Products Industries Services and Support Training Community Developer Partner About

Ask a Question Write a Blog Post

Login

Chandrashekhar Mahajan

September 24, 2014 14 minute read

Let’s code association/navigation and data provider expand in OData service!

Follow RSS feed Like

42 Likes 101,152 Views 34 Comments

Introduction

In my earlier blog Let’s code CRUDQ and Function Import operations in OData service! we understood the basic operation performed in OData service.

In this blog I will explain creation of simple SAP Gateway OData service having association and navigation between entities. Also we will see how to implement it through code based approach and finally conclude with implantation of GET_EXPANDED_ENTITYSET and GET_EXPANDED_ENTITY

invoked by $expand.

Note – Steps mentioned in this blog are performed in SAP Gateway System SP08 (Embedded Architecture)

Let’s see what is meant by association and navigation property.

> ***Associations** define the relationship between two or more Entity Types (for example, Employee WorksFor Department). Instances of associations are grouped in **Association Sets**.*
>
> ***Navigation Properties** are special properties on Entity Types which are bound to a specific association and can be used to refer to associations of an entity.*
>
> *Finally, all instance containers (Entity Sets and Association Sets) are grouped in an **Entity Container**.*

Reference – Overview | Open Data Protocol

Also let’s understand the difference between association/navigation and $expand. In short, it is as below,

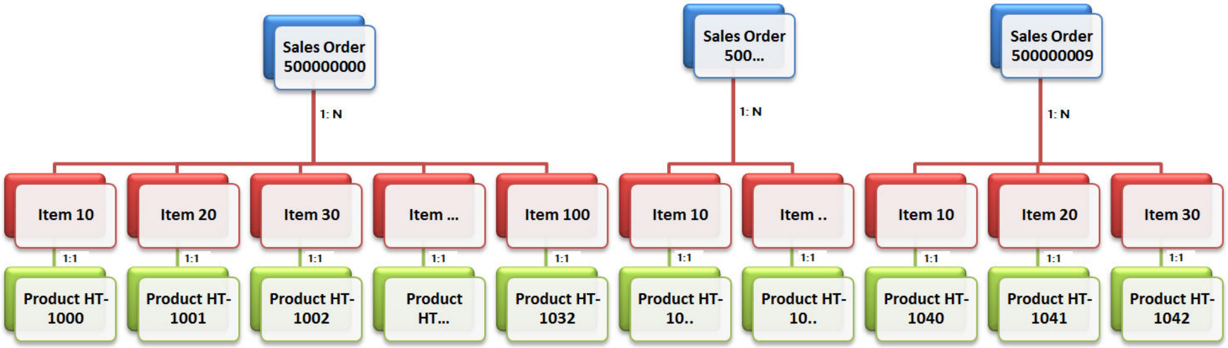
Association/Navigation	Give me associated (dependent) entity/entities using Navigation property	http://services.odata.org/OData/OData.svc/Categories(1)/Products?$format=json

$expand	Give me associated (dependent) entity/entities + **Principal entity/entities** using Navigation property	http://services.odata.org/OData/OData.svc/Categories(1)?$expand=Products&$format=json

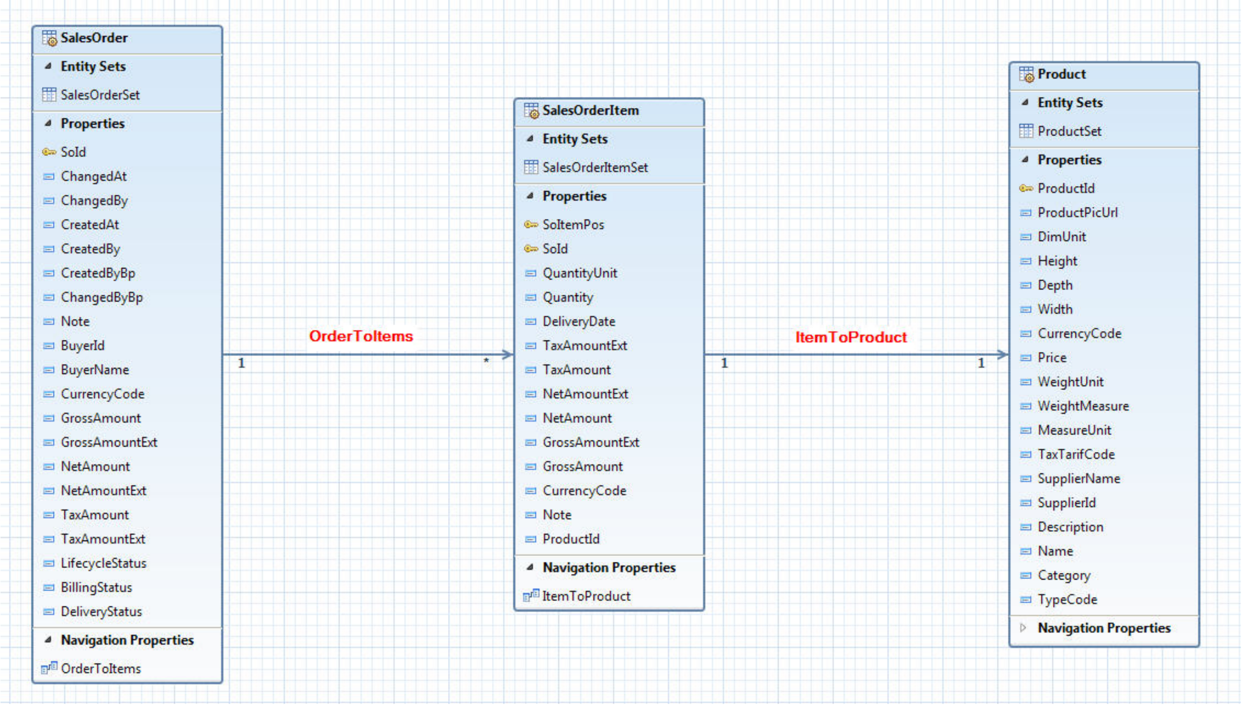
you can also refer this nice blog Implementing Expand Entity/Entity Set by Srikanth Gajula

Scenario

We will read Sales order, items and product data from Enterprise Procurement Model (EPM). This is the pictorial representation of sales order, items and product with their association.



We will have 3 entities as displayed in below EDM diagram.



SalesOrder will be associated with SalesOrderItem and SalesOrderItem with Product.

Principle Entity	Dependent Entity	Navigation Property	Cardinality
SalesOrder	SalesOrderItem	OrderToItems	1:N
SalesOrderItem	Product	ItemToProduct	1:1

We will use below BAPIs to get the Sales Order, Items and Product data in DPC_EXT class methods.

- BAPI_EPM_SO_GET_LIST
- BAPI_EPM_SO_GET_DETAIL
- BAPI_EPM_PRODUCT_GET_DETAIL

We will code for association/navigation and data provider expand scenario and will also understand the framework expand.

