Uladzimir Sapazhkou

March 29, 2018 | 6 minute read

C-create, R-read, U-update, D-delete OData Services Creation Using ABAP CDS Views

Introduction

This article describes the fastest way of OData Services creation using ABAP Core Data Services. Details on @OData.publish annotation utilization, reference to Data Source CDS-Entity and import from DDIC Structure are given. Create, Update and Delete actions are shown on the custom transparent table data just to simplify the article and do not move focus from OData Service creation to business process requirements implementation.

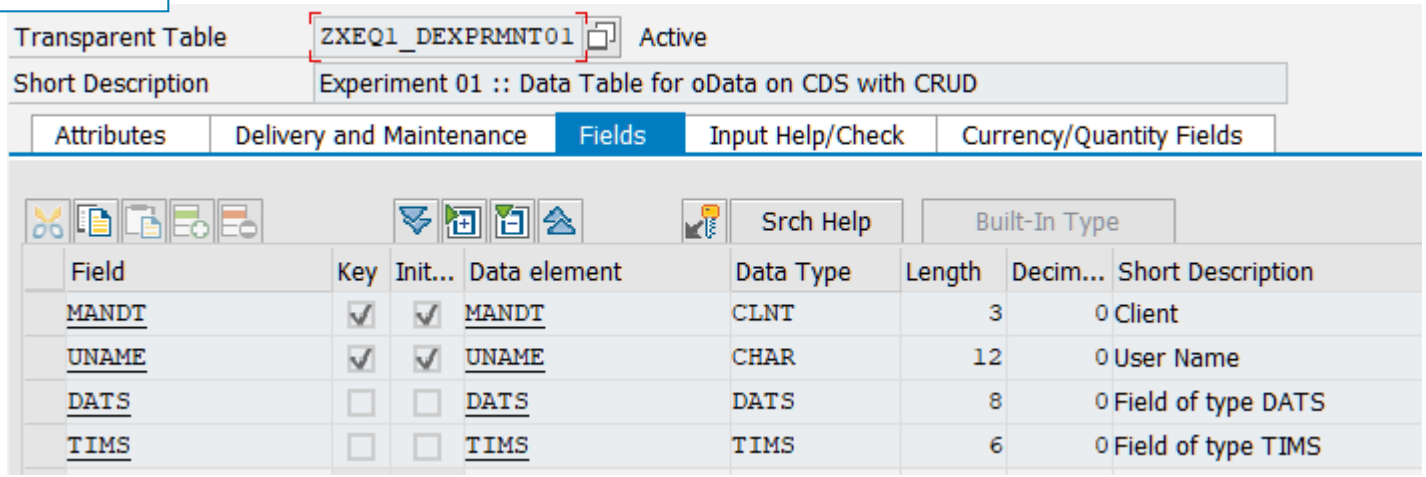
“B” Type OData Service Creation. With using of @OData.publish annotation in ABAP CDS

Follow

Like

RSS Feed

Let us assume that we have any data table (probably in the custom namespace, just to simplify the demo case and get OData service only, and not on SAP functional modules and class usage)



Transparent Table ZXEQ1_DEXPRMNT01 Active

Short Description Experiment 01 :: Data Table for oData on CDS with CRUD

Attributes | Delivery and Maintenance | Fields | Input Help/Check | Currency/Quantity Fields

Srch Help | Built-In Type

Field	Key	Init...	Data element	Data Type	Length	Decim...	Short Description
MANDT	✓	✓	MANDT	CLNT	3	0	Client
UNAME	✓	✓	UNAME	CHAR	12	0	User Name
DATS			DATS	DATS	8	0	Field of type DATS
TIMS			TIMS	TIMS	6	0	Field of type TIMS

It is possible to create ABAP CDS view for it using SAP HANA Studio

*[SH1] ZXEQ1_LEXPRMNT01

```
        atalog.sqlViewName: 'ZXEQ1_QEXPRMNT01'
        atalog.compiler.compareFilter: true
  @AccessControl.authorizationCheck: #CHECK
      erText.label: 'Experiment 01'
      .publish: true

  define view ZXEQ1_LEXPRMNT01
      elect
      om zxeq1_dexprmnt01  as Data
  {
    key Data.uname,
        Data.dats,
        Data.tims
  }
```

During CDS activation OData service will be generated in the background

Service document looks like this:

```
<?xml version="1.0" encoding="UTF-8"?>
- <app:service xml:lang="en" xmlns:sap="http://www.sap.com/Protocols/SAPData"
xmlns:m="http://schemas.microsoft.com/ado/2007/08/dataservices/metadata" xmlns:atom="http://www.w3.org/2005/Atom"
xmlns:app="http://www.w3.org/2007/app"
xml:base="http://hosting5.skybuffer.com:8013/sap/opu/odata/sap/ZXEQ1S_EXPRMNT01_SRV/">
  - <app:workspace>
      <atom:title type="text">Data</atom:title>
    - <app:collection href="ZXEQ1_LEXPRMNT01" sap:content-version="1" sap:deletable="false" sap:updatable="false"
      sap:creatable="false">
        <atom:title type="text">ZXEQ1_LEXPRMNT01</atom:title>
        <sap:member-title>Experiment 01</sap:member-title>
      </app:collection>
  </app:workspace>
  <atom:link href="http://hosting5.skybuffer.com:8013/sap/opu/odata/sap/ZXEQ1S_EXPRMNT01_SRV/" rel="self"/>
  <atom:link href="http://hosting5.skybuffer.com:8013/sap/opu/odata/sap/ZXEQ1S_EXPRMNT01_SRV/" rel="latest-version"/>
</app:service>
```

