checked="" type="checkbox"/>	OFF	Material	CHAR (40)	I_SalesOrderItemCube.Material	Material		⚠️
<input type="checkbox"/>	OFF	Plant	CHAR (4)	I_SalesOrderItemCube.Plant	Plant		

5 Publish Check Cancel Preview Data Browser Inventory

So at this point you might think that it feels a little like the dog chasing his own tail scenario, and it is. This means that however we look at this, we will need to do something in the Custom CDS app, then process changes in the custom Analytical query app and there do all the changes that prevent us from deleting the deprecated fields from the custom CDS app, then go back to the custom CDS app to finally remove the deprecated fields. Truth be told, whilst we will not present it here, we could also have started the changes from the Custom Analytical Query app (by removing the deprecated fields from display), but this would not have spared us the same round trip of having to come back to the custom analytical query (to add the successor fields). We prefer to use the workflow we will go through in this document as it allows for us to better check incrementally that our changes will not present any regression.

Step 1 - Amending the Custom CDS and adding the successor elements

We start by going to the Custom CDS app and switch to edit mode in our custom CDS. Here we will add all the Successor objects, that is the customer CDS entity `I_CustomerMaterial_2` and the successor fields `Product` and `NetAmount_2` from the CDS `I_SalesOrderItemCube`.

We cannot delete or simply amend the name of the deprecated CDS entity at this moment as it is used.

The screenshot displays the SAP Custom CDS configuration interface. At the top, navigation tabs include Data Sources, Parameters, Elements, Element Properties, Filter, Log, and Tasks. The 'Data Sources (2)' section contains a table with the following data:

Name	Alias	Type	Cardinality	Tasks	Status	Parameters	Join Condition
<code>I_SalesOrderItemCube</code>	<code>I_SalesOrderItemCube</code>	Primary Datasource			Released	<input checked="" type="checkbox"/>	
<code>I_CustomerMaterial</code>	<code>_I_CustomerMaterial</code>	Associated Datasource	Zero or One [0..1]		Deprecated		

Below this is the 'Define Join Conditions' section. It shows the configuration for the selected data source:

- Name: `I_CustomerMaterial_2`
- Alias: `_I_CustomerMaterial_2`
- Type: Associated Datasource
- Cardinality: Zero or One [0..1]

The 'Join Conditions (3)' section shows a table with the following data:

Data Source Field	Type	Operator	Value Type	Value
<code>SalesOrganization</code>	CHAR (4)	Equal	Field	<code>I_SalesOrderItemCube.S...</code>
<code>Product</code>	CHAR (40)	Equal	Field	<code>I_SalesOrderItemCube.Product</code>
<code>Customer</code>	CHAR (10)	Equal	Field	<code>I_SalesOrderItemCube.S...</code>

Red arrows in the original image point from the 'Delete' button in the Data Sources table to the 'I_CustomerMaterial' row, and from the 'Join Conditions' table to the 'Product' row.

We will add the successor CDS entity, taking care of not using the field `Material` in the join, as we earlier noted that it was used, and instead use the successor field which is `Product`.

At this point we will have two associations to the primary data source (one association with the deprecated CDS, and one association to the successor CDS). We will revisit this later, when we are able to delete the join to the deprecated CDS entity.

Name	Alias	Type	Cardinality	Tasks	Status	Parameters	Join Condition
I_SalesOrderItemCube	<input type="text" value="I_SalesOrderItemCube"/>	Primary Datasource			Released	<input checked="" type="checkbox"/>	
I_CustomerMaterial	<input type="text" value="_I_CustomerMaterial"/>	Associated Datasource	Zero or One [0..1]		Deprecated		Link
I_CustomerMaterial_2	<input type="text" value="_I_CustomerMaterial_2"/>	Associated Datasource			Released		Link

We then go to the Elements tab, and here we will add all the new successor fields (as previously noted we cannot yet delete the deprecated fields, as they are still used in the custom Analytical Query). Similarly, we cannot simply amend the formula of our calculation field which uses the deprecated field **NetAmount** as it is also used in the custom analytical query, so we will also need to add a new calculation field, making sure to use the successor field.

Name: *

Label:

+ - * / ()

1 I_SalesOrderItemCube.**NetAmount_2** - I_SalesOrderItemCube.CostAmount

If we look at the list of Elements, we see all the new - successor - fields we have just added. Interesting things to point out are:

1. For the calculation field we added, we suffixed the alias with `_2` (we could have named the alias anything we wanted except `Y_Difference`, since that is the name of the original calculation field and it is still present in the CDS - and you cannot have more than one alias with the same name!)
2. We see that we have added the field `MaterialByCustomer` from the successor CDS entity `I_CustomerMaterial_2` but we also see
3. That the alias of this field has been suffixed with `_1` automatically, because we still have the same field (provided by the deprecated CDS entity with the same name) in our custom CDS. The same reason applies here, we cannot have two aliases with the same name.

The screenshot shows the SAP Fiori Elements list interface. At the top, there is a navigation bar with icons for Data Sources, Parameters, Elements (highlighted), Element Properties, Filter, Log, Tasks, and Next Steps. Below the navigation bar is a header for the 'Elements (43)' list, including a search bar and 'Add', 'Delete', and refresh icons. The main content is a table with the following columns: Key, Alias, Type, Path, Label, Calculation, and Task. The table contains several rows, with three rows highlighted by colored boxes and numbered callouts:

- A blue box highlights the row with Alias `Y_Difference_2` and a callout '1' pointing to the Alias field.
- A red box highlights the row with Alias `MaterialByCustomer_1` and a callout '2' pointing to the Path field.
- A red box highlights the row with Alias `MaterialByCustomer_1` and a callout '3' pointing to the Alias field.

Key	Alias	Type	Path	Label	Calculation	Task
<input type="checkbox"/>	Y_Difference			Amount Difference		
<input type="checkbox"/>	Product	CHAR (40)	I_SalesOrderItemCube.Product	Product		
<input type="checkbox"/>	NetAmount_2	CURR (19,2)	I_SalesOrderItemCube.NetAmount_2	Order Item Net Value		
<input type="checkbox"/>	MaterialByCustomer_1	CHAR (35)	_I_CustomerMaterial_2.MaterialByCustomer	Customer Material		
<input type="checkbox"/>	Y_Difference_2			Amount Difference		
<input type="checkbox"/>	SalesOrder	Association	I_SalesOrderItemCube_SalesOrder	SalesOrder		

You may also need to adjust the element properties, but at this point we will perform a check to make sure there are no errors, and if not publish the changes.

Step 2 - Check the non regression of your changes in the Custom CDS

Before we move on to the next step, we want to check that the field changes we made to the Custom CDS still deliver the information we will subsequently need. For this we call the Data Browser, right from the Custom CDS. We then adjust the fields and output and check that the successor fields deliver information, and that it is identical to the deprecated fields, which in this case it is!

SALESORDER: SALESORDERITEM: SALESORDERTYPE: SALESORDERITEMCATEGORY: Maximum Rows to Be Display...: MATERIALBYCUSTOMER: [Go](#) [Adapt Filters](#)

Deprecated Objects **Successor Objects**

YY1_PRE_DEPR_00 (100) Search

SALESORDER	SALESORDERITEM	SALESORGANIZATION	MATERIAL	MATERIALBYCUSTOMER	NETAMOUNT	Y_DIFFERENCE	PRODUCT	MATERIALBYCUSTOMER_1	NETAMOUNT_2	Y_DIFFERENCE_2
0000064331	000010	1710	MZ-FG-M500	M500-US-CU	1778.00	422.60	MZ-FG-M500	M500-US-CU	1778.00	422.60
0000069062	000010	1710	MZ-FG-M500	M500-US-CU	1778.00	422.60	MZ-FG-M500	M500-US-CU	1778.00	422.60
0000073455	000010	1710	MZ-FG-M500	M500-US-CU	1778.00	422.60	MZ-FG-M500	M500-US-CU	1778.00	422.60
0000037702	000010	1710	MZ-FG-M500	M500-US-CU	1778.00	422.60	MZ-FG-M500	M500-US-CU	1778.00	422.60
0000027336	000010	1710	MZ-FG-M500	M500-US-CU	4445.00	1056.50	MZ-FG-M500	M500-US-CU	4445.00	1056.50
0000050113	000010	1710	MZ-FG-M500	M500-US-CU	1778.00	422.60	MZ-FG-M500	M500-US-CU	1778.00	422.60
0000050697	000010	1710	MZ-FG-M500	M500-US-CU	4445.00	1056.50	MZ-FG-M500	M500-US-CU	4445.00	1056.50
0000071324	000010	1710	MZ-FG-M500	M500-US-CU	4445.00	1056.50	MZ-FG-M500	M500-US-CU	4445.00	1056.50
0000035984	000010	1710	MZ-FG-M500	M500-US-CU	5334.00	1267.80	MZ-FG-M500	M500-US-CU	5334.00	1267.80
0000044209	000010	1710	MZ-FG-M500	M500-US-CU	5334.00	1267.80	MZ-FG-M500	M500-US-CU	5334.00	1267.80

Step 3 - Amending the Custom Analytical Query

We now move on to the Custom Analytical Queries app, to amend our query. Even without going into edit mode, we can already in the Field Selection tab see the fields we just added to the Custom CDS. So let's go into edit mode and add the successor fields to our query, but do not uncheck the deprecated fields just yet.

Deprecation CAQ - Sales 00 (YY1_PRE_DEPR_CAQ_00)

General >>> **Field Selection** >>> Display >>> Filters

Available Fields (30)

Type	Label	Name	Selection
📄	Customer Material	MaterialByCustomer_1	<input type="checkbox"/>
🗨	Order Item Net Value	NetAmount ⚠️ Deprecated	<input checked="" type="checkbox"/>
🗨	Net Value	NetAmountInDisplayCurrency	<input checked="" type="checkbox"/>
🗨	Order Item Net Value	NetAmount_2	<input type="checkbox"/>
🗨	Order Quantity	OrderQuantity	<input checked="" type="checkbox"/>
📄	Sales Unit	OrderQuantityUnit	<input type="checkbox"/>
📄	Plant	Plant	<input checked="" type="checkbox"/>
📄	Product	Product	<input type="checkbox"/>
📄	Sales Order	SalesOrder	<input checked="" type="checkbox"/>
📄	Month of Order	SalesOrderDateYearMonth	<input checked="" type="checkbox"/>
📄	Item	SalesOrderItem	<input checked="" type="checkbox"/>
📄	Sls Order Item Cat.	SalesOrderItemCategory	<input checked="" type="checkbox"/>
📄	Sales Document Type	SalesOrderType	<input checked="" type="checkbox"/>
📄	Sales Organization	SalesOrganization	<input checked="" type="checkbox"/>
📄	Sold-to Party	SoldToParty	<input checked="" type="checkbox"/>
📄	Sold-to Party Name	SoldToPartyName	<input checked="" type="checkbox"/>
📄	Document Currency	TransactionCurrency	<input type="checkbox"/>
📄	Description	UserDescription	<input type="checkbox"/>
📄	User ID	UserID	<input type="checkbox"/>
🗨	Amount Difference	Y_Difference	<input checked="" type="checkbox"/>
🗨	Amount Difference	Y_Difference_2	<input type="checkbox"/>

After going into edit mode, we check the successor fields (right), but we also do not uncheck the deprecated ones just yet (left).

Deprecation CAQ - Sales 00 (YY1_PRE_DEPR_CAQ_00)

General >>> **Field Selection** >>> Display >>> Filters

Available Fields (30)

Type	Label	Name	Selection
📄	Customer Material	MaterialByCustomer_1	<input checked="" type="checkbox"/>
🗨	Order Item Net Value	NetAmount ⚠️ Deprecated	<input checked="" type="checkbox"/>
🗨	Net Value	NetAmountInDisplayCurrency	<input checked="" type="checkbox"/>
🗨	Order Item Net Value	NetAmount_2	<input checked="" type="checkbox"/>
🗨	Order Quantity	OrderQuantity	<input checked="" type="checkbox"/>
📄	Sales Unit	OrderQuantityUnit	<input type="checkbox"/>
📄	Plant	Plant	<input checked="" type="checkbox"/>
📄	Product	Product	<input checked="" type="checkbox"/>
📄	Sales Order	SalesOrder	<input checked="" type="checkbox"/>
📄	Month of Order	SalesOrderDateYearMonth	<input checked="" type="checkbox"/>
📄	Item	SalesOrderItem	<input checked="" type="checkbox"/>
📄	Sls Order Item Cat.	SalesOrderItemCategory	<input checked="" type="checkbox"/>
📄	Sales Document Type	SalesOrderType	<input checked="" type="checkbox"/>
📄	Sales Organization	SalesOrganization	<input checked="" type="checkbox"/>
📄	Sold-to Party	SoldToParty	<input checked="" type="checkbox"/>
📄	Sold-to Party Name	SoldToPartyName	<input checked="" type="checkbox"/>
📄	Document Currency	TransactionCurrency	<input type="checkbox"/>
📄	Description	UserDescription	<input type="checkbox"/>
📄	User ID	UserID	<input type="checkbox"/>
🗨	Amount Difference	Y_Difference	<input checked="" type="checkbox"/>
🗨	Amount Difference	Y_Difference_2	<input checked="" type="checkbox"/>

Checking these fields, will have the effect of making them visible in the Display tab of the query. Depending on the kind of field, it will either be added to the Measures structure, or just be available in the Free section. It is up to you to decide if you want to leave them there, or if you want to have them appear by default in the row or column axis of the ensuing report. It is of course up to you to decide how to proceed, but we would recommend that you move the successor fields from the the 'Free' area of the query to be in the immediate vicinity of the fields they will replace, as below.

Deprecation CAQ - Sales 00 (YY1_PRE_DEPR_CAQ_00)

General >>> Field Selection >>> **Display** >>> Filters

Display Fields (26) Up Down Add

Type	Label	Name	Display	Filter	Remove
	Sold-to Party	SoldToParty	☉	▼	⊗
	Sales Document Type	SalesOrderType	☉	▼	⊗
	Sales Order	SalesOrder	☉	▼	⊗
	Sold-to Party Name	SoldToPartyName	☉	▼	⊗
	Item	SalesOrderItem	☉	▼	⊗
	Material	Material	☉	▼	⊗
	Plant	Plant	☉	▼	⊗
	Customer Material	MaterialByCustomer	☉	▼	⊗
	Created On	CreationDate	☉	▼	⊗

Column

Measure Structure

	Order Item Net Value	NetAmount	☉	▼	⊗
	Cost	CostAmount	☉	▼	⊗
	Amount Difference	Y_Difference	☉	▼	⊗
	Net Value	NetAmountInDisplayCurrency	☉	▼	⊗
	Order Quantity	OrderQuantity	☉	▼	⊗
	Margin	Y_Margin	☉	▼	⊗
	Order Item Net Value	NetAmount_2	☉	▼	⊗
	Amount Difference	Y_Difference_2	☉	▼	⊗

Free

	Exchange Rate Type	P_ExchangeRateType	☉	▼	⊗
	Display Currency	P_DisplayCurrency	☉	▼	⊗
	Created By	CreatedByUser	☉	▼	⊗
	Sls Order Item Cat.	SalesOrderItemCategory	☉	▼	⊗
	Customer Material	MaterialByCustomer_1	☉	▼	⊗
	Product	Product	☉	▼	⊗

We moved the successor fields in the vicinity of those that will be removed. We have chosen to display Deprecated fields first, Successor fields second. That way, when fields that have the same names are displayed, we will know which is which in the preview report.

Deprecation CAQ - Sales 00 (YY1_PRE_DEPR_CAQ_00)

General >>> Field Selection >>> **Display** >>> Filters

Display Fields (26) Up Down Add

Type	Label	Name	Display	Filter	Remove
	Sold-to Party	SoldToParty	☉	▼	⊗
	Sales Document Type	SalesOrderType	☉	▼	⊗
	Sales Order	SalesOrder	☉	▼	⊗
	Sold-to Party Name	SoldToPartyName	☉	▼	⊗
	Item	SalesOrderItem	☉	▼	⊗
	Material	Material	☉	▼	⊗
	Product	Product	☉	▼	⊗
	Plant	Plant	☉	▼	⊗
	Customer Material	MaterialByCustomer	☉	▼	⊗
	Customer Material	MaterialByCustomer_1	☉	▼	⊗
	Created On	CreationDate	☉	▼	⊗

Column

Measure Structure

	Order Item Net Value	NetAmount	☉	▼	⊗
	Order Item Net Value	NetAmount_2	☉	▼	⊗
	Cost	CostAmount	☉	▼	⊗
	Amount Difference	Y_Difference	☉	▼	⊗
	Amount Difference	Y_Difference_2	☉	▼	⊗
	Net Value	NetAmountInDisplayCurrency	☉	▼	⊗
	Order Quantity	OrderQuantity	☉	▼	⊗
	Margin	Y_Margin	☉	▼	⊗

Free

	Exchange Rate Type	P_ExchangeRateType	☉	▼	⊗
	Display Currency	P_DisplayCurrency	☉	▼	⊗
	Created By	CreatedByUser	☉	▼	⊗
	Sls Order Item Cat.	SalesOrderItemCategory	☉	▼	⊗

Lastly, we also need to check the additional fields that we added directly in the query. One such field is the **Margin** field, which is a calculated measure and makes use of the deprecated field **NetAmount**. Because we previously made a full data extract from the custom analytical query before we made any changes, we will directly change the formula here, and replace the deprecated field with the successor field. Using the previous data download to compare with the data produced after we make our changes, will help confirm that our changes are correct.

Note: There will be no error or warning message if your formulas use deprecated fields, nor will you have a message once you have removed the deprecated field from the query, so you need to do this check!

