Technical Articles

Andrea Schlotthauer

September 2, 2020 | 3 minute read

A new generation of CDS views: CDS view entities

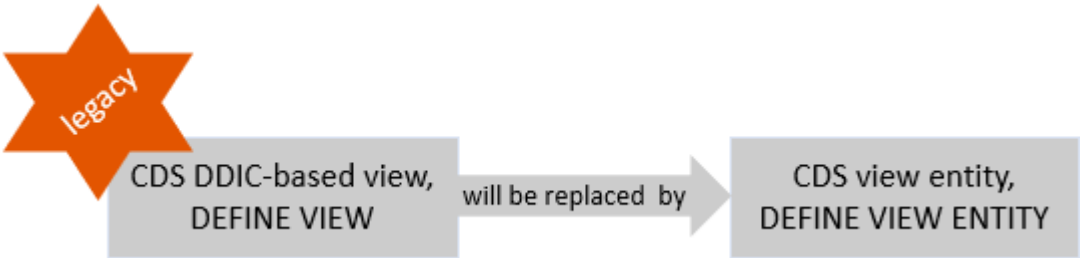
41 61 15,157

Follow

Like

RSS Feed

With ABAP release 7.55, a new type of CDS view is available: in official terminology, it's called **CDS view entity**. And it has come to replace the "classic" CDS DDIC-based views that have been around for years.



This blog post provides the following information:

1. Motivation: why has SAP developed a new type of CDS view?
2. Key differences between DDIC-based views and CDS view entities
3. Outlook: transition from DDIC-based views to CDS view entities

Motivation: why has SAP developed a new type of CDS view?

A CDS DDIC-based view is defined using the statement DEFINE VIEW. This kind of CDS view was first released with release 7.40, SP05 and it was for many years the only available kind of CDS view.

Since release 7.55, **CDS view entities** are available. They are defined using the statement DEFINE VIEW ENTITY. They have evolved from CDS DDIC-based views, serve the same purpose and have the same structure as their predecessor. But they offer several advantages over the "classic" CDS DDIC-based views, such as the following:

- No additional ABAP Dictionary view is created on activation.
- Improved performance during view activation.
- Optimization and simplification of syntax.
- Stricter syntax checks indicate problematic situations more explicitly, for example, annotation checks.

Key differences between DDIC-based views and CDS view entities

CDS view entities are a new and improved version of CDS DDIC-based views. Although very similar, CDS view entities are easier to use, and offer many small improvements and enhanced features.

Here are some principal ways in which CDS view entities differ from CDS DDIC-based views:

- There's no SQL view. The annotation *@AbapCatalog.sqlViewName* is not required and each view has only one name.
- Fewer annotations are required. For example, client handling takes place implicitly and doesn't require any development effort.
- Annotations are checked to ensure that only annotations defined as CDS objects in a CDS annotation definition can be used.
- Expressions can be nested within each other. Situations that previously required a view stack can now be implemented within a single view.
- Operand positions, such as the WHERE-clause, allow a greater variety of operands.
- Some features that haven't been widely used are no longer supported in view entities. Here are some examples (but the list is not exhaustive):
    - In DDIC-based views it is possible to assign alternative element names to elements of a SELECT-list using a name list. CDS view entities don't support name lists.
    - Domain fixed values in front of literals cannot be defined in CDS view entities.

- The syntax SELECT * to select all elements from the data source is supported in CDS DDIC-based views, but not in CDS view entities.

Additional features, such as typed literals to enhance type safety, and optimized buffer handling, are planned for future releases.

The following example compares a CDS DDIC-based view with a CDS view entity:

CDS DDIC-based view	CDS view er

`@AbapCatalog.sqlViewName: 'DEMO_CDS_JOIN'`
`@AbapCatalog.compiler.compareFilter: true`
`@ClientHandling.algorithm: #SESSION_VARIABLE`
`@AbapCatalog.preserveKey: true`
`@AccessControl.authorizationCheck: #NOT_ALLOWED`
`define view demo_cds_scarr_spfli`
`  as select from spfli`
`  join scarr on scarr.carrid = spfli.carrid`
`{`
`  key spfli.carrid  as id,`
`  key scarr.carrname  as carrier,`
`  key spfli.connid  as flight,`
`  spfli.cityfrom as departure,`
`  spfli.cityto  as destination`
`}`
	.
	.
	.
	.
	@Access
	define
	as sele
	j
	{
	k
	k
	k
	
	}

Differences:

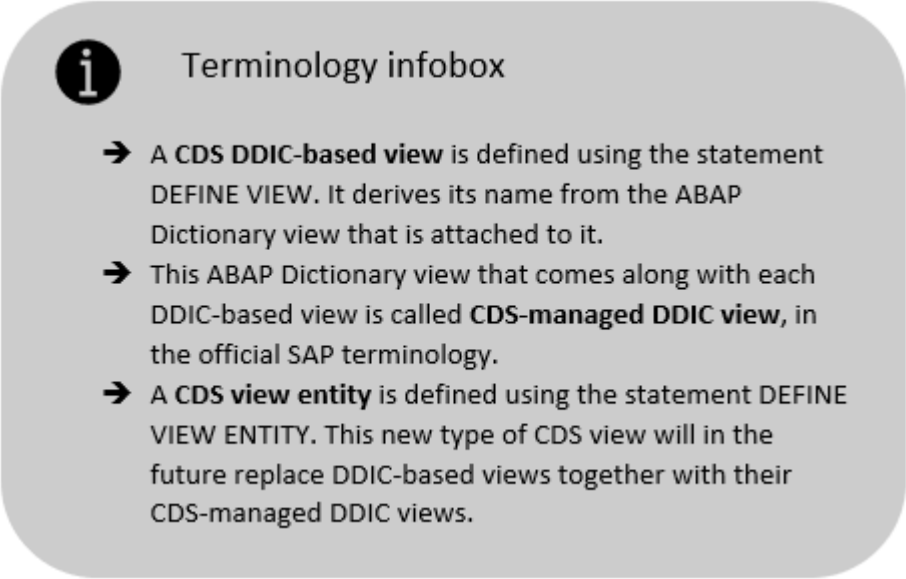
- The view entity doesn't require the annotation *@AbapCatalog.sqlViewName.*
- The view entity doesn't require the annotation *@AbapCatalog.compiler.compareFilter: true,* because the filter is implicitly and automatically compared.
- The view entity doesn't require the annotation *@ClientHandling.algorithm*, since client handling takes place implicitly.
- The view entity doesn't require the annotation *@AbapCatalog.preserveKey: true*, because there's no ABAP Dictionary view attached to a CDS view entity.
- The view entity is defined using the statement DEFINE VIEW ENTITY.