“R” Type OData Service Creation. Using reference on ABAP CDS

Follow

Like

RSS Feed

Let us assume that we have any data table (probably in custom namespace, just to simplify demo case and get focused on OData service only, and not on SAP functional modules and class usage)

Transparent Table ZXEQ1_DEXPRMNT01 Active

Short Description Experiment 01 :: Data Table for oData on CDS with CRUD

Attributes | Delivery and Maintenance | Fields | Input Help/Check | Currency/Quantity Fields

Srch Help | Built-In Type

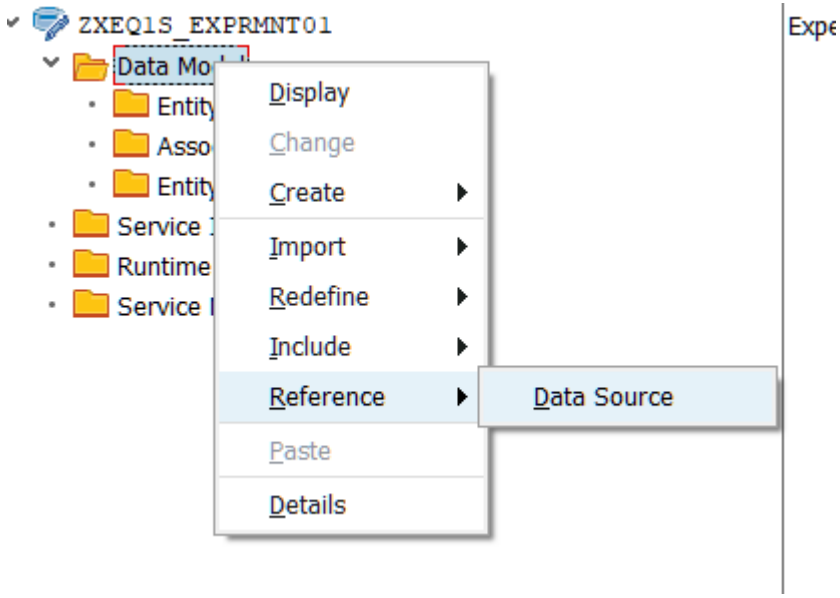
Field	Key	Init...	Data element	Data Type	Length	Decim...	Short Description
MANDT	✓	✓	MANDT	CLNT	3	0	Client
UNAME	✓	✓	UNAME	CHAR	12	0	User Name
DATS			DATS	DATS	8	0	Field of type DATS
TIMS			TIMS	TIMS	6	0	Field of type TIMS

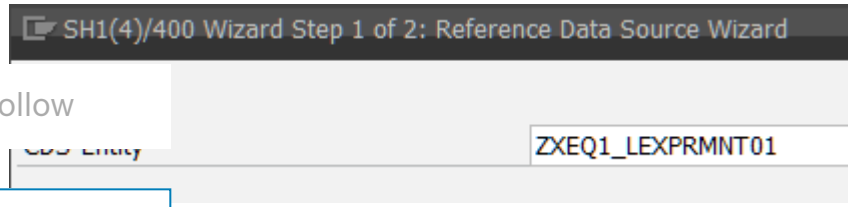
It is possible to create ABAP CDS view for it using SAP HANA Studio. Please consider that in this case there is no @OData annotation in use