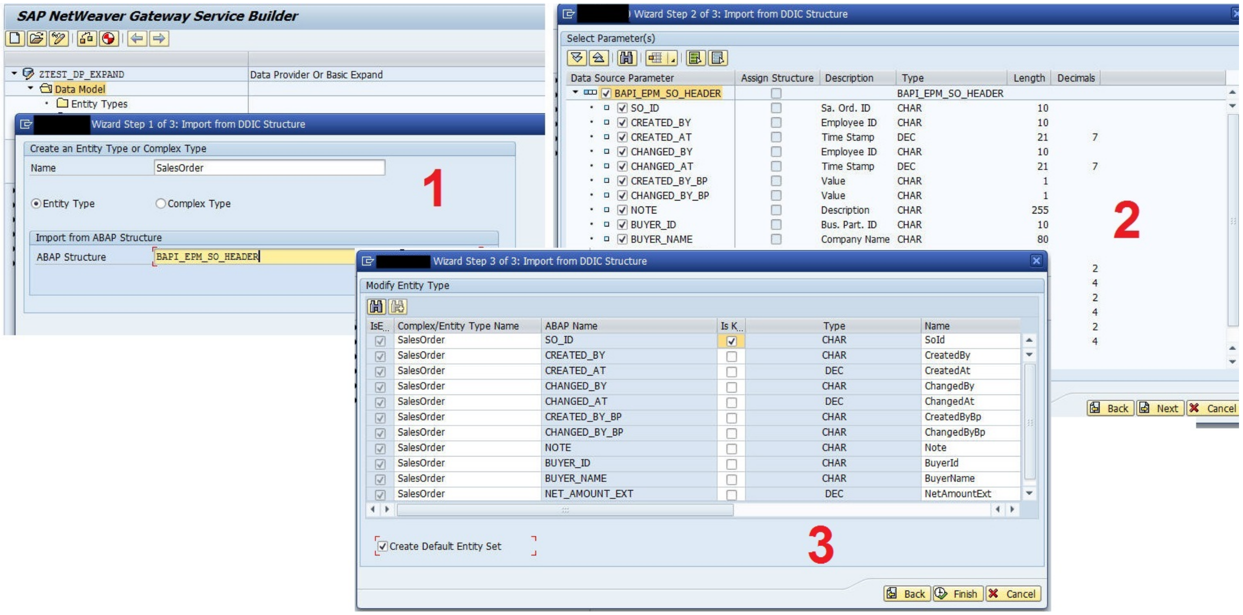
Before that just look at difference between Data provider expand and framework expand

Framework Expand	Data Provider Expand
Formerly called as generic expand	Formerly called as basic expand
Requires no implementation effort	Requires implementation effort
As this is handled by framework, same logic may be called multiple times in loop resulting in poor performance	In some cases, this provides better performance depending on how the code is implemented

Reference – Expand in Framework and Data Provider – SAP NetWeaver Gateway – SAP Library

Procedure

Create entity SalesOrder by importing DDIC structure as shown below. Please note that Entity set will be created by default if the check box “Create Default Entity Set” is checked.


SAP NetWeaver Gateway Service Builder

ZTEST_DP_EXPAND — Data Provider Or Basic Expand
- Data Model
  - Entity Types

[redacted] Wizard Step 1 of 3: Import from DDIC Structure

Create an Entity Type or Complex Type

Name: SalesOrder

● Entity Type ○ Complex Type

Import from ABAP Structure

ABAP Structure: BAPI_EPM_SO_HEADER

[redacted] Wizard Step 2 of 3: Import from DDIC Structure

Select Parameter(s)

Data Source Parameter	Assign Structure	Description	Type	Length	Decimals
☑ BAPI_EPM_SO_HEADER	□		BAPI_EPM_SO_HEADER		
☑ SO_ID	□	Sa. Ord. ID	CHAR	10	
☑ CREATED_BY	□	Employee ID	CHAR	10	
☑ CREATED_AT	□	Time Stamp	DEC	21	7
☑ CHANGED_BY	□	Employee ID	CHAR	10	
☑ CHANGED_AT	□	Time Stamp	DEC	21	7
☑ CREATED_BY_BP	□	Value	CHAR	1	
☑ CHANGED_BY_BP	□	Value	CHAR	1	
☑ NOTE	□	Description	CHAR	255	
☑ BUYER_ID	□	Bus. Part. ID	CHAR	10	
☑ BUYER_NAME	□	Company Name	CHAR	80	

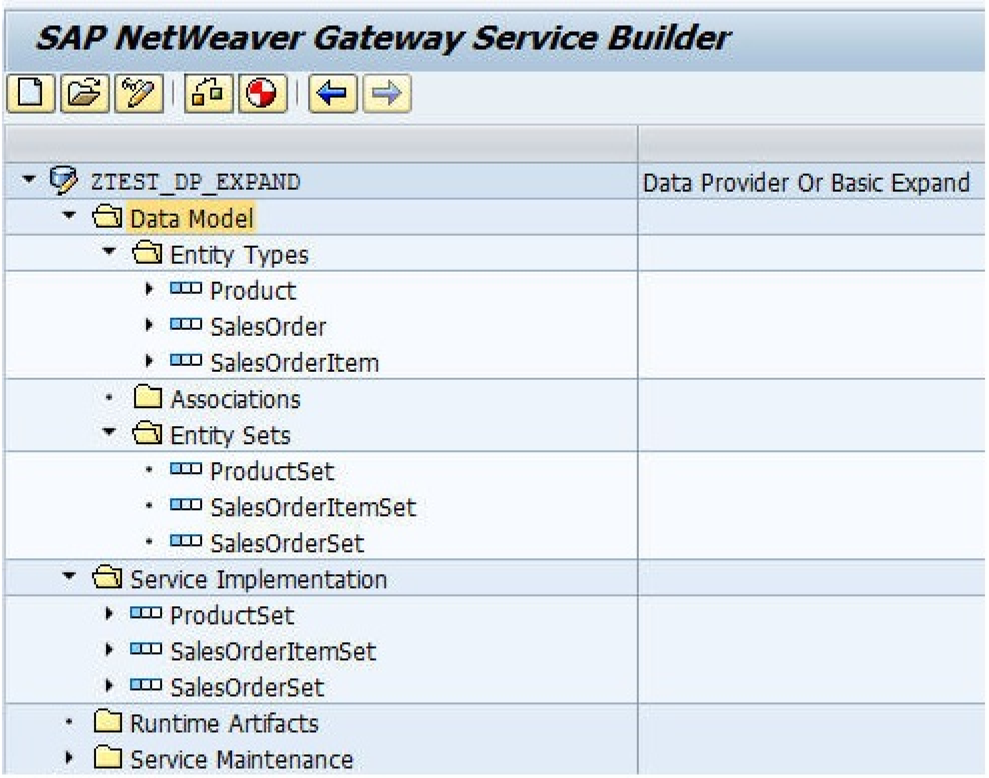
[redacted] Wizard Step 3 of 3: Import from DDIC Structure

Modify Entity Type

IsE...	Complex/Entity Type Name	ABAP Name	Is K...	Type	Name
☑	SalesOrder	SO_ID	☑	CHAR	SoId
☑	SalesOrder	CREATED_BY	□	CHAR	CreatedBy
☑	SalesOrder	CREATED_AT	□	DEC	CreatedAt
☑	SalesOrder	CHANGED_BY	□	CHAR	ChangedBy
☑	SalesOrder	CHANGED_AT	□	DEC	ChangedAt
☑	SalesOrder	CREATED_BY_BP	□	CHAR	CreatedByBp
☑	SalesOrder	CHANGED_BY_BP	□	CHAR	ChangedByBp
☑	SalesOrder	NOTE	□	CHAR	Note
☑	SalesOrder	BUYER_ID	□	CHAR	BuyerId
☑	SalesOrder	BUYER_NAME	□	CHAR	BuyerName
☑	SalesOrder	NET_AMOUNT_EXT	□	DEC	NetAmountExt

☑ Create Default Entity Set

Repeat the process for entities SalesOrderItem and Product. End result will be as displayed below.