For a comprehensive description of CDS view entities, refer to ABAP Keyword Documentation (F1 Help in SAP GUI and ADT).

Outlook: transition from CDS DDIC-based views to CDS view entities

CDS view entities are the future of ABAP CDS. A migration tool that enables the migration from DDIC-based views to CDS view entities is under development. As soon as it is available, it is planned to migrate as many of the existing "classic" views to CDS view entities as possible (due to some incompatible changes, it is unlikely that *all* of the existing views will be migrated, though).

So, if you work with CDS views, familiarize yourself with CDS view entities. They'll be around soon.



Further information:

- ABAP Keyword Documentation (F1 Help, also available in SAP GUI and ADT)
- CDS view entities – the new CDS views – blog post about CDS view entities with further details

Alert Moderator

Assigned tags

ABAP Development

abap cds views

CDS projection views

Similar Blog Posts

CDS view entities - The new CDS views

By Harish Bokkasam Sep 04, 2020

Getting Started with ABAP Core Data Services (CDS)

By Carine Tchoutouo Djomo Feb 01, 2016

New type of CDS entity available: CDS projection views

By Andrea Schlotthauer Sep 16, 2020

Related Questions

Best practice for using CDS Views in structures

By David Henn Sep 18, 2020

CDS views with ABAP : 'The Parameter is not bound' error while activating ABAP program .

By Sharda P Oct 04, 2021

ABAP CDS View error : Field "HANAVIEW-ABC" is still being used as a view field in view "HANAVIEW"

By Sainath C Aug 10, 2018

Join the Conversation

SAP TechEd

Tune in for tech talk. Stay for inspiration. Upskill your future.

Coffee Corner

Join the new Coffee Corner Discussion Group.

41 Comments

You must be Logged on to comment or reply to a post.

Syambabu Allu
September 3, 2020 at 7:46 am

Hi Andrea,

much awaited info.Thanks for sharing info.

Thank you,

Syam

Like 1 | Share

Tarun Arora
September 4, 2020 at 1:03 pm

Thanks Andrea, One thing which i want to clarify "Nested Expressions", will it includes Subqueries also?

Like 0 | Share

Andrea Schlotthauer | Blog Post Author
September 4, 2020 at 1:27 pm

Hi Tarun, no, subqueries are not supported in view entities and they are currently not planned.

"Nested expressions" rather refers to, for example:

- within an arithmetic expression, a cast expression, case expression, aggregate expression or built-in function can be used (DDIC-based views don't offer the full range).

Same for aggregate expressions, case, cast, built-in functions, and so on - they all allow other expressions as operands.

Best,

Andrea

Like 0 | Share

Anup Kulkami
September 5, 2020 at 3:01 am

HI Andrea,

Can you throw some light on what would be impact on ODP transient provider? They do require a SQL view with prefix 2C.

Thanks

Like 0 | Share

Andrea Schlotthauer | Blog Post Author
September 8, 2020 at 4:43 pm

Hi Anup,

For CDS view entities, the analytical query name is the name of the view entity with the prefix 2C: 2C<name of CDS view entity in UpperCase>.

As the BW name should not be longer than 30 characters, the CDS view entity name should be less than 28 characters.

There is also the annotation @Analytics.technicalName (String(16)). If this annotation is used in a CDS view entity, then the analytical query will be „2C<technicalName>“.

This annotation should be used especially when converting a CDS DDIC-based view to a CDS view entity and when you want the BW name to stay the same and not change during conversion.

Best,

Andrea

Like 0 | Share

Anup Kulkami
September 9, 2020 at 12:42 am

Thanks Andrea!

Does this also imply that (since no SQL view gets generated), CDS view entity cannot be accessed natively in HANA database? There was a note 2511210 in the past mentioning that it "Should Not Be" . But now it appears that it "cannot be".

Anup

Like 0 | Share

Andrea Schlotthauer | Blog Post Author
September 14, 2020 at 5:28 am

Hi Anup,

CDS view entities do also generate a HANA view with the name of the CDS view entity. Hence, it is possible to access a view entity natively on HANA database. However, the same restrictions apply as described in note 2511210.

Best,

Andrea

Like 0 | Share

Viren Devi
July 5, 2021 at 11:42 am

Hello Andrea,

During migration will SAP keep name of the SAP provided extraction enabled CDS's views same? Or names will change ?

Thank you.

Like 0 | Share

Andrea Schlotthauer | Blog Post Author
July 12, 2021 at 12:02 pm

Hi Viren Devi,

the name stays the same during the migration. As a precondition for the migration, the name of the DDLS source and of the CDS entity must be identical. The newly generated view entity has the name of this CDS entity on the database.

