



Tushar Sharma

October 9, 2017 | 5 minute read

ABAP Core Data Services – Part 3(Virtual Data model types)

9 20 64,915

Follow



Like



RSS Feed

Table of Content:

S.No	Topics
1.	Introduction
2.	VDM View types
3.	Analytics Data Category
4.	More Blog's (Follow)
5.	Credits

This blog is continuation of my previous blog's :

[ABAP Core Data Services – Introduction \(ABAP CDS view\)](#) in this detailed introduction is given about the ABAP Core Data services.

[ABAP Core Data Services – Part 1 \(ABAP CDS Entities\)](#) in this detailed introduction is given about the ABAP CDS Entities.

[ABAP Core Data Services – Part 2 \(Types of CDS views\)](#) in this detailed introduction is given about different types of ABAP CDS views.

Before, start reading this I would request you to go through above mentioned blog for better and smooth understanding.

I, hope you have got the basic understanding about ABAP CDS views. In this blog, we'll see the different Virtual data model types in ABAP CDS views .

[Let's Start !!](#)

Introduction

A [Virtual Data Model](#) (VDM) is a structured representation of HANA database views which was introduced to be used in SAP HANA Live for SAP Business Suite. It provides direct access to SAP business data using standard SQL or OData requests. The VDM exposes the Business Data of an SAP system as an **Understandable**, **Comprehensive** and **Executable** model for consumers in Transactional applications, Analytics, External interfaces fulfilling important product qualities.

VDM is intended to be interpreted by view browsers and other functionality which is based on the virtual data model. So, these views can be consumed directly by rich client UIs (such as HTML5, SAP BusinessObjects, and Excel) without any additional software layer (for example, ABAP application system) in between (two-tier architecture).

Now, to take advantage of powerful functionality of virtual data model in ABAP CDS views, SAP introduced [@VDM annotations](#) to define an ABAP CDS view which can be consumed and interpreted by Analytics engine used in different tools for Analysis. It offers additional features

Eg: Better Performance, Extensibility, Security and Privacy, Consistency as a reusable, stable and compatible platform for SAP internal Development, Partners, Customers thus increasing development efficiency and fostering ease of consumption.

VDM View types

The classification of VDM is used only for SAP internal structuring and interpretation of the CDS views. Releasing CDS views for customers and partners is controlled by additional internal classification of the views.

Different VDM view types are:

1. Basic
2. Composite
3. Consumption

Basic View

A basic view is created on top of DDIC tables/views, which means only a basic view is a view which interacts with database for fetching data. The fetching of data directly from database results as core data for other type of view types without any data redundancies.

Data source for basic view is raw data directly from DDIC tables/views.

Syntax: @VDM.Viewtype: #BASIC

Composite View

A composite view is created on top of basic views, which means composite views doesn't interact with database directly for data, but instead thorough dimension result set of basic views. The multiple basic views can be used with join / association for fetching data and doing calculations (eg:. using built-in-expressions and functions) on retrieved data to consume in different analytics tools using consumption view.

Data source for composite view is basic views.

Syntax: @VDM.Viewtype: #COMPOSITE

Consumption View

A consumption view is created on top of composite view, which means consumption view doesn't interact with DDIC tables/views or basic views. The consumption view is written on top of composite view which provides factual and cube data. [This view is used to expose the data to different analytics tools](#) (such as HTML, SAP BusinessObjects, Analysis for Excel, SAP Lumira, etc..) which helps to create reports and different visualization for presenting Business data.

Data source for consumption view is composite views.

Syntax: @VDM.Viewtype: #CONSUMPTION

We have already seen, different virtual data model view types that are used for Analytics purpose. But only using VDM view types is not just enough for consuming CDS view in analytics tools because till now we have just defined data model for CDS view, nothing related to type of data CDS view is returning. So, to let the Analytical

manager know how to interpret each CDS entity we have to declare **Data Category** for each particular ABAP CDS views with VDM view type annotations.

Analytics Data Category

To enable the analytic manager for multidimensional data consumption, performing data aggregation, and slicing and dicing data. BI frontends like Design Studio and Analysis Office can consume the data via the analytic manager.

The different types of Analytics data category are:

1. Dimension.
2. Fact.
3. Cube.
4. Aggregation level.

Dimension

The view that represents master data is annotated as `@ObjectModel.dataCategory: #DIMENSION`. If we have to use any of the view for replication or for a query, we should use `@Analytics.dataCategory: #DIMENSION`.

Fact

The view represents transactional data (center of star schema). Usually, it contains measures. Typically, these views are necessary for replication, therefore, views listed as Fact data category should not be joined with master data views.

Syntax: `@Analytics.dataCategory: #FACT`

Cube

The Cube view (like #FACT) represents factual data, but cube view does not have to be without redundancy, which means joins with master data are possible. The consumption views are built on top of CUBE views, where data is replicated from facts.


















Syntax: `@Analytics.dataCategory: #CUBE`

Aggregation Level

The view using data category as aggregation level indicates that CDS view is of type projection. For, this type of view analytics manager offers write-back functionality. In this type of view associations are not allowed and elements can be renamed.

Data source of this type of views is a CDS view with dataCategory = #CUBE, which supports Analytics.writeBack.className annotation.

