

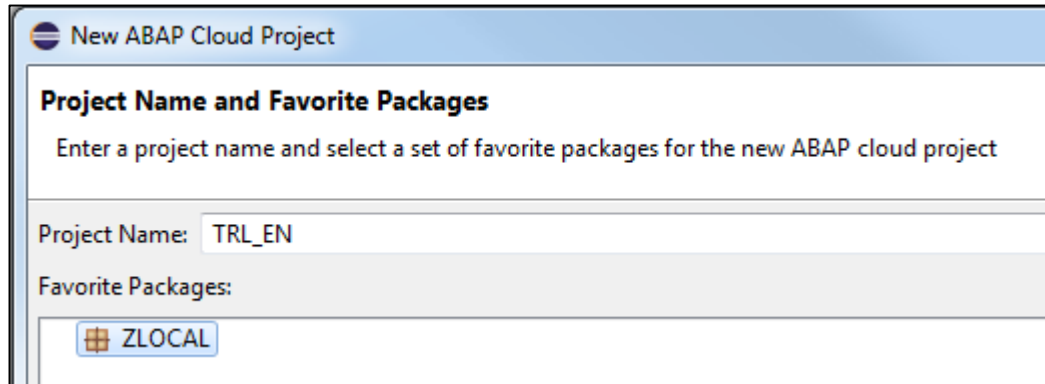
Core Data Services (CDS)

Tutorial 01: Use a Core Data Services for the access to a database table

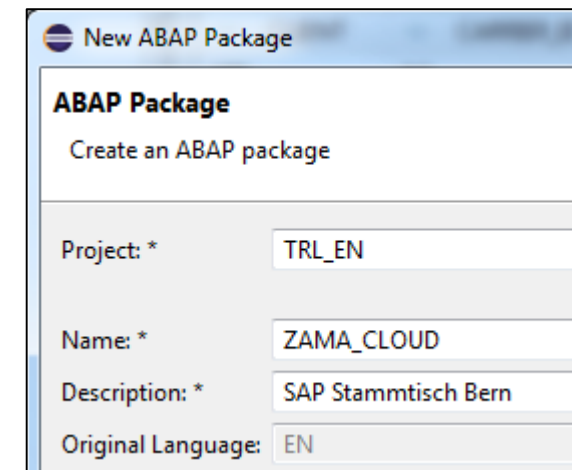
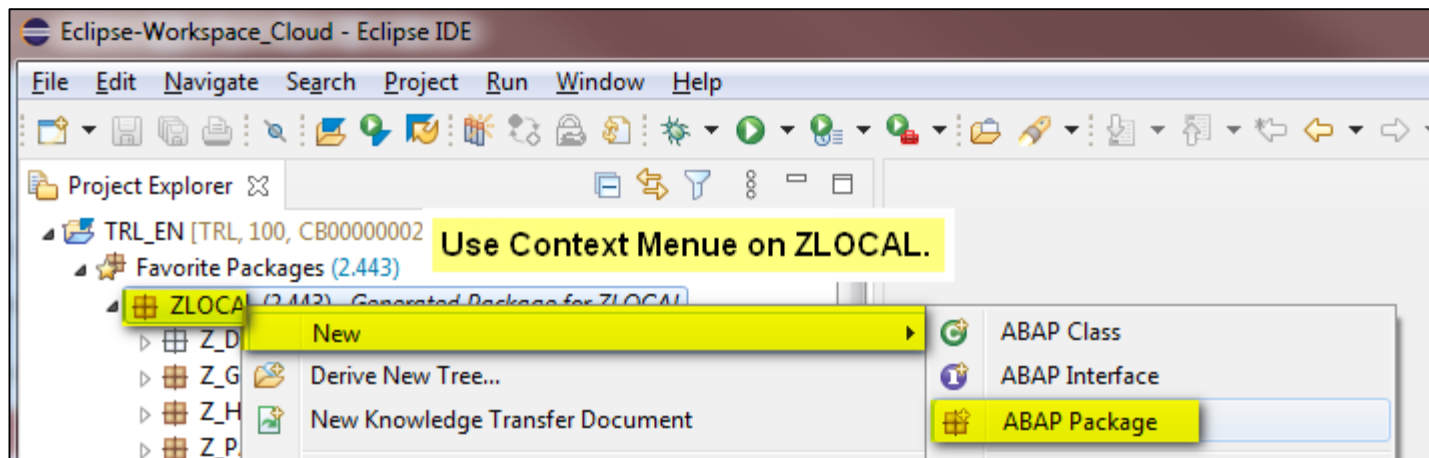
Step 01 - Create an own subpackage



- During the Preparation Phase you have defined a favorite package ZLOCAL for your ABAP Cloud Project.