Amount Difference	Y_Difference	👁	▼	⊗
Amount Difference	Y_Difference_2	👁	▼	⊗
Net Value	NetAmountInDisplayCurrency	👁	▼	⊗
Order Quantity	OrderQuantity	👁	▼	⊗
Margin	Y_Margin	👁	▼	⊗

Expression

```
Formula: NODIM ( ( ( "Order Item Net Value(NetAmount)" - "Cost" ) / "Order Item Net Value(NetAmount)" ) * 100 )
```

Edit

Expression Editor

```
1 Order Item Net Value(NetAmount_2) - "Cost" / "Order Item Net Value(NetAmount_2)" * 100
```

Amount Difference	Y_Difference	👁	▼	⊗
Amount Difference	Y_Difference_2	👁	▼	⊗
Net Value	NetAmountInDisplayCurrency	👁	▼	⊗
Order Quantity	OrderQuantity	👁	▼	⊗
Margin	Y_Margin	👁	▼	⊗

Expression

```
Formula: NODIM ( ( ( "Order Item Net Value(NetAmount_2)" - "Cost" ) / "Order Item Net Value(NetAmount_2)" ) * 100 )
```

Edit

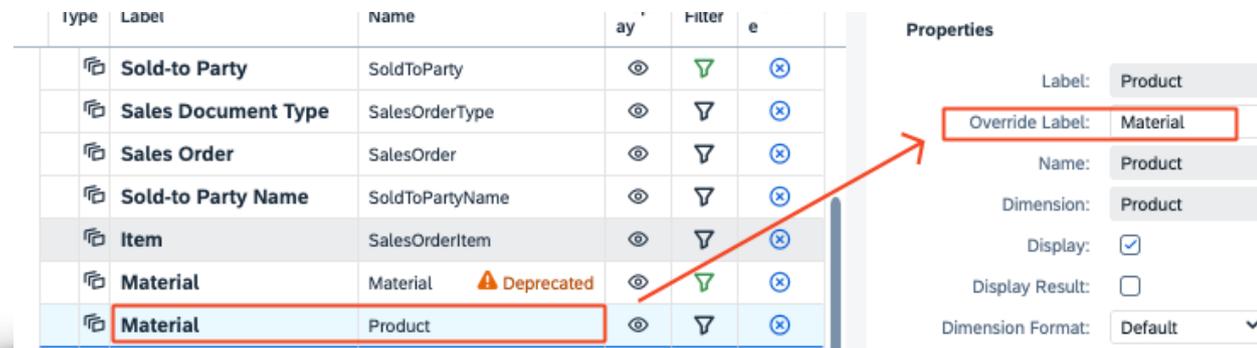
The formula was amended, and we replaced the deprecated field **NetAmount** and replaced it with the successor field **NetAmount_2**.

Next, before we proceed with saving our changes, we can again look at the layout of our report that we had saved before making changes to make sure we have not missed anything.

Looking at the layout we see that the fields **Material** and **Customer Material** are available as filters. So we must adapt the field properties of the newly added fields so that they can be positioned as filters as well. In addition we also note, that one field has a changed name. Indeed, one of the deprecated fields was the field **Material** and the successor to it is called **Product**. Will this be a problem for the business users? You therefore have a choice of either keeping the name **Product** and educating the business users as to this change (preferred) or, you can override this label, and choose to continue using **Material**.



If you wanted to keep the label **Material**, then you would need in the Display tab of the query, for the field **Product**, to maintain the override label there. In our case though, we will keep the new term **Product**, and as part of change management activities, let our users know of the wording change.



We now just need to position the new successor fields **Product** and **Customer Material** as filters, and remove the deprecated fields **Material** and **Customer Material** from the filters. For this we need to go to the Filters tab of the query.

General >>> Field Selection >>> Display >>> **Filters**

Filter Fields (26)

Type	Label	Name	Display	Filter	Remove
Sales Organization	SalesOrganization	SalesOrganization	👁	📏	🗑
Distribution Channel	DistributionChannel	DistributionChannel	👁	📏	🗑
Month of Order	SalesOrderDateYearMonth	SalesOrderDateYearMonth	👁	📏	🗑
Sold-to Party	SoldToParty	SoldToParty	👁	📏	🗑
Sales Document Type	SalesOrderType	SalesOrderType	👁	📏	🗑
Sales Order	SalesOrder	SalesOrder	👁	📏	🗑
Sold-to Party Name	SoldToPartyName	SoldToPartyName	👁	📏	🗑
Item	SalesOrderItem	SalesOrderItem	👁	📏	🗑
Material	Material	Material ⚠️ Deprecated	👁	📏	🗑
Product	Product	Product	👁	📏	🗑
Plant	Plant	Plant	👁	📏	🗑

Product (Dimension)

No Filters
 Fixed Values

Include: +
Exclude: +

User Input Values

Selection Type: Single

Multiple Selection:

Mandatory:

Variable Sequence: - 0 +

Default Value:

Derivation Functions

Lookup Entity:

We made the new successor fields **Product** and **Customer Material** filters, to replace the to be removed fields, and at the same time removed the filtering property from the deprecated fields **Material** and **Customer Material**.

General >>> Field Selection >>> Display >>> **Filters**

Filter Fields (26)

Type	Label	Name	Display	Filter	Remove
Sales Organization	SalesOrganization	SalesOrganization	👁	📏	🗑
Distribution Channel	DistributionChannel	DistributionChannel	👁	📏	🗑
Month of Order	SalesOrderDateYearMonth	SalesOrderDateYearMonth	👁	📏	🗑
Sold-to Party	SoldToParty	SoldToParty	👁	📏	🗑
Sales Document Type	SalesOrderType	SalesOrderType	👁	📏	🗑
Sales Order	SalesOrder	SalesOrder	👁	📏	🗑
Sold-to Party Name	SoldToPartyName	SoldToPartyName	👁	📏	🗑
Item	SalesOrderItem	SalesOrderItem	👁	📏	🗑
Material	Material	Material ⚠️ Deprecated	👁	📏	🗑
Product	Product	Product	👁	📏	🗑
Plant	Plant	Plant	👁	📏	🗑
Customer Material	MaterialByCustomer	MaterialByCustomer	👁	📏	🗑
Customer Material	MaterialByCustomer_1	MaterialByCustomer_1	👁	📏	🗑
Created On	CreationDate	CreationDate	👁	📏	🗑

Material (Dimension)

No Filters
 Fixed Values

Include: +
Exclude: +

User Input Values

Selection Type: Single

Multiple Selection:

Mandatory:

Variable Sequence: - 0 +

Default Value:

Derivation Functions

Lookup Entity:

Hide:

Mandatory:

General >>> Field Selection >>> Display >>> **Filters**

Filter Fields (26)

Type	Label	Name	Display	Filter	Remove
Sales Organization	SalesOrganization	SalesOrganization	👁	📏	🗑
Distribution Channel	DistributionChannel	DistributionChannel	👁	📏	🗑
Month of Order	SalesOrderDateYearMonth	SalesOrderDateYearMonth	👁	📏	🗑
Sold-to Party	SoldToParty	SoldToParty	👁	📏	🗑
Sales Document Type	SalesOrderType	SalesOrderType	👁	📏	🗑
Sales Order	SalesOrder	SalesOrder	👁	📏	🗑
Sold-to Party Name	SoldToPartyName	SoldToPartyName	👁	📏	🗑
Item	SalesOrderItem	SalesOrderItem	👁	📏	🗑
Material	Material	Material ⚠️ Deprecated	👁	📏	🗑
Product	Product	Product	👁	📏	🗑
Plant	Plant	Plant	👁	📏	🗑
Customer Material	MaterialByCustomer	MaterialByCustomer	👁	📏	🗑
Customer Material	MaterialByCustomer_1	MaterialByCustomer_1	👁	📏	🗑
Created On	CreationDate	CreationDate	👁	📏	🗑

Material (Dimension)

No Filters
 Fixed Values

Include: +
Exclude: +

User Input Values

Selection Type: Single

Multiple Selection:

Mandatory:

Variable Sequence: - 0 +

Default Value:

Derivation Functions

Lookup Entity:

Hide:

Mandatory:

General >>> Field Selection >>> Display >>> **Filters**

Filter Fields (26)

Type	Label	Name	Display	Filter	Remove
Sales Organization	SalesOrganization	SalesOrganization	👁	📏	🗑
Distribution Channel	DistributionChannel	DistributionChannel	👁	📏	🗑
Month of Order	SalesOrderDateYearMonth	SalesOrderDateYearMonth	👁	📏	🗑
Sold-to Party	SoldToParty	SoldToParty	👁	📏	🗑
Sales Document Type	SalesOrderType	SalesOrderType	👁	📏	🗑
Sales Order	SalesOrder	SalesOrder	👁	📏	🗑
Sold-to Party Name	SoldToPartyName	SoldToPartyName	👁	📏	🗑
Item	SalesOrderItem	SalesOrderItem	👁	📏	🗑
Material	Material	Material ⚠️ Deprecated	👁	📏	🗑
Product	Product	Product	👁	📏	🗑
Plant	Plant	Plant	👁	📏	🗑
Customer Material	MaterialByCustomer	MaterialByCustomer	👁	📏	🗑
Customer Material	MaterialByCustomer_1	MaterialByCustomer_1	👁	📏	🗑
Created On	CreationDate	CreationDate	👁	📏	🗑

Customer Material (Dimension)

No Filters
 Fixed Values

Include: +
Exclude: +

User Input Values

Selection Type: Single

Multiple Selection:

Mandatory:

Variable Sequence: - 0 +

Default Value:

Derivation Functions

Lookup Entity:

Hide:

Mandatory:

Now that we have processed all the changes (except removal of the deprecated fields), we can save our query, publish it, preview the data and compare both the layout and content, to make sure the data that will be offered to the business users is still as expected. Comparing our data preview, with the excel download we had done before making the changes we can see that we are still serving up the same data.

1. The same filters are available
2. The **Product** and **Material** field provide the same data
3. The **Customer Material** fields provide the same data
4. The **NetAmount** and **NetAmount_2** fields provide the same data
5. The **Amount Difference** fields provide the same data (these fields came from the Custom CDS)
6. The **Margin** field, for which we changed the formula, calculates the same value

It appears our changes are correct.

The screenshot shows the SAP Fiori interface for 'Deprecation CAQ - Sales 00'. The top navigation bar includes 'SAP', 'Standard *', and search filters. The main area displays a data table with columns for sales data, material details, and financial metrics. Red boxes and arrows highlight specific areas:

- Annotation 1:** Points to the 'Product' and 'Customer Material' filter fields at the top right.
- Annotation 2:** Points to the 'Material' and 'Product' columns in the table.
- Annotation 3:** Points to the 'Customer Material' column in the table.
- Annotation 4:** Points to the 'Net Value' and 'Net Amount' columns.
- Annotation 5:** Points to the 'Amount Difference' columns.
- Annotation 6:** Points to the 'Margin' column.

Below the SAP interface is an Excel export of the same data. Red boxes and arrows in the Excel file correspond to the annotations in the SAP interface, showing that the data values are identical. The Excel table has columns: Sales Order, Sold-to Party Name, Item, Material, Plant, Customer Material, Created On, Order Item Net Value, Cost, Amount Difference, Net Value, Order Quantity, and Margin.

We can now go back to the query and uncheck from it, all the fields that we no longer want to see in it (remember, we are wanting to remove their use in the query, so that we may remove them from the custom CDS). So let's go back to the query in edit mode and uncheck from the Field Selection tab, all the deprecated and replaced fields. Doing so will automatically also removed them from wherever they are used in the display tab (in the Row, Column or free section). Once we are done, we can once again, save our changes and publish the query. You can also preview the data again, which should this time be a mirror image of your report, before you made any changes.

Deprecation CAQ - Sales 00 (YY1_PRE_DEPR_CAQ_00)

General >>> **Field Selection** >>> Display >>> Filters

Type	Label	Name	Selection
📄	Cross-plant CP	CrossPlantConfigurableProduct	<input type="checkbox"/>
📄	Display Currency	DisplayCurrency	<input type="checkbox"/>
📄	Distribution Channel	DistributionChannel	<input checked="" type="checkbox"/>
📄	Material	Material ⚠️ Deprecated	<input checked="" type="checkbox"/>
📄	Customer Material	MaterialByCustomer	<input checked="" type="checkbox"/>
📄	Customer Material	MaterialByCustomer_1	<input checked="" type="checkbox"/>
📄	Order Item Net Value	NetAmount ⚠️ Deprecated	<input checked="" type="checkbox"/>
📄	Net Value	NetAmountInDisplayCurrency	<input checked="" type="checkbox"/>
📄	Order Item Net Value	NetAmount_2	<input checked="" type="checkbox"/>
📄	Order Quantity	OrderQuantity	<input checked="" type="checkbox"/>
📄	Sales Unit	OrderQuantityUnit	<input type="checkbox"/>
📄	Plant	Plant	<input checked="" type="checkbox"/>
📄	Product	Product	<input checked="" type="checkbox"/>
📄	Sales Order	SalesOrder	<input checked="" type="checkbox"/>
📄	Month of Order	SalesOrderDateYearMonth	<input checked="" type="checkbox"/>
📄	Item	SalesOrderItem	<input checked="" type="checkbox"/>
📄	Sls Order Item Cat.	SalesOrderItemCategory	<input checked="" type="checkbox"/>
📄	Sales Document Type	SalesOrderType	<input checked="" type="checkbox"/>
📄	Sales Organization	SalesOrganization	<input checked="" type="checkbox"/>
📄	Sold-to Party	SoldToParty	<input checked="" type="checkbox"/>
📄	Sold-to Party Name	SoldToPartyName	<input checked="" type="checkbox"/>
📄	Document Currency	TransactionCurrency	<input type="checkbox"/>
📄	Description	UserDescription	<input type="checkbox"/>
📄	User ID	UserID	<input type="checkbox"/>
📄	Amount Difference	Y_Difference	<input checked="" type="checkbox"/>
📄	Amount Difference	Y_Difference_2	<input checked="" type="checkbox"/>

Deprecation CAQ - Sales 00 (YY1_PRE_DEPR_CAQ_00)

General >>> **Field Selection** >>> Display >>> Filters

Type	Label	Name	Selection
📄	Cross-plant CP	CrossPlant	<input type="checkbox"/>
📄	Display Currency	DisplayCurrency	<input type="checkbox"/>
📄	Distribution Channel	DistributionChannel	<input checked="" type="checkbox"/>
📄	Material	Material ⚠️ Deprecated	<input type="checkbox"/>
📄	Customer Material	MaterialByCustomer	<input type="checkbox"/>
📄	Customer Material	MaterialByCustomer_1	<input checked="" type="checkbox"/>
📄	Order Item Net Value	NetAmount ⚠️ Deprecated	<input type="checkbox"/>
📄	Net Value	NetAmountInDisplayCurrency	<input checked="" type="checkbox"/>
📄	Order Item Net Value	NetAmount_2	<input checked="" type="checkbox"/>
📄	Order Quantity	OrderQuantity	<input checked="" type="checkbox"/>
📄	Sales Unit	OrderQuantityUnit	<input type="checkbox"/>
📄	Plant	Plant	<input checked="" type="checkbox"/>
📄	Product	Product	<input checked="" type="checkbox"/>
📄	Sales Order	SalesOrder	<input checked="" type="checkbox"/>
📄	Month of Order	SalesOrderDateYearMonth	<input checked="" type="checkbox"/>
📄	Item	SalesOrderItem	<input checked="" type="checkbox"/>
📄	Sls Order Item Cat.	SalesOrderItemCategory	<input checked="" type="checkbox"/>
📄	Sales Document Type	SalesOrderType	<input checked="" type="checkbox"/>
📄	Sales Organization	SalesOrganization	<input checked="" type="checkbox"/>
📄	Sold-to Party	SoldToParty	<input checked="" type="checkbox"/>
📄	Sold-to Party Name	SoldToPartyName	<input checked="" type="checkbox"/>
📄	Document Currency	TransactionCurrency	<input type="checkbox"/>
📄	Description	UserDescription	<input type="checkbox"/>
📄	User ID	UserID	<input type="checkbox"/>
📄	Amount Difference	Y_Difference	<input checked="" type="checkbox"/>
📄	Amount Difference	Y_Difference_2	<input checked="" type="checkbox"/>

Fields need to be unchecked in the Query, in order to be able to remove them from the Custom CDS

Now that the query has been fully updated and changed, we might be tempted to now go back to the custom CDS and remove all the fields that are deprecated and causing for rework task messages to be displayed. Indeed, if we go to the custom CDS, and switch to edit mode, we can see the deprecated fields (and generally those that we want to remove), can now be deleted from the custom CDS - that is because they are no longer used by the query. This is also what I meant earlier when I referred to hard and loose coupling. I.e these fields are still used in the eSAC story, but that is not stopping me from deleting them here and now!

So at this point, let's leave the fields in the Custom CDS. Cancel switching to edit mode and discard the draft on the custom CDS.

We will delete those fields at the very end.

Deprecation Custom CDS - Sales 00
 Dependencies: 1 (Compatible Changes Allowed)

Elements (43)

<input type="checkbox"/>	Key	Alias	Type	Path	Label	Calculation	Task
<input type="checkbox"/>	OFF	SalesOrderDateYearMonth	NUMC (6)	I_SalesOrderItemCube.SalesOrderDateYearMonth	Month of Order		
<input type="checkbox"/>	OFF	SalesOrganization	CHAR (4)	I_SalesOrderItemCube.SalesOrganization	Sales Organization		
<input type="checkbox"/>	OFF	DistributionChannel	CHAR (2)	I_SalesOrderItemCube.DistributionChannel	Distribution Channel		
<input checked="" type="checkbox"/>	OFF	Material	CHAR (40)	I_SalesOrderItemCube.Material	Material		⚠
<input type="checkbox"/>	OFF	Plant	CHAR (4)	I_SalesOrderItemCube.Plant	Plant		
<input type="checkbox"/>	OFF	SoldToParty	CHAR (10)	I_SalesOrderItemCube.SoldToParty	Sold-to Party		
<input type="checkbox"/>	OFF	SoldToPartyName	CHAR (80)	I_SalesOrderItemCube.SoldToPartyName	Sold-to Party Name		
<input checked="" type="checkbox"/>	OFF	NetAmount	CURR (15,2)	I_SalesOrderItemCube.NetAmount	Order Item Net Value		⚠
<input type="checkbox"/>	OFF	NetAmountInDisplayCurrency	CURR (19,2)	I_SalesOrderItemCube.NetAmountInDisplayCurrency	Net Value		
<input type="checkbox"/>	OFF	CostAmount	CURR (13,2)	I_SalesOrderItemCube.CostAmount	Cost		
<input type="checkbox"/>	OFF	UserID	CHAR (12)	I_SalesOrderItemCube._CreatedByUser.UserID	User ID		
<input type="checkbox"/>	OFF	OrderQuantity	QUAN (15,3)	I_SalesOrderItemCube.OrderQuantity	Order Quantity		
<input type="checkbox"/>	OFF	TransactionCurrency	CUKY (5)	I_SalesOrderItemCube.TransactionCurrency	Document Currency		
<input type="checkbox"/>	OFF	DisplayCurrency	CUKY (5)	I_SalesOrderItemCube.DisplayCurrency	Display Currency		
<input type="checkbox"/>	OFF	UserDescription	CHAR (80)	I_SalesOrderItemCube._CreatedByUser.UserDescription	Description		
<input type="checkbox"/>	OFF	OrderQuantityUnit	UNIT (3)	I_SalesOrderItemCube.OrderQuantityUnit	Sales Unit		
<input checked="" type="checkbox"/>	OFF	MaterialByCustomer	CHAR (35)	_I_CustomerMaterial.MaterialByCustomer	Customer Material		
<input type="checkbox"/>	OFF	CrossPlantConfigurableProduct	CHAR (40)	I_SalesOrderItemCube._Product.CrossPlantConfigurableProduct	Cross-plant CP		

Step 4 - Amending the Query report - Multi Dimensional Report

Actually, we don't need to do anything here. The custom analytical query, we had previously exposed as a multidimensional report (using the Manage KPI & Reports) application, which we then exposed as a tile on the Fiori Launchpad, by doing a custom catalog extension was already changed, when I changed the Query. So if I click on the tile that is on the Fiori Launchpad, I should then in theory launch a report that is in effect identical to the preview we saw in the custom analytical query! So let's try it. Indeed we can see that our report works as expected. However, note that some report variants may need to be reworked if the old removed fields were used in them.