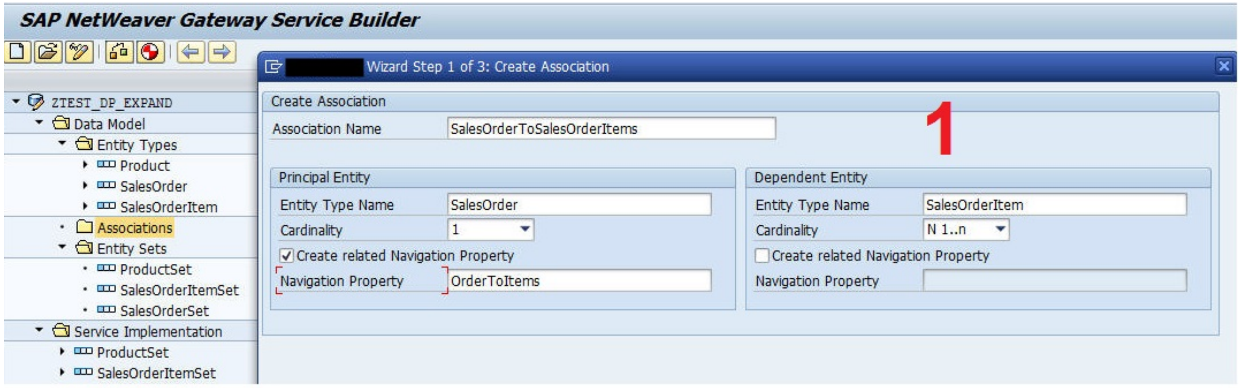


SAP NetWeaver Gateway Service Builder

- ZTEST_DP_EXPAND — Data Provider Or Basic Expand
  - Data Model
    - Entity Types
      - Product
      - SalesOrder
      - SalesOrderItem
    - Associations
    - Entity Sets
      - ProductSet
      - SalesOrderItemSet
      - SalesOrderSet
  - Service Implementation
    - ProductSet
    - SalesOrderItemSet
    - SalesOrderSet
  - Runtime Artifacts
  - Service Maintenance

Now let's create association, navigation etc. By using Create Association wizard, it is just 3 steps as displayed below. This will create Navigation property, Association set etc.

Create association between entities SalesOrder and SalesOrderItem with navigation property as OrderToItems.



[redacted] Wizard Step 1 of 3: Create Association

Create Association

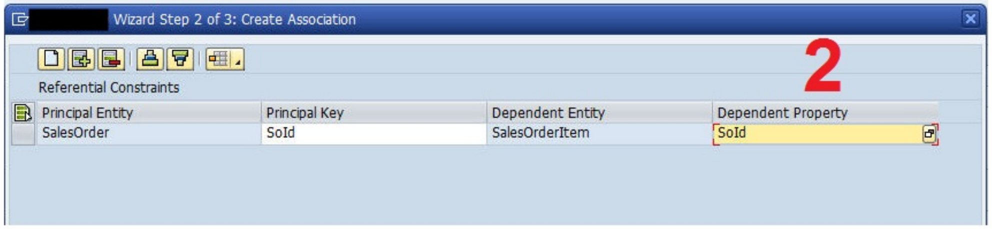
Association Name: SalesOrderToSalesOrderItems

Principal Entity

- Entity Type Name: SalesOrder
- Cardinality: 1
- ☑ Create related Navigation Property
- Navigation Property: OrderToItems

Dependent Entity

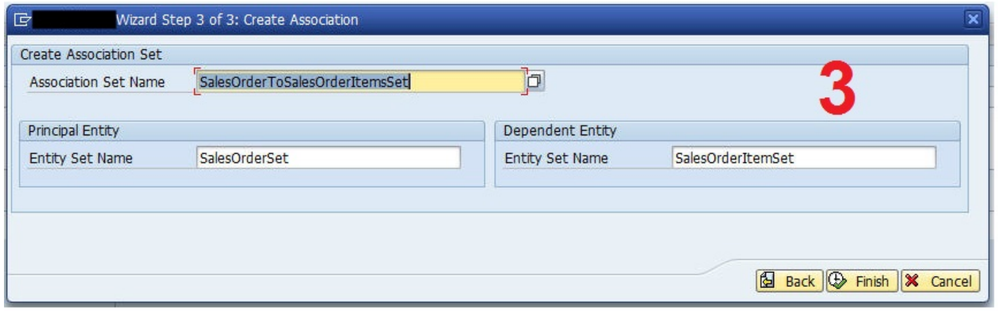
- Entity Type Name: SalesOrderItem
- Cardinality: N 1..n
- ☐ Create related Navigation Property
- Navigation Property:



[redacted] Wizard Step 2 of 3: Create Association

Referential Constraints

Principal Entity	Principal Key	Dependent Entity	Dependent Property
SalesOrder	SoId	SalesOrderItem	SoId



[redacted] Wizard Step 3 of 3: Create Association

Create Association Set

Association Set Name: SalesOrderToSalesOrderItemsSet

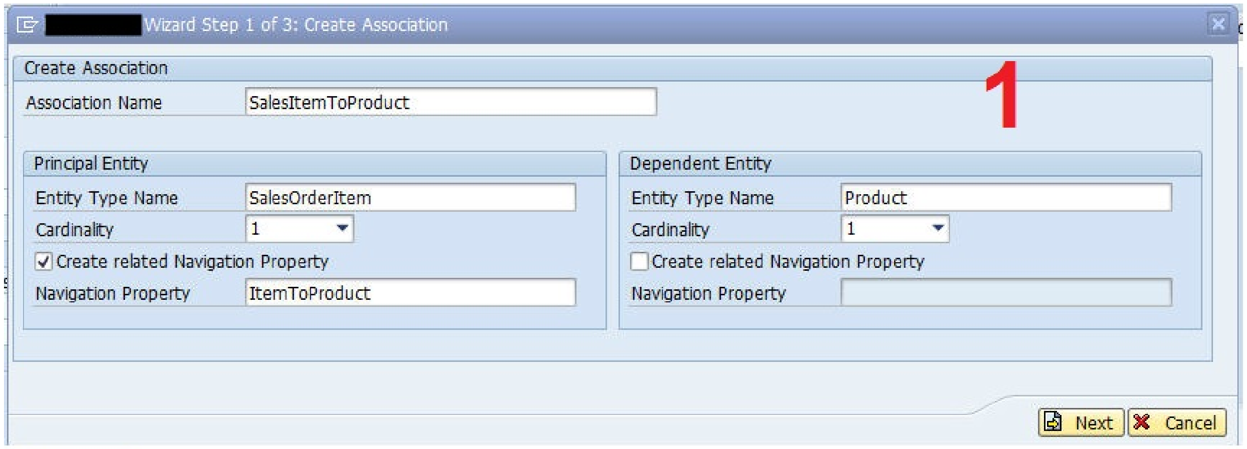
Principal Entity

- Entity Set Name: SalesOrderSet

Dependent Entity

- Entity Set Name: SalesOrderItemSet

On similar lines, create association between entities SalesOrderItem and Product with navigation property as ItemToProduct.