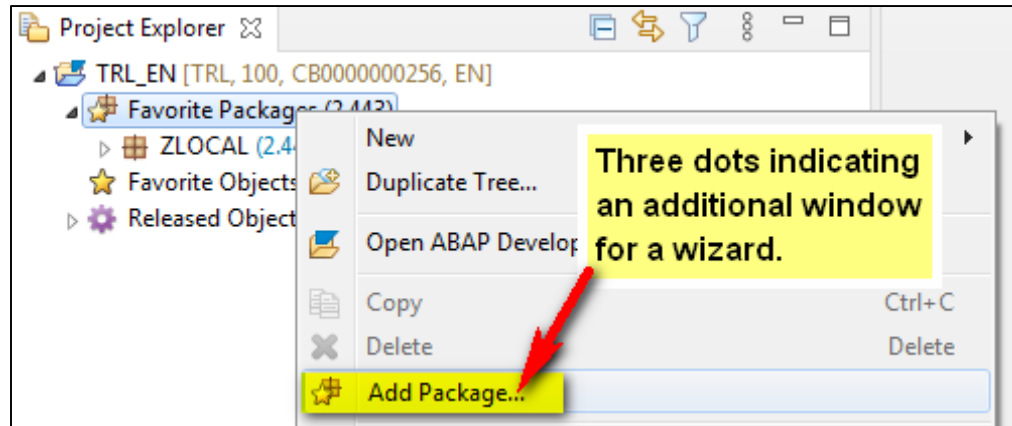
- Create an own Subpackage with a unique name in ZLOCAL.



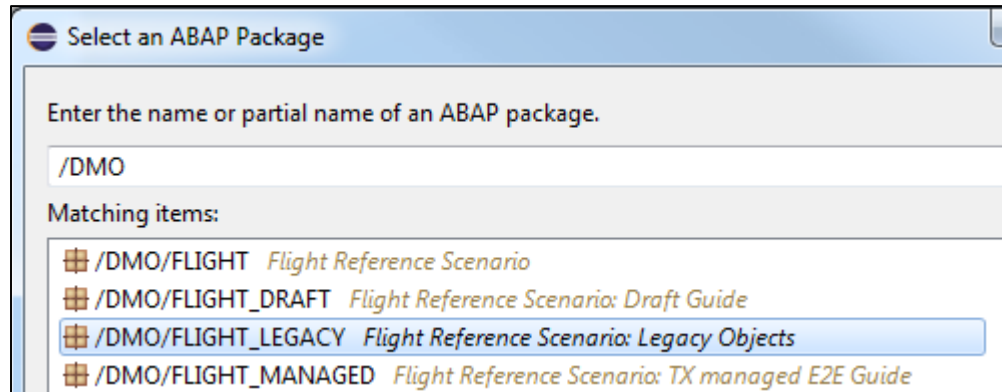
Step 02 - Make the Flight-Package a favorite one



- Add a package using the context menu.



- Add package /DMO/FLIGHT_LEGACY using the context menu.

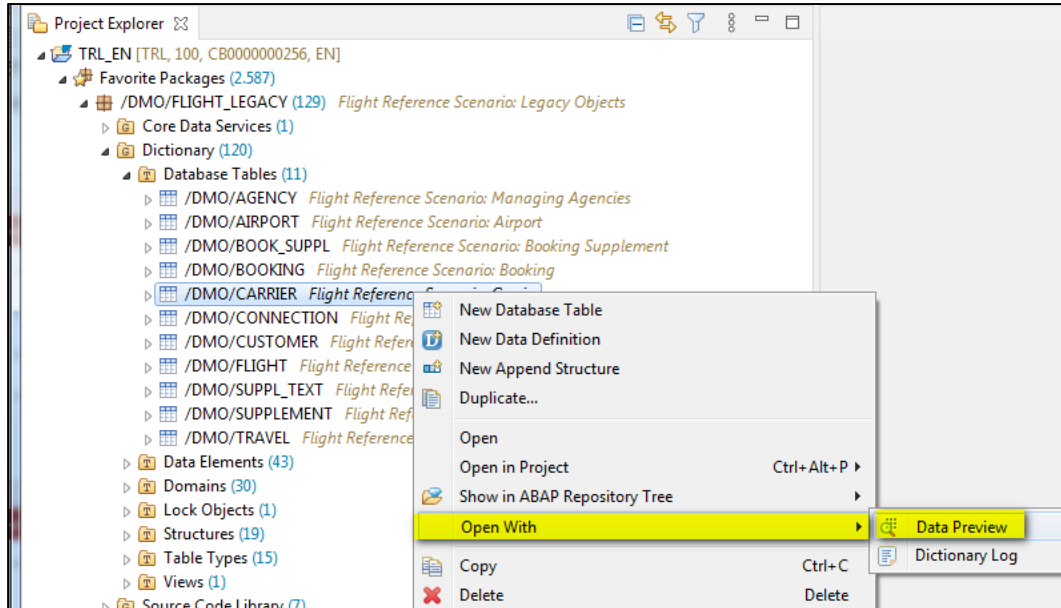


Step 03 - Have a look at the data model



➤ Have a look at the available data.