The screenshot shows the SAP Fiori Analytics interface for a report titled "Deprecation Sales" by Pascal. The report is displayed in a table view under the "Data Analysis" tab. The top navigation bar includes filters for "Rate Type", "Display Currency" (set to EUR), "Sales Organization" (1010), "Sold-to Party", "Product", and "Customer Material". The table below shows data for various sales orders and items, with columns for "Product", "Plant", "Customer Material", "Net Value", "Cost", "Amount Difference", "Net Value", "Order Quantity", and "Margin".

Sales Order	Sold-to Party Name	Item	Product	Plant	Customer Material	Net Value	Cost	Amount Difference	Net Value	Order Quantity	Margin
SO01	Hypercom AG	1	P002	1010	#	23,400.00 EUR	0.00 EUR	23,400.00 EUR	23,400.00 EUR	0 AU	100.00%
OR	Hypercom AG	10	TG11	1010	#	351.00 EUR	270.00 EUR	81.00 EUR	351.00 EUR	20 PC	23.00%
OR	Hypercom AG	10	TG11	1010	#	702.00 EUR	540.00 EUR	162.00 EUR	702.00 EUR	40 PC	23.00%
OR	Hypercom AG	10	TG11	1010	#	702.00 EUR	540.00 EUR	162.00 EUR	702.00 EUR	40 PC	23.00%
SO01	TechInc	1	P001	1010	#	50,000.00 EUR	0.00 EUR	50,000.00 EUR	50,000.00 EUR	0 AU	100.00%
SO01	TechInc	2	P002	1010	#	180,000.00 EUR	0.00 EUR	180,000.00 EUR	180,000.00 EUR	0 AU	100.00%
SO01	TechInc	3	P003	1010	#	20,000.00 EUR	0.00 EUR	20,000.00 EUR	20,000.00 EUR	0 AU	100.00%
SO01	TechInc	4	U001	1010	#	20,000.00 EUR	0.00 EUR	20,000.00 EUR	20,000.00 EUR	0 PC	100.00%
OR	Hypercom AG	10	TG12	DE20	HANACT1	17.55 EUR	15.80 EUR	1.75 EUR	17.55 EUR	1 PC	9.9%
OR	Hypercom AG	10	TG12	DE20	HANACT1	17.55 EUR	15.80 EUR	1.75 EUR	17.55 EUR	1 PC	9.9%
OR	Hypercom AG	10	TG12	DE20	HANACT1	17.55 EUR	15.80 EUR	1.75 EUR	17.55 EUR	1 PC	9.9%
OR	TechInc	10	TG12	1710	#	17.55 EUR	15.80 EUR	1.75 EUR	17.55 EUR	1 PC	9.9%
SO01	TechInc	1	P002	1010	#	1,200.00 EUR	0.00 EUR	1,200.00 EUR	1,200.00 EUR	0 AU	100.00%
SO01	TechInc	1	P002	1010	#	42,000.00 EUR	0.00 EUR	42,000.00 EUR	42,000.00 EUR	0 AU	100.00%
SO01	TechInc	1	P002	1010	#	42,000.00 EUR	0.00 EUR	42,000.00 EUR	42,000.00 EUR	0 AU	100.00%

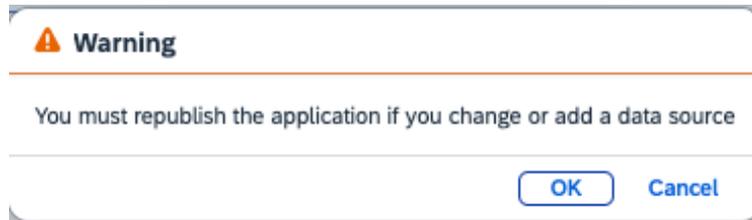
Step 5 - Amending the Embedded SAC story

We now go to the Manage KPIs & Reports app, go the Stories tab, select our story and click Edit.

The screenshot shows the SAP Manage KPIs and Reports app interface. At the top, the 'Manage KPIs and Reports' menu is open. Below it, the 'Stories' icon is highlighted with a red box. The main content area shows a search bar and filters for Source, Tags, and CDS View. Below these filters, there are two tabs: 'Embedded Stories (36)' and 'Stand-alone Stories'. The 'Embedded Stories' tab is active, and a table of stories is displayed. The table has columns for Source, Story Name, Description, Data Source, Tags, Job Status, and Applications. The 'Deprecated Story' row is selected, and its 'Edit' button is highlighted with a red box.

Source	Story Name	Description	Data Source	Tags	Job Status	Applications
	Engagement Project Profit Margin	PFS Project Profit Margin Analysis	C_EngmtProjMargAnlysQuery	PFS	Export Successful	1 >
	Deprecated Story		YY1_PRE_DEPR_CAQ_00		Export Successful	1 >

Doing so will open a warning pop-up making you aware of having to republish the application if you change or add a data source. We click OK to proceed. Once past this screen, the eSAC story will load.



Once the eSAC story launches (top), and compare it to a previously saved (bottom) screenshot (before making any changes), we can see that something is wrong. Whilst the measures seem to all be there, we can immediately see that the **Material** and **Customer Material** columns are not in the tabular report view, and thus we need to add them back in the report.

The top screenshot shows a report with the following measures: Order Item Net Value, Cost, Amount Difference, Net Value, Order Quantity, and Margin. The bottom screenshot shows the same report but with 'Material' and 'Customer Material' columns missing from the measures. Red arrows indicate that these columns were present in the previous state but are now absent.

Sales Organization	Distribution Channel	Month of Order	Sold-to Party	Sales Document Type	Sales Order	Sold-to Party Name	Item	Plant	Order Item Net Value	Cost	Amount Difference	Net Value	Order Quantity	Margin
1010	10	02/2021	PFSLC1001	SO01	217596	Fastkorb AG	1	1010	EUR1,924,500.00	EUR0.00	EUR1,924,500.00	EUR1,924,500.00	0 AU	100.00
		11/2023	10100001	OR	385517	Hypercom AG	10	1010	EUR995,000.00	EUR40,000.00	EUR955,000.00	EUR995,000.00	10 PC	95.98
			10100004	OR	386769	Inlandskunde DE 4	10	1010	EUR211,198.80	EUR0.00	EUR211,198.80	EUR211,198.80	172 KWH	100.00
					386583	Inlandskunde DE 4	10	1010	EUR200,839.24	EUR0.00	EUR200,839.24	EUR200,839.24	172 KWH	100.00
					386751	Inlandskunde DE 4	10	1010	EUR200,724.00	EUR0.00	EUR200,724.00	EUR200,724.00	172 KWH	100.00
		06/2017	10100002	SO01	36733	TechInc	2	1010	EUR180,000.00	EUR0.00	EUR180,000.00	EUR180,000.00	0 AU	100.00
		11/2023	10100001	OR	386456	Hypercom AG	20	1010	EUR109,989.00	EUR3,168.00	EUR106,821.00	EUR109,989.00	99 PC	97.12

Distribution Channel	Month of Order	Sold-to Party	Sales Document Type	Sales Order	Sold-to Party Name	Item	Material	Plant	Customer Material	Order Item Net Value	Cost	Amount Difference	Net Value	Order Quantity	Margin
10	02/2021	PFSLC1001	SO01	217596	Fastkorb AG	1	P002	1010	#	EUR1,924,500.00	EUR0.00	EUR1,924,500.00	EUR1,924,500.00	0 AU	100.00
	11/2023	10100001	OR	385517	Hypercom AG	10	AVC_RBT_ROBOT2	1010	#	EUR995,000.00	EUR40,000.00	EUR955,000.00	EUR995,000.00	10 PC	95.98
		10100004	OR	386583	Inlandskunde DE 4	10	SM100	1010	#	EUR200,839.24	EUR0.00	EUR200,839.24	EUR200,839.24	172 KWH	100.00
	06/2017	10100002	SO01	36733	TechInc	2	P002	1010	#	EUR180,000.00	EUR0.00	EUR180,000.00	EUR180,000.00	0 AU	100.00
	11/2023	10100001	OR	386456	Hypercom AG	20	FG130	1010	#	EUR109,989.00	EUR3,168.00	EUR106,821.00	EUR109,989.00	99 PC	97.12

We therefore select the report and in designer mode, we add the missing dimensions and place them in the correct order, as they were before.

The first screenshot shows the report designer interface. The 'Rows' list includes Sales Organization, Distribution Channel, Month of Order, Sold-to Party, Sales Document Type, Sales Order, Sold-to Party Name, Item, and Plant. The 'Add Dimensions' button is highlighted with a red box. A red arrow points to the second screenshot.

The second screenshot shows the 'Add Dimensions' dialog box. The 'Product' checkbox is checked and highlighted with a red box. A red arrow points to the third screenshot.

The third screenshot shows the report layout. The 'Measures' section now includes 'Customer Material' in addition to 'Order Item Net Value'. A red arrow points from the 'Product' checkbox in the second screenshot to the 'Product' dimension in the report layout.

We once again check the report to make sure that everything that needed to be changed has been changed.

Deprecated Sales Demo

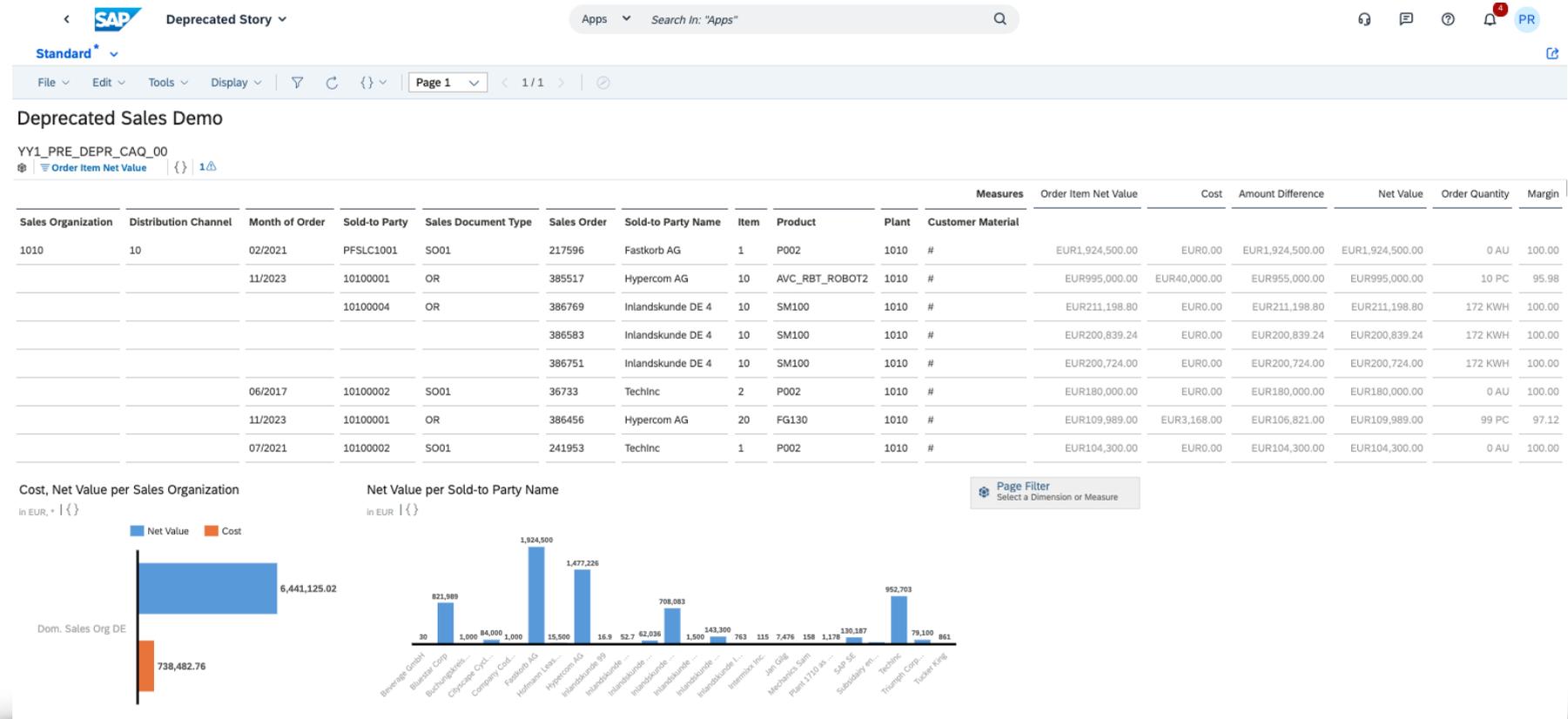
YY1_PRE_DEPR_CAQ_00

Order Item Net Value

Item	Product	Plant	Customer Material	Order Item Net Value	Cost	Amount Difference	Net Value	Order Quantity	Margin
1	P002	1010	#	EUR1,924,500.00	EUR0.00	EUR1,924,500.00	EUR1,924,500.00	0 AU	100.00
10	AVC_RBT_ROBOT2	1010	#	EUR995,000.00	EUR40,000.00	EUR955,000.00	EUR995,000.00	10 PC	95.98
10	SM100	1010	#	EUR211,198.80	EUR0.00	EUR211,198.80	EUR211,198.80	172 KWH	100.00
10	SM100	1010	#	EUR200,839.24	EUR0.00	EUR200,839.24	EUR200,839.24		
2	P002	1010	#	EUR180,000.00	EUR0.00	EUR180,000.00	EUR180,000.00		
20	FG130	1010	#	EUR109,989.00	EUR0.00	EUR109,989.00	EUR109,989.00		
1	P002	1010	#	EUR104,300.00	EUR0.00	EUR104,300.00	EUR104,300.00		

Once we are happy with our changes, and checked that all the data presented is as it was before we made changes, we can activate the eSAC story. If we launch the eSAC story by using the tile on our Fiori Launchpad, our story is presented and displays as desired.

Note that in our case, we only needed to add a couple of fields. Depending on the sophistication of your eSAC story, the changes might be more extensive. Again we cannot stress enough the importance of being able to do before/after comparisons of both layout and content.



Step 6 - Amending the custom CDS

Now that we have propagated all the required changes, we are back where everything started. We open our custom CDS with the Custom CDS app, switch to edit mode and go to the Elements tab, where we select and delete all the deprecated fields and fields we want to remove.

Custom CDS View Details - Depracation Custom CDS - Sales 00

Dependencies: 1 (Compatible Changes Allowed)

Elements (43)

<input type="checkbox"/>	Key	Alias	Type	Path	Label	Calculation	Task
<input type="checkbox"/>	OFF	CreatedByUserName	CHAR (80)	I_SalesOrderItemCube.CreatedByUserName	Created By User Name		
<input type="checkbox"/>	OFF	CreationDate	DATS (8)	I_SalesOrderItemCube.CreationDate	Created On		
<input type="checkbox"/>	OFF	SalesOrderDateYearM...	NUMC (6)	I_SalesOrderItemCube.SalesOrderDateYearMonth	Month of Order		
<input type="checkbox"/>	OFF	SalesOrganization	CHAR (4)	I_SalesOrderItemCube.SalesOrganization	Sales Organization		
<input type="checkbox"/>	OFF	DistributionChannel	CHAR (2)	I_SalesOrderItemCube.DistributionChannel	Distribution Channel		
<input checked="" type="checkbox"/>	OFF	Material	CHAR (40)	I_SalesOrderItemCube.Material	Material		
<input type="checkbox"/>	OFF	Plant	CHAR (4)	I_SalesOrderItemCube.Plant	Plant		
<input type="checkbox"/>	OFF	SoldToParty	CHAR (10)	I_SalesOrderItemCube.SoldToParty	Sold-to Party		
<input type="checkbox"/>	OFF	SoldToPartyName	CHAR (80)	I_SalesOrderItemCube.SoldToPartyName	Sold-to Party Name		
<input checked="" type="checkbox"/>	OFF	NetAmount	CURR (15,2)	I_SalesOrderItemCube.NetAmount	Order Item Net Value		
<input type="checkbox"/>	OFF	NetAmountInDisplayC...	CURR (19,2)	I_SalesOrderItemCube.NetAmountInDisplayCurrency	Net Value		
<input type="checkbox"/>	OFF	CostAmount	CURR (13,2)	I_SalesOrderItemCube.CostAmount	Cost		
<input type="checkbox"/>	OFF	UserID	CHAR (12)	I_SalesOrderItemCube._CreatedByUser.UserID	User ID		
<input type="checkbox"/>	OFF	OrderQuantity	QUAN (15,3)	I_SalesOrderItemCube.OrderQuantity	Order Quantity		
<input type="checkbox"/>	OFF	TransactionCurrency	CUKY (5)	I_SalesOrderItemCube.TransactionCurrency	Document Currency		
<input type="checkbox"/>	OFF	DisplayCurrency	CUKY (5)	I_SalesOrderItemCube.DisplayCurrency	Display Currency		
<input type="checkbox"/>	OFF	UserDescription	CHAR (80)	I_SalesOrderItemCube._CreatedByUser.UserDescription	Description		
<input type="checkbox"/>	OFF	OrderQuantityUnit	UNIT (3)	I_SalesOrderItemCube.OrderQuantityUnit	Sales Unit		
<input checked="" type="checkbox"/>	OFF	MaterialByCustomer	CHAR (35)	_I_CustomerMaterial.MaterialByCustomer	Customer Material		
<input type="checkbox"/>	OFF	CrossPlantConfigurable...	CHAR (40)	I_SalesOrderItemCube._Product.CrossPlantConfigurableProduct	Cross-plant CP		
<input checked="" type="checkbox"/>	OFF	Y_Difference			Amount Difference		
<input type="checkbox"/>	OFF	Product	CHAR (40)	I_SalesOrderItemCube.Product	Product		