The name of the DDIC SQL view is dropped during the migration. Currently, it is not recommended to reuse the name of the DDIC SQL view of a migrated view for another CDS object.

Documentation regarding the migration will be released soon.

Best

Andrea

Like 0 | Share

Vijay Sharma
September 5, 2020 at 10:21 am

Nice.. thanks for sharing !!

Like 0 | Share

David Alvarez
September 10, 2020 at 2:17 pm

Hallo andrea,

> *due to some incompatible changes, it is unlikely that all of the existing views will be migrated*

In our SAP CPI-DS project, we are using CDSs to create complex logic. In this way, we release CPI-DS system of making complex calculations that could take more than 15 minutes in CPI-DS (vs seconds in CDS).

However, CPI-DS can only consume the view (database layer).

Should we get worried about it? I am afraid "Unlikely" could mean a lot of things. Removing these views or making mandatory not using them could ruin other parallel projects that cannot call to a CDS directly .

Do you have information about it or who could clarify this point?

Thanks in advance,
David

Like 0 | Share

Andrea Schlotthauer | Blog Post Author
September 14, 2020 at 7:16 am

Hi David,

how do you consume the CDS views? Do you use ABAP SQL? If yes, then you can use the CDS view entity directly, and not the SQL view.

Does that help?

Best,

Andrea

Like 0 | Share

David Alvarez
September 17, 2020 at 8:37 pm

Hi Andrea,

I am afraid SAP CPI-DS is not the best ETL tool in the market (but it's SAP recommendation when working with IBP).

It is only able to "see" tables or views at database level when connecting to a backend system. That means CDSs does not appear as a source object. However, we link the CDS view in CPI-DS and this is how we are working: generating complex SELECTs in CDS and linking them to CPI-DS through its view.

Our concern is:

- If these new CDS view entity is going to replace completely classic CDS or they are going to coexist
- No option to continue creation of classic CDSs once new CDS view entity is available

Please, we are not asking for a work around. We need just confirmation of the roadmap to know if we need to change our way of work.

Thanks in advance,
David

Like 0 | Share

Andrea Schlotthauer | Blog Post Author
September 18, 2020 at 7:03 am

Hi David,

The classic, DDIC-based CDS views will still be supported. It is not planned to deprecate them. Both types of views will coexist.

You as owner of the view can decide whether you want to migrate your existing views to view entities or not. There won't be any automatic conversion.

You can continue creating classic CDS view, even when CDS view entities are available. SAP only recommends to switch to view entities, because once they are feature complete, they are easier to work with and offer more options.

Best,

Andrea

Like 1 | Share

Avinash D M
September 29, 2020 at 5:52 am

Hi ,

Is it possible to extend the CDS Entity like we have for old CDS View?

Thanks and regards,

Avinash

Like 0 | Share

Andrea Schlotthauer | Blog Post Author
September 29, 2020 at 7:24 am

Hi Avinash,

Yes, it is possible to extend CDS view entities in the same way as CDS DDIC-based views.

Some details:

- A boundary condition has been removed: an extend for a CDS view entity can contain only new associations. It is not required (as for the old CDS views) to include at least one field in an extension.
- View extensions are possible with EXTEND VIEW and there's also the new statement EXTEND VIEW ENTITY. View extends created with EXTEND VIEW ENTITY don't create a DDIC SQL view append and they have all the advantages of the new view entities, such as faster activation.