[SH1] ZXEQ1_LEXPRMNT01

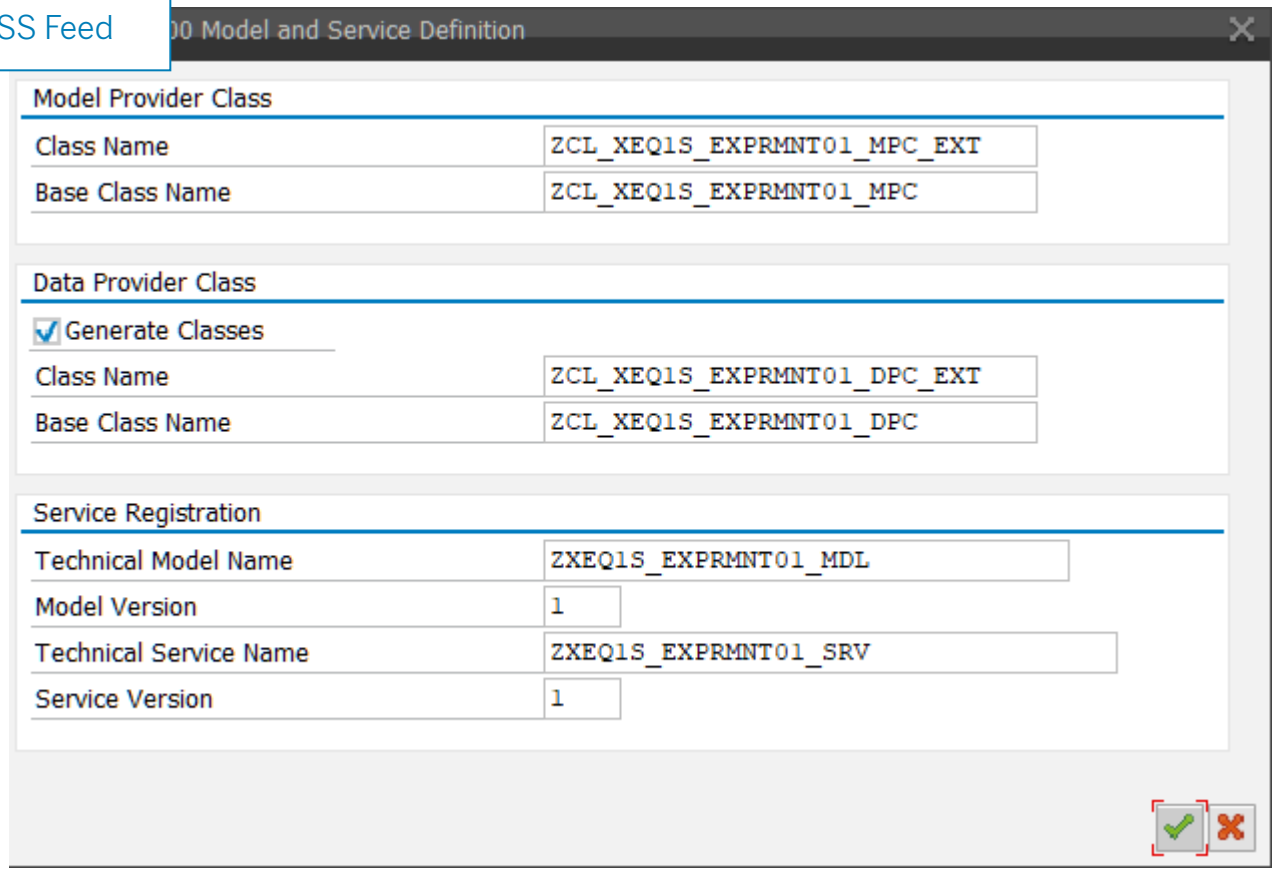
```
atalog.sqlViewName: 'ZXEQ1_QEXPRMNT01'
atalog.compiler.compareFilter: true
@AccessControl.authorizationCheck: #CHECK
erText.label: 'Experiment 01'
 view ZXEQ1_LEXPRMNT01
elect
from zxeq1_dexprmnt01  as Data

Data.uname,
Data.dats,
Data.tims
}
```

During our next step we will Create Gateway Project via transaction code **SEGW** and refer it to the created in the previous step ABAP CDS





erate runtime objects based on reference to ABAP CDS

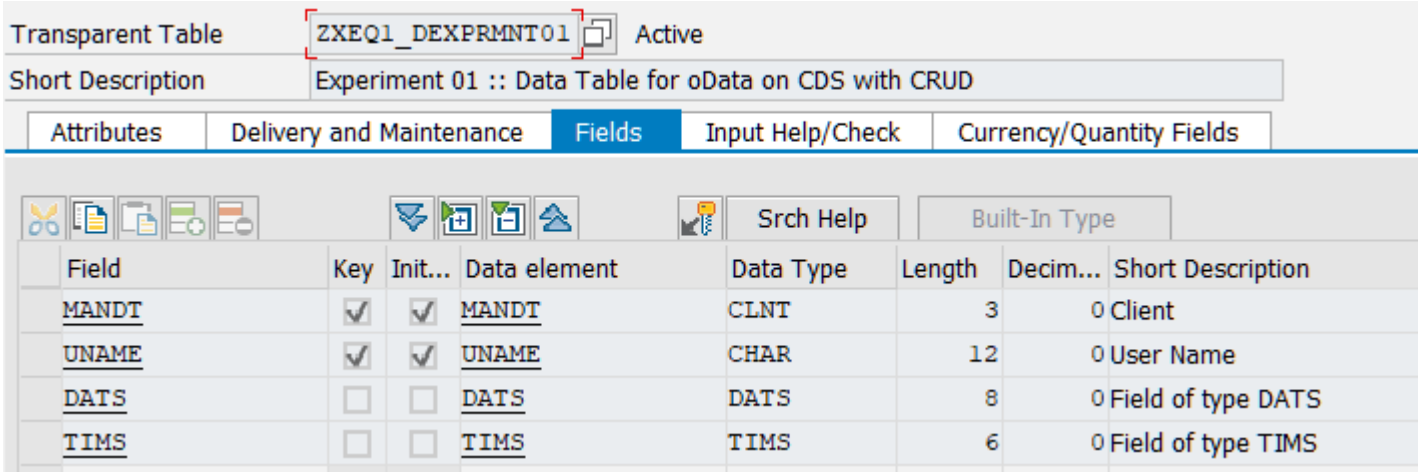


The service document looks like this:

```
<?xml version="1.0" encoding="UTF-8"?>
- <app:service xml:lang="en" xmlns:sap="http://www.sap.com/Protocols/SAPData"
ttp://schemas.microsoft.com/ado/2007/08/dataservices/metadata" xmlns:atom="http://www.w3.org/2005/Atom"
http://www.w3.org/2007/app"
ttp://hosting5.skybuffer.com:8013/sap/opu/odata/sap/ZXEQ1S_EXPRMNT01_SRV/">
rkspace>
<atom:title type="text">Data</atom:title>
:collection href="ZXEQ1_LEXPRMNT01" sap:content-version="1" sap:deletable="false" sap:updatable="false"
creatable="false">
<atom:title type="text">ZXEQ1_LEXPRMNT01</atom:title>
<sap:member-title>Experiment 01</sap:member-title>
</app:collection>
</app:workspace>
nk href="http://hosting5.skybuffer.com:8013/sap/opu/odata/sap/ZXEQ1S_EXPRMNT01_SRV/" rel="self"/>
nk href="http://hosting5.skybuffer.com:8013/sap/opu/odata/sap/ZXEQ1S_EXPRMNT01_SRV/" rel="latest-version"/>
e>
```