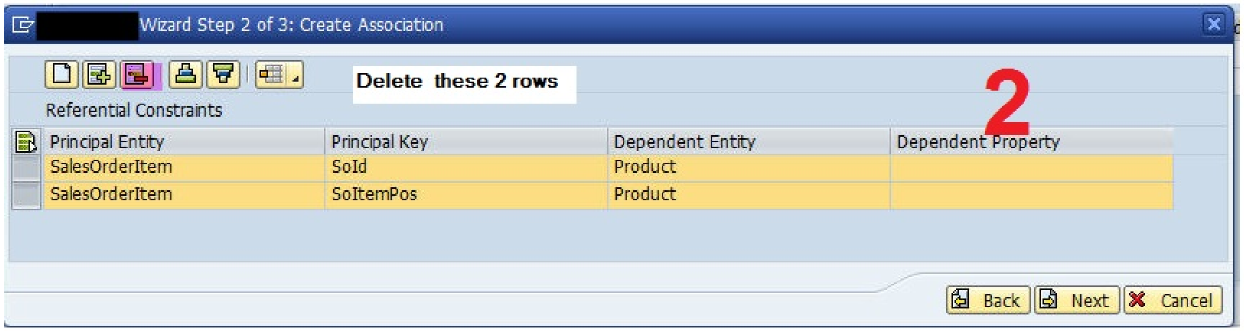

[redacted] Wizard Step 1 of 3: Create Association

1

Create Association

Association Name: SalesItemToProduct

Principal Entity		Dependent Entity	
Entity Type Name	SalesOrderItem	Entity Type Name	Product
Cardinality	1	Cardinality	1
☑ Create related Navigation Property		□ Create related Navigation Property	
Navigation Property	ItemToProduct	Navigation Property	

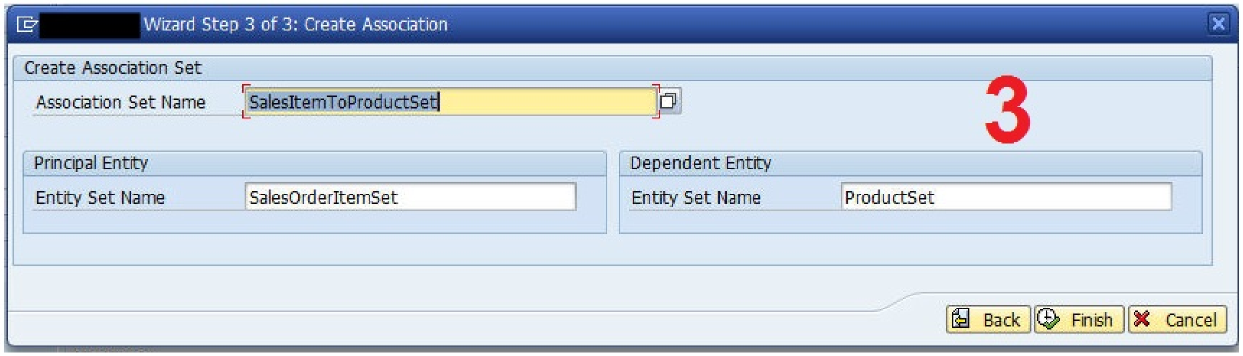

[redacted] Wizard Step 2 of 3: Create Association

Delete these 2 rows

2

Referential Constraints

Principal Entity	Principal Key	Dependent Entity	Dependent Property
SalesOrderItem	SoId	Product	
SalesOrderItem	SoItemPos	Product	


[redacted] Wizard Step 3 of 3: Create Association

3

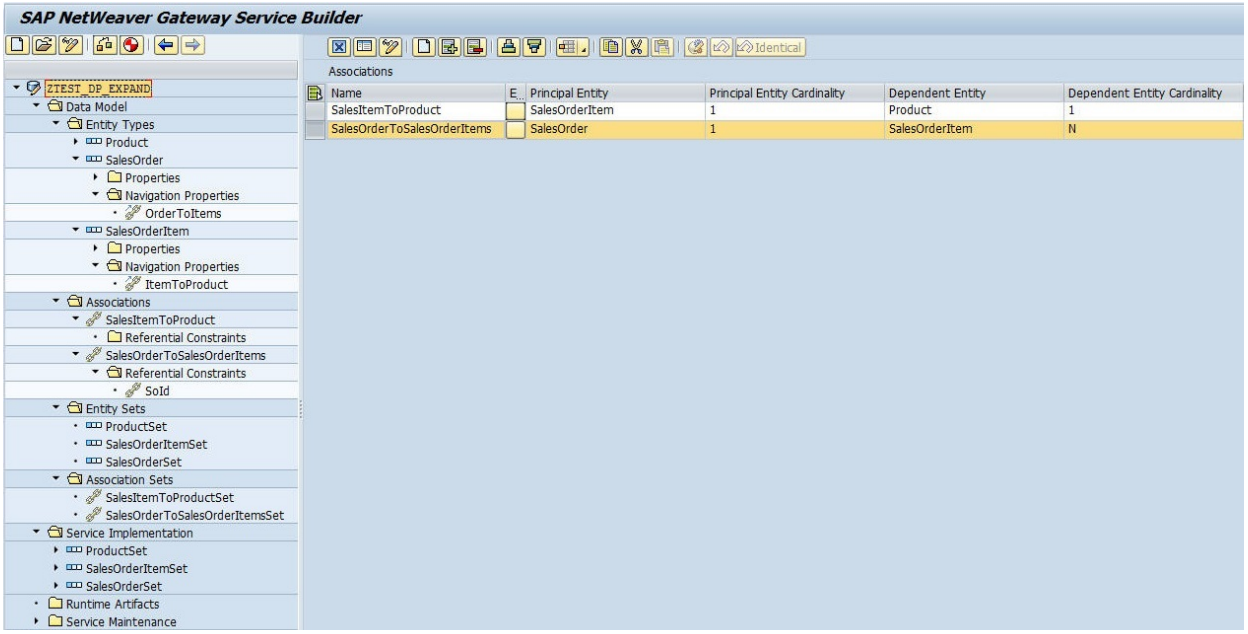
Create Association Set

Association Set Name: SalesItemToProductSet

Principal Entity		Dependent Entity	
Entity Set Name	SalesOrderItemSet	Entity Set Name	ProductSet

Please note that we do not have referential constraints created between SalesOrderItem and Product.

The final OData modeling will look like as below,



Associations

Name	E...	Principal Entity	Principal Entity Cardinality	Dependent Entity	Dependent Entity Cardinality
SalesItemToProduct	□	SalesOrderItem	1	Product	1
SalesOrderToSalesOrderItems	□	SalesOrder	1	SalesOrderItem	N

Coding

In this section, we will redefine methods in DPC_EXT class. Please note that code provided in this blog is in simplest form. You may need to consider proper error handling and other best practices while writing the code.

Association/Navigation

First we will implement logic in method SALESORDERSET_GET_ENTITYSET by redefining it.

We will use below execution URI in Gateway Client (Transaction /IWFND/GW_CLIENT) to check the response payload.

/sap/opu/odata/sap/ZTEST_DP_EXPAND_SRV/SalesOrderSet

```
METHOD salesorderset_get_entityset.

  DATA: lt_salesorder TYPE TABLE OF bapi_epm_so_header,
         ls_salesorder LIKE LINE OF lt_salesorder,
         ls_entity   LIKE LINE OF et_entityset,
         l_max_rows TYPE bapi_epm_max_rows.
  l_max_rows-bapimaxrow = '10'.
  CALL FUNCTION 'BAPI_EPM_SO_GET_LIST'
    EXPORTING
      max_rows    = l_max_rows
    TABLES
      soheaderdata = lt_salesorder.
*Fill ET_ENTITYSET
  LOOP AT lt_salesorder INTO  ls_salesorder .
    MOVE-CORRESPONDING ls_salesorder TO ls_entity.
    APPEND ls_entity TO et_entityset.
  ENDLOOP.
ENDMETHOD.
```

Redefine method SALESORDERSET_GET_ENTITY as below and then execute with below URI