Best,

Andrea

Like 0 | Share

Santhosh Kadiyala
October 15, 2020 at 6:54 pm

Hi Andrea Schlotthauer

i have few queries regarding the new CDS Entity-

our system is on SAP Latest 2020

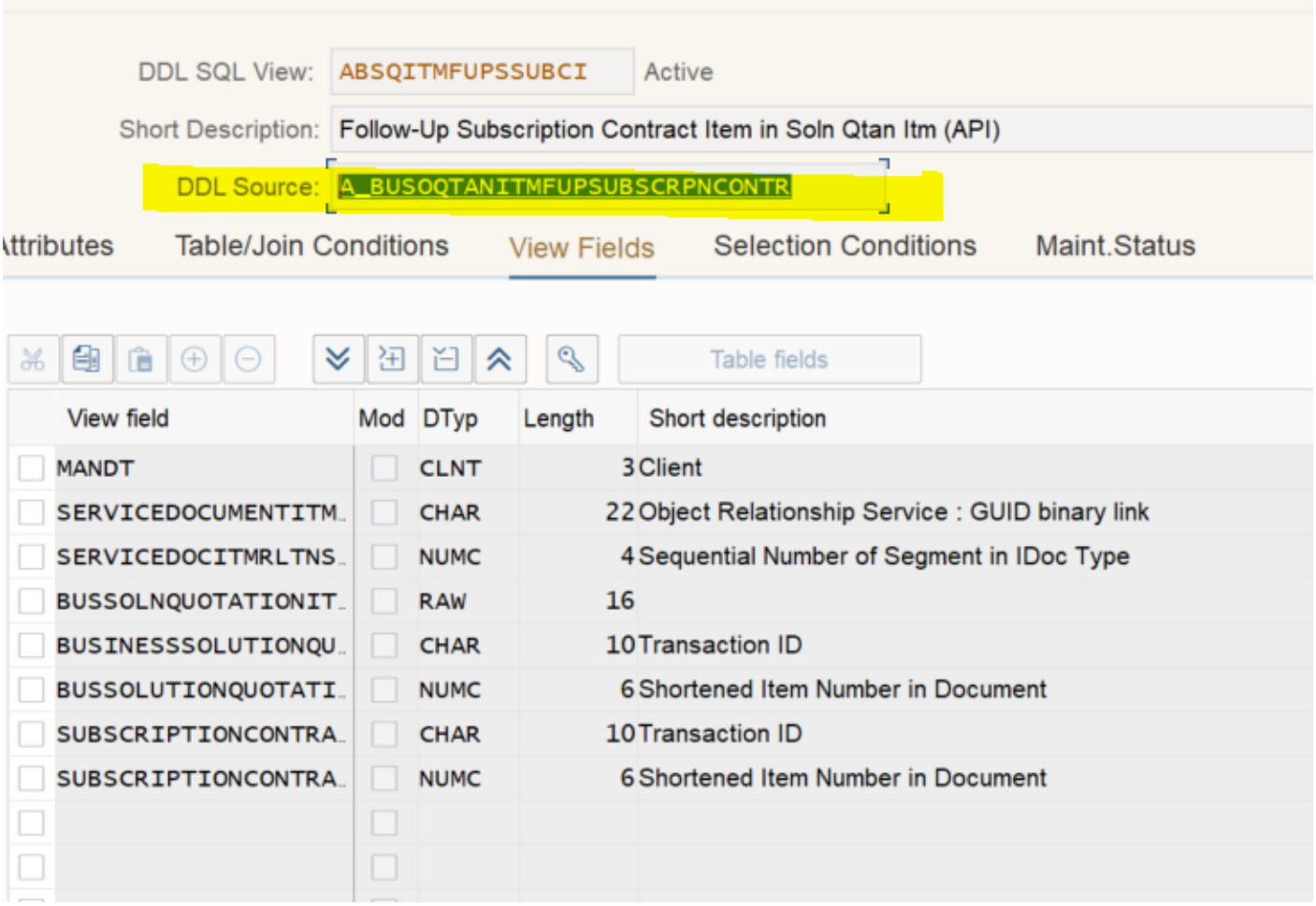
is there a way i display the DDL Source in SAP SE11 directly other than by DDL SQL View?

i see VIEW for this object in DDLDEPENDENCY

```
@ClientHandling.algorithm:#SESSION_VARIABLE
@EndUserText.label: 'Follow-up Subscription Contract Item'
@VDM: {
  viewType: #COMPOSITE,
  lifecycle.contract.type: #PUBLIC_REMOTE_API
}
@AbapCatalog: {
  sqlViewName: 'ABSQITMFUPSSUBCI',
  compiler.compareFilter: true,
  preserveKey: true
}
@AccessControl: {
  authorizationCheck: #CHECK,
  personalData.blocking: #('TRANSACTIONAL_DATA')
}
@ObjectModel: {
   createEnabled: false,
   updateEnabled: false,
   deleteEnabled: false,
   usageType: {
     dataClass:      #TRANSACTIONAL,
     serviceQuality: #C,
     sizeCategory:   #L
   }
}
@Metadata.ignorePropagatedAnnotations: true

define view A_BuSoQtanItmFUPSubscrpnContr
  as select from I_ServiceDocumentItemSuccssr_2
  association [1..1] to I_ServiceDocumentItemEnhcd  as _SourceItemDc
  association [1..1] to A_BusinessSolutionQuotation as _BusinessSolu
```

if you see there is an abap view for this



View field	Mod	DTyp	Length	Short description
MANDT		CLNT	3	Client
SERVICEDOCUMENTITM		CHAR	22	Object Relationship Service : GUID binary link
SERVICEDOCITMRLTNS		NUMC	4	Sequential Number of Segment in IDoc Type
BUSSOLNQUOTATIONIT		RAW	16	
BUSINESSSOLUTIONQU		CHAR	10	Transaction ID
BUSSOLUTIONQUOTATI		NUMC	6	Shortened Item Number in Document
SUBSCRIPTIONCONTRA		CHAR	10	Transaction ID
SUBSCRIPTIONCONTRA		NUMC	6	Shortened Item Number in Document

is there a way i can access this directly without going to view

Data Definition: A_BUSOQTANITMFUPSUBSCRPNCONTR Active

rties Source Code

ADT-Link: adt://SBS/sap/bc/adt/ddic/ddl/sources/a_busoqtanitmfupsubscrpncontr

```
@ClientHandling.algorithm:#SESSION_VARIABLE
@EndUserText.label: 'Follow-Up Subscription Contract Item in Soln Qtan Itm (API)'
@VDM: {
  viewType: #COMPOSITE,
  lifecycle.contract.type: #PUBLIC_REMOTE_API
}
@AbapCatalog: {
  sqlViewName: 'ABSQITMFUPSSUBCI',
  compiler.compareFilter: true,
  preserveKey: true
}
@AccessControl: {
  authorizationCheck: #CHECK,
  personalData.blocking: #('TRANSACTIONAL_DATA')
}
@ObjectModel: {
   usageType: {
     dataClass:      #TRANSACTIONAL,
     serviceQuality: #C,
     sizeCategory:   #L
   }
}
@Metadata.ignorePropagatedAnnotations: true

define view A_BuSoQtanItmFUPSubscrpnContr
  as select from I_ServiceDocumentItemSuccessor
  association [1..1] to I_ServiceDocumentItemEnhcd  as _SourceItemDoc
  association [1..1] to A_BusinessSolutionQuotation as _BusinessSolutionQuotation
{
  key ServiceDocumentItmRelationUUID,
  key ServiceDocItmRltnSqncNumber,

      ServiceDocumentItemUUID                      as BusSolnQuotationItemUUID,
      _SourceItemDoc.ServiceDocument               as BusinessSolutionQuotation,
      _SourceItemDoc.ServiceDocumentItem           as BusSolutionQuotationItem,

      _ServiceDocItm.ServiceDocument               as SubscriptionContract,
      _ServiceDocItm.ServiceDocumentItem           as SubscriptionContractItem,

      _BusinessSolutionQuotation
}
where
      _SourceItemDoc.ServiceObjectType = 'BUS2000116'
  and _ServiceDocItm.ServiceObjectType = 'BUS2000266'
  and CharcLength                      = 32
```