<https://help.sap.com/viewer/923180ddb98240829d935862025004d6/Cloud/en-US/a9d7c7c140a0408dbc5966c52d156b49.html>

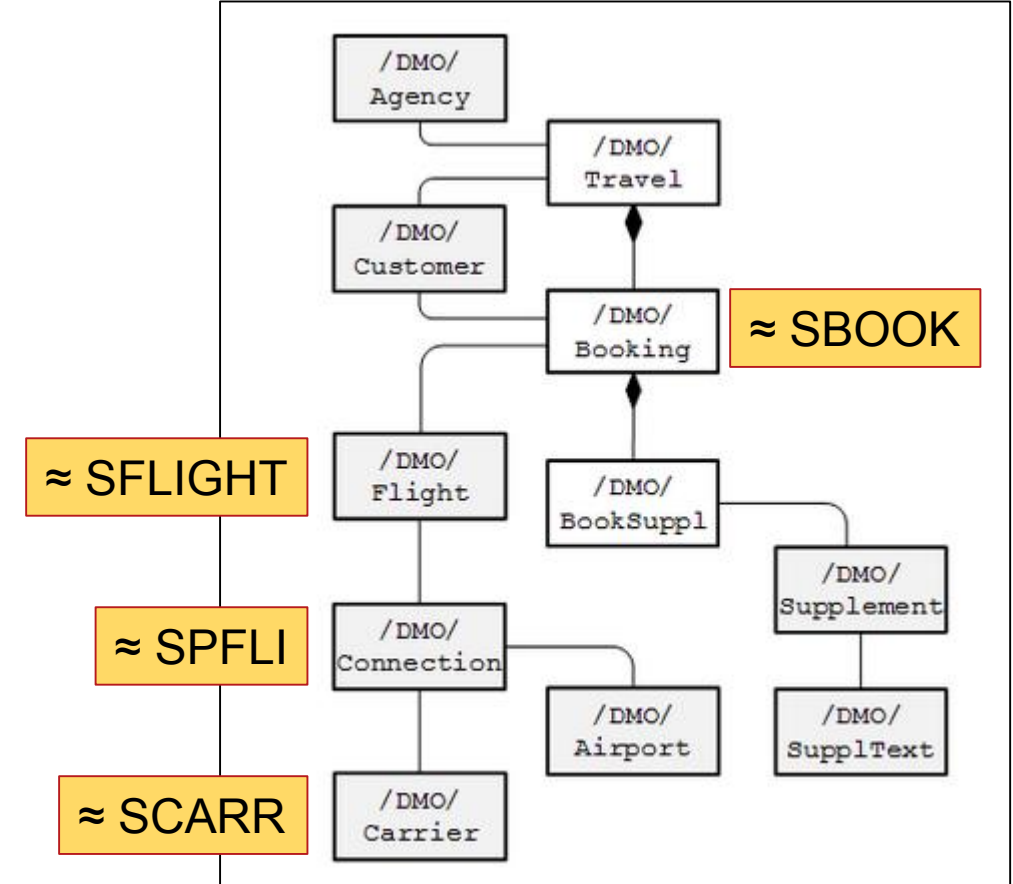


[TRL] /DMO/CARRIER

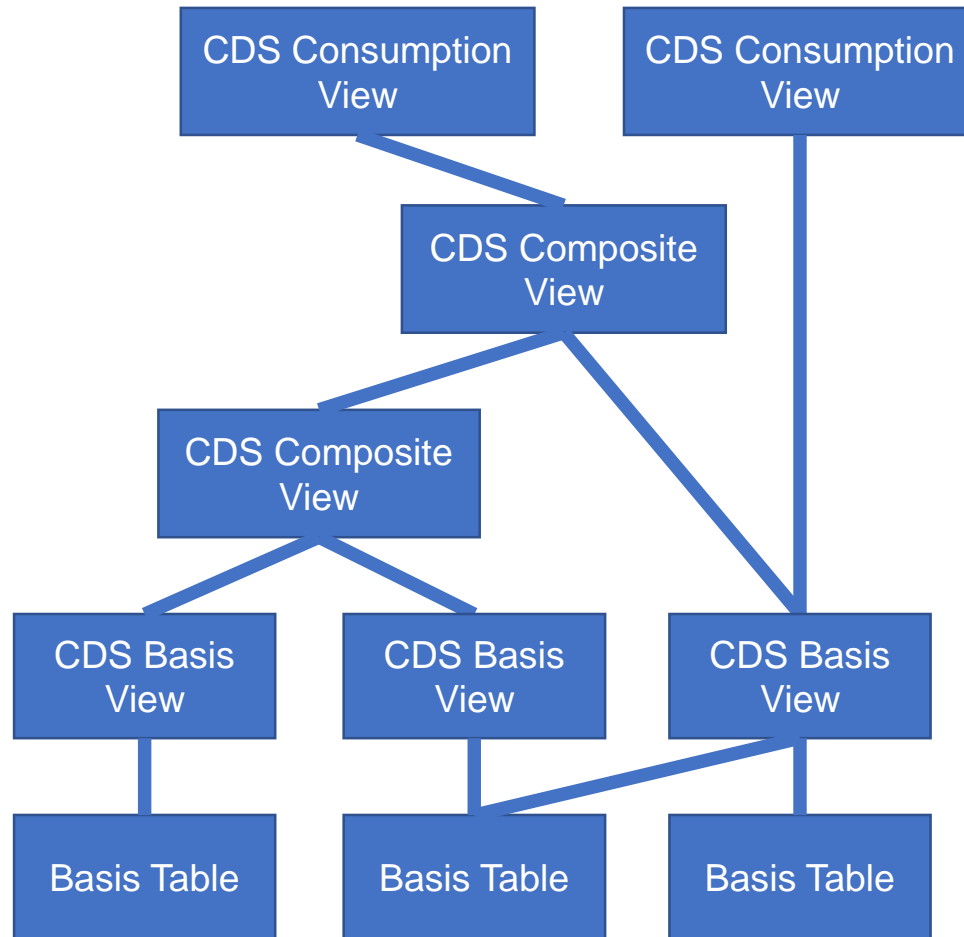
Raw Data

Filter pattern 16 rows retrieved - 2 ms

AB	CLIENT	AB	CARRIER_ID	AB	NAME	AB	CURRENCY_CODE
	100		AA		American Airlines Inc.		USD
	100		AC		Air Canada		CAD
	100		AF		Air France		EUR
	100		AZ		Alitalia Societa Aerea Italiana S.p.A.		EUR



CDS Viewtypes of the SAP Virtual Data Model (VDM)



CDS Consumption View: Expose the data to the access of different consumers (ABAP reports, analytics tools, ...).