“CRUD” Type OData Service Creation

Let us assume that we have any data table (probably in custom namespace, just to simplify the demo case and get focused on OData service only, and not on SAP functional modules and class usage)



Transparent Table ZXEQ1_DEXPRMNT01 Active

Short Description: Experiment 01 :: Data Table for oData on CDS with CRUD

Attributes | Delivery and Maintenance | Fields | Input Help/Check | Currency/Quantity Fields

Srch Help | Built-In Type

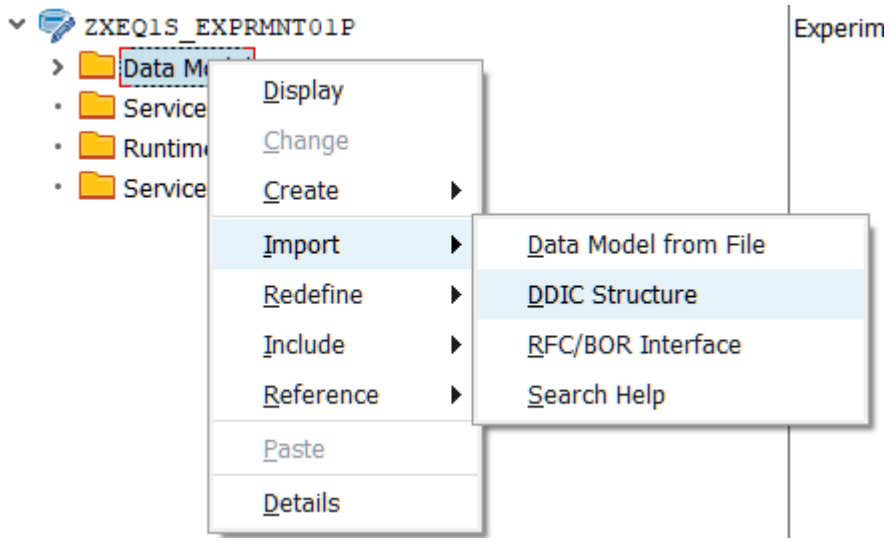
Field	Key	Init...	Data element	Data Type	Length	Decim...	Short Description
MANDT	✓	✓	MANDT	CLNT	3	0	Client
UNAME	✓	✓	UNAME	CHAR	12	0	User Name
DATS			DATS	DATS	8	0	Field of type DATS
TIMS			TIMS	TIMS	6	0	Field of type TIMS

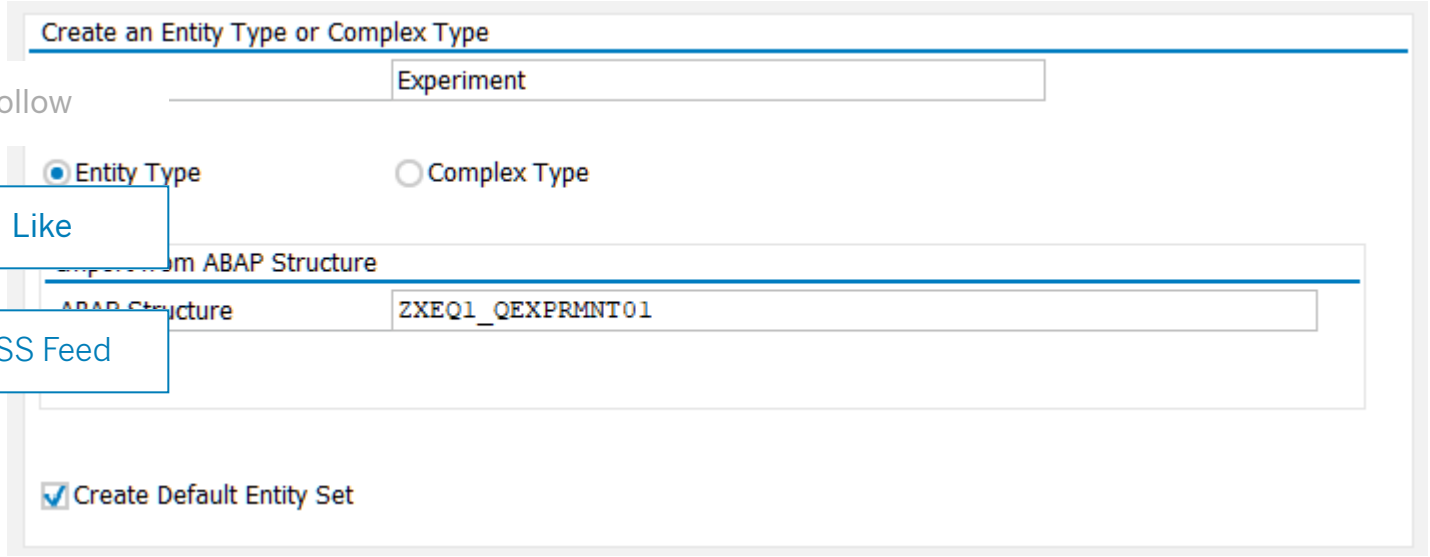
It is possible to create ABAP CDS view for it using SAP HANA Studio (no @OData annotation is in use)