Custom CDS View Details - Depracation Custom CDS - Sales 00

Dependencies: 1 (Compatible Changes Allowed)

Elements (39)

<input type="checkbox"/>	Key	Alias	Type	Path	Label	Calculation	Task
<input type="checkbox"/>	ON	SalesOrder	CHAR (10)	I_SalesOrderItemCube.SalesOrder	Sales Order		
<input type="checkbox"/>	ON	SalesOrderItem	NUMC (6)	I_SalesOrderItemCube.SalesOrderItem	Item		
<input type="checkbox"/>	OFF	SalesOrderType	CHAR (4)	I_SalesOrderItemCube.SalesOrderType	Sales Document Type		
<input type="checkbox"/>	OFF	SalesOrderItemCategory	CHAR (4)	I_SalesOrderItemCube.SalesOrderItemCategory	SIS Order Item Cat.		
<input type="checkbox"/>	OFF	CreatedByUser	CHAR (12)	I_SalesOrderItemCube.CreatedByUser	Created By		
<input type="checkbox"/>	OFF	CreatedByUserName	CHAR (80)	I_SalesOrderItemCube.CreatedByUserName	Created By User Name		
<input type="checkbox"/>	OFF	CreationDate	DATS (8)	I_SalesOrderItemCube.CreationDate	Created On		
<input type="checkbox"/>	OFF	SalesOrderDateYearM...	NUMC (6)	I_SalesOrderItemCube.SalesOrderDateYearMonth	Month of Order		
<input type="checkbox"/>	OFF	SalesOrganization	CHAR (4)	I_SalesOrderItemCube.SalesOrganization	Sales Organization		
<input type="checkbox"/>	OFF	DistributionChannel	CHAR (2)	I_SalesOrderItemCube.DistributionChannel	Distribution Channel		
<input type="checkbox"/>	OFF	Plant	CHAR (4)	I_SalesOrderItemCube.Plant	Plant		
<input type="checkbox"/>	OFF	SoldToParty	CHAR (10)	I_SalesOrderItemCube.SoldToParty	Sold-to Party		
<input type="checkbox"/>	OFF	SoldToPartyName	CHAR (80)	I_SalesOrderItemCube.SoldToPartyName	Sold-to Party Name		
<input type="checkbox"/>	OFF	NetAmountInDisplayC...	CURR (19,2)	I_SalesOrderItemCube.NetAmountInDisplayCurrency	Net Value		
<input type="checkbox"/>	OFF	CostAmount	CURR (13,2)	I_SalesOrderItemCube.CostAmount	Cost		
<input type="checkbox"/>	OFF	UserID	CHAR (12)	I_SalesOrderItemCube._CreatedByUser.UserID	User ID		
<input type="checkbox"/>	OFF	OrderQuantity	QUAN (15,3)	I_SalesOrderItemCube.OrderQuantity	Order Quantity		
<input type="checkbox"/>	OFF	TransactionCurrency	CUKY (5)	I_SalesOrderItemCube.TransactionCurrency	Document Currency		
<input type="checkbox"/>	OFF	DisplayCurrency	CUKY (5)	I_SalesOrderItemCube.DisplayCurrency	Display Currency		
<input type="checkbox"/>	OFF	UserDescription	CHAR (80)	I_SalesOrderItemCube._CreatedByUser.UserDescription	Description		
<input type="checkbox"/>	OFF	OrderQuantityUnit	UNIT (3)	I_SalesOrderItemCube.OrderQuantityUnit	Sales Unit		

Now that the `Customer Material` field that was provided by the Deprecated CDS Entity has been removed, we can now go to the Data Sources tab and remove the association to the Deprecated CDS entity.

The screenshot shows the SAP Data Sources configuration interface. The breadcrumb navigation includes: Data Sources >> Parameters >> Elements >> Element Properties >> Filter >> Log >> Tasks >> Next Steps. The 'Data Sources (3)' section contains a table with the following data:

Name	Alias	Type	Cardinality	Tasks	Status	Parameters	Join Condition
<input type="radio"/> I_SalesOrderItemCube	I_SalesOrderItemCube	Primary Datasource			Released	<input checked="" type="checkbox"/>	
<input checked="" type="radio"/> I_CustomerMaterial	I_CustomerMaterial	Associated Datasource	Zero or One [0..1]		Deprecated		
<input type="radio"/> I_CustomerMaterial_2	I_CustomerMaterial_2	Associated Datasource	Zero or One [0..1]		Released		

A red box highlights the 'Delete' button in the top right corner. A red arrow points from the 'Delete' button to the 'I_CustomerMaterial' row, indicating the removal of this deprecated association.

The deprecated CDS entity has now been removed.

The screenshot shows the SAP Data Sources configuration interface after the removal of the deprecated CDS entity. The breadcrumb navigation is the same as in the previous screenshot. The 'Data Sources (2)' section contains a table with the following data:

Name	Alias	Type	Cardinality	Tasks	Status	Parameters	Join Condition
<input type="radio"/> I_SalesOrderItemCube	I_SalesOrderItemCube	Primary Datasource			Released	<input checked="" type="checkbox"/>	
<input checked="" type="radio"/> I_CustomerMaterial_2	I_CustomerMaterial_2	Associated Datasource	Zero or One [0..1]		Released		

A red callout box with the text "Deprecated CDS entity usage has now been removed" points to the 'I_CustomerMaterial' row, which is no longer present in the table.

We can perform an integrity check by clicking the Check button and if the check is successful, we can then publish our custom CDS by clicking the Publish button. One done, when we go back the Custom CDS view list screen, we see that there are no outstanding rework tasks and our custom CDS is no longer flagged as being with Deprecation! We are done. Or are we?

Custom CDS Views (3)								Create	Copy	Delete	
Name	Scenario	Protection	Exported	Status	Task Priority	Rework	Task Category				
<input type="radio"/> YY1_PRE_API_00 Label: Deprecation API Sales CDS - 00	External API	Protected	No	Published	✘ High	<input checked="" type="checkbox"/>	Deprecation				
<input type="radio"/> YY1_PRE_DEPR_00 Label: Deprecation Custom CDS - Sales 00	Analytical Cube	Protected	No	Published							
<input type="radio"/> YY1_PRE_DEPR_01 Label: Deprecation Custom CDS - Sales 01	Analytical Cube	Protected	No	Published	✘ High	<input checked="" type="checkbox"/>	Deprecation				

Step 7 - Better safe than sorry

We have step by step tested our changes, so we could call it a day. However, in the spirit of being better safe than sorry, I would recommend doing a final pass, by once again launching the customer data browser, the Query report and eSAC story, to make sure that nothing was missed. If not, you are then ready to move your changes in the landscape until it gets to production.

SAP Customer Data Browser - Data Details

Parameters Total Records View Technical Names

Exchange Rate Type: M x Display Currency: EUR (Euro) x Sales Organization: 1010 (Dom. Sales Org DE) x Sold to Party: Product: Customer Material: Adapt Filters (3) Go

Search: YY1_PRE_DEPR_00 (100)

Sales Order	Sales Order Item	Sales Document Type	Sales Order Item Category	Created By	User Description	Record Created On	Month of Order	Sales Organization	Distribution Channel
0000386875	000010	TA	TAN	CB998005069	Mike Roza	11/30/2023	202311	1710	10
0000064331	000010	ZCM	TAN	SAP_SYSTEM	SAP System Pr...	04/30/2018	201804	1710	10
0000069062	000010	ZCM	TAN	SAP_SYSTEM	SAP System Pr...	06/12/2018	201806	1710	10
0000073455	000010	ZCM	TAN	SAP_SYSTEM	SAP System Pr...	07/25/2018	201807	1710	10
0000037702	000010	ZCM	TAN	SAP_SYSTEM	SAP System Pr...	07/14/2017	201707	1710	10
0000027336	000010	ZCM	TAN	SAP_SYSTEM	SAP System Pr...	02/06/2017	201702	1710	10
0000050113	000010	ZCM	TAN	SAP_SYSTEM	SAP System Pr...	12/05/2017	201712	1710	10
0000050697	000010	ZCM	TAN	SAP_SYSTEM	SAP System Pr...	12/18/2017	201712	1710	10
0000071324	000010	ZCM	TAN	SAP_SYSTEM	SAP System Pr...	07/04/2018	201807	1710	10
0000035984	000010	ZCM	TAN	SAP_SYSTEM	SAP System Pr...	06/08/2017	201706	1710	10

SAP Deprecation CAQ - Sales 00

Standard

Exchange Rate Type: M x Display Currency: EUR (Euro) x Sales Organization: 1010 (Dom. Sales Org DE) x Sold to Party: Product: Customer Material: Adapt Filters (3) Go

Data Analysis Graphical Display Query Information

<No Bookmark Loaded> Filter Sort Hierarchy Drilldown Display Measures Totals Navigate To

Item	Product	Plant	Customer Mater	Item Net Value	Cost	Amount Difference	Net Value	Ort
1	P002	1010	#	23,400.00 EUR	0.00 EUR	23,400.00 EUR	23,400.00 EUR	
10	TG11	1010	#	351.00 EUR	270.00 EUR	81.00 EUR	351.00 EUR	
10	TG11	1010	#	702.00 EUR	540.00 EUR	162.00 EUR	702.00 EUR	
1	P001	1010	#	50,000.00 EUR	0.00 EUR	50,000.00 EUR	50,000.00 EUR	
2	P002	1010	#	180,000.00 EUR	0.00 EUR	180,000.00 EUR	180,000.00 EUR	
3	P003	1010	#	20,000.00 EUR	0.00 EUR	20,000.00 EUR	20,000.00 EUR	
4	U001	1010	#	20,000.00 EUR	0.00 EUR	20,000.00 EUR	20,000.00 EUR	
10	TG12	DE20	HANACT1	17.55 EUR	15.80 EUR	1.75 EUR	17.55 EUR	
10	TG12	DE20	HANACT1	17.55 EUR	15.80 EUR	1.75 EUR	17.55 EUR	
10	TG12	DE20	HANACT1	17.55 EUR	15.80 EUR	1.75 EUR	17.55 EUR	
10	TG12	1710	#	17.55 EUR	15.80 EUR	1.75 EUR	17.55 EUR	
1	P002	1010	#	1,200.00 EUR	0.00 EUR	1,200.00 EUR	1,200.00 EUR	
1	P002	1010	#	42,000.00 EUR	0.00 EUR	42,000.00 EUR	42,000.00 EUR	
1	P002	1010	#	42,000.00 EUR	0.00 EUR	42,000.00 EUR	42,000.00 EUR	
1	P002	1010	#	1,000.00 EUR	1,000.00 EUR	0.00 EUR	1,000.00 EUR	
10	TG12	1010	#	52.65 EUR	40.50 EUR	12.15 EUR	52.65 EUR	
1	P002	1010	#	1,000.00 EUR	1,000.00 EUR	0.00 EUR	1,000.00 EUR	
10	TG11	DE20	#	35.10 EUR	27.10 EUR	8.00 EUR	35.10 EUR	
1	P003	1010	#	800.00 EUR	0.00 EUR	800.00 EUR	800.00 EUR	
10	ME7TG32	1010	#	3,500.00 EUR	332.50 EUR	3,167.50 EUR	3,500.00 EUR	
1	PIF1	1010	#	1,600.00 EUR	0.00 EUR	1,600.00 EUR	1,600.00 EUR	
2	PIF3	1010	#	2,400.00 EUR	0.00 EUR	2,400.00 EUR	2,400.00 EUR	
10	TG11	1010	#	35.10 EUR	26.98 EUR	8.12 EUR	35.10 EUR	
20	TG12	1010	HANACT1	35.10 EUR	26.98 EUR	8.12 EUR	35.10 EUR	
30	TG21	1010	#	35.10 EUR	27.54 EUR	7.56 EUR	35.10 EUR	

SAP Depreciated Story

Standard

File Edit Tools Display Page 1 1/1

Deprecated Sales Demo

YY1_PRE_DEPR_CAQ_00

Order Item Net Value Cost Amount Difference

Channel	Month of Order	Sold to Party	Sales Document Type	Sales Order	Sold to Party Name	Item	Product	Plant	Customer Material	Order Item Net Value	Cost	Amount Difference
	02/2021	PFSLC1001	S001	217596	Fastorb AG	1	P002	1010	#	EUR1,924,500.00	EUR0.00	EUR1,924,500.00
	11/2023	10100001	OR	385517	Hypercom AG	10	AVC_RBT_ROBOT2	1010	#	EUR995,000.00	EUR40,000.00	EUR955,000.00
		10100004	OR	386769	Inlandskunde DE 4	10	SM100	1010	#	EUR211,198.80	EUR0.00	EUR211,198.80
				386583	Inlandskunde DE 4	10	SM100	1010	#	EUR200,839.24	EUR0.00	EUR200,839.24
				386751	Inlandskunde DE 4	10	SM100	1010	#	EUR200,724.00	EUR0.00	EUR200,724.00
	06/2017	10100002	S001	36733	Technic	2	P002	1010	#	EUR180,000.00	EUR0.00	EUR180,000.00

Cost, Net Value per Sales Organization Net Value per Sold-to Party Name

Bar chart showing Net Value (blue) and Cost (orange) for various sales organizations. The chart shows a significant net value for 'Dom. Sales Org DE' (6,443,125.62) and a cost of 786,482.75.

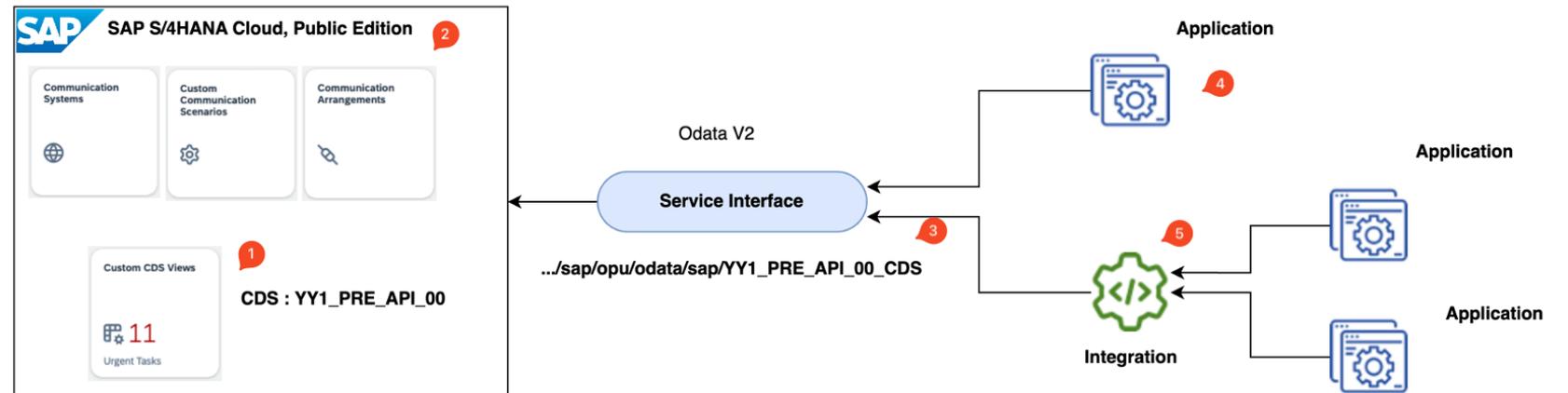
Introduction to the use case 2 - External API

Setting the scene

To work through this deprecation process, we have devised the following hypothetical storyline:

In our demo organisation, we have a landscape that requires us to make available on-demand some basic sales information to sister companies, for the purpose of consolidating sales information and forecasting. Some of these applications make a direct call to the SAP S/4HANA Cloud service, some leverage a middleware/integration layer in between. In order to provide this information, we created a custom CDS and then exposed it as an Odata v2 service. A high level diagram of this setup is shown below.

1. We created a custom CDS
2. Using the usual apps available to expose the CDS as a Odata service
3. That creates an URL entry point into our S/4HANA Cloud system
4. That is used by an application directly
5. Or via an integration layer (eg:BTP)



At the time of the requirement, based on the intended end-use of this service (on-demand querying of sales data) a custom CDS, with the scenario External API was created. All the data required was provided by a single CDS:

- **I_SalesOrderItemCube** - Sales Order Item - Cube, which provided us with sales order item data (dates, product, quantities, etc..)

The table below shows which fields were selected as Elements to be exposed via this CDS.

I_SalesOrderItemCube		
Alias	Path	Label
SalesOrder	I_SalesOrderItemCube.SalesOrder	Sales Order
SalesOrderItem	I_SalesOrderItemCube.SalesOrderItem	Item
SalesOrderType	I_SalesOrderItemCube.SalesOrderType	Sales Document Type
SalesOrderItemCategory	I_SalesOrderItemCube.SalesOrderItemCategory	Sls Order Item Cat.
CreationDateYearMonth	I_SalesOrderItemCube.CreationDateYearMonth	Yr/Mo. of Creation
SalesOrganization	I_SalesOrderItemCube.SalesOrganization	Sales Organization
DistributionChannel	I_SalesOrderItemCube.DistributionChannel	Distribution Channel
Material	I_SalesOrderItemCube.Material	Material
MaterialGroup	I_SalesOrderItemCube.MaterialGroup	Material Group
Plant	I_SalesOrderItemCube.Plant	Plant
SoldToParty	I_SalesOrderItemCube.SoldToParty	Sold-to Party
SoldToPartyName	I_SalesOrderItemCube.SoldToPartyName	Sold-to Party Name
OrderQuantity	I_SalesOrderItemCube.OrderQuantity	Order Quantity
OrderQuantityUnit	I_SalesOrderItemCube.OrderQuantityUnit	Sales Unit

Processing the CDS Deprecation

The deprecation announcement

Having in the previous chapter set the scene for our scenario, we now fast forward to a hypothetical upgrade of SAP S/4HANA Cloud, public edition and we have taken the time to take stock of the various announcements that go with such an event, and in particular the advertised deprecations. Being responsible for 'Data & Integration' we have in particular looked at the deprecations of CDS entities and CDS fields. We have done our initial due diligence:

- We have looked at the What's New viewer and filtered the data to look at deprecations of CDSs only,
- We have an understanding of the Deprecation Policy of CDSs, and we know that the Deprecated objects will still reside in our system for at least the next 12 months - this gives us time to plan and execute this deprecation, fully understanding which are the connected systems that call our custom API.
- Whilst not a mission critical API, we know that this custom API is, for the system that makes the most frequent use of it, called every Sunday night (so we can plan for rolling out the changes)

The diagram on the right, for the sake of setting the scene, shows how the deprecation announcements affect our Custom CDS. We have two deprecated fields that we need to replace with their Successors.