i am able to access SQL View directly in abap as shown below

```
*&---------------------------------------------------------------------*
*& Report ZTEST_T1
*&---------------------------------------------------------------------*
*&
*&---------------------------------------------------------------------*
REPORT ztest_t1.

DATA lt_table TYPE TABLE OF absqitmfupssubci.

SELECT * FROM absqitmfupssubci INTO TABLE @lt_table.

IF NOT lt_table[] IS INITIAL.
ENDIF.
```

Variables 1 | Variables 2 | Locals | Globals | Auto | Memory Analysis

St...	Variable	Val...	Val.	Ch...	Hexa
	LT_TABLE		[762x8(138)]Standard Table		

its also giving me data if i give cds entity as shown below

```
DATA lt_table TYPE TABLE OF a_busoqtanitmfupsubscrpncontr.

SELECT * FROM a_busoqtanitmfupsubscrpncontr INTO TABLE @lt_table.

IF NOT lt_table[] IS INITIAL.
ENDIF.
```

St...	Variable	Val...	Val.	Ch...	Hexadecimal
	LT_TABLE		[762x7(132)]Standard Table		

why is this throwing an error when i declare lt_Table as cds view and trying to access data from the same cds entity ? cds view and cds entity are related by 1:1 right, why is this discrepancy. please clarify
thanks

San

Like 0 | Share

Andrea Schlotthauer | Blog Post Author
October 26, 2020 at 9:58 am

Hi San,

For CDS views and CDS view entities, it is recommended to work in ABAP Development tools (ADT). Otherwise, please use transaction sddlar to display your CDS views and CDS view entities. For CDS views, you can use the view name directly, no need to use the SQL view name.

SE11 has limitations: views (classic views as well as CDS view entities) can only be displayed when their name has no more than 16 characters.
Moreover, it is not recommended to access a DDL source in SE11 using the SQL view (warning message).

Regarding your second question: The SQL view can contain an additional client field and therefore, an error message is returned. You can avoid it by using same name (either SQL view name OR DDL source name) for typing and after FROM.

You can also use an inline declaration, such as the following:

```
select *
  from andrea_v2_support
  into table @data(lt_table).
```

Hope this helps. If not, please contact me again.

Best,

Andrea

Like 0 | Share

Diptarup Majumder
November 3, 2020 at 8:41 am

Hi Andrea,

Nice blog!

Need some guidance in the light of CDS root view entity.

We are in S4 HANA 1909 SP02 on-prem version hosted on HEC. SAP_ABA is on 75E. SAP_BASIS is on 754. Still, it is not giving an option to create CDS view entity. After upgrading the Eclipse version 2020-09, ADT, I somehow managed to get the CDS view entity option. However, when I am using the following code snippet to activate it, I am getting the error as 'Keyword Entity not Allowed'.

```
@AccessControl.authorizationCheck: #NOT_REQUIRED
@EndUserText.label: 'View '
define root view  entity Y_RPA_002
as select from  yraptesttable
{
key sonumber,
creationdate,
customer
}
```

```
1 @AccessControl.authorizationCheck: #CHECK
2 @EndUserText.label: 'Test View'
3 define root view entity y_rpa_004 as select from yraptesttable
4   {
5   key sonumber,
6   creationdate,
7   customer
8 }
```

Syntax error: Keyword ENTITY not allowed

I am following the below blog:

https://blogs.sap.com/2019/10/25/getting-started-with-the-abap-restful-programming-model/

My questions are :

1. In the version provided above, thought we are in ABA 75E, would we be able to get onboarded with the RAP model provided above ? If not, can we use CDS view to serve the purpose ?
2. Any clue why are we getting the error 'Keyword Entity not allowed' ?

Thanks for your help!

Like 0 | Share

Andrea Schlotthauer | Blog Post Author
November 4, 2020 at 9:30 am

Hi Diptarup,
CDS view entities are available since S4HANA 2005, on premise release 7.55. To work with CDS view entities, it is necessary to update your system. But RAP works also with classic CDS views.
You can find further details in the RAP development guide under https://help.sap.com/viewer/923180ddb98240829d935862025004d6/Cloud/en-US/289477a81eec4d4e84c0302fb6835035.html.
Hope this answers your question.
Best,
Andrea

Like 0 | Share

Kyo Choi
February 19, 2021 at 10:31 am

I am getting same message for ABAP version 7.5E. Is this lower than 7.55?

Like 0 | Share

Andrea Schlotthauer | Blog Post Author

February 19, 2021 at 12:59 pm

Hi Kyo Choi,

yes indeed. SAP_ABA 7.5E is ABAP Platform Release 1909 (see ABAP Platform – Part 1 – Evolution from SAP Netweaver | SAP Blogs) and 7.55 is ABAP Platform Release 2020 (see ABAP Keyword Documentation (sap.com)).