C_<CDS name>,
ZC_<CDS name>

CDS Composite (Interface) View: Middle layer to combine, manipulate and process data of different CDS Basic Views.

I_<CDS name>,
ZI_<CDS name>

CDS Basic (Interface) View: Fetching the raw data from the real database tables by filtering the access to columns and doing an initial processing.

I_<CDS name>,
ZI_<CDS name>

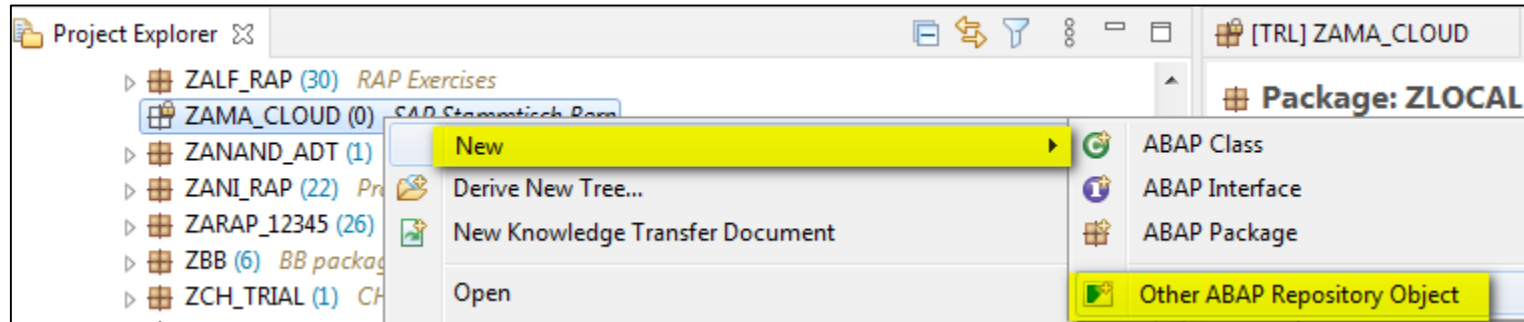
Basis Tables: The database tables containing the raw data.

Naming conventions e.g. here:
<https://blog.sap-press.com/how-to-name-a-virtual-data-model-in-sap-s4hana>