I_SalesOrderItemCube		
Alias	Path	Label
SalesOrder	I_SalesOrderItemCube.SalesOrder	Sales Order
SalesOrderItem	I_SalesOrderItemCube.SalesOrderItem	Item
SalesOrderType	I_SalesOrderItemCube.SalesOrderType	Sales Document Type
SalesOrderItemCategory	I_SalesOrderItemCube.SalesOrderItemCategory	Sls Order Item Cat.
CreationDateYearMonth	I_SalesOrderItemCube.CreationDateYearMonth	Yr/Mo. of Creation
SalesOrganization	I_SalesOrderItemCube.SalesOrganization	Sales Organization
DistributionChannel	I_SalesOrderItemCube.DistributionChannel	Distribution Channel
Material	I_SalesOrderItemCube.Material	Material
MaterialGroup	I_SalesOrderItemCube.MaterialGroup	Material Group
Plant	I_SalesOrderItemCube.Plant	Plant
SoldToParty	I_SalesOrderItemCube.SoldToParty	Sold-to Party
SoldToPartyName	I_SalesOrderItemCube.SoldToPartyName	Sold-to Party Name
OrderQuantity	I_SalesOrderItemCube.OrderQuantity	Order Quantity
OrderQuantityUnit	I_SalesOrderItemCube.OrderQuantityUnit	Sales Unit

Understanding the size of the rework

In the previous use case, we started this task by looking at the Custom CDS app. We could do so again here, but for the sake of variety, we will start by opening up the Extensibility Inventory App. We apply the filters to limit the scope to custom CDS deprecations.

The screenshot shows the SAP Extensibility Inventory application interface. At the top, there is a navigation bar with the SAP logo and 'Extensibility Inventory' dropdown. Below this, there are several filter fields: 'Item Description', 'Last Changed On' (with a date range 'e.g. 12/22/2023-12/31...'), 'Last Changed By', 'Deleted', 'Software Collection', 'Task Priority' (set to 'Low (4) x 3 More'), and 'Item Type' (set to 'Custom CDS Vie x'). A 'Task Category' dropdown is also visible, set to '2 Items'. A 'Go' button and 'Adapt Filters (3)' are located to the right of the filters. Below the filters, there is a table titled 'Items (10)' with a search bar and sorting options. The table has columns: Name, Type, Last Changed On, Uses, Used By, Uses SAP Objects, Task Priority, Rework Priority Sort ..., and Task Category. The table contains four rows of data, with the third row, 'Deprecation API Sales CDS - 00', highlighted with a red box and an arrow pointing to it.

Name	Type	Last Changed On	Uses	Used By	Uses SAP Objects	Task Priority	Rework Priority Sort ...	Task Category
Analytics - Sales Volume Cube YY1 BILLINGDOCUMENTITEMCUB	Custom CDS View	10/16/2023, 09:59:36 PM			7	⚠ Medium	2	Deprecation
DEMO_15516 G/L Account Line Item Cube YY1_GLACCOUNTLINEITEM	Custom CDS View	08/31/2018, 05:47:56 PM	1	1	19	⚠ Medium	2	Multiple Categories
Deprecation API Sales CDS - 00 YY1_PRE_API_00	Custom CDS View	11/27/2023, 10:46:03 AM		1	3	⚠ High	3	Deprecation
Deprecation Custom CDS - Sales 01 YY1_PRE_DFPR_01	Custom CDS View	11/22/2023, 06:10:55 PM		1	16	⚠ High	3	Deprecation

We can click on the line corresponding to our CDS, as highlighted in the previous picture. This will then open up the inventory view of our CDS, and already here, we see that our CDS is used in a Custom Communication Scenario.

If we then click on the bound box containing the name of the custom CDS, the box will change colour (1) and reveal a couple of icons. One of which (2) will allow us to navigate to the custom CDS app. Let's click on it to go to the Custom CDS app.

As soon as we get into the detail screen of the custom CDS, there is no evident deprecation information being brought to our attention. However, we have in the top left an indication that Dependencies exist and Compatible changes are allowed. This means that this custom CDS is used by another object and we can only make changes that will not affect or break the object that is using our custom CDS. For example, you would not want to be able to just delete a field from your custom CDS, if it was used by, as in this case a communication scenario!

Clicking on the hyperlink titled (Compatible Changes Allowed), will open a pop-up window showing you the actual name and object type of the object(s) that uses the custom CDS.

The image shows a sequence of three screenshots illustrating the navigation and dependency information in SAP S/4HANA.

Top Screenshot: Shows a 'Custom CDS View' for 'Deprecation API Sales CDS - 00' with 'High Deprecation' status. An arrow points to a 'Custom Communication Scenario' for 'Communication Scenario CDS PRE'.

Middle Screenshot: The 'Custom CDS View' is highlighted in blue (1). A red circle with '2' highlights the navigation icon (a double right arrow) next to the CDS name.

Bottom Screenshot: Shows the detail screen for 'Deprecation API Sales CDS - 00'. A red arrow points to the 'Dependencies: 1 (Compatible Changes Allowed)' link. A pop-up window shows the following table:

Dependent Objects			
Dependencies (1)			
View Name	Dependent Object	Draft Indicator	Publish Indicator
YY1_PRE_API_DEP_00	Custom Communication Scenario	No	Yes

Below the table, the following information is displayed:

- Name: YY1_PRE_API_00
- Representative Key: Not required
- Scenario: External API
- Published By: [User]
- Published On: 11/27/2023, 10:46:02 AM
- Protection: Protected
- Created: 11/24/2023
- Status: Published

At the bottom, the 'Data Sources (1)' table is shown:

Name	Alias	Type	Cardinality	Tasks
I_SalesOrderItemCube	I_SalesOrderItemCube	Primary Datasource		

We then move on to the Tasks tab, and in the Rework Tasks section, get more prescriptive information, clearly telling us what is deprecated as well as the successor object(s). Note that in this case we have rework tasks that pertain to CDS elements only.

Scenario: External API Protection: Protected Status: Published

Navigation: Data Sources >> Parameters >> Elements >> Element Properties >> Filter >> Log >> **Tasks** >> More

Rework Tasks (2)

Task Priority	Task Category	Description	Registered On
High	Deprecation	Element Material of CDS view I_SalesOrderItemCube is deprecated. Use element PRODUCT instead.	11/24/2023, 11:28:12 AM
High	Deprecation	Element MaterialGroup of CDS view I_SalesOrderItemCube is deprecated. Use element PRODUCTGROUP instead.	11/24/2023, 11:28:12 AM

On the basis of the information found in the Tasks tab, we can update our Custom CDS data schema, and add to it the Successor information we have now confirmed. Next to the deprecated objects, we have added the successor information.

I_SalesOrderItemCube		
Alias	Path	Label
SalesOrder	I_SalesOrderItemCube.SalesOrder	Sales Order
SalesOrderItem	I_SalesOrderItemCube.SalesOrderItem	Item
SalesOrderType	I_SalesOrderItemCube.SalesOrderType	Sales Document Type
SalesOrderItemCategory	I_SalesOrderItemCube.SalesOrderItemCategory	Sls Order Item Cat.
CreationDateYearMonth	I_SalesOrderItemCube.CreationDateYearMonth	Yr/Mo. of Creation
SalesOrganization	I_SalesOrderItemCube.SalesOrganization	Sales Organization
DistributionChannel	I_SalesOrderItemCube.DistributionChannel	Distribution Channel
Material	I_SalesOrderItemCube.Material	Material
MaterialGroup	I_SalesOrderItemCube.MaterialGroup	Material Group
Plant	I_SalesOrderItemCube.Plant	Plant
SoldToParty	I_SalesOrderItemCube.SoldToParty	Sold-to Party
SoldToPartyName	I_SalesOrderItemCube.SoldToPartyName	Sold-to Party Name
OrderQuantity	I_SalesOrderItemCube.OrderQuantity	Order Quantity
OrderQuantityUnit	I_SalesOrderItemCube.OrderQuantityUnit	Sales Unit

PRODUCT
PRODUCTGROUP

Deprecated Object
Successor Object

With respect to the deprecated CDS elements, it could be that the successor CDS elements are different. For sure we know the name will be different, but it can also rarely happen that the data format will be different - for example, is the length of the successor field still X characters long? Is the field still of type CHAR? As this can have an effect on how the data will be presented and interpreted in the calling application, this needs to be checked. As we covered earlier, you could do this by checking the View Browser app in your system, or you could check the [SAP Business Accelerator Hub](#), or you could use the Excel tool we have purposefully created for this purpose. You can [download it here](#). We used this tool to pull the CDS entity information. We can see that the successor information is advertised (we know which field is the successor) so we can quickly compare it with the deprecated field.

We can below, see that our deprecated and successor fields are identical in terms of data format and length. That will be one less thing to worry about!

CDS Comparison					
Material	Material Number	Deprecated	Product	CHAR	40
Product	Product Number			CHAR	40
MaterialGroup	Material Group	Deprecated	ProductGroup	CHAR	9
ProductGroup	Product Group			CHAR	9

We can also look at the Elements tab, which also provides us with essential information. We can see:

1. A clear indication of those fields that are deprecated, as well as an indication of the successor field that we should be using.

The screenshot shows the SAP Custom CDS View Details interface for 'Deprecation API Sales CDS - 00'. The 'Elements' tab is active, displaying a table of 14 elements. The 'Material' element is highlighted with a red box, and a red circle with the number '1' is placed above its 'Sales Document Type' field. A tooltip window is open over this field, providing information about the deprecation. The tooltip contains the following text:

Information

There is additional information for this element

Label: Material

Alias: Material

Deprecated Element

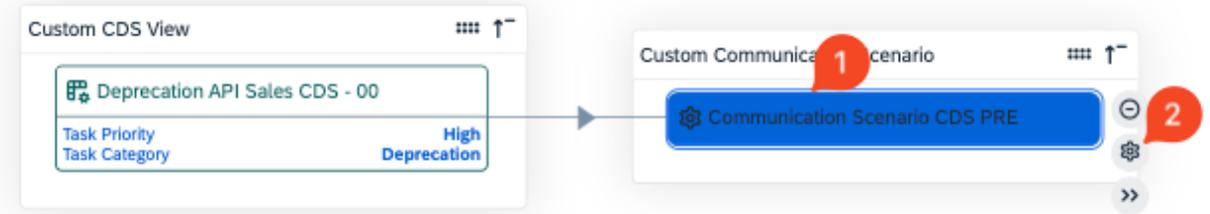
The field is deprecated and should not be used. Please use the successor instead.

Successor: Product

The table below shows the elements listed in the screenshot:

Key	Alias	Type	Path	Label	Calculation	Task	Successor
<input type="checkbox"/>	<input checked="" type="checkbox"/> ON	SalesOrder	CHAR (10)	I_SalesOrderItemCube.SalesOrder			
<input type="checkbox"/>	<input checked="" type="checkbox"/> ON	SalesOrderItem	NUMC (6)	I_SalesOrderItemCube.SalesOrderItem			
<input type="checkbox"/>	<input type="checkbox"/> OFF	SalesOrderType	CHAR (4)	I_SalesOrderItemCube.SalesOrderType			Sales Document Type
<input type="checkbox"/>	<input type="checkbox"/> OFF	SalesOrderItemCategory	CHAR (4)	I_SalesOrderItemCube.SalesOrderItemCategory			
<input type="checkbox"/>	<input type="checkbox"/> OFF	CreationDateYearMonth	NUMC (6)	I_SalesOrderItemCube.CreationDateYearMonth			
<input type="checkbox"/>	<input type="checkbox"/> OFF	SalesOrganization	CHAR (4)	I_SalesOrderItemCube.SalesOrganization			
<input type="checkbox"/>	<input type="checkbox"/> OFF	DistributionChannel	CHAR (2)	I_SalesOrderItemCube.DistributionChannel			
<input type="checkbox"/>	<input type="checkbox"/> OFF	Material	CHAR (40)	I_SalesOrderItemCube.Material			Product
<input type="checkbox"/>	<input type="checkbox"/> OFF	MaterialGroup	CHAR (9)	I_SalesOrderItemCube.MaterialGroup			ProductGroup
<input type="checkbox"/>	<input type="checkbox"/> OFF	Plant	CHAR (4)	I_SalesOrderItemCube.Plant			
<input type="checkbox"/>	<input type="checkbox"/> OFF	SoldToParty	CHAR (10)	I_SalesOrderItemCube.SoldToParty			
<input type="checkbox"/>	<input type="checkbox"/> OFF	SoldToPartyName	CHAR (80)	I_SalesOrderItemCube.SoldToPartyName			
<input type="checkbox"/>	<input type="checkbox"/> OFF	OrderQuantity	QUAN (15,3)	I_SalesOrderItemCube.OrderQuantity			Order Quantity

We can then navigate back to the Extensibility Inventory app. This time, we will select the bound box holding the name of the custom communication scenario, which will change its colour (1) and click on the navigation icon (2) as shown to navigate to the Custom Communication Scenario.



Once we reach the Custom Communication scenario, we see further important information such as the inbound service ID, as well as the communication arrangements that can make use of this inbound service! This is of course important as it will help you to identify which systems have been setup to access this service and verify if and how they also need to be amended, so as to continue to be able to parse the information the service will deliver once it is changed!

Communication Scenario CDS PRE Edit Check Create Arrangement Display Publishing Process ...

YY1_PRE_API_DEP_00

Last Changed By: Last Changed On: 11/27/2023 Exported: No

Status: **Published** Editing Status: **Active** Allowed Instances: One Instance per Scenario and Communication System

Inbound Services (1) **Outbound Services (0)** **Authorizations (0)** **Communication Arrangements (2)**

Inbound Services (1)

Inbound ID	Description	Status	Service Type	Last Changed On	Last Changed By
YY1_PRE_API_00_CDS	Deprecation API Sales CDS - 00		OData V2	11/27/2023	SAP System Processing

Communication Arrangements (2)

Communication Arrangements	Communication System	Status
YY1_PRE_API_DEP_00	PREMEL	Active
YY1_PRE_API_DEP_01	ITTECH	Active

Much like the previous use case, before you make any changes on the SAP side, we would recommend that you extract a sample of your data so as to be able to make before/after comparisons of your changes. In this case, as we are looking at an API, we would recommend that you make an extract of your data, using the Data Browser (directly from the CDS), as well as sample payloads, to make sure/check that the syntax is still the same, or you are able to identify the differences. You would also want to have sufficient data to verify each of your scenarios. I.e do you have Materials codes that have different formats (some are only numerical, some are character or some are alphanumeric). In our case we have made several extracts of both xml and json payloads.

```
GET https://my300448-api.s4hana.ondemand.com/sap/opu/odata/sap/YY1_PRE_API_00_CDS/YY1_PRE_API_00/?$top=5&$format=json

Body Cookies (2) Headers (15) Test Results Status: 200 OK Time: 2.13 s Size: 4.12 KB

Pretty Raw Preview Visualize JSON

1  "d": {
2
3    "results": [
4      {
5        "_metadata": {
6          "id": "https://my300448-api.s4hana.ondemand.com/sap/opu/odata/sap/YY1_PRE_API_00_CDS/YY1_PRE_API_00('1-0000000001.2-000010')",
7          "uri": "https://my300448-api.s4hana.ondemand.com/sap/opu/odata/sap/YY1_PRE_API_00_CDS/YY1_PRE_API_00('1-0000000001.2-000010')",
8          "type": "YY1_PRE_API_00_CDS.YY1_PRE_API_00Type"
9        },
10       "ID": "1-0000000001.2-000010",
11       "SalesOrder": "1",
12       "SalesOrderItem": "000010",
13       "SalesOrderType": "OR",
14       "SalesOrderItemCategory": "TAN",
15       "CreationDateYearMonth": "201608",
16       "SalesOrganization": "1710",
17       "DistributionChannel": "10",
18       "Material": "TG11",
19       "MaterialGroup": "L001",
20       "Plant": "1710",
21       "SoldToParty": "17100001",
22       "SoldToPartyName": "Domestic US Customer 1",
23       "OrderQuantity": "3",
24       "OrderQuantityUnit": "PC"
25     }
26   ]
27 }
```

```
GET https://my300448-api.s4hana.ondemand.com/sap/opu/odata/sap/YY1_PRE_API_00_CDS/YY1_PRE_API_00/?$top=5

Body Cookies (2) Headers (13) Test Results Status: 200 OK Time: 576 ms

Pretty Raw Preview Visualize XML

1  <feed xmlns="http://www.w3.org/2005/Atom" xmlns:m="http://schemas.microsoft.com/ado/2007/08/dataservices/metadata"
2    xmlns:d="http://schemas.microsoft.com/ado/2007/08/dataservices" xml:base="https://my300448-api.s4hana.ondemand.com/sap/opu/odata/sap/YY1_PRE_API_00_CDS/YY1_PRE_API_00">
3    <id>https://my300448-api.s4hana.ondemand.com/sap/opu/odata/sap/YY1_PRE_API_00_CDS/YY1_PRE_API_00</id>
4    <title type="text">YY1_PRE_API_00</title>
5    <updated>2023-12-01T05:45:59Z</updated>
6    <author>
7      <name/>
8    </author>
9    <link href="YY1_PRE_API_00/" rel="self" title="YY1_PRE_API_00"/>
10   <entry>
11     <id>https://my300448-api.s4hana.ondemand.com/sap/opu/odata/sap/YY1_PRE_API_00_CDS/YY1_PRE_API_00('1-0000000001.2-000010')</id>
12     <title type="text">YY1_PRE_API_00('1-0000000001.2-000010')</title>
13     <updated>2023-12-01T05:45:59Z</updated>
14     <category term="YY1_PRE_API_00_CDS.YY1_PRE_API_00Type" scheme="http://schemas.microsoft.com/ado/2007/08/dataservices/metadata" rel="self" title="YY1_PRE_API_00Type"/>
15     <content type="application/xml">
16       <m:properties xmlns:m="http://schemas.microsoft.com/ado/2007/08/dataservices/metadata" xmlns:d="http://schemas.microsoft.com/ado/2007/08/dataservices">
17         <d:ID>1-0000000001.2-000010</d:ID>
18         <d:SalesOrder>1</d:SalesOrder>
19         <d:SalesOrderItem>000010</d:SalesOrderItem>
20         <d:SalesOrderType>OR</d:SalesOrderType>
21         <d:SalesOrderItemCategory>TAN</d:SalesOrderItemCategory>
22         <d:CreationDateYearMonth>201608</d:CreationDateYearMonth>
23         <d:SalesOrganization>1710</d:SalesOrganization>
24         <d:DistributionChannel>10</d:DistributionChannel>
25         <d:Material>TG11</d:Material>
26         <d:MaterialGroup>L001</d:MaterialGroup>
27         <d:Plant>1710</d:Plant>
28         <d:SoldToParty>17100001</d:SoldToParty>
29         <d:SoldToPartyName>Domestic US Customer 1</d:SoldToPartyName>
30         <d:OrderQuantity>3</d:OrderQuantity>
31         <d:OrderQuantityUnit>PC</d:OrderQuantityUnit>
32       </m:properties>
33     </content>
34   </entry>
35 </feed>
```

We have also made a customer data browser extract that includes at least the deprecated fields.