/sap/opu/odata/sap/ZTEST_DP_EXPAND_SRV/SalesOrderSet('500000009')

```
METHOD salesorderset_get_entity.
  DATA: ls_salesorder TYPE bapi_epm_so_header,
        ls_key_tab    TYPE /iwbep/s_mgw_name_value_pair,
        lv_soid       TYPE bapi_epm_so_id.
*Get the key property values
  READ TABLE it_key_tab WITH KEY name = 'SoId' INTO ls_key_tab.
  lv_soid = ls_key_tab-value.
  CALL FUNCTION 'CONVERSION_EXIT_ALPHA_INPUT'
    EXPORTING
      input = lv_soid
    IMPORTING
      output = lv_soid.
  CALL FUNCTION 'BAPI_EPM_SO_GET_DETAIL'
    EXPORTING
      so_id     = lv_soid
    IMPORTING
      headerdata = ls_salesorder.
```

```
*Fill ER_ENTITY

  MOVE-CORRESPONDING ls_salesorder TO er_entity.

ENDMETHOD.
```

Redefine method SALESORDERITEMSE_GET_ENTITYSET as below with URI

/sap/opu/odata/sap/ZTEST_DP_EXPAND_SRV/SalesOrderSet(‘500000008’)/OrderToItems

```
METHOD salesorderitemse_get_entityset.

  DATA: ls_salesorder TYPE bapi_epm_so_header,

        lt_itemdata   TYPE TABLE OF bapi_epm_so_item,

        ls_itemdata   TYPE  bapi_epm_so_item,

        ls_entity     LIKE LINE OF et_entityset.

  DATA:  ls_key_tab  TYPE /iwbep/s_mgw_name_value_pair,

          lv_soid    TYPE bapi_epm_so_id.

*Get the key property values

  READ TABLE it_key_tab WITH KEY name = 'SoId' INTO ls_key_tab.

  lv_soid = ls_key_tab-value.
  CALL FUNCTION 'CONVERSION_EXIT_ALPHA_INPUT'
    EXPORTING
      input = lv_soid
    IMPORTING
      output = lv_soid.
  CALL FUNCTION 'BAPI_EPM_SO_GET_DETAIL'
    EXPORTING
      so_id      = lv_soid
    IMPORTING
      headerdata = ls_salesorder
    TABLES
      itemdata   = lt_itemdata.
  LOOP AT lt_itemdata INTO ls_itemdata.
    MOVE-CORRESPONDING ls_itemdata TO ls_entity  .
    APPEND ls_entity  TO et_entityset.
  ENDLOOP.
ENDMETHOD.
```

*Notice that we used navigation property **OrderToItems** to get the associated entities.*

Framework expand

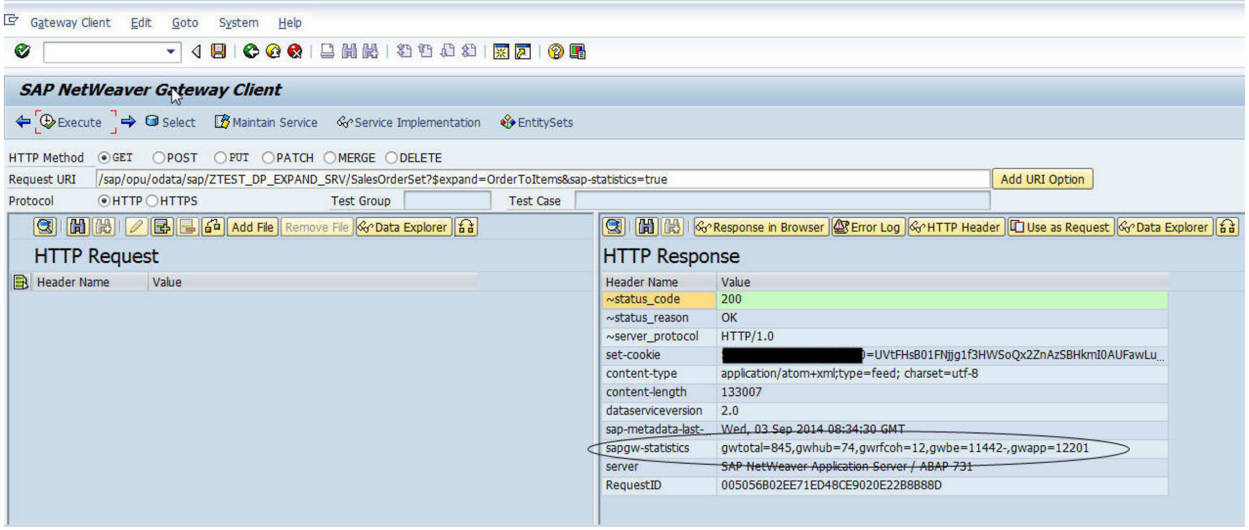
By executing below URI, it will call framework expand by default.

/sap/opu/odata/sap/ZTEST_DP_EXPAND_SRV/SalesOrderSet?$expand=OrderToItems

*This calls method SALESORDERSET_GET_ENTITYSET and SALESORDERITEMSE_GET_ENTITYSET **in loop**.*

Check the values in header name sapgw-statistics for URI /sap/opu/odata/sap/ZTEST_DP_EXPAND_SRV/SalesOrderSet?$expand=OrderToItems&sap-statistics=true

This will give you performance statistics for OData request. For more information, refer Some new features in SAP NW Gateway 2.0 SP08



HTTP Method ● GET ○ POST ○ PUT ○ PATCH ○ MERGE ○ DELETE

Request URI /sap/opu/odata/sap/ZTEST_DP_EXPAND_SRV/SalesOrderSet?$expand=OrderToItems&sap-statistics=true

Protocol ● HTTP ○ HTTPS Test Group Test Case

HTTP Request

Header Name	Value

HTTP Response

Header Name	Value
~status_code	200
~status_reason	OK
~server_protocol	HTTP/1.0
set-cookie	[redacted]=UVtFHsB01FNjjg1f3HWSoQx2ZnAzSBHkmI0AUFawLu...
content-type	application/atom+xml;type=feed; charset=utf-8
content-length	133007
dataserviceversion	2.0
sap-metadata-last-...	Wed, 03 Sep 2014 08:34:30 GMT
sapgw-statistics	gwtotal=845,gwhub=74,gwrfcoh=12,gwbe=11442-,gwapp=12201
server	SAP NetWeaver Application Server / ABAP 731
RequestID	005056B02EE71ED48CE9020E22B8B88D

If we execute URI /sap/opu/odata/sap/ZTEST_DP_EXPAND_SRV/SalesOrderSet('500000009')?$expand=OrderToItems

It will call method SALESORDERSET_GET_ENTITY and SALESORDERITEMSE_GET_ENTITYSET.

For below URI to work

/sap/opu/odata/sap/ZTEST_DP_EXPAND_SRV/SalesOrderSet(‘500000000’)/OrderToItems(SoId=’0500000000’,SoItemPos=’0000000010’)

We need to implement logic with navigation property keys.