Syntax: @Analytics.dataCategory: #AGGREGATIONLEVEL

Data Category	Represents Master data	Represent Transaction data	Can be Joined with master data	Offers Write back	Contains Measure
Dimension					
Fact					
Cube					
Aggregation Level					

To keep the focus on core objective (Core Data Services) of this blog, I tried to make it as small as possible.

Suggestions and questions are welcomed !!

Follow:-

- [ABAP Core Data Services – Introduction \(ABAP CDS Views\).](#)
- [ABAP Core Data Services – Part 1 \(ABAP CDS Entities\).](#)
- [ABAP Core Data Services – Part 2 \(Types of CDS views\)](#)

Thank you.

Credits:-
[Virtual Data Model](#)

[VDM Annotations](#)

[Analytics Annotations](#)

Alert Moderator

Assigned tags

SAP S/4HANA
ABAP Development
ABAP CDS VDM types
ABAP CDS view
ABAP Core Data Service Views
beginner
Virtual data models

Similar Blog Posts

[SAP S/4HANA Embedded Analytics](#)

By Esha Rajpal Jun 29, 2017

[Core Data Services \[CDS\] in SAP S/4 HANA](#)

By Shivankit Bisht Sep 26, 2016

[ABAP Core Data Services | SAP S/4HANA - Best Practice Guide](#)

By Bernd Kohler Jan 22, 2019

Related Questions

[SAP HANA Certification Core Data Services](#)

By Bhakti joshi Nov 22, 2017

[Best Practices For ABAP CDS VDM development and front-end development](#)

By Pierfrancesco La Spada Oct 04, 2017

ABAP Migration to S/4 HANA and the new Data Model

By Felix Fichtner Oct 18, 2016

9 Comments

You must be [Logged on](#) to comment or reply to a post.

Stephen Marshall

October 9, 2017 at 2:23 pm

Amazing post

Like 2 | Share

Syam Babu

October 12, 2017 at 4:05 am

Good blog..with all necessary information of ABAP CDS views with data models and analytical query's

Thanks,

Syam

Like 2 | Share

Joachim Rees

January 15, 2018 at 3:32 pm

Hey Tushar,

thanks for sharing!

I tried to map what you describe in theory with an actual example I came across today.

I start with C_MaterialStockActual, which is the consumption view, and I turn the sequence you gave around, putting the Consumption view on the top of the list and the Basic-View on the bottom.

-> This seems a better fit to me, as the Basic-view reads from the database, and that is underneath everything else.

So this is how I went exploring: Open C_MaterialStockActual in AdT, and then just see where it selects from - follow that path using the F3-key.

This is the 'stack' that unfolds this way:

C_MaterialStockActual (reads from:)

I_MaterialStock (reads from:)

I_MaterialStock_Aggr (reads from:)

I_MaterialDocumentRecord (reads from:)

MATDOC (-> the database table).

Here I show parts of the annotations, especially the @VDM.viewType:

C_MaterialStockActual:

```
@VDM.viewType: #CONSUMPTION
@Analytics.query : true

define view C_MaterialStockActual
```


I_MaterialStock:

```
@VDM.viewType: #COMPOSITE
@VDM.private: false
@Analytics.dataCategory:#CUBE

define view I_MaterialStock
  as select from I_MaterialStock_Aggr
```

I_MaterialStock_Aggr:

```
@VDM.viewType: #COMPOSITE
@VDM.private: false
@Analytics.dataCategory:#FACT

define view I_MaterialStock_Aggr
```

I_MaterialDocumentRecord:

```
@VDM.private: false
@VDM.viewType: #BASIC

define view I_MaterialDocumentRecord
```

So I think I learned the following in addition to your blog:

1. It's not always exactly 3 levels (Basic, Aggregate, Consumption), it also can be more .
2. "A composite view is created on top of basic views," **or** on top of **another composite view** .

Great to dig deeper in the big field of CDS-Views and to learn navigation my way around them !

best
Joachim

Like 1 | Share

Tushar Sharma | Blog Post Author
January 31, 2018 at 8:31 am

Hi [Joachim Rees](#)

I'm glad you liked the blog !

Thanks for sharing the findings and for a great example.

Cheers,

Tushar Sharma

Like 1 | Share

Former Member
January 19, 2018 at 5:58 pm

Hello,

I'm trying to find an implementation for writeback functionality, but there is no documentation for that. Has someone an example for the implementation of the interface IF_RODPS_ODP_WRITEBACK for ABAP CDS

thank you

Ahmed.

Like 0 | Share

Joachim Rees
February 1, 2018 at 12:57 pm

Helpfull blog on the VDM for S/4HANA:

<https://blogs.sap.com/2018/01/30/implementation-patterns-for-cds-views-in-sap-s4hana/>

Like 1 | Share

Kuldeep Jain

August 1, 2018 at 8:38 am

Thanks Tushar, Good Blog and very-much helpful .

Like 1 | Share

Vipul Gulati

August 21, 2018 at 9:00 am

Hello,

Please Share a solution for CDS with Select-Options.Is it possible?

Thanks

Like 0 | Share

Tushar Sharma | Blog Post Author

September 10, 2018 at 5:32 am

Hi Vipul,

As of now we don't have an option to pass parameters as select options.

Thanks

Like 0 | Share

Find us on

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