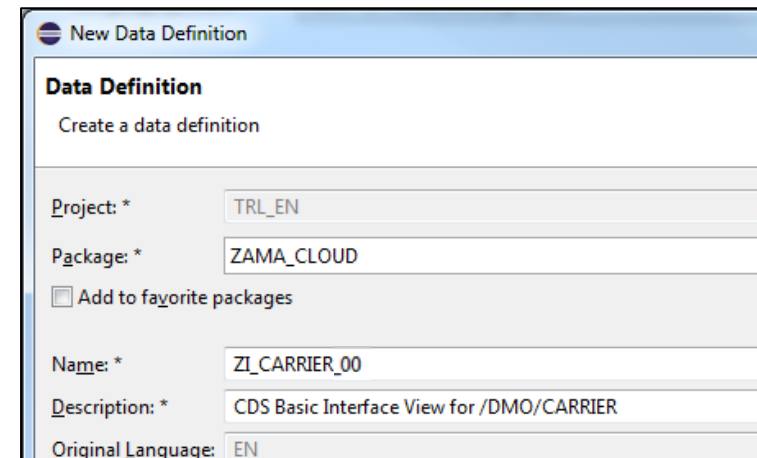
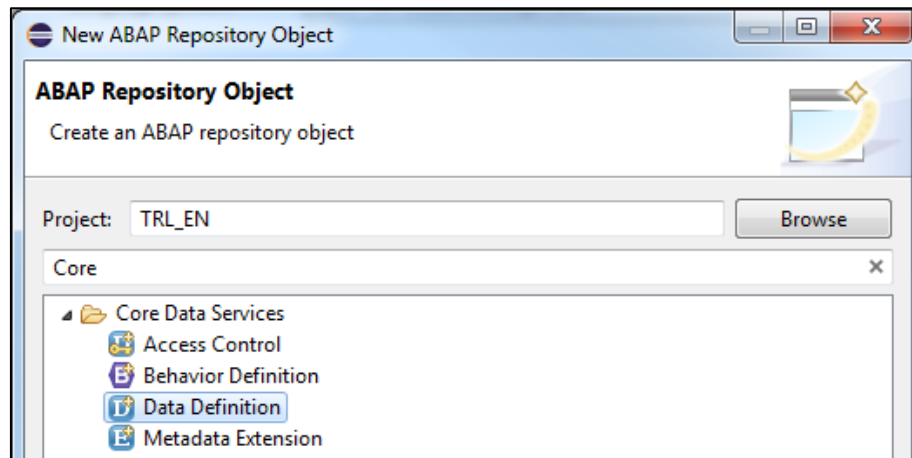
Step 04 - Create a first CDS-View (1)



- Create a new ABAP Repository Object.



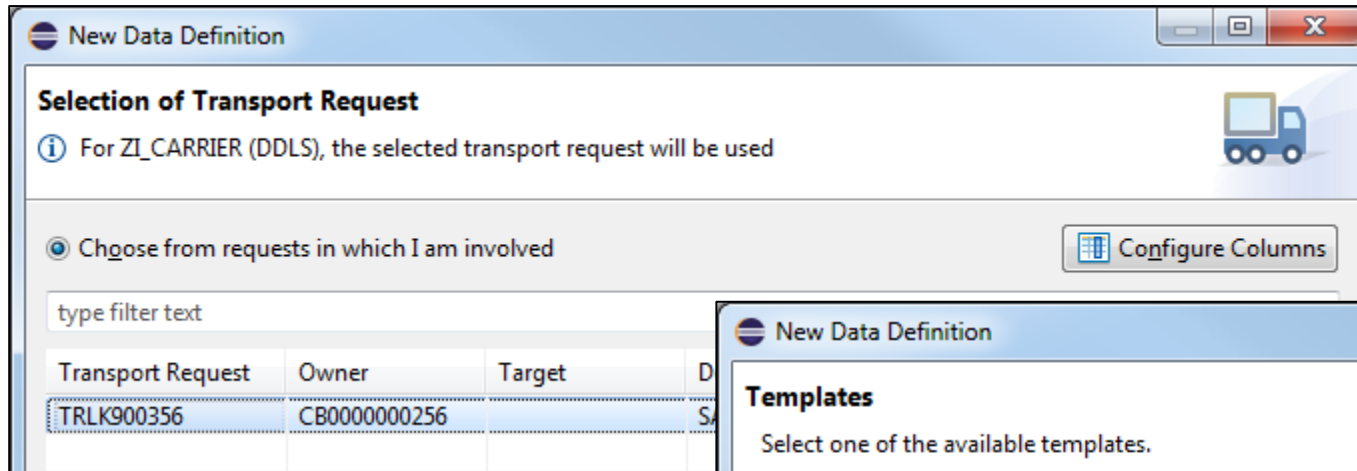
- Choose *Core Data Services* -> *Data Definition* and create a first CDS Basic Interface View `ZI_CARRIER_##`, following the naming convention.