Alternatively we can read as /sap/opu/odata/sap/ZTEST_DP_EXPAND_SRV/SalesOrderItemSet(SoId=’0500000000’,SoItemPos=’0000000010’)

Let’s implement SALESORDERITEMSE_GET_ENTITY so that we can read data for above URI as

```
METHOD salesorderitemse_get_entity.

  DATA: ls_salesorder TYPE bapi_epm_so_header.

  DATA:  ls_key_tab   TYPE /iwbep/s_mgw_name_value_pair,

          lv_soid  TYPE bapi_epm_so_id,

          lv_soitempos TYPE snwd_so_item_pos,

          lt_itemdata TYPE TABLE OF bapi_epm_so_item,

         ls_itemdata TYPE  bapi_epm_so_item.

*Get the key property values

  READ TABLE it_key_tab WITH KEY name = 'SoId' INTO ls_key_tab.

  lv_soid = ls_key_tab-value.

  READ TABLE it_key_tab WITH KEY name = 'SoItemPos' INTO ls_key_tab.

  lv_soitempos = ls_key_tab-value.
  CALL FUNCTION 'CONVERSION_EXIT_ALPHA_INPUT'
    EXPORTING
      input = lv_soid
    IMPORTING
      output = lv_soid.
  CALL FUNCTION 'CONVERSION_EXIT_ALPHA_INPUT'
    EXPORTING
      input = lv_soitempos
    IMPORTING
      output = lv_soitempos.
*Get data from BAPI
  CALL FUNCTION 'BAPI_EPM_SO_GET_DETAIL'
    EXPORTING
      so_id     = lv_soid
    IMPORTING
      headerdata = ls_salesorder
    TABLES
```

```
        itemdata  = lt_itemdata.

    READ TABLE  lt_itemdata INTO  ls_itemdata WITH KEY so_id = lv_soid

                                                        so_item_pos = lv_soitempos.

    IF sy-subrc = 0.

      MOVE-CORRESPONDING ls_itemdata TO er_entity.

    ENDIF.

  ENDMETHOD.
```

Use of Navigation Path to read navigation keys

To read the data for below URI, we need to implement logic as below in method PRODUCTSET_GET_ENTITY

Execution URI – /sap/opu/odata/sap/ZTEST_DP_EXPAND_SRV/SalesOrderSet(‘500000001’)/OrderToItems(SoId=’500000001’,SoItemPos=’0000000010’)/ItemToProduct

```
  METHOD productset_get_entity.

    DATA: ls_salesorder TYPE bapi_epm_so_header.

    DATA:   ls_key_tab    TYPE /iwbep/s_mgw_name_value_pair,

            lv_soid   TYPE bapi_epm_so_id,

            lv_soitempos TYPE snwd_so_item_pos,
            lt_itemdata TYPE TABLE OF bapi_epm_so_item,
            ls_itemdata TYPE  bapi_epm_so_item.
    DATA: ls_navigation    TYPE /iwbep/s_mgw_navigation_path,
          lv_property                 TYPE string.
    DATA: lv_product_id TYPE bapi_epm_product_id,
          lv_product_header TYPE bapi_epm_product_header.
    IF iv_source_name = iv_entity_name.
  *Get the key property values
      READ TABLE it_key_tab WITH KEY name = 'ProductId' INTO ls_key_tab.
      IF sy-subrc = 0.
  * MOVE-CORRESPONDING ls_itemdata to er_entity.
        lv_product_id = ls_key_tab-value.
        CALL FUNCTION 'BAPI_EPM_PRODUCT_GET_DETAIL'
          EXPORTING
            product_id = lv_product_id
          IMPORTING
            headerdata = lv_product_header.
        MOVE-CORRESPONDING lv_product_header TO er_entity.
      ENDIF.
    ELSE.
      IF it_navigation_path IS NOT INITIAL.
        READ TABLE it_navigation_path INTO ls_navigation INDEX 1.
        IF sy-subrc EQ 0.
          CASE ls_navigation-nav_prop.
            WHEN 'OrderToItems'.
              LOOP AT ls_navigation-key_tab INTO ls_key_tab.
                CASE ls_key_tab-name.
                  WHEN 'SoId'.
                    lv_soid = ls_key_tab-value.
                  WHEN 'SoItemPos'.
                    lv_soitempos = ls_key_tab-value.
                  WHEN OTHERS.
                ENDCASE.
              ENDLOOP.
          ENDCASE.
        ENDIF.
      ENDIF.
      CALL FUNCTION 'CONVERSION_EXIT_ALPHA_INPUT'
        EXPORTING
          input = lv_soid
        IMPORTING
          output = lv_soid.
      CALL FUNCTION 'CONVERSION_EXIT_ALPHA_INPUT'
        EXPORTING
          input = lv_soitempos
        IMPORTING
          output = lv_soitempos.
  *Get data from BAPI_EPM_SO_GET_DETAIL
      CALL FUNCTION 'BAPI_EPM_SO_GET_DETAIL'
        EXPORTING
          so_id     = lv_soid
        IMPORTING
          headerdata = ls_salesorder
        TABLES
          itemdata  = lt_itemdata.
  *Fill ER_ENTITY
      READ TABLE  lt_itemdata INTO  ls_itemdata WITH KEY so_id = lv_soid
                                                          so_item_pos = lv_soitempos.
      IF sy-subrc = 0.
        lv_product_id-product_id = ls_itemdata-product_id.
        CALL FUNCTION 'BAPI_EPM_PRODUCT_GET_DETAIL'
          EXPORTING
            product_id = lv_product_id
          IMPORTING
            headerdata = lv_product_header.
        MOVE-CORRESPONDING lv_product_header TO er_entity.
      ENDIF.
    ENDIF.
  ENDMETHOD.
```

Also note that we can read product directly using URI /sap/opu/odata/sap/ZTEST_DP_EXPAND_SRV/ProductSet('HT-1030')

Now try with this URI /sap/opu/odata/sap/ZTEST_DP_EXPAND_SRV/SalesOrderSet('500000000')?$expand=OrderToItems/ItemToProduct

Product details will not filled as in the navigation keys are empty because we do not have referential constraint.

Also try with /sap/opu/odata/sap/ZTEST_DP_EXPAND_SRV/SalesOrderItemSet(SoId='0500000000',SoItemPos='0000000020')/ItemToProduct and check why it is not working.

Additionally you can query as

/sap/opu/odata/sap/ZTEST_DP_EXPAND_SRV/SalesOrderSet('500000000')/$links/OrderToItems

/sap/opu/odata/sap/ZTEST_DP_EXPAND_SRV/SalesOrderSet('500000009')/$links/OrderToItems/$count

Data Provider Expand

In this section, we will see how to implement data provider expand by redefining GET_EXPANDED_ENTITYSET and GET_EXPANDED_ENTITY.

Implementing GET_EXPANDED_ENTITYSET

Let's redefine GET_EXPANDED_ENTITYSET. With redefinition (just blank code) execute again the URI /sap/opu/odata/sap/ZTEST_DP_EXPAND_SRV/SalesOrderSet?$expand=OrderToItems

You will not get any response as you need to implement the code yourself.

One of the important point while implementing logic is Data declaration! Based on level till which you want to expand, you need to define your internal table having nested structure or table type.