Parameters Total Records View Technical Names

YY1_PRE_API_00 (100)

Search

Deprecated Fields

Sales Order	Sales Order Item	Sales Document Type	Material Number	Material Group
0000059950	000010	ZCM	MZ-TG-Y240	ZYOUTH
0000060175	000010	ZCM	MZ-TG-Y240	ZYOUTH
0000060288	000010	ZCM	MZ-TG-Y240	ZYOUTH
0000061980	000010	ZCM	MZ-TG-Y240	ZYOUTH
0000060527	000010	ZCM	MZ-TG-Y240	ZYOUTH
0000063468	000010	ZCM	MZ-TG-Y240	ZYOUTH
0000062180	000010	ZCM	MZ-TG-Y240	ZYOUTH
0000060731	000010	ZCM	MZ-TG-Y240	ZYOUTH
0000063883	000010	ZCM	MZ-TG-Y240	ZYOUTH
0000063983	000010	ZCM	MZ-TG-Y240	ZYOUTH

The Process

Now that we have done our analysis, we know what we need to change and where we need to change things, and we have also made backups of our data so we can make before change and after change comparisons at every step of the way.

Logically, one might think that we will start at the bottom and make our way up the chain of objects. By that I mean that, one might think that it would be easy enough to remove the deprecated fields from the custom CDS, add the successor fields in the custom CDS, save and publish and we are done. Unfortunately, in practice it does not work quite like this, because our custom communication scenario is using all the fields from our CDS.

To illustrate this, let's for example go to our custom CDS into edit mode and try to remove one of the deprecated fields —> **Material**. As you can see, the Delete button remains invariably inaccessible - because deleting this field would be an incompatible change (as the field is used in the Custom Communication Scenario). However, we can see that the Add option is available, so we can add - successor fields.

Deprecation API Sales CDS - 00
Dependencies: 1 (Compatible Changes Allowed) Draft

Navigation: Data Sources >> Parameters >> **Elements** >> Element Properties >> Filter >> Log >> Tasks >> Next Steps

Elements (14) [Add](#) [Delete](#) [Settings](#) [Refresh](#)

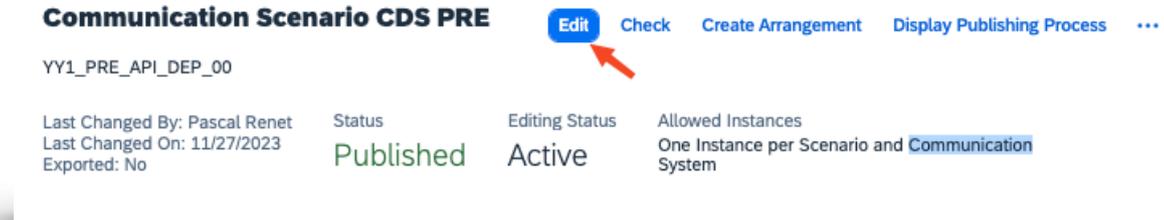
<input type="checkbox"/>	Key	Alias	Type	Path	Label	Calculation	Task
<input type="checkbox"/>	OFF	DistributionChannel	CHAR (2)	I_SalesOrderItemCube.DistributionChannel	Distribution Channel		
<input checked="" type="checkbox"/>	OFF	Material	CHAR (40)	I_SalesOrderItemCube.Material	Material		⚠
<input checked="" type="checkbox"/>	OFF	MaterialGroup	CHAR (9)	I_SalesOrderItemCube.MaterialGroup	Material Group		⚠
<input type="checkbox"/>	OFF	Plant	CHAR (4)	I_SalesOrderItemCube.Plant	Plant		

So, like in the previous use case, we must do a change somewhere else first that will then allow us to remove the deprecated fields from the Custom CDS. That somewhere else, is the custom communication scenario.

In the previous use case, we started by making some changes in the custom CDS (we added the successor field), then went to the analytical query, and it is only at the very end that we came back to the CDS to finally remove the deprecated fields. This time we will operate differently and leave the custom CDS alone for the time being.

Step 1 - Amending the Custom communication scenario

We start by going to the Custom Communication scenario app because that is what is keeping us from changing the underlying custom CDS. We go into our communication scenario, and switch to edit mode, so that we can make some changes to it.



Communication Scenario CDS PRE Edit Check Create Arrangement Display Publishing Process ...

YY1_PRE_API_DEP_00

Last Changed By: Pascal Renet
Last Changed On: 11/27/2023
Exported: No

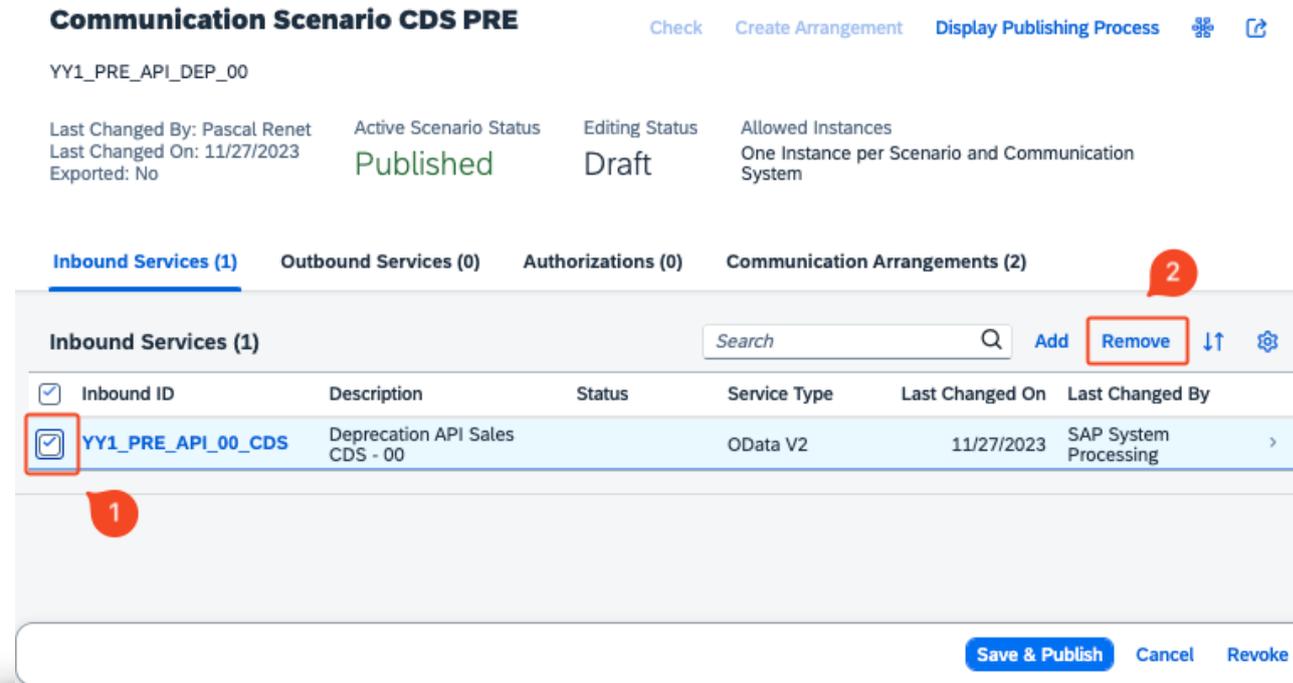
Status: **Published**

Editing Status: **Active**

Allowed Instances: One Instance per Scenario and **Communication System**

We are then going to:

1. Select the inbound service and
2. Click Remove, to remove it



Communication Scenario CDS PRE Check Create Arrangement Display Publishing Process

YY1_PRE_API_DEP_00

Last Changed By: Pascal Renet
Last Changed On: 11/27/2023
Exported: No

Active Scenario Status: **Published**

Editing Status: **Draft**

Allowed Instances: One Instance per Scenario and Communication System

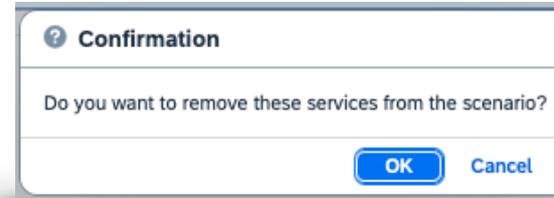
Inbound Services (1) Outbound Services (0) Authorizations (0) Communication Arrangements (2)

Inbound Services (1) Search Add Remove

<input checked="" type="checkbox"/>	Inbound ID	Description	Status	Service Type	Last Changed On	Last Changed By
<input checked="" type="checkbox"/>	YY1_PRE_API_00_CDS	Deprecation API Sales CDS - 00		OData V2	11/27/2023	SAP System Processing

Save & Publish Cancel Revoke

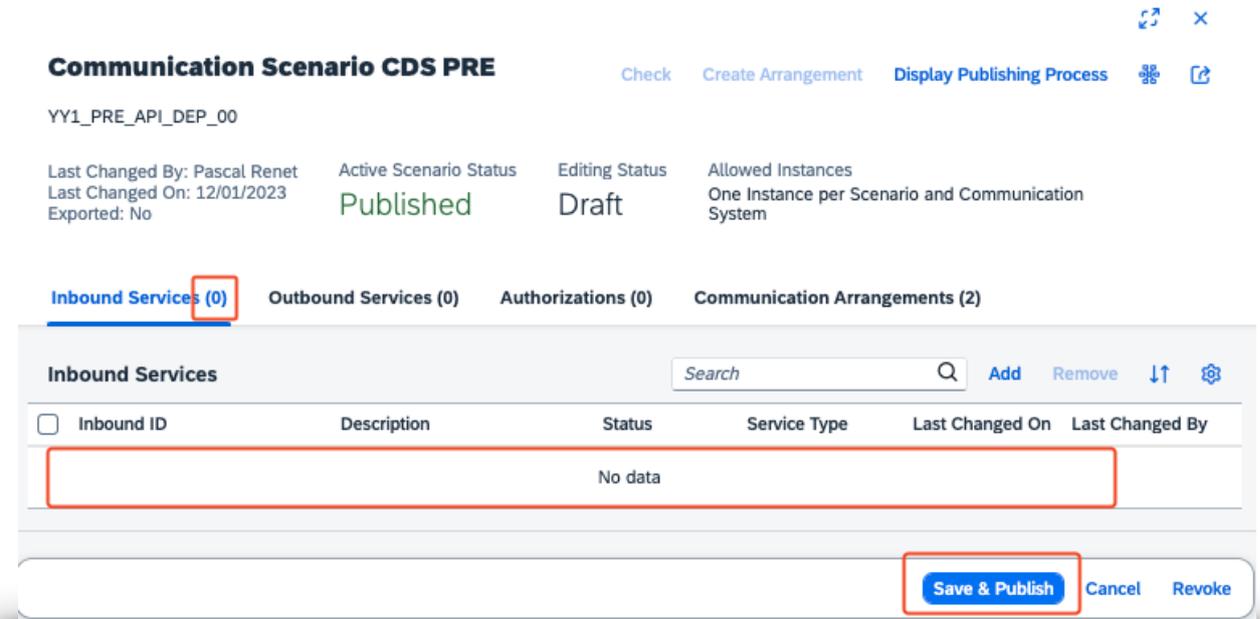
You will then be presented with this popup. Click OK



The Inbound service will then be removed.

Save your changes, and wait for the publishing process to finish.

At this moment, the custom CDS is no longer used by the communication scenario, so we return to the custom CDS app to proceed with the changes we want to make there.

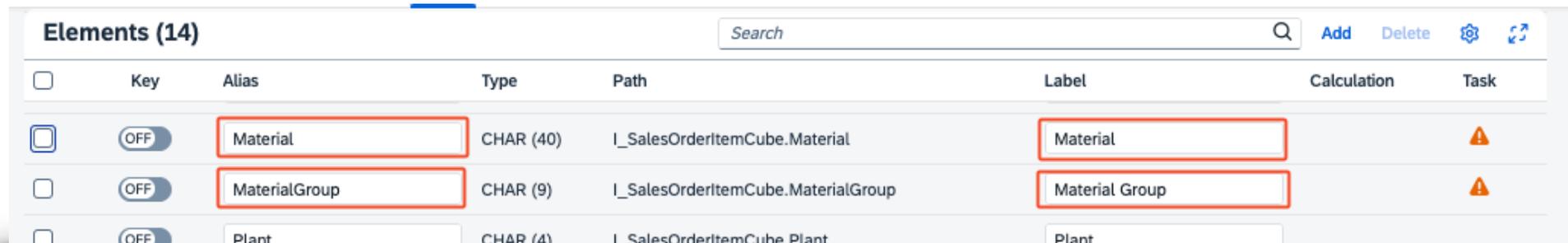
The interface for a "Communication Scenario CDS PRE" is shown. At the top right are icons for refresh and close. Below the title are action buttons: "Check", "Create Arrangement", and "Display Publishing Process". The scenario ID is "YY1_PRE_API_DEP_00". A metadata section shows: "Last Changed By: Pascal Renet", "Last Changed On: 12/01/2023", "Exported: No", "Active Scenario Status: Published", "Editing Status: Draft", and "Allowed Instances: One Instance per Scenario and Communication System". A navigation bar shows "Inbound Services (0)", "Outbound Services (0)", "Authorizations (0)", and "Communication Arrangements (2)". The "Inbound Services" section is active, showing a table with columns: Inbound ID, Description, Status, Service Type, Last Changed On, and Last Changed By. The table is currently empty, displaying "No data". At the bottom right, there are buttons for "Save & Publish", "Cancel", and "Revoke".

Step 2 - Amending the Custom CDS

We open the Custom CDS app and switch to edit mode, to start making changes. Before we start making changes, we can ask ourselves how we want to proceed, and the answer to this question might depend on the amount of fields we are going to change and the kind of fields we are going to change (i.e do we have calculations that we need to check). If you recall, in the previous use case, we added the successor fields, but did not right away remove the deprecated ones...because we want to be able to compare them step by step. In this case it is a little different because a) I have already saved a sample of my data before making any changes (for comparison) and b) the next time my data will be used, will be by an external application and c) if I can, I want to avoid having to make ANY changes to my 3rd party applications or integration mappings.

Another thing to consider is that I don't really want to spend more time than I need to changing my integration flow. By that I mean that if I remove the field **MATERIAL** and replace it with the field **PRODUCT**, my integration flow will break. I.e the calling application will be expecting to receive in the payload an element with the name `<Material>` but will not receive it. Can we avoid this? We had the same conundrum in the previous use case, where we could have changed the label of the UI field **Product** with **Material** (keeping the old screen name), so as to not perturb the business users, but we decided that it was best to change and use the new name. However in this case it is different. Does the calling application really need to know that this has changed? Probably not. So in this case, whilst we will be removing the deprecated fields and changing them to use the successors, we will keep the 'old - deprecated' name in the integration.

So now that we are clear, let's go to the Elements tab of our custom CDS and first note the **Alias** and **Label** of the deprecated fields that we are going to remove.



Elements (14)									
	Key	Alias	Type	Path	Label	Calculation	Task		
<input checked="" type="checkbox"/>	<input type="checkbox"/>	Material	CHAR (40)	I_SalesOrderItemCube.Material	Material				
<input type="checkbox"/>	<input type="checkbox"/>	MaterialGroup	CHAR (9)	I_SalesOrderItemCube.MaterialGroup	Material Group				
<input type="checkbox"/>	<input type="checkbox"/>	Plant	CHAR (4)	I_SalesOrderItemCube.Plant	Plant				

Then, select the deprecated fields and Delete them to remove them from the custom CDS. Click ok on the popup that will be presented to confirm. Click 'Check' to generate a draft.

The screenshot shows the 'Elements (14)' view in SAP Fiori. The breadcrumb navigation includes Data Sources, Parameters, Elements, Element Properties, Filter, Log, Tasks, and Next Steps. The table below lists elements with columns for Key, Alias, Type, Path, Label, Calculation, and Task. Two rows are selected: 'Material' and 'MaterialGroup'. A red box highlights the 'Delete' button in the top right corner, and a red arrow points from it to the selected rows.

Key	Alias	Type	Path	Label	Calculation	Task
<input checked="" type="checkbox"/>	Material	CHAR (40)	I_SalesOrderItemCube.Material	Material		⚠
<input checked="" type="checkbox"/>	MaterialGroup	CHAR (9)	I_SalesOrderItemCube.MaterialGroup	Material Group		⚠

Now click 'Add' to add the successor elements (which were Product and Product Group).

NOTE: It is worthwhile noting that the successor fields we just added will be tacked to the end of the CDS entity (i.e. not positioned in the same place where the deprecated fields were). This is done on purpose, but we will later show how you can move the placement of fields.

The screenshot shows the 'Select: Elements' dialog box. It includes a search bar and a 'Show Used' dropdown set to 'No'. The 'Items' table lists various elements, with 'Product' and 'ProductGroup' selected. Below the table, the 'Selected Items and Conditions (2)' section shows 'Product' and 'ProductGroup' in a list. The 'OK' button is highlighted in blue.