Best,

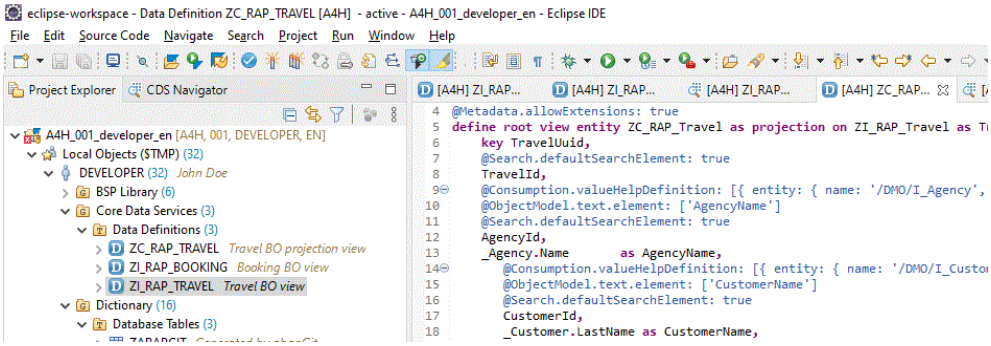
Andrea

Like 1 | Share

Kyo Choi

February 19, 2021 at 1:28 pm

On BO Project View, "entity" worked fine. Not sure why CDS view has issue. BTW. I am using the developer edition 1909.



Like 0 | Share

Andrea Schlotthauer | Blog Post Author

February 19, 2021 at 2:19 pm

CDS projection views have been available since ABAP release 7.76, which is the same as ABAP Platform clout 1905. In on-premise, they have been available since S/4HANA 1909.

See New type of CDS entity available: CDS projection views | SAP Blogs.

ABAP Cloud Release	ABAP platform cloud	ABAP On-Premise Release	ABAP platform	Kernel Release
7.79	2002	-	-	7.79
7.78	1911	-	-	7.78
7.77	1908	7.54	1909	7.77
7.76	1905	-	-	7.76

Like 1 | Share

Amit Kumar
May 27, 2021 at 6:12 pm

Hi Andrea,

I tried to go through the given blog or help page you mentioned above but not sure how to co-relate ABAP release 7.55 with ABAP platform 1809,1909, 2020 etc or SAP_ABA 7.5D, E, F etc? if you could help us understanding this connection and where do we see **ABAP release 7.55** etc in the system?

Thanks,

Amit Kumar

Like 0 | Share

Harish Bokkasam
November 4, 2020 at 10:32 am

Hallo Diptarup,

As Andrea mentioned, you can build RAP BO with classic CDS views..define CDS view syntax

Greetings,

Harish B

Like 0 | Share

Sebastien Phan
November 4, 2020 at 5:29 pm

Hi Andrea!

I just created my first CDS entity view. We had for regular cds view the tool "dependency analyzer" in abap in eclipse. Now, If I try to use this tool, it says "dependency analyzer only support cds view"

I use abap in eclipse Version: 3.15.0.

Is there a similar tool for this?

Alternate question: for HANA studio, we could query the CDS view by using the DDIC name in the select. For entity view, there are no ddic. Would it be possible as well?

Best regards,

Sebastien

Like 0 | Share

Andrea Schlotthauer | Blog Post Author
November 5, 2020 at 12:56 pm

Hi Sebastien,

dependency analyzer is currently not available for CDS view entities. Development is working on it, hopefully it will be available soon. There's no alternative tool, unfortunately.

CDS view entities can be accessed in HANA Studio by their DDLS name (same as the entity name). Like for classic DDIC-based views, it might be required to specify the schema.

Best,

Andrea

Like 1 | Share

Miroslaw Grygolec
February 8, 2021 at 4:12 pm

hi Andrea,

It's a while since we've got view entities at our disposal and to most case I found them fully functional. However recently I accidentally faced restriction to migrate CDS which includes $session in the conditional

expression:

> *case when (UserRole.ValidFromDate <= $session.system_date and UserRole.ValidToDate >= $session.system_date) then 'X' else '' end as IsValid,*

The original DDIC-base CDS was fully functional, same syntax with view entity results in the error message: "Expression type SESSION_VAR_EXPR not allowed in expression context COMPARISON, clause type COMP_RIGHT".

I was lucky to overcome this case with $parameters, though result is not as elegant as originally and generally this was unexpected obstacle for migration from legacy.

So my question would be - is there any reference of the restrictions (not supported syntax, pending features) associated to new view entities?

Thank you in advance.

Mirosław

Like 0 | Share

Andrea Schlotthauer | Blog Post Author
February 9, 2021 at 8:42 am

Hi Mirosław,

your conditional expression should work in CDS view entities. I've tested it and tried to reproduce the error, but in my example, there is no error message.

The topic about CASE WHEN in the ABAP Keyword Documentation lists all possible operands and it says the the right side of *CASE WHEN* can be a session variable.

So this looks like a bug, please open an incident to component **BC-DWB-DIC** if you would like this to be further investigated.

The ABAP Keyword Documentation (F1 Help) describes all supported syntax and all restrictions compared to CDS DDIC-based views. That's the only externally available source.