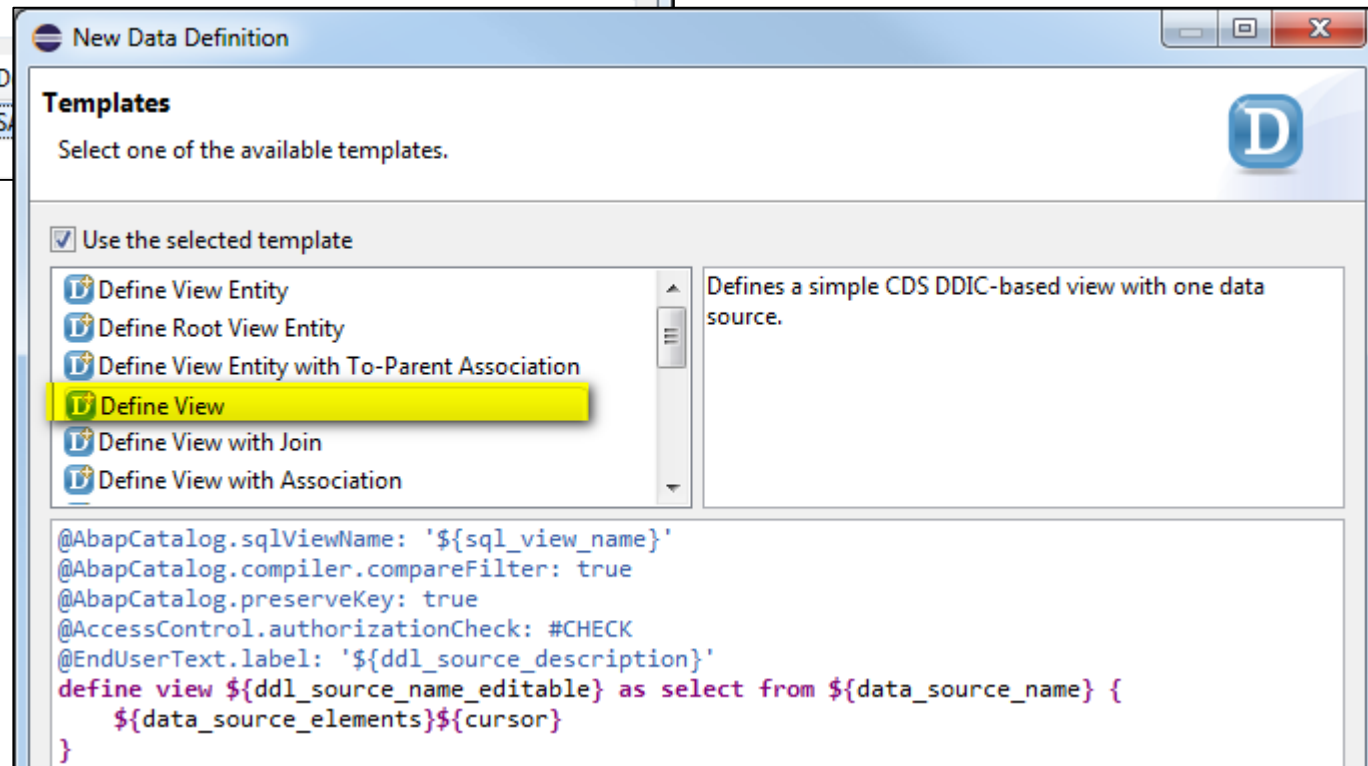
Step 04 - Create a first CDS-View (2)



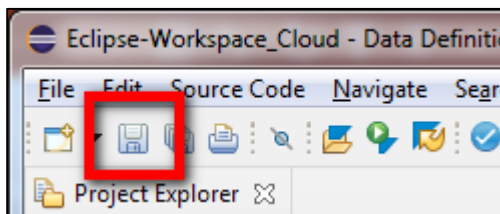
- Confirm the transport - and choose *Next*, **not** *Finish*!



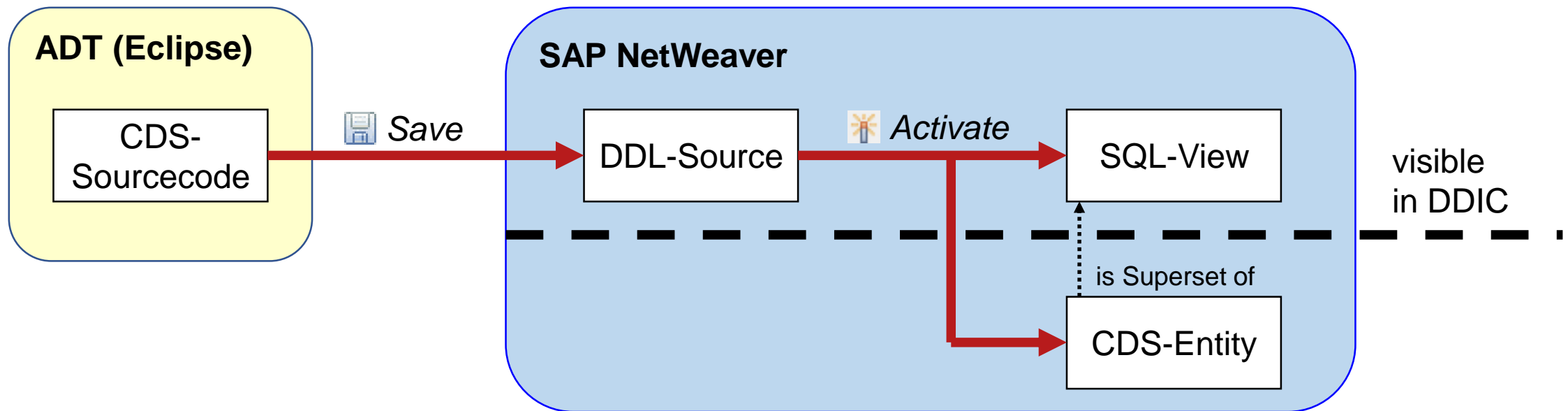
- Select template *Define View*.



- Don't forget to *Save* afterwards.



- **ABAP Core Data Services (ABAP-CDS)** are ABAP Repository Objects which consists of different smaller parts.