Alias	Key	Type	Label	Status	Used
<input checked="" type="checkbox"/> Product		CHAR (40)	Product		
<input type="checkbox"/> OriginallyRequestedMaterial		CHAR (40)	Material Entered		
<input type="checkbox"/> MaterialByCustomer		CHAR (35)	Customer Material		
<input type="checkbox"/> InternationalArticleNumber		CHAR (18)	EAN/UPC		
<input type="checkbox"/> Batch		CHAR (10)	Batch		
<input type="checkbox"/> ProductHierarchyNode		CHAR (18)	Product Hierarchy		
<input type="checkbox"/> ProductCatalog		CHAR (10)	Product Catalog		
<input type="checkbox"/> MaterialSubstitutionReason		CHAR (4)	Substitution Reason		
<input type="checkbox"/> MaterialGroup		CHAR (9)	Material Group	⚠	
<input checked="" type="checkbox"/> ProductGroup		CHAR (9)	Product Group		

Selected Items and Conditions (2): Product × ProductGroup ×

OK Cancel

Our successor fields are now added to the Custom CDS.

We now need to rename them, so that their names (alias and label) are identical to the deprecated fields we removed.

We now have field aliases and labels that are identical to the deprecated fields, but the source of the data is indeed the successor element! Do a check, and if OK, publish your changes.

The screenshot shows the 'Elements (14)' table in SAP. The table has columns: Key, Alias, Type, Path, Label, Calculation, and Task. Three rows are visible, with the last two highlighted by a red box:

Key	Alias	Type	Path	Label	Calculation	Task
<input type="checkbox"/>	OrderQuantityUnit	UNIT (3)	I_SalesOrderItemCube.OrderQuantityUnit	Sales Unit		
<input type="checkbox"/>	Product	CHAR (40)	I_SalesOrderItemCube.Product	Product		
<input type="checkbox"/>	ProductGroup	CHAR (9)	I_SalesOrderItemCube.ProductGroup	Product Group		

The screenshot shows the 'Elements (14)' table in SAP after updates. The table has columns: Key, Alias, Type, Path, Label, Calculation, and Task. Three rows are visible, with the last two highlighted by a red box. The 'Alias' and 'Label' fields for the last two rows have been updated to 'Material' and 'Material Group' respectively. The 'Path' fields for the last two rows have been updated to 'I_SalesOrderItemCube.Product' and 'I_SalesOrderItemCube.ProductGroup' respectively.

Key	Alias	Type	Path	Label	Calculation	Task
<input type="checkbox"/>	OrderQuantityUnit	UNIT (3)	I_SalesOrderItemCube.OrderQuantityUnit	Sales Unit		
<input type="checkbox"/>	Material	CHAR (40)	I_SalesOrderItemCube.Product	Material		
<input type="checkbox"/>	MaterialGroup	CHAR (9)	I_SalesOrderItemCube.ProductGroup	Material Group		

Before we leave the custom CDS, let's do a before after change comparison to make sure the required data is still coming through, by using the Data Browser. We can see that the before change (left) data we saved is identical to the after (right) change data. We also see that the UI name of the successor fields is showing the Successor names, but if we switch to the Technical Names view, we see that the deprecated technical names are used to describe the successor fields !

Parameters Total Records View Technical Names

YY1_PRE_API_00 (100)

Search

Deprecated Fields

Sales Order	Sales Order Item	Sales Document Type	Material Number	Material Group
0000059950	000010	ZCM	MZ-TG-Y240	ZYOUTH
0000060175	000010	ZCM	MZ-TG-Y240	ZYOUTH
0000060288	000010	ZCM	MZ-TG-Y240	ZYOUTH
0000061980	000010	ZCM	MZ-TG-Y240	ZYOUTH
0000060527	000010	ZCM	MZ-TG-Y240	ZYOUTH
0000063468	000010	ZCM	MZ-TG-Y240	ZYOUTH
0000062180	000010	ZCM	MZ-TG-Y240	ZYOUTH
0000060731	000010	ZCM	MZ-TG-Y240	ZYOUTH
0000063883	000010	ZCM	MZ-TG-Y240	ZYOUTH
0000063983	000010	ZCM	MZ-TG-Y240	ZYOUTH

Parameters Total Records View Technical Names

YY1_PRE_API_00 (100)

Search

Successor Fields

Sales Order	Sales Order Item	Sales Document Type	Product Group	Product Number
0000059950	000010	ZCM	ZYOUTH	MZ-TG-Y240
0000060175	000010	ZCM	ZYOUTH	MZ-TG-Y240
0000060288	000010	ZCM	ZYOUTH	MZ-TG-Y240
0000061980	000010	ZCM	ZYOUTH	MZ-TG-Y240
0000060527	000010	ZCM	ZYOUTH	MZ-TG-Y240
0000063468	000010	ZCM	ZYOUTH	MZ-TG-Y240
0000062180	000010	ZCM	ZYOUTH	MZ-TG-Y240
0000060731	000010	ZCM	ZYOUTH	MZ-TG-Y240
0000063883	000010	ZCM	ZYOUTH	MZ-TG-Y240
0000063983	000010	ZCM	ZYOUTH	MZ-TG-Y240

Search

Sales Order Item	Sales Document Type	Material Group	Material
000010	ZCM	ZYOUTH	MZ-TG-Y240

Lastly, we can also confirm in the custom CDS list view that, our custom CDS no longer has any Deprecation rework tasks assigned to it !

My CDS ▼ Delete Non-Migrated Views Migrate Views 10

Show Filter Bar Filters (1)

↑ ★

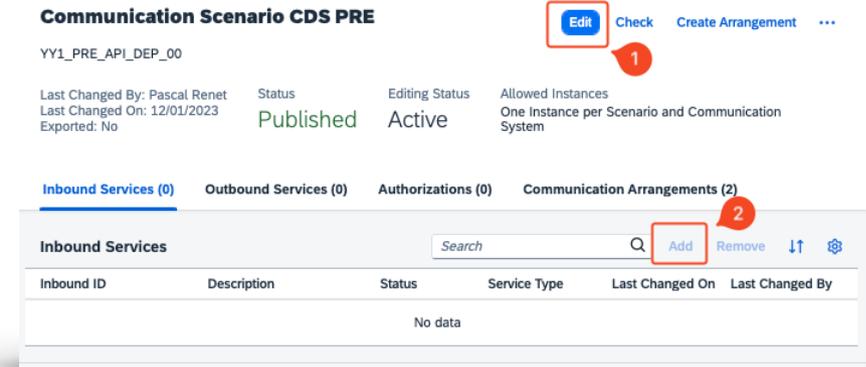
Custom CDS Views (3) Create Copy Delete Settings (Cmd+,)

Name	Scenario	Protection	Exported	Status	Task Priority
<input type="radio"/> YY1_PRE_API_00 <small>Label: Deprecation API Sales CDS - 00</small>	External API	Protected	No	Published	<div style="border: 2px solid red; width: 100px; height: 40px; display: flex; align-items: center; justify-content: center;">></div>
<input type="radio"/> YY1_PRE_DEPR_00 <small>Label:</small>	Analytical Cube	Protected	No	Published	>

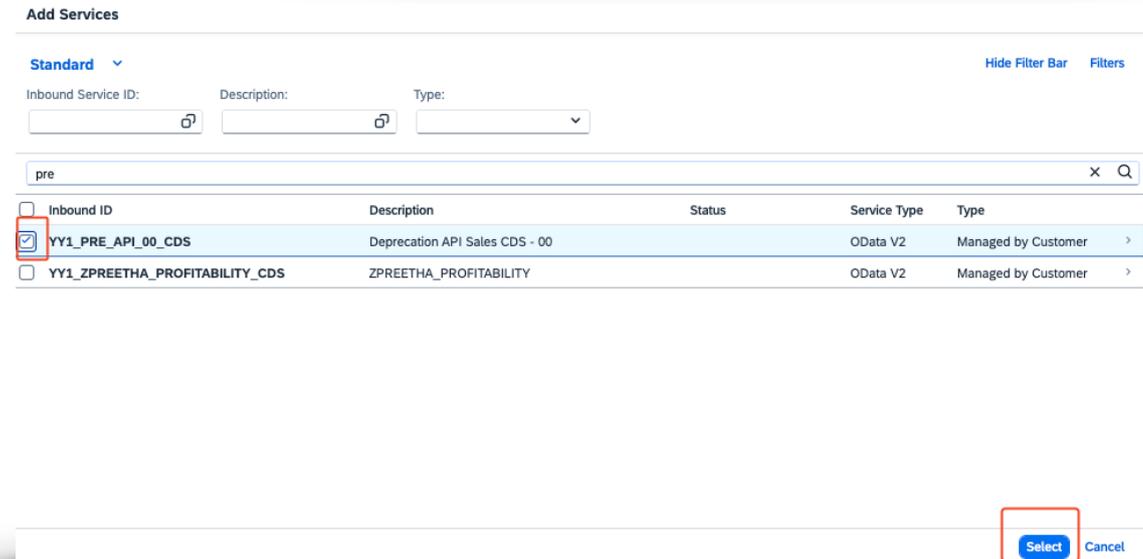
Step 3 - Amending the Custom Communication Scenario

Our Custom CDS changes completed and after having confirmed that the correct data is still coming through, we open the Custom Communication App, so as to once again add the inbound service back to it.

We start by switching to edit mode (1), and then, once it becomes active, we select Add (2).



Search and select your Custom CDS and click the Select button.



Once your Custom CDS appears in the Inbound Services section, click Save & Publish.

Communication Scenario CDS PRE [Check](#) [Create Arrangement](#) [Display Publishing Process](#) [...](#)

YY1_PRE_API_DEP_00

Last Changed By: Pascal Renet Active Scenario Status: **Published** Editing Status: Draft
Last Changed On: 12/01/2023
Exported: No
Allowed Instances: One Instance per Scenario and Communication System

Inbound Services (1) Outbound Services (0) Authorizations (0) Communication Arrangements (2)

Inbound Services (1) [Add](#) [Remove](#) [↑↓](#) [⚙️](#)

<input type="checkbox"/>	Inbound ID	Description	Status	Service Type	Last Changed On
<input type="checkbox"/>	YY1_PRE_API_00_CDS	Deprecation API Sales CDS - 00		OData V2	11/24/2023 >

Last Changed By: SAP_SYSTEM

[Save & Publish](#) [Cancel](#) [Revoke](#)

You then have to wait for the publishing to finish. You can monitor the publishing by selecting the 'Display Publishing Process' or wait for the Status to change to Published.

Communication Scenario CDS PRE [Edit](#) [Check](#) [Create Arrangement](#) [...](#)

YY1_PRE_API_DEP_00

Last Changed By: Pascal Renet Status: **Publishing** Editing Status: **Published**
Last Changed On: 12/01/2023
Exported: No
Allowed Instances: One Instance per Scenario and Communication System

[Display Publishing Process](#)

YY1_PRE_API_DEP_00
Last Changed By: Pascal Renet Status: **Published** Editing Status: Active
Last Changed On: 12/01/2023
Exported: No

Step 4 - Test the Integration flow

Now that all the changes have been done on the SAP side and our custom communication scenario is published, it's the moment of truth!

Communication Scenario CDS PRE

YY1_PRE_API_DEP_00

Last Changed By: Pascal Renet
Last Changed On: 12/01/2023
Exported: No

Status: **Published**

Editing Status: **Active**

Allowed Instances: One Instance per Scenario and Communication System

[Edit](#) [Check](#) [Create Arrangement](#) [Display Publishing Process](#)

[Inbound Services \(1\)](#) [Outbound Services \(0\)](#) [Authorizations \(0\)](#) [Communication Arrangements \(2\)](#)

Inbound Services (1)					
Inbound ID	Description	Status	Service Type	Last Changed On	Last Changed By
YY1_PRE_API_00_CDS	Deprecation API Sales CDS - 00		OData V2	12/01/2023	SAP System Processing

[Publish](#) [Revoke](#)

Let's once again test the APIs and check what the payloads look like. For this we go back to our REST client, and we test the GET operation on our API. Note that the URL has not changed.

First off, we'll do a before (left) and after (right) comparison of our json payload. Whilst we observe that the two successor fields we added have been pushed to the end of the payload, we can see that the content is identical from a field name and content point of view.

The positioning of the successor fields at the end of the API, is because when we added the successor fields to the CDS we did not re-position them - we left them at the end of the Elements list. In the next section we will show you how this can be avoided - if you so wish.

The image displays two screenshots of a REST client interface, comparing JSON payloads before and after a change. Both screenshots show a GET request to an SAP OData endpoint. The left screenshot shows the original payload, and the right screenshot shows the updated payload. A yellow arrow points from the 'Material' and 'MaterialGroup' fields in the left screenshot to their positions in the right screenshot, indicating they have been pushed to the end of the payload.

```
1 "d": {
2   "results": [
3     {
4       "__metadata": {
5         "id": "https://my.../sap/opu/odata/sap/YY1_PRE_API_00_CDS/YY1_PRE_API_00('1-0000000001.2-000010",
6         "uri": "https://my.../sap/opu/odata/sap/YY1_PRE_API_00_CDS/YY1_PRE_API_00('1-0000000001.2-000010",
7         "type": "YY1_PRE_API_00_CDS.YY1_PRE_API_00Type"
8       },
9       "ID": "1-0000000001.2-000010",
10      "SalesOrder": "1",
11      "SalesOrderItem": "000010",
12      "SalesOrderType": "OR",
13      "SalesOrderItemCategory": "TAN",
14      "CreationDateYearMonth": "201608",
15      "SalesOrganization": "1710",
16      "DistributionChannel": "10",
17      "Material": "TG11",
18      "MaterialGroup": "L001",
19      "Plant": "1710",
20      "SoldToParty": "17100001",
21      "SoldToPartyName": "Domestic US Customer 1",
22      "OrderQuantity": "3",
23      "OrderQuantityUnit": "PC"
24    },
25  ]
26 }
```

```
1 "d": {
2   "results": [
3     {
4       "__metadata": {
5         "id": "https://my.../sap/opu/odata/sap/YY1_PRE_API_00_CDS/YY1_PRE_API_00('1-0000000001.2-000010",
6         "uri": "https://my.../sap/opu/odata/sap/YY1_PRE_API_00_CDS/YY1_PRE_API_00('1-0000000001.2-000010",
7         "type": "YY1_PRE_API_00_CDS.YY1_PRE_API_00Type"
8       },
9       "ID": "1-0000000001.2-000010",
10      "SalesOrder": "1",
11      "SalesOrderItem": "000010",
12      "SalesOrderType": "OR",
13      "SalesOrderItemCategory": "TAN",
14      "CreationDateYearMonth": "201608",
15      "SalesOrganization": "1710",
16      "DistributionChannel": "10",
17      "Plant": "1710",
18      "SoldToParty": "17100001",
19      "SoldToPartyName": "Domestic US Customer 1",
20      "OrderQuantity": "3",
21      "OrderQuantityUnit": "PC",
22      "Material": "TG11",
23      "MaterialGroup": "L001"
24    },
25  ]
26 }
```

Then, we'll do a before (left) and after (right) comparison of our xml payload. Just like the json payload, whilst we observe that the two successor fields we added have been pushed to the end of the payload, we can see that the content is identical from a field name and content point of view.

```
GET https://my300448-api.s4hana.ondemand.com/sap/opu/odata/sap/YY1_PRE_API_00_CDS/YY1_PRE_API_00/?$top=5

Body
Pretty Raw Preview Visualize XML
1 <feed xmlns="http://www.w3.org/2005/Atom" xmlns:m="http://schemas.microsoft.com/ado/2007/08/dataservices/metadata"
2   xmlns:d="http://schemas.microsoft.com/ado/2007/08/dataservices" xml:base="https://my300448-api.s4hana.ondemand.com/sap/opu/odata/sap/YY1_PRE_API_
3   <id>https://my300448-api.s4hana.ondemand.com/sap/opu/odata/sap/YY1_PRE_API_00_CDS/YY1_PRE_API_00/</id>
4   <title type="text">YY1_PRE_API_00</title>
5   <updated>2023-12-01T05:45:59Z</updated>
6   <author>
7     <name/>
8   </author>
9   <link href="YY1_PRE_API_00/" rel="self" title="YY1_PRE_API_00"/>
10  <entry>
11    <id>https://my300448-api.s4hana.ondemand.com/sap/opu/odata/sap/YY1_PRE_API_00_CDS/YY1_PRE_API_00('1-000000
12    <title type="text">YY1_PRE_API_00('1-0000000001.2-000010')</title>
13    <updated>2023-12-01T05:45:59Z</updated>
14    <category term="YY1_PRE_API_00_CDS.YY1_PRE_API_00Type" scheme="http://schemas.microsoft.com/ado/2007/08/dat
15    <link href="YY1_PRE_API_00('1-0000000001.2-000010')</link> rel="self" title="YY1_PRE_API_00Type"/>
16    <content type="application/xml">
17      <m:properties xmlns:m="http://schemas.microsoft.com/ado/2007/08/dataservices/metadata" xmlns:d="http://
18        <dataservices>
19          <d:ID>.1-0000000001.2-000010</d:ID>
20          <d:SalesOrder>1</d:SalesOrder>
21          <d:SalesOrderItem>000010</d:SalesOrderItem>
22          <d:SalesOrderType>OR</d:SalesOrderType>
23          <d:SalesOrderItemCategory>TAN</d:SalesOrderItemCategory>
24          <d:CreationDateYearMonth>201608</d:CreationDateYearMonth>
25          <d:SalesOrganization>1710</d:SalesOrganization>
26          <d:DistributionChannel>10</d:DistributionChannel>
27          <d:Material>TG11</d:Material>
28          <d:MaterialGroup>L001</d:MaterialGroup>
29          <d:Plant>1710</d:Plant>
30          <d:SoldToParty>17100001</d:SoldToParty>
31          <d:SoldToPartyName>Domestic US Customer 1</d:SoldToPartyName>
32          <d:OrderQuantity>3</d:OrderQuantity>
33          <d:OrderQuantityUnit>PC</d:OrderQuantityUnit>
34        </m:properties>
35      </content>
36    </entry>
37  </feed>
```

```
GET https://my300448-api.s4hana.ondemand.com/sap/opu/odata/sap/YY1_PRE_API_00_CDS/YY1_PRE_API_00/?$top=5

Body
Pretty Raw Preview Visualize XML
1 <feed xmlns="http://www.w3.org/2005/Atom" xmlns:m="http://schemas.microsoft.com/ado/2007/08/dataservices/metadata"
2   xmlns:d="http://schemas.microsoft.com/ado/2007/08/dataservices" xml:base="https://my300448-api.s4hana.ondemand.com/sap/opu/odata/sap/YY1_PRE_API_
3   <id>https://my300448-api.s4hana.ondemand.com/sap/opu/odata/sap/YY1_PRE_API_00_CDS/YY1_PRE_API_00/</id>
4   <title type="text">YY1_PRE_API_00</title>
5   <updated>2023-12-01T09:41:40Z</updated>
6   <author>
7     <name/>
8   </author>
9   <link href="YY1_PRE_API_00/" rel="self" title="YY1_PRE_API_00"/>
10  <entry>
11    <id>https://my300448-api.s4hana.ondemand.com/sap/opu/odata/sap/YY1_PRE_API_00_CDS/YY1_PRE_API_00('1-000000
12    <title type="text">YY1_PRE_API_00('1-0000000001.2-000010')</title>
13    <updated>2023-12-01T09:41:40Z</updated>
14    <category term="YY1_PRE_API_00_CDS.YY1_PRE_API_00Type" scheme="http://schemas.microsoft.com/ado/2007/08/dat
15    <link href="YY1_PRE_API_00('1-0000000001.2-000010')</link> rel="self" title="YY1_PRE_API_00Type"/>
16    <content type="application/xml">
17      <m:properties xmlns:m="http://schemas.microsoft.com/ado/2007/08/dataservices/metadata" xmlns:d="http://
18        <dataservices>
19          <d:ID>.1-0000000001.2-000010</d:ID>
20          <d:SalesOrder>1</d:SalesOrder>
21          <d:SalesOrderItem>000010</d:SalesOrderItem>
22          <d:SalesOrderType>OR</d:SalesOrderType>
23          <d:SalesOrderItemCategory>TAN</d:SalesOrderItemCategory>
24          <d:CreationDateYearMonth>201608</d:CreationDateYearMonth>
25          <d:SalesOrganization>1710</d:SalesOrganization>
26          <d:DistributionChannel>10</d:DistributionChannel>
27          <d:Plant>1710</d:Plant>
28          <d:SoldToParty>17100001</d:SoldToParty>
29          <d:SoldToPartyName>Domestic US Customer 1</d:SoldToPartyName>
30          <d:OrderQuantity>3</d:OrderQuantity>
31          <d:OrderQuantityUnit>PC</d:OrderQuantityUnit>
32          <d:Material>TG11</d:Material>
33          <d:MaterialGroup>L001</d:MaterialGroup>
34        </m:properties>
35      </content>
36    </entry>
37  </feed>
```