Best,

Andrea

Like 1 | Share

Miroslaw Grygolec

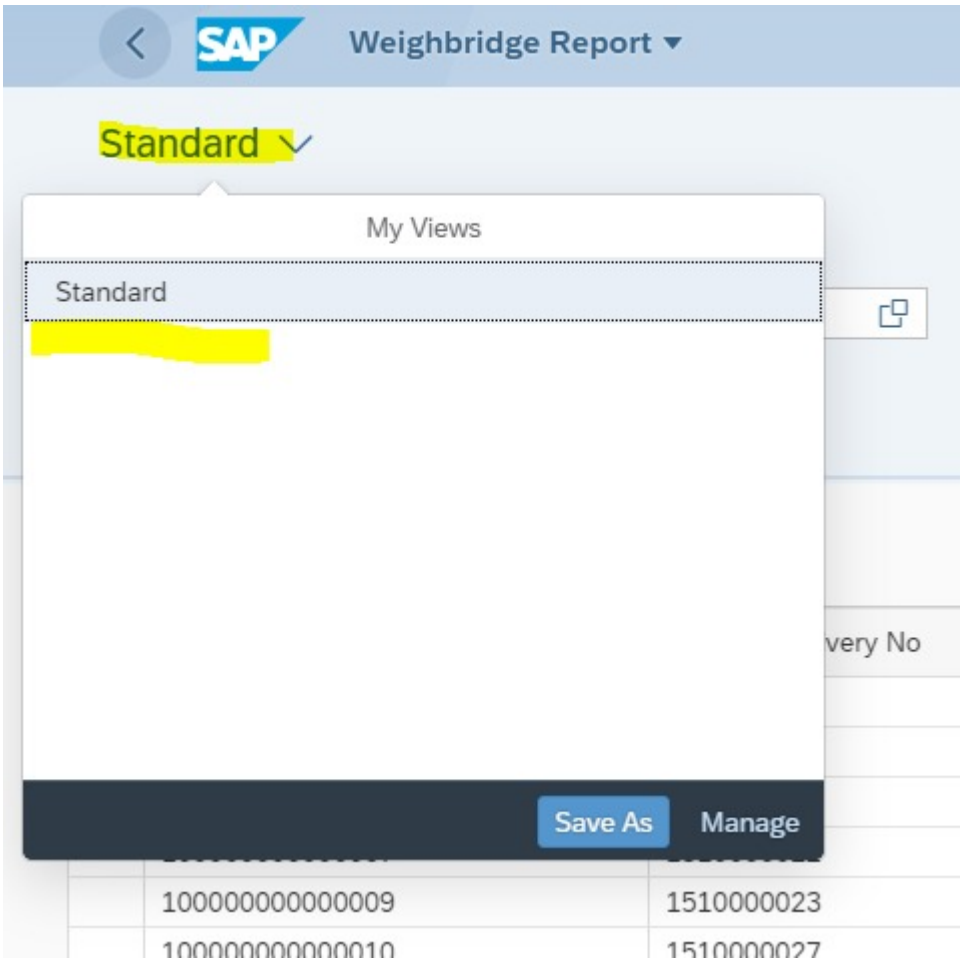
February 9, 2021 at 9:49 am

Thank you for quick answer! 🙂 Will follow as you advise.

Like 0 | Share

Shilpa R
May 19, 2021 at 4:29 am

Hi Andrea Schlotthauer, this blog is really helpful, and even I have used this cds view entities in web ide and I am using s/4 hana databse for the same, but can we implement variant using this cds view entity?



Like 0 | Share

Andrea Schlotthauer | Blog Post Author
May 21, 2021 at 5:55 am

Hi Shilpa R,

as far as I know, implementing variants is not possible using CDS annotations.

For further questions, please open a BCP incident on component BC-ESI-ESF-GW.

Best,

Andrea

Like 1 | Share

Shilpa R
May 27, 2021 at 4:51 pm

Ok thank you so much for your reply Andrea Schlotthauer

Like 0 | Share

Amit Kumar
May 27, 2021 at 6:12 pm

Hi Andrea,

I tried to go through the given blog or help page you mentioned above but not sure how to co-relate ABAP release 7.55 with ABAP platform 1809,1909, 2020 etc or SAP_ABA 7.5D, E, F etc? if you could help us understanding this connection and where do we see **ABAP release 7.55** etc in the system?

Thanks,

Amit Kumar

Like 0 | Share

Andrea Schlotthauer | Blog Post Author
May 28, 2021 at 8:06 am

Hi Amit,

The on-premise docu on SAP Help Portal contains a table that maps ABAP releases to ABAP Platform releases, SAP S/4HANA on premise releases and kernel releases: ABAP Keyword Documentation (sap.com)

Release	ABAP Platform	OP Release	Kernel Release
7.55	2020	2020	7.81
7.54	1909	1909	7.77
7.53	1809	1809	7.73
7.52	-	1709	7.53
7.51	-	1610	7.49
7.50	-	1511	7.45

Regarding your second question:

- do you mean where you can see the ABAP release information in your system? You find this information under *System > Status >* section *SAP System Data > Product Version >* tab I*nstalled Product Version.*
- are you looking for a version history of the ABAP Keyword Documentation? Here's the URL to a specific version: https://help.sap.com/doc/abapdocu_755_index_htm/7.55/en-US/index.htm. Just replace "755" with the version you would like to see.
- to see a different version of the documentation in your system, the only way I know is to log on to a corresponding system.

Best

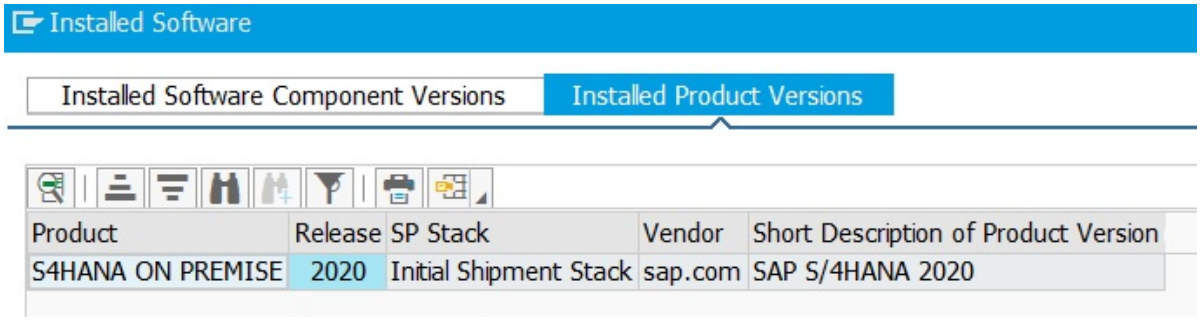
Andrea

Like 1 | Share

Amit Kumar
May 28, 2021 at 10:25 am

Thanks for the details.