ONE single
SQL-SELECT-
Statement

```
D [TRL] ZI_CARRIER_00 ⌘  
1 @AbapCatalog.sqlViewName: 'ZI_CARRIER_00_A'  
2 @AbapCatalog.compiler.compareFilter: true  
3 @AbapCatalog.preserveKey: true  
4 @AccessControl.authorizationCheck: #CHECK  
5 @EndUserText.label: 'CDS Basic Ifc View for /DMO/CARRIER'  
6 define view ZI_CARRIER_00  
7   as select from /dmo/carrier  
8 {  
9   key carrier_id    as CarrierId,  
10     name            as Name,  
11     currency_code   as CurrencyCode  
12 }  
13
```

Annotations
to add metadata
to the statement

Documentation:

<https://help.sap.com/viewer/cc0c305d2fab47bd808adcad3ca7ee9d/7.5.9/en-US/630ce9b386b84e80bfade96779fbaeec.html>

CDS-Sourcecode = DDL-Source

SQL-View

CDS-Entity
(sometimes:
CDS-View)

 [TRL] ZI_CARRIER_00 

```
1 @AbapCatalog.sqlViewName: 'ZI_CARRIER_00_A'
2 @AbapCatalog.compiler.compareFilter: true
3 @AbapCatalog.preserveKey: true
4 @AccessControl.authorizationCheck: #CHECK
5 @EndUserText.label: 'CDS Basic Ifc View for /DMO/CARRIER'
6 define view ZI_CARRIER_00
7   as select from /dmo/carrier
8 {
9   key carrier_id    as CarrierId,
10     name            as Name,
11     currency_code   as CurrencyCode
12 }
13
```

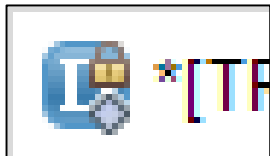
Step 05 - Define the name of the SQL-View



- Define the name of the SQL-View.

```
*[TRL] ZI_CARRIER_00 ⌘  
1 @AbapCatalog.sqlViewName: 'ZI_CARRIER_00_A'  
2 @AbapCatalog.compiler.compareFilter: true  
3 @AbapCatalog.preserveKey: true  
4 @AccessControl.authorizationCheck: #CHECK  
5 @EndUserText.label: 'CDS Basic Ifc View for /DMO/CARRIER'  
6 define view ZI_CARRIER_00 as select from /dmo/carrier {  
7  
8 }
```

- Notice the flags of the *Data Definition* icon.



- Lock: Locked against changing in SAPGUI-tools.
- Rhombus: not activated yet
- Star: not saved yet

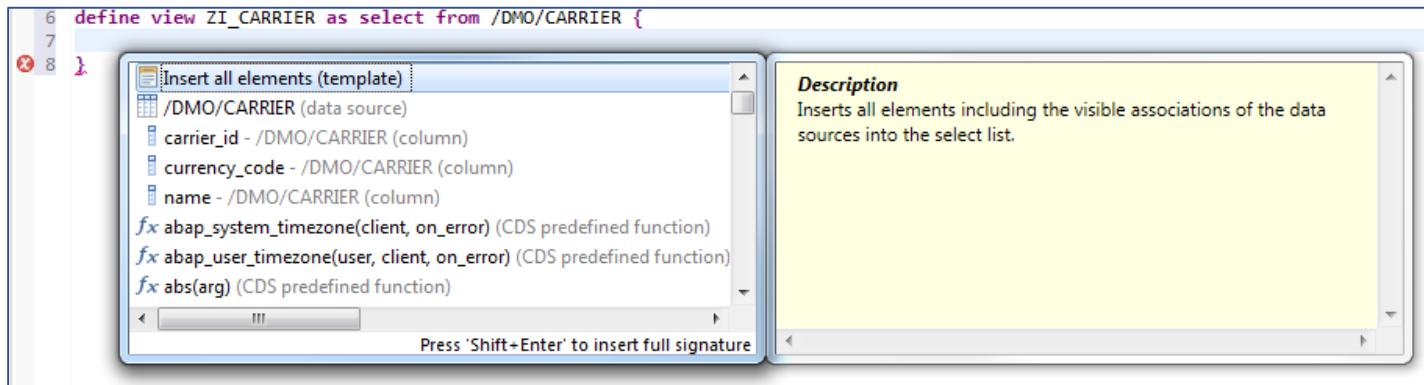
Step 05 - Define SQL SELECT statement



- Enter the name of the data source.

```
5 @EndUserText.label: 'CDS Basic Interface View for /DMO/CARRIER'  
6 define view ZI_CARRIER_00 as select from /dmo/carrier {
```

- Notice the code completion functionality (**Ctrl** + **blank** in curly brackets).