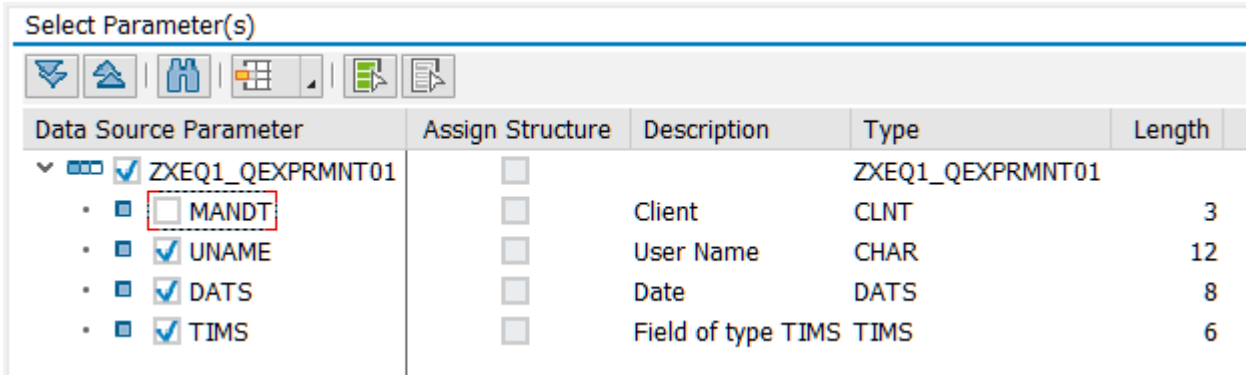
[SH1] ZXEQ1_LEXPRMNT01

```
atalog.sqlViewName: 'ZXEQ1_QEXPRMNT01'
atalog.compiler.compareFilter: true
@AccessControl.authorizationCheck: #CHECK
erText.label: 'Experiment 01'
 view ZXEQ1_LEXPRMNT01
elect
from zxeq1_dexprmnt01  as Data

Data.uname,
Data.dats,
Data.tims
}
```

During our next step we will Create Gateway Project via transaction code **SEGW** and import DDIC Structure (use @AbapCatalog.sqlViewName value from ABAP CDS, please see the picture above)