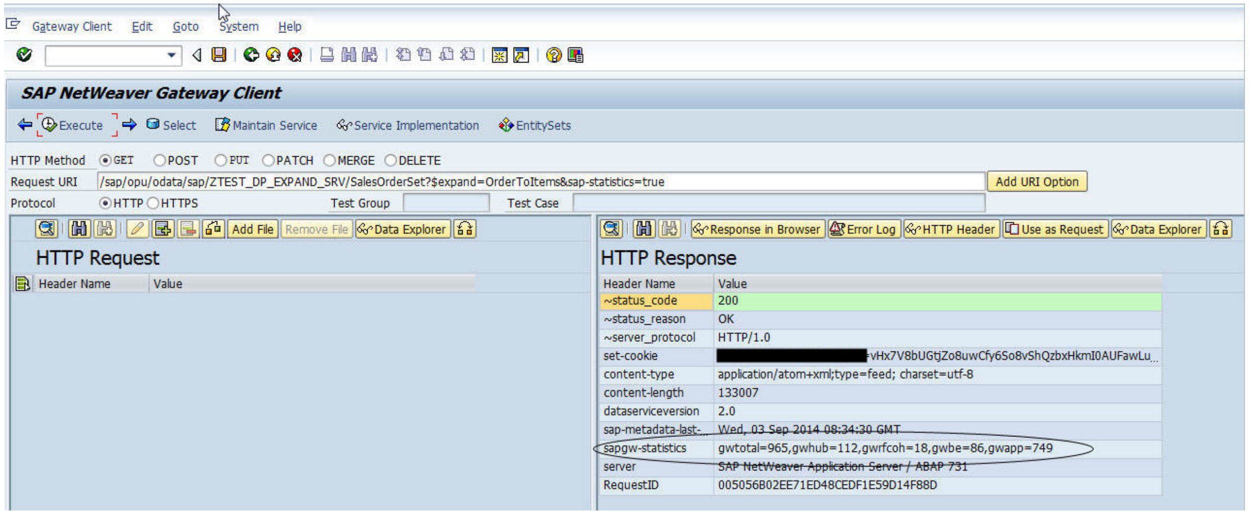
In below code, we want to expand Sales Order and its items. Hence the expand technical clause will be ORDERTOITEMS.

```
METHOD /iwbep/if_mgw_appl_srv_runtime~get_expanded_entityset.

  DATA:  BEGIN OF t_expand_so.

  INCLUDE             TYPE zcl_ztest_dp_expand_mpc_ext=>ts_salesorder.

  DATA: ordertoitems  TYPE zcl_ztest_dp_expand_mpc_ext=>tt_salesorderitem,

       END OF t_expand_so.

  DATA: lt_expand_so   LIKE  TABLE OF t_expand_so,

        ls_expand_so   LIKE t_expand_so,

        ls_item        TYPE zcl_ztest_dp_expand_mpc_ext=>ts_salesorderitem.

  DATA: lt_salesorder  TYPE TABLE OF bapi_epm_so_header,

        ls_salesorder LIKE LINE OF lt_salesorder,

        lt_itemdata   TYPE TABLE OF bapi_epm_so_item,

        ls_itemdata   TYPE  bapi_epm_so_item,

        l_max_rows    TYPE bapi_epm_max_rows.

  CONSTANTS: lc_expand_tech_clause TYPE string VALUE 'ORDERTOITEMS'.

* Read Sales Order and Item data

  l_max_rows-bapimaxrow = '10'.

  CALL FUNCTION 'BAPI_EPM_SO_GET_LIST'
```

```
    EXPORTING
      max_rows    = l_max_rows
    TABLES
      soheaderdata = lt_salesorder
      soitemdata  = lt_itemdata.
* Data processing logic
  LOOP AT lt_salesorder INTO ls_salesorder.
    MOVE-CORRESPONDING ls_salesorder TO ls_expand_so  .
    LOOP AT lt_itemdata INTO ls_itemdata WHERE so_id = ls_salesorder-so_id .
      MOVE-CORRESPONDING ls_itemdata TO ls_item  .
      APPEND ls_item  TO ls_expand_so-ordertoitems.
      CLEAR: ls_item.
    ENDLOOP.
    APPEND ls_expand_so  TO lt_expand_so.
    CLEAR: ls_expand_so.
  ENDLOOP.
* Fill EE_ENTITYSET
```

```
  copy_data_to_ref(
    EXPORTING
      is_data = lt_expand_so
    CHANGING
      cr_data = er_entityset ).
* Insert Navigation property into ET_EXPANDED_TECH_CLAUSES
  INSERT lc_expand_tech_clause INTO TABLE et_expanded_tech_clauses.
ENDMETHOD.
```

Query with URI /sap/opu/odata/sap/ZTEST_DP_EXPAND_SRV/SalesOrderSet?$expand=OrderToItems&sap-statistics=true to check the runtime statistics and compare the values with before implementing data provider expand.



HTTP Method ● GET ○ POST ○ PUT ○ PATCH ○ MERGE ○ DELETE

Request URI /sap/opu/odata/sap/ZTEST_DP_EXPAND_SRV/SalesOrderSet?$expand=OrderToItems&sap-statistics=true

Protocol ● HTTP ○ HTTPS Test Group Test Case

HTTP Request

Header Name	Value

HTTP Response

Header Name	Value
~status_code	200
~status_reason	OK
~server_protocol	HTTP/1.0
set-cookie	[redacted]=vHx7V8bUGtjZo8uwCfy6So8vShQzbxHkmI0AUFawLu...
content-type	application/atom+xml;type=feed; charset=utf-8
content-length	133007
dataserviceversion	2.0
sap-metadata-last-...	Wed, 03 Sep 2014 08:34:30 GMT
sapgw-statistics	gwtotal=965,gwhub=112,gwrfcoh=18,gwbe=86,gwapp=749
server	SAP NetWeaver Application Server / ABAP 731
RequestID	005056B02EE71ED48CEDF1E59D14F88D

Now we will try to expand to one level more i.e. we will expand sales order, its items and product of each item. In this case the expand technical clause will be ORDERTOITEMS/ITEMTOPRODUCT.

Below is the code we need to put to achieve the desired result.

```
METHOD /iwbep/if_mgw_appl_srv_runtime~get_expanded_entityset.

  DATA:  BEGIN OF t_orderitems.

  INCLUDE              TYPE  zcl_ztest_dp_expand_mpc_ext=>ts_salesorderitem.

  DATA: itemtoproduct TYPE  zcl_ztest_dp_expand_mpc_ext=>ts_product,

        END OF t_orderitems.

  DATA:  BEGIN OF t_expand_so.

  INCLUDE             TYPE zcl_ztest_dp_expand_mpc_ext=>ts_salesorder.

  DATA: ordertoitems  LIKE TABLE OF t_orderitems,

       END OF t_expand_so.

  DATA: lt_expand_so   LIKE  TABLE OF t_expand_so,

        ls_expand_so   LIKE t_expand_so,

        ls_item        LIKE t_orderitems.

  DATA: lt_salesorder  TYPE TABLE OF bapi_epm_so_header,

        ls_salesorder LIKE LINE OF lt_salesorder,
        lt_itemdata   TYPE TABLE OF bapi_epm_so_item,
        ls_itemdata   TYPE  bapi_epm_so_item,
        l_max_rows    TYPE bapi_epm_max_rows.
  DATA: lv_product_id TYPE bapi_epm_product_id,
         ls_product_header TYPE bapi_epm_product_header.
  CONSTANTS: lc_expand_tech_clause  TYPE string VALUE 'ORDERTOITEMS/ITEMTOPRODUCT'.
* Read Sales Order and Item data
  l_max_rows-bapimaxrow = '10'.
  CALL FUNCTION 'BAPI_EPM_SO_GET_LIST'
    EXPORTING
      max_rows    = l_max_rows
    TABLES
      soheaderdata = lt_salesorder
      soitemdata  = lt_itemdata.
* Data processing logic
  LOOP AT lt_salesorder INTO ls_salesorder.
    MOVE-CORRESPONDING ls_salesorder TO ls_expand_so  .
    LOOP AT lt_itemdata INTO ls_itemdata WHERE so_id = ls_salesorder-so_id .
```

```
      MOVE-CORRESPONDING ls_itemdata TO ls_item  .

      lv_product_id = ls_itemdata-product_id.

      CALL FUNCTION 'BAPI_EPM_PRODUCT_GET_DETAIL'

        EXPORTING

          product_id = lv_product_id

        IMPORTING

          headerdata = ls_product_header.

      MOVE-CORRESPONDING ls_product_header TO ls_item-itemtoproduct.

      APPEND ls_item  TO ls_expand_so-ordertoitems.

      CLEAR: ls_item.

    ENDLOOP.

    APPEND ls_expand_so  TO lt_expand_so.

    CLEAR: ls_expand_so, lv_product_id.

  ENDLOOP.

* Fill EE_ENTITYSET

  copy_data_to_ref(

    EXPORTING
```

```
      is_data = lt_expand_so
    CHANGING
      cr_data = er_entityset ).
* Insert Navigation property into ET_EXPANDED_TECH_CLAUSES
  INSERT lc_expand_tech_clause INTO TABLE et_expanded_tech_clauses.
ENDMETHOD.
```

Please note that we need to insert complete expand clause in `et_expanded_tech_clauses`

Implementing GET_EXPANDED_ENTITY

Let's implement GET_EXPANDED_ENTITY.

