

# Create an oData service from CDS

**Author** : Tobias Hofmann

**Date** : November 6, 2019

This blog is about how to create an oData service from a CDS View. The code and example follow closely SAP Help documentation and the included example on this topic:

- [Develop CDS View](#)
- [Expose as OData service](#)

I only cut the documentation overhead and make the information available in a single blog. As you can see in the above two links, the task consists of 2 steps:

1. Create CDS View
2. Expose OData service

For the example, I used NW ABAP 7.52 Developer Edition and ABAP in Eclipse (ADT) tools. If you have a "real" SAP NW ABAP System available, you may also implement the sample service there.

## Create CDS Data Source

In ADT, create a new CDS Data Definition.

Name: ZDEMO\_CDS\_SalesOrderItem Description: List Reporting for Sales Order Item

Click on next to go through the wizard.

Paste the following code in the new created file:

[https://github.com/tobiashofmann/cds\\_sample\\_service/blob/master/ZDEMO\\_CDS\\_SalesOrderItem](https://github.com/tobiashofmann/cds_sample_service/blob/master/ZDEMO_CDS_SalesOrderItem)

```
@AbapCatalog.sqlViewName: 'ZDEMO_SOI_001' @AbapCatalog.compiler.compareFilter:
  true @AbapCatalog.preserveKey: true @AccessControl.authorizationCheck: #CHECK
  @EndUserText.label: 'List Reporting for Sales Order Item' @OData.publish: true
e  define view ZDEMO_CDS_SalesOrderItem as select from SEPM_I_SalesOrderItem_E
  as Item {
    key Item.SalesOrder as SalesOrderID,
    key Item.SalesOrderItem as ItemPosition,
    Item._SalesOrder._Customer.CompanyName as CompanyName,
    Item.Product as Product,
    @Semantics.currencyCode: true Item.TransactionCurrency as CurrencyCode,
    @Semantics.amount.currencyCode: 'CurrencyCode' Item.GrossAmountInTransacCurrency as GrossAmount,
    @Semantics.amount.currencyCode: 'CurrencyCode' Item.NetAmountInTransactionCurrency as NetAmount,
    @Semantics.amount.currencyCode: 'CurrencyCode' Item.TaxAmountInTransactionCurrency as TaxAmount,
    Item.ProductAvailabilityStatus as ProductAvailabilityStatus }

```

Save and activate the CDS View.

## Activate OData Service

The above created a CDS Data Definition and when activating, some magic happened. What is missing is to activate the OData service. ADT won't do this for you, this needs to be done manually in the Gateway System.

Tx: /n/IWFND/MAINT\_SERVICE

Click on Add Service

Search for services in the local system.

System Alias: LOCAL Technical Service Name: ZDEMO\_CDS\_SALESORDERITEM\_CDS

Click on Get Services

The CDS Service is shown.

Select the service and click on Add selected Services

Add service dialog.

Package Assignment: \$TMP (click on Local Object)

## Test service

After performing the above steps, the CDS View is implemented and the OData service exposing the data is activate and can be used. You may now test the service to see if everything is working as expected.

Tx: /n/IWFND/MAINT\_SERVICE Select the service: ZDEMO\_CDS\_SALESORDERITEM\_CDS

Click on SAP Gateway Client. To test the service, use the URL:

`/sap/opu/odata/sap/ZDEMO_CDS_SALESORDERITEM_CDS/$metadata`

## It's full of stars!

*Where documentation meets reality*

---

Available entity sets can be seen by clicking on EntitySets.

Available options by clicking on Add URI Option

## It's full of stars!

*Where documentation meets reality*

---

To see the top 2 results in json, the URL is:

```
/sap/opu/odata/sap/ZDEMO_CDS_SALESORDERITEM_CDS/ZDEMO_CDS_SalesOrderItem?$top=2&$format=json
```