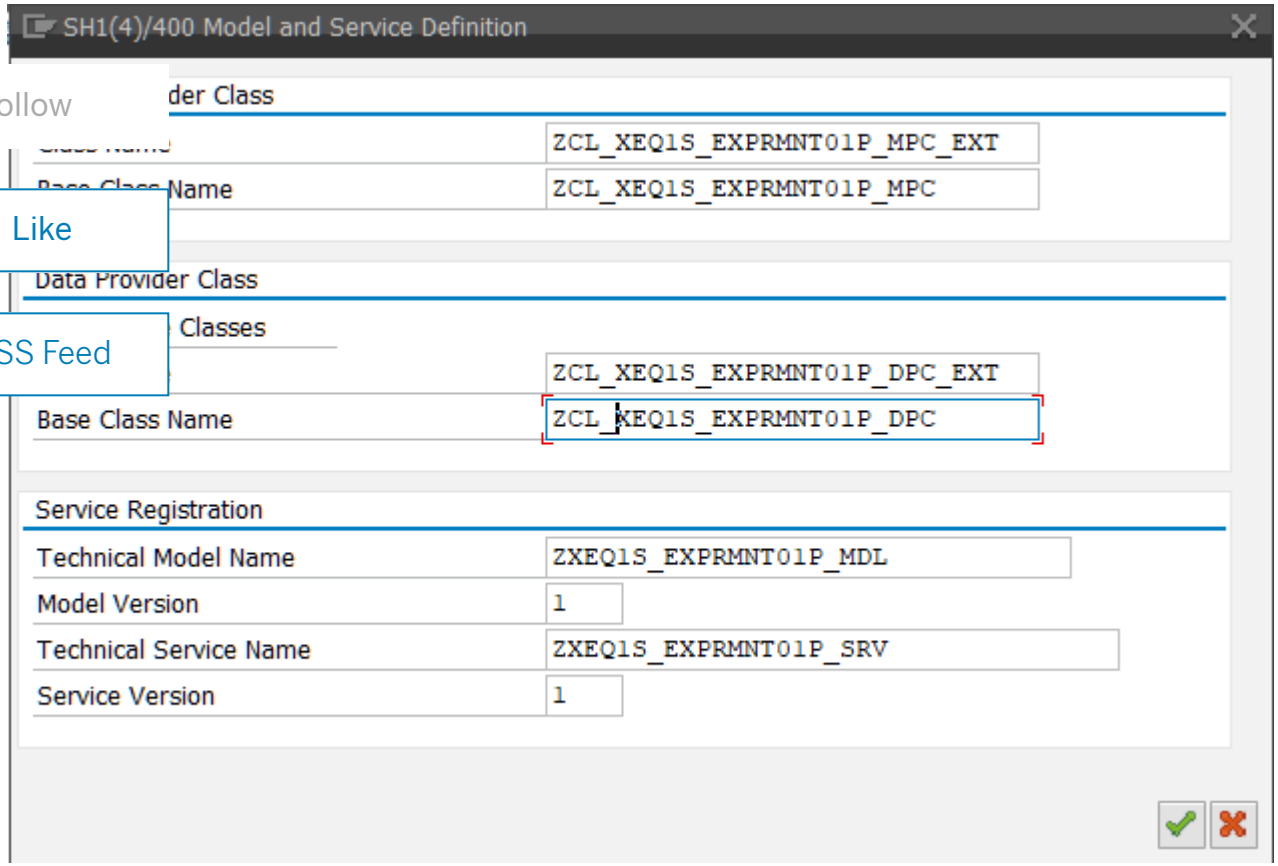






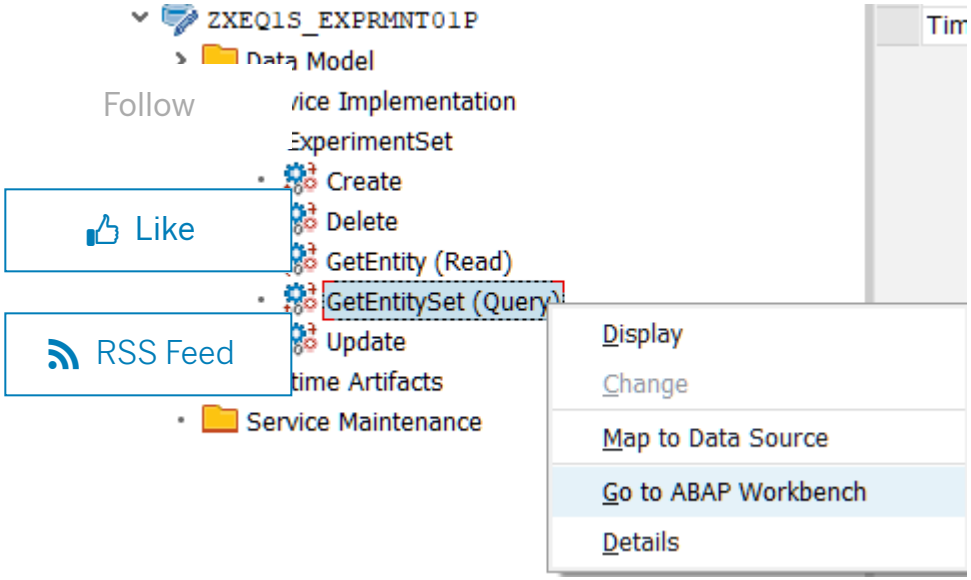
If you are using complex CDS with associations (for Master-Details view, for example) you should additionally create Associations and Navigation Properties. This section will be added.

SAP will generate runtime objects



For “CRUD” type of OData Services we should set parameters of Entity Set and then implement each of them

Navigate to ABAP Workbench and redefine methods of *DPC_EXT class for CRUD operations



Here it is possible to point out SELECT statement directly to ABAP CDS for **Get_EntitySet** method

```
METHOD experimentset_get_entityset.
   SELECT *
     FROM zxeq1_lexprmnt01
     INTO CORRESPONDING FIELDS OF TABLE @et_entityset
     ORDER BY PRIMARY KEY.
 ENDMETHOD.
```

If you set Searchable parameter of Entity Set then ABAP code implementation should look like the following

```
METHOD experimentset_get_entityset.
   DATA: lv_osql_where_clause TYPE string.
   "Prepare where clause
   lv_osql_where_clause = io_tech_request_context->get_osql_where_clause( ).
   "Select data
   SELECT *
```

```
    FROM zxeq1_lexprmnt01
    INTO CORRESPONDING FIELDS OF TABLE @et_entityset
    WHERE (lv_osql_where_clause)
    ORDER BY PRIMARY KEY.
ENDMETHOD.
```

Follow

Like

RSS Feed

...nally set Pageable parameter of Entity Set then ABAP code should look like the following

```
METHOD experimentset_get_entityset.
   DATA: lv_osql_where_clause TYPE string,
         lv_top               TYPE i,
         lv_skip              TYPE i,
         lv_max_index         TYPE i.
   "Prepare top and skip
   lv_top = io_tech_request_context->get_top( ).
   lv_skip = io_tech_request_context->get_skip( ).
   IF lv_top IS NOT INITIAL.
     lv_max_index = lv_top + lv_skip.
   ENDIF.
   "Prepare where clause
   lv_osql_where_clause = io_tech_request_context->get_osql_where_clause( ).
   "Select data
   SELECT *
     FROM zxeq1_lexprmnt01
     INTO CORRESPONDING FIELDS OF TABLE @et_entityset
     UP TO @lv_max_index ROWS
     WHERE (lv_osql_where_clause)
     ORDER BY PRIMARY KEY.
   "Process skip
   IF lv_skip IS NOT INITIAL.
     DELETE et_entityset TO lv_skip.
   ENDIF.
   "Process inline couter
```

```
    IF io_tech_request_context->has_inlinecount( ) = abap_true.
      SELECT COUNT(*)
        FROM zxeq1_lexprmnt01 WHERE (lv_osql_where_clause).
      es_response_context-inlinecount = sy-dbcnt.
    ELSE.
      CLEAR es_response_context-inlinecount.
    ENDIF.
  ENDMETHOD.
```

In **Get_Entity** method we should use ABAP CDS and other data type related objects and key field names

```
METHOD experimentset_get_entity.
   DATA: ls_data       TYPE zcl_xeq1s_exprmnt01p_mpc=>ts_experiment.
   "Convert keys to data
   CALL METHOD io_tech_request_context->get_converted_keys
     IMPORTING
       es_key_values = ls_data.
   "Select data by keys
   SELECT SINGLE *
     INTO CORRESPONDING FIELDS OF @er_entity
     FROM zxeq1_lexprmnt01
     WHERE uname = @ls_data-uname.
 ENDMETHOD.
```