Regarding the second point, yes…I meant to ask ABAP release information in the system- I see only S4HANA release (Not the ABAP release 7.55). Here is the screenshot of what I see if I follow the steps *System > Status >* section *SAP System Data > Product Version >* tab I*nstalled Product Version.*



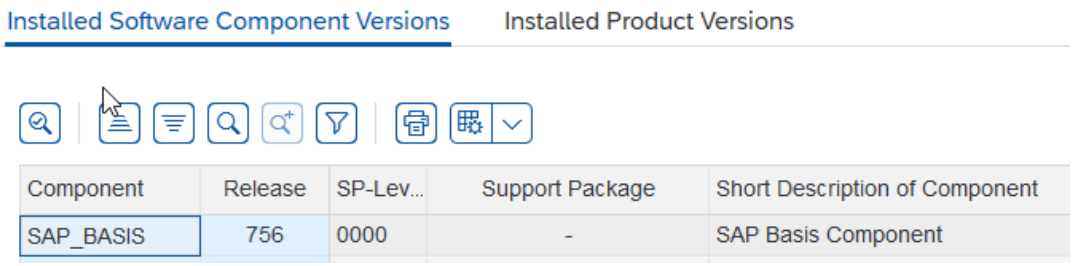
Tab Installed Product Version

I was hoping to see ABAP release 7.55 somewhere directly in the system. But mapping table you provided (via help link) is also fine. Thanks!

Like 0 | Share

Andrea Schlotthauer | Blog Post Author
May 28, 2021 at 11:00 am

Maybe check tab *Installed Software Component Versions.* I find the required information there:

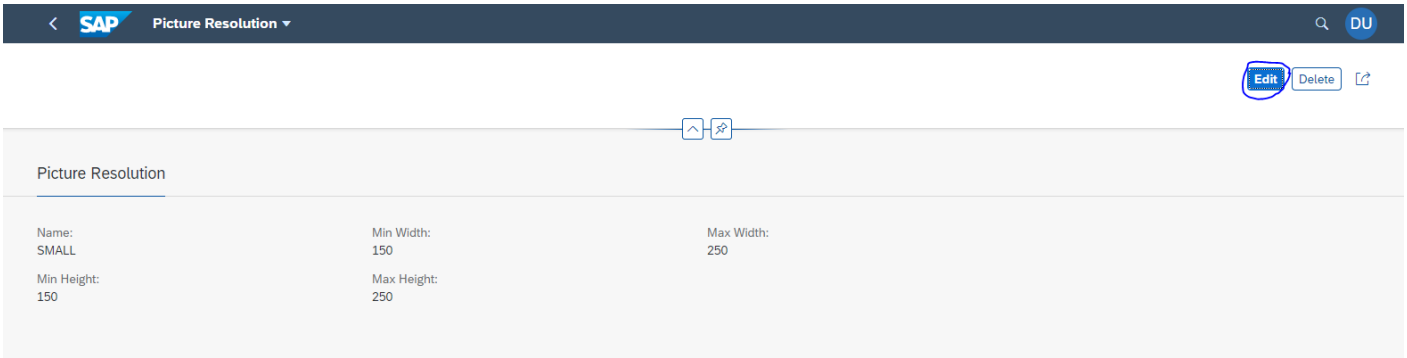


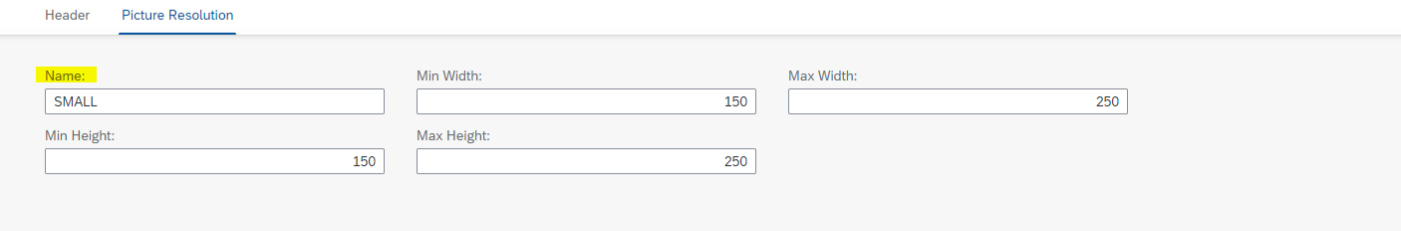
Like 0 | Share

Shilpa R
June 1, 2021 at 5:52 pm

Hi Andrea Schlotthauer,

I wanted to make name field as non editable inside the object page in my capm project





Andrea Schlotthauer | Blog Post Author
June 2, 2021 at 12:45 pm

How carHi Shilpa R,

Like 0 | Share

thanks for your message.

Unfortunately, I can't answer this question. My area of expertise is ABAP Core Data Services, basis development. This question seems to be related to a specific application and to UI design. UI annotations are not part of basis development.

Please check the corresponding product documentation.

Best

Andrea

Like 1 | Share

Shilpa R
June 2, 2021 at 1:33 pm

Hi Andrea Schlotthauer Thank you so much for your reply

Like 0 | Share

Find us on

Privacy	Terms of Use
Legal Disclosure	Copyright
Trademark	Cookie Preferences
Newsletter	Support