Whilst the order of the fields may have changed, I would not expect this to be an issue from an integration point of view. I.e the calling application is not looking for the first, second or nth field in the payload, but rather an element included between a tag, e.g <tag>. The net result is that I had to not have to do any kind of changes in my middleware or in the calling application as the message payload has not changed - we did not adopt the successor field names. Of course you would still want to test the integration before rolling out the changes productively, but that should be a formality.

Amending the order of the field names

As we have [previously noted](#), when we added the successor fields to the custom CDS, we just let the successor fields be added at the end of the elements list. This explains why the successor fields changed position and ended up at the end of the json and xml payloads. However, know that if this should be a problem, you can move the elements in the custom CDS and position where you want them to be (in the place of the deprecated fields for example). To do this, we go back to the custom CDS app (you have to be in Edit mode).

We can see our successor fields at the bottom of the list, and the arrow, showing where we want them to be (below distribution channel).

Deprecation API Sales CDS - 00
Dependencies: 1 (Compatible Changes Allowed)

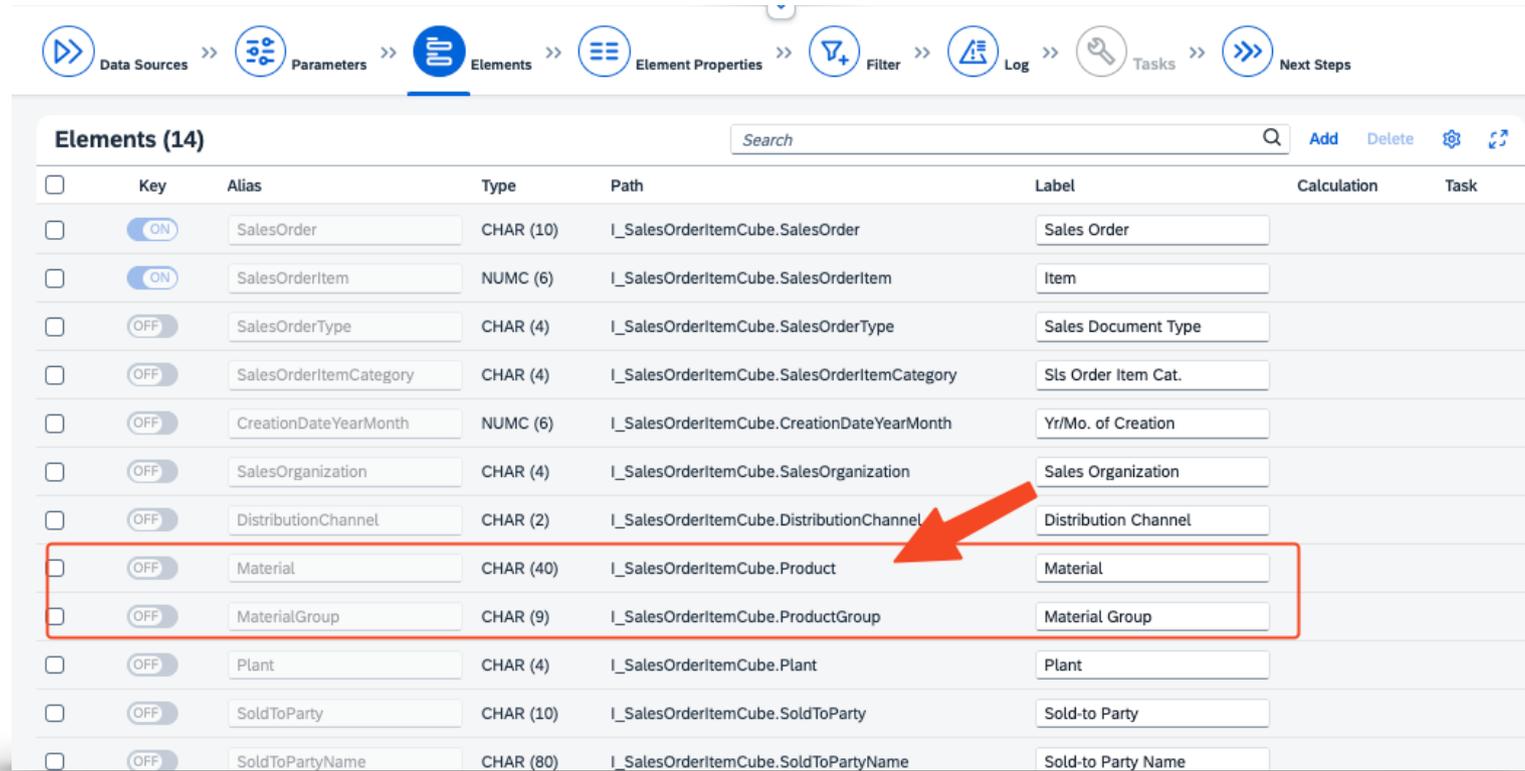
Data Sources >> Parameters >> **Elements** >> Element Properties >> Filter >> Log >> Tasks >> Next Step

Elements (14) [Add](#)

<input type="checkbox"/>	Key	Alias	Type	Path	Label	Calcul
<input type="checkbox"/>	<input type="checkbox"/> ON	SalesOrder	CHAR (10)	I_SalesOrderItemCube.SalesOrder	Sales Order	
<input type="checkbox"/>	<input type="checkbox"/> ON	SalesOrderItem	NUMC (6)	I_SalesOrderItemCube.SalesOrderItem	Item	
<input type="checkbox"/>	<input type="checkbox"/> OFF	SalesOrderType	CHAR (4)	I_SalesOrderItemCube.SalesOrderType	Sales Document Type	
<input type="checkbox"/>	<input type="checkbox"/> OFF	SalesOrderItemCategory	CHAR (4)	I_SalesOrderItemCube.SalesOrderItemCategory	Sls Order Item Cat.	
<input type="checkbox"/>	<input type="checkbox"/> OFF	CreationDateYearMonth	NUMC (6)	I_SalesOrderItemCube.CreationDateYearMonth	Yr/Mo. of Creation	
<input type="checkbox"/>	<input type="checkbox"/> OFF	SalesOrganization	CHAR (4)	I_SalesOrderItemCube.SalesOrganization	Sales Organization	
<input type="checkbox"/>	<input type="checkbox"/> OFF	DistributionChannel	CHAR (2)	I_SalesOrderItemCube.DistributionChannel	Distribution Channel	
<input type="checkbox"/>	<input type="checkbox"/> OFF	Plant	CHAR (4)	I_SalesOrderItemCube.Plant	Plant	
<input type="checkbox"/>	<input type="checkbox"/> OFF	SoldToParty	CHAR (10)	I_SalesOrderItemCube.SoldToParty	Sold-to Party	
<input type="checkbox"/>	<input type="checkbox"/> OFF	SoldToPartyName	CHAR (80)	I_SalesOrderItemCube.SoldToPartyName	Sold-to Party Name	
<input type="checkbox"/>	<input type="checkbox"/> OFF	OrderQuantity	QUAN (15,3)	I_SalesOrderItemCube.OrderQuantity	Order Quantity	
<input type="checkbox"/>	<input type="checkbox"/> OFF	OrderQuantityUnit	UNIT (3)	I_SalesOrderItemCube.OrderQuantityUnit	Sales Unit	
<input type="checkbox"/>	<input type="checkbox"/> OFF	Material	CHAR (40)	I_SalesOrderItemCube.Product	Material	
<input type="checkbox"/>	<input type="checkbox"/> OFF	MaterialGroup	CHAR (9)	I_SalesOrderItemCube.ProductGroup	Material Group	

As we hover over the field we want to move, a 'hand' icon will appear. We simply need to click to select it, then drag it to where we want it to be.

We repeat this for all the fields we want to move.



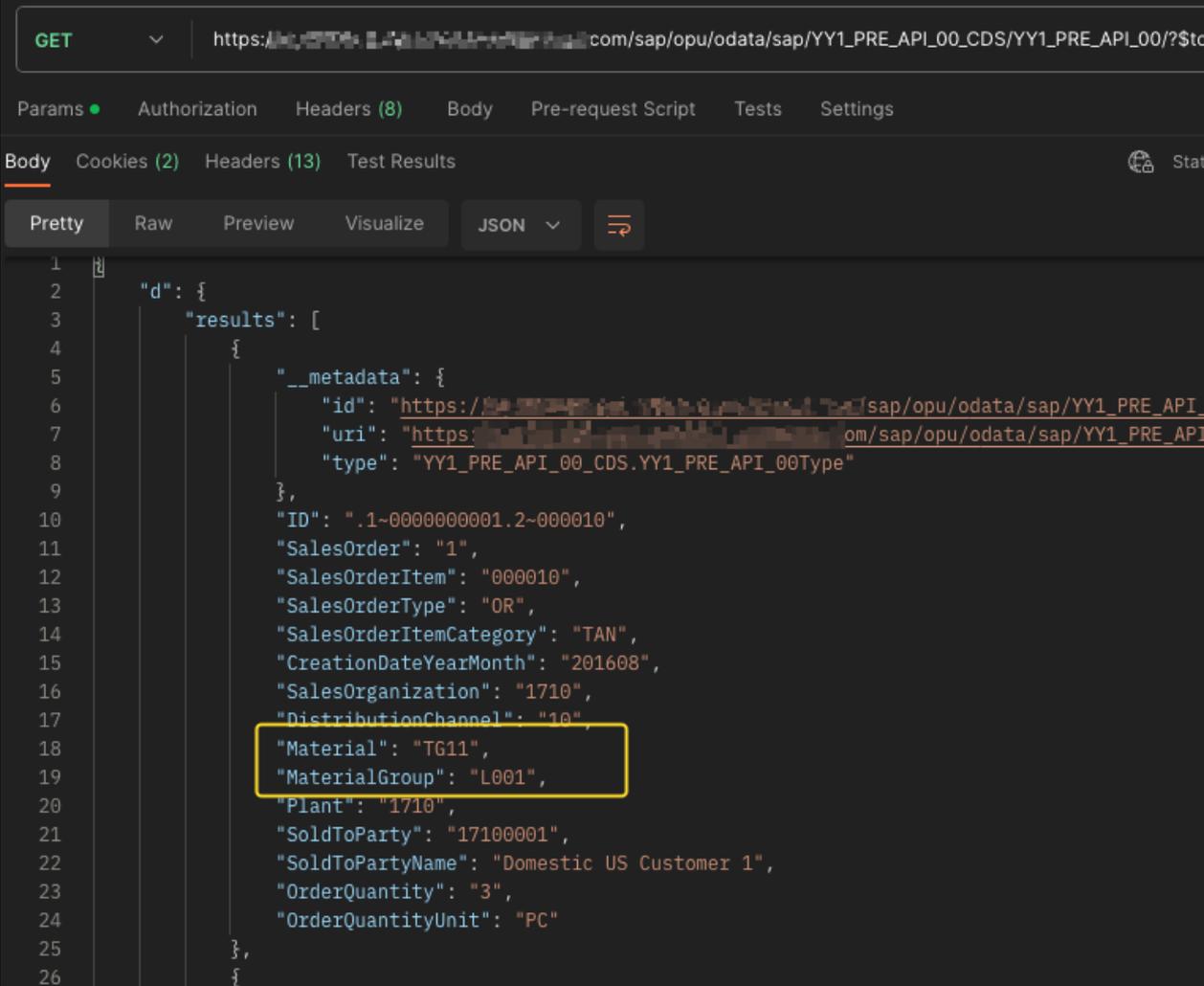
The screenshot shows the 'Elements' configuration page. The breadcrumb navigation at the top includes: Data Sources, Parameters, Elements, Element Properties, Filter, Log, Tasks, and Next Steps. The main content area is titled 'Elements (14)' and contains a table with the following data:

<input type="checkbox"/>	Key	Alias	Type	Path	Label	Calculation	Task
<input type="checkbox"/>	<input checked="" type="checkbox"/>	SalesOrder	CHAR (10)	I_SalesOrderItemCube.SalesOrder	Sales Order		
<input type="checkbox"/>	<input checked="" type="checkbox"/>	SalesOrderItem	NUMC (6)	I_SalesOrderItemCube.SalesOrderItem	Item		
<input type="checkbox"/>	<input type="checkbox"/>	SalesOrderType	CHAR (4)	I_SalesOrderItemCube.SalesOrderType	Sales Document Type		
<input type="checkbox"/>	<input type="checkbox"/>	SalesOrderItemCategory	CHAR (4)	I_SalesOrderItemCube.SalesOrderItemCategory	Sls Order Item Cat.		
<input type="checkbox"/>	<input type="checkbox"/>	CreationDateYearMonth	NUMC (6)	I_SalesOrderItemCube.CreationDateYearMonth	Yr/Mo. of Creation		
<input type="checkbox"/>	<input type="checkbox"/>	SalesOrganization	CHAR (4)	I_SalesOrderItemCube.SalesOrganization	Sales Organization		
<input type="checkbox"/>	<input type="checkbox"/>	DistributionChannel	CHAR (2)	I_SalesOrderItemCube.DistributionChannel	Distribution Channel		
<input type="checkbox"/>	<input type="checkbox"/>	Material	CHAR (40)	I_SalesOrderItemCube.Product	Material		
<input type="checkbox"/>	<input type="checkbox"/>	MaterialGroup	CHAR (9)	I_SalesOrderItemCube.ProductGroup	Material Group		
<input type="checkbox"/>	<input type="checkbox"/>	Plant	CHAR (4)	I_SalesOrderItemCube.Plant	Plant		
<input type="checkbox"/>	<input type="checkbox"/>	SoldToParty	CHAR (10)	I_SalesOrderItemCube.SoldToParty	Sold-to Party		
<input type="checkbox"/>	<input type="checkbox"/>	SoldToPartyName	CHAR (80)	I_SalesOrderItemCube.SoldToPartyName	Sold-to Party Name		

We can then see that our fields **Product** and **ProductGroup** have been re-positioned below the **Distribution Channel**. We of course check and publish the changes, and then head back to our REST client to check the effect of our changes.

We execute a GET operation on the same, usual API endpoint, and we see that the changing of the field order in the custom CDS has been replicated as desired in the resulting API payload.

The payload is at this point identical to what it was before we made any changes to any objects in SAP S/4HANA Cloud and we have not made any changes to our integration middleware or calling applications.



```
GET https://.../sap/opu/odata/sap/YY1_PRE_API_00_CDS/YY1_PRE_API_00/?$stc

Params • Authorization Headers (8) Body Pre-request Script Tests Settings
Body Cookies (2) Headers (13) Test Results
Pretty Raw Preview Visualize JSON ↕

1 2 "d": {
3   "results": [
4     {
5       "__metadata": {
6         "id": "https://.../sap/opu/odata/sap/YY1_PRE_API_
7         "uri": "https://.../sap/opu/odata/sap/YY1_PRE_API_
8         "type": "YY1_PRE_API_00_CDS.YY1_PRE_API_00Type"
9       },
10      "ID": ".1~0000000001.2~000010",
11      "SalesOrder": "1",
12      "SalesOrderItem": "000010",
13      "SalesOrderType": "OR",
14      "SalesOrderItemCategory": "TAN",
15      "CreationDateYearMonth": "201608",
16      "SalesOrganization": "1710",
17      "DistributionChannel": "10",
18      "Material": "TG11",
19      "MaterialGroup": "L001",
20      "Plant": "1710",
21      "SoldToParty": "17100001",
22      "SoldToPartyName": "Domestic US Customer 1",
23      "OrderQuantity": "3",
24      "OrderQuantityUnit": "PC"
25    },
26  ]
}
```

Wrap-up

If you have made it this far, Thank You! I hope you got something out of this document and that it has helped you to understand deprecations a little better, and maybe learnt one or two things along the way that you will be able to apply during the next upgrade.

Before you go though, I would be immensely grateful if you could leave a [comment here](#), letting me know if this document was helpful to you or if there is content that you would like to see added to it, they will help to motivate and improve future editions.