If you set **Creatable** parameter of Entity Set then you should implement **Create_Entity** method and ABAP code could look like the following. In our current example we are using database table as target to INSERT data (in more complex and realistic business case SAP BAPI or functional modules or ABAP Classes should be used here to implement required business logic)

```
METHOD experimentset_create_entity.
   DATA: ls_data       TYPE zxeq1_dexprmnt01.
   "Insert data in database
   MOVE-CORRESPONDING er_entity TO ls_data.
   INSERT zxeq1_dexprmnt01
     FROM @ls_data.
ENDMETHOD.
```

If you set **Updatable** parameter of Entity Set then you should implement **Update_Entity** method like the following sample. In our current example we are using database table as target to UPDATE data (in more complex and realistic business case SAP BAPI or functional modules or ABAP Classes should be used here to implement required business logic)

```
METHOD experimentset_update_entity.
   DATA: ls_data       TYPE zxeq1_dexprmnt01.
   "Update data in database
   MOVE-CORRESPONDING er_entity TO ls_data.
   UPDATE zxeq1_dexprmnt01
     FROM @ls_data.
 ENDMETHOD.
```

If you set **Deletable** parameter of Entity Set then you should implement **Delete_Entity** method like the following sample. In our current example we are using database table as target to DELETE data (in more complex and realistic business case SAP BAPI or functional modules or ABAP Classes should be used here to implement required business logic)

```
METHOD experimentset_delete_entity.
   DATA: ls_data       TYPE zcl_xeq1s_exprmnt01p_mpc=>ts_experiment.
   "Convert keys to data
   CALL METHOD io_tech_request_context->get_converted_keys
     IMPORTING
```

```
            es_key_values = ls_data.
        "Delete data from database
          DELETE FROM zxeq1_dexprmnt01
            WHERE uname = @ls_data-uname.
      ENDMETHOD.
```

Follow

Like

RSS Feed

here is the following OData Service Document created

```
<?xml version="1.0" encoding="UTF-8"?>
- <app:service xml:lang="en" xmlns:sap="http://www.sap.com/Protocols/SAPData"
xmlns:m="http://schemas.microsoft.com/ado/2007/08/dataservices/metadata" xmlns:atom="http://www.w3.org/2005/Atom"
xmlns:app="http://www.w3.org/2007/app"
xml:base="http://hosting5.skybuffer.com:8013/sap/opu/odata/sap/ZXEQ1S_EXPRMNT01P_SRV/">
  - <app:workspace>
      <atom:title type="text">Data</atom:title>
    - <app:collection href="ExperimentSet" sap:content-version="1" sap:addressable="false" sap:searchable="true">
        <atom:title type="text">ExperimentSet</atom:title>
        <sap:member-title>Experiment</sap:member-title>
        <atom:link title="searchExperimentSet" type="application/opensearchdescription+xml"
          href="ExperimentSet/OpenSearchDescription.xml" rel="search"/>
      </app:collection>
    - <app:w      pace>
    - <n:l      ref="http://hosting5.skybuffer.com:8013/sap/opu/odata/sap/ZXEQ1S_EXPRMNT01P_SRV/" rel="self"/>
    <atom:link href="http://hosting5.skybuffer.com:8013/sap/opu/odata/sap/ZXEQ1S_EXPRMNT01P_SRV/" rel="latest-version"/>
</app:service>
```

Alert Moderator

Assigned tags

OData

ABAP Development

SAP HANA

SAP HANA studio

abap

abap cds

ABAP CDS view

Conclusion

This article shows a fast and simple way of OData Service creation with full CRUD functions support based on ABAP CDS technology. It is possible to see that wrapper for CRUD OData Service based on ABAP CDS could be created as fast as a few hours activity. What is needed is simply to add ABAP logic inside it that is mostly the same kind of activities we've done in ABAP development before web services usage. Additionally, OData Service created on ABAP CDS views is ready to be consumed by SAP HTML5 (UI5) Application and could either be deployed to your company local SAP Fiori Launchpad or to SAP Fiori Cloud Launchpad (Portal) from SAP Cloud Platform.

View more... 〉

Similar Blog Posts

Follow

[Implement custom ODATA service for Standard CDS View](#)

By Vighnesh Kamath Oct 29, 2016

Like

[Integration User for Odata services in SAP Cloud for Customer](#)

By Praveen Kumar Dwivedi Dec 03, 2019

RSS Feed

[Creation of Odata services for beginners](#)

By Ravi Verma Oct 07, 2019

Related Questions

[ABAP CDS View - Odata- Sharepoint 2013](#)

By Alexander Oertel May 24, 2017

[read write update delete operations using odata service](#)

By Former Member Feb 01, 2018

[sapui5 odata remove deep entity](#)

By Shweta Mourya Feb 16, 2018

20 Comments

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

Follow

Mahesh kumar palavalli
March 29, 2018 at 7:34 am

Hi Uladzimir Sapazhkou,

Nice Blog. There is one more way, where you can directly bind the CDS view to the entity. So we dont need to write the code to fetch the data. SADL layer will fetch the data for us with top, skip, filters.., applied automatically.

In the service implementation, right click on the entity and map to data source, here map the same to CDS view.