- Write down the SELECT statement. Note the optional semicolon at the end of the statement.

```
6 define view ZI_CARRIER_00  
7   as select from /dmo/carrier  
8 {  
9   key carrier_id    as CarrierId,  
10     name            as Name,  
11     currency_code  as CurrencyCode  
12 }
```

➤ Notice length restrictions!

```
*[TRL] ZI_CARRIER_00 ⓘ
1 @AbapCatalog.sqlViewName: 'ZI_CARRIER_00_A'
2 @AbapCatalog.compiler.compareFilter: true
3 @AbapCatalog.preserveKey: true
4 @AccessControl.authorizationCheck: #CHECK
5 @EndUserText.label: 'CDS Basic Interface View for /DMO/CARRIER'
6 define view ZI_CARRIER_00
7   as select from /dmo/car
8 {
9   key carrier_id    as CarrierId,
10     name            as Name,
11     currency_code   as CurrencyCode
12 }
13
```

Multiple markers at this line:
- Token "'CDS Basic Interface View for /DMO/CARRIER'" contains more than 40 characters (41)
- Annotation value 'CDS Basic Interface View for /DMO/CARRIER' length (41) too long. Type must be 'String(40)'

Name of SQL View (ZI_CARRIER_00_A): up to 16 characters

Name of CDS Entity (ZI_CARRIER_00): up to 40 characters

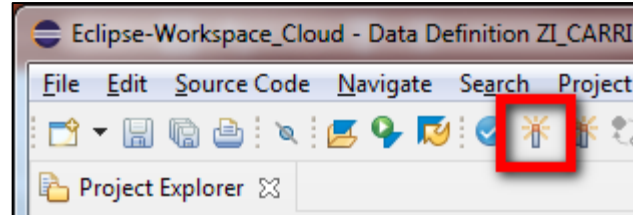
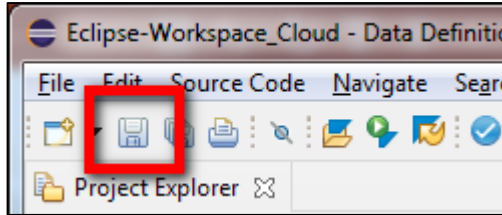
Name of DDL Source (ZI_CARRIER_00): up to 40 characters

Annotations: length is variable, depending on the Annotation.

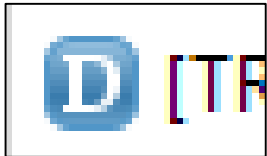
Step 06: Save and activate



- Save and activate the CDS View.



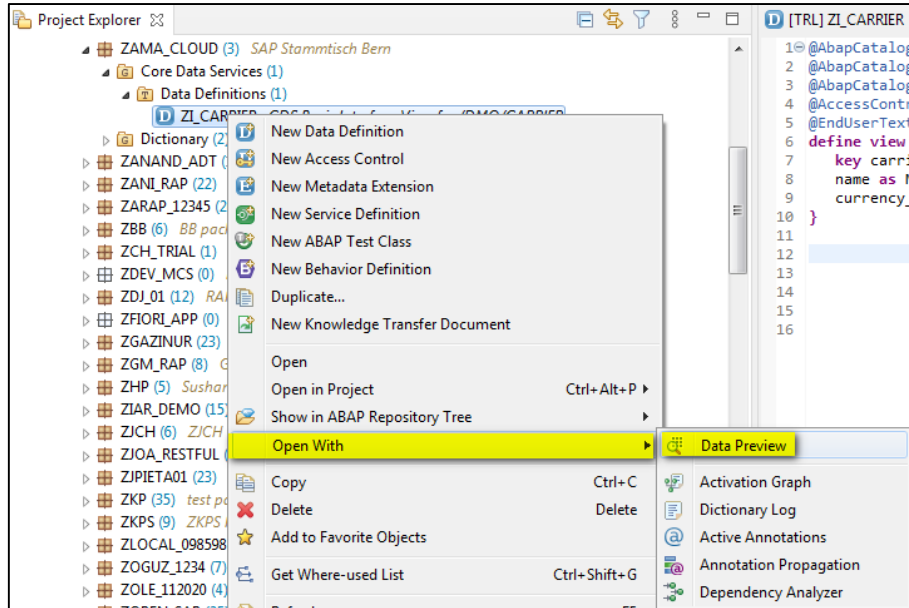
- Notice the change of the flags of the *Data Definition* icon.



Step 07: Data Preview



- Do a preview of the data of the CDS.



[TRL] ZI_CARRIER_00

ZI_CARRIER_00

Raw Data

Filter pattern 16 rows retrieved - 21 ms

CarrierId	Name	CurrencyCode
AA	American A...	USD
AC	Air Canada	CAD
AF	Air France	EUR
AZ	Alitalia Soci...	EUR
BA	British Airw...	GBP
FJ	Air Pacific L...	USD
CO	Cobaltair Lt...	USD

Appendix - Sourcecode



```
@AbapCatalog.sqlViewName: 'ZI_CARRIER_00_A'
@AbapCatalog.compiler.compareFilter: true
@AbapCatalog.preserveKey: true
@AccessControl.authorizationCheck: #CHECK
@endUserText.label: 'CDS Basic Ifc View for /DMO/CARRIER'
define view ZI_CARRIER_00
  as select from /dmo/carrier
{
  key carrier_id    as CarrierId,
    name            as Name,
    currency_code   as CurrencyCode
}
```