Thanks & Best Regards,

Mahesh

Like 6 | Share

Uladzimir Sapazhkou | Blog Post Author
March 29, 2018 at 10:18 am

Hi Mahesh Kumar Palavalli,

Thank you for useful information!

Like 1 | Share

Zubair D3 squad
April 24, 2018 at 6:59 am

Thanks mahesh.

Like 1 | Share

Deborshi De Sarkar
June 19, 2019 at 9:24 am

Hello, Mahesh Kumar Palavalli,

Which is the other way we can directly bind CDS view to the entity?

Like 0 | Share

Mahesh Kumar Palavalli

June 19, 2019 at 5:12 pm

Deborshi De Sarkaryou need to do the sadl mapping, import the cds view as entity types in the odata project and go to entity set and map it to the cds view. right click on the entityset you will get the option.

Like 0 | Share

Deborshi De Sarkar

June 20, 2019 at 7:29 am

Hi Mahesh Kumar Palavalli Is this BOPF ?

Like 0 | Share

Mahesh Kumar Palavalli

June 20, 2019 at 7:25 pm

No, it is'nt.. it's just normal mapping of cds view to the entity set.. You can even map a database table or view..

Like 0 | Share

Sarbjeet Singh

June 25, 2019 at 3:27 pm

Hey Mahesh,

Is there any way to pass on the CDS UI annotations automatically using this mapping editor method?

Regards,

Sarbjeet Singh

Like 0 | Share

Mahesh Kumar Palavalli

June 25, 2019 at 6:33 pm

Hi Sarbjeet Singh

I don't think it is possible if we do like that. I've only observed the annotations flowing from the Odata service that is generated from the CDS view or via the referenced odata source.

BR,

Mahesh

Like 1 | Share

Sarbjeet Singh

June 25, 2019 at 6:47 pm

Thanks.

Like 0 | Share

Thomas Mundt

November 14, 2019 at 1:02 pm

Hi @Mahesh Kumar Palavalli

Is it possible to redefine methods for CDS, to apply addational data/logic not being covered by CDS?

I try to call method to include more data in entity while GET request is initiated for the entity.

Your help is much appreciated!

Thomas

Like 0 | Share

Mahesh Kumar Palavalli

November 14, 2019 at 6:21 pm

If the service is not generated from the CDS view then yes it is possible as you will be doing it via the reference odata source or entity set binding to individual cds view.

So in the dpc_ext class, redefine the required enitityset method and call the super class method again so that the SADL layer will take care of the actual read then after that manipulate the response.

is that what you are expecting?

Like 1 | Share

Thomas Mundt

November 15, 2019 at 12:16 pm

Yes I do. You confirmed something I only could guess about. Thank you very much for your valuable response!
Is it possible to access data from a database table of another system in this method? What I need to achieve is to fetch data from another odata service and its entity.

Like 0 | Share

Thomas Mundt

December 27, 2019 at 11:00 am

FYI:
I skipped data retrieval on the ABAP/backend side. My solution is to the stuff on frontend, accessing two different OData services.

Like 0 | Share

Che Eky

October 3, 2020 at 9:43 am

Hi @Mahesh Kumar Palavalli,
Say the OData is generated via the reference odata source and the CDS contains aggregation annotation. If you manipulate the response in the dpc_ext class could it affect the guid generated for aggregation? or is the aggregation and guid generated after manipulating the response in the dpc_ext class?

Like 0 | Share

Che Eky

October 6, 2020 at 10:44 am

I tested this and the aggregation has already been done by the time the data is available in the redefined get_entityset. So you cannot add or filter out rows of data as it will invalidate the totals. What you can do is calculate additional fields for display that will not affect the totals, e.g. fetch a text, etc. Because of pagination only a subset of data is available in the get_entityset, not all. This has the benefit that you can perform calculations on a subset of data which should be faster than the whole dataset.

Like 0 | Share

Bodhisattwa Pal

January 11, 2021 at 12:09 pm

Hi Mahesh

Thanks for raising this topic .

I was under the impression that SADL technique can be used only for fetching data .Not for making any modification (Update , Delete) .

Could you kindly share any article how to make modifications in a CDS view , using SADL .

The only two options I am aware of , for making modifications (Update , Delete) are BOPF and the one mentioned by @Uladzimir .

Thanks

Bodhisattwa Pal

Like 0 | Share

Mahesh Kumar Palavalli

January 12, 2021 at 12:50 pm

I think those are the only ways with CDS views.

Reference data source/cds data source for entity & manually overriding the crud methods

abap programming model for fiori & usng bopf

ABAP RAP

Like 0 | Share

Muthukumaran Pandian

April 8, 2019 at 1:49 pm

Thanks Mahesh.

Like 1 | Share

Follow

Idris Ahmed Khan
May 23, 2019 at 8:44 am

Hi Uladzimir,

Thanks for the great document.
I tried to implement the above-explained method. will it only work for the custom dictionary object because in delete method you mentioned to delete it specifically?
I have a requirement to use standard I_MaterialStock view and it is required to remove or hide a field in Fiori query browser, can we achieve delete fields for standard views using the above method?
we have a requirement to hide/remove a certain field from standard view "C_MaterialStockActual". is there any other method to hide fields in a fiory application?

Regards,
Idris.

Like 0 | Share

Legal Disclosure	Copyright
Follow	Cookie Preferences
Like	Support

RSS Feed