Below is the query to execute and code to for implementation.

/sap/opu/odata/sap/ZTEST_DP_EXPAND_SRV/SalesOrderSet('500000005')?$expand=OrderToItems/ItemToProduct

```
METHOD /iwbep/if_mgw_appl_srv_runtime~get_expanded_entity.
  DATA:  BEGIN OF t_orderitems.
  INCLUDE              TYPE  zcl_ztest_dp_expand_mpc_ext=>ts_salesorderitem.
  DATA: itemtoproduct TYPE  zcl_ztest_dp_expand_mpc_ext=>ts_product,
        END OF t_orderitems.

  DATA:  BEGIN OF t_expand_so.

  INCLUDE             TYPE zcl_ztest_dp_expand_mpc_ext=>ts_salesorder.

  DATA: ordertoitems  LIKE TABLE OF t_orderitems,

       END OF t_expand_so.

  DATA: ls_expand_so   LIKE t_expand_so,

        ls_item LIKE t_orderitems.

  DATA:  ls_salesorder TYPE bapi_epm_so_header,

         lt_itemdata   TYPE TABLE OF bapi_epm_so_item,

         ls_itemdata   TYPE  bapi_epm_so_item.

  DATA:  ls_key_tab   TYPE /iwbep/s_mgw_name_value_pair,

          lv_soid  TYPE bapi_epm_so_id.

  DATA: lv_product_id     TYPE bapi_epm_product_id,

        ls_product_header TYPE bapi_epm_product_header.

  CONSTANTS: lc_expand_tech_clause  TYPE string VALUE 'ORDERTOITEMS/ITEMTOPRODUCT'.

*Get the key property values and Read Sales Order and Item data

  READ TABLE it_key_tab WITH KEY name = 'SoId' INTO ls_key_tab.

  lv_soid = ls_key_tab-value.
  CALL FUNCTION 'CONVERSION_EXIT_ALPHA_INPUT'
    EXPORTING
      input = lv_soid
    IMPORTING
      output = lv_soid.
  CALL FUNCTION 'BAPI_EPM_SO_GET_DETAIL'
    EXPORTING
      so_id     = lv_soid
    IMPORTING
      headerdata = ls_salesorder
    TABLES
      itemdata  = lt_itemdata.
* Data processing logic
  MOVE-CORRESPONDING ls_salesorder TO ls_expand_so  .
  LOOP AT lt_itemdata INTO ls_itemdata WHERE so_id = ls_salesorder-so_id .
    MOVE-CORRESPONDING ls_itemdata TO ls_item  .
    lv_product_id = ls_itemdata-product_id.
    CALL FUNCTION 'BAPI_EPM_PRODUCT_GET_DETAIL'
      EXPORTING
        product_id = lv_product_id
      IMPORTING
        headerdata = ls_product_header.
    MOVE-CORRESPONDING ls_product_header TO ls_item-itemtoproduct.
    APPEND ls_item  TO ls_expand_so-ordertoitems.
  ENDLOOP.
* Fill ER_ENTITY
  copy_data_to_ref(
    EXPORTING
      is_data = ls_expand_so
    CHANGING
      cr_data = er_entity ).
* Insert Navigation property into ET_EXPANDED_TECH_CLAUSES
  INSERT lc_expand_tech_clause INTO TABLE et_expanded_tech_clauses.
ENDMETHOD.
```

Closing Remarks

Important points to be considered while coding for association/navigation and data provider $expand,

- *Use of navigation path and navigation key. Entities can be directly accessed or via navigation property. Code for both scenario using navigation path and navigation keys.*
- *Data declaration of internal tables in case of data provider expand. Understand the relations between entities. while declaring internal table, use navigation property name to address dependent entity structure.* ***It should be same****. Check below data declaration.*

```
DATA:  BEGIN OF t_orderitems.

 INCLUDE              TYPE  zcl_ztest_fw_expand_mpc_ext=>ts_salesorderitem.

 DATA: itemtoproduct TYPE  zcl_ztest_fw_expand_mpc_ext=>ts_product,

       END OF t_orderitems.

 DATA:  BEGIN OF t_expand.

 INCLUDE             TYPE zcl_ztest_fw_expand_mpc_ext=>ts_salesorder.

 DATA: ordertoitems  LIKE TABLE OF t_orderitems,

      END OF t_expand.

 DATA: lt_so    LIKE  TABLE OF t_expand,

       ls_so    LIKE t_expand,

       ls_item LIKE t_orderitems.
```

- *Parent and its immediate child will be separated using "/" for e.g $expand=OrderToItems/ItemToProduct if we would have 2nd sibling at order level for e.g Partners of Sales order then we could have navigation property as OrderToPartners with say its child as PartnerToAddress then we need to access it as OrderToPartners/PartnerToAddress. To get the expanded result for both hierarchies, the expand clause will look as $expand=OrderToItems/ItemToProduct,OrderToPartners/PartnerToAddress (separated by ",")*
- *Navigation property separated with "/" will be inserted into expanded technical clause. 2nd hierarchy will be appended in expanded technical clause*

In short, for example mentioned in point 3, it would be,

```
ls_expanded_clause_items = 'ORDERTOITEMS/ITEMTOPRODUCT'.

ls_expanded_clause_partners = 'ORDERTOPARTNERS/PARTNERTOADDRESS'.

APPEND ls_expanded_clause_items TO et_expanded_tech_clauses.

APPEND ls_expanded_clause_partners TO et_expanded_tech_clauses.
```

also refer this thread Error with $expand and navigation: Resource not found for the segment 'NavAtp'

- *whether framework expand or data provider expand provides better result will depends on your scenario. Please notice that with $expand, we are making single call to read the parent and child data but at the same time ensure that the data to be retrieved is not too large.*

I hope you enjoyed reading this blog and now will be able to play with association/navigation and data provider $expand!

Please feel free if you have any different thoughts to improve any section of this blog.

Happy Learning & Coding 🙂 🙂 🙂

Alert Moderator

Assigned tags

SAP Gateway | association | basic expand | data provider expand | expand |

View more...

Related Blog Posts

Recommended instructions for OData Service Development

By **Former Member** , Jul 21, 2016

Cost Effective SAP NetWeaver Gateway Trial

By **Uladzislau Pralat** , Jan 13, 2016

Expand Parent Child Entities in a Single Expansion – GET_EXPANDED_ENTITYSET

By **Kanwardeep Singh Gill** , Nov 18, 2014

Related Questions

$expand passing query value as '00000000' instead of original value

By **SUBHAJIT DAS** , Jan 11, 2018

OData Performance issue: $expand

By **Gaurav Khare** , Mar 30, 2019

Pass Expanded Entity without Using $EXPAND in URL

By **Christoffer Fuss** , May 31, 2018

34 Comments

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

Former Member

September 24, 2014 at 3:58 pm

Hello Chandra,

Great and very well explained by

emphasizing on the scenarios like *SalesOrderSet('500000005')?$expand=OrderToItems/ItemToProduct* as we would encounter those kind of scenarios as well

Regards,

Ashwin

Like (0)

Chandrashekhar Mahajan | Post author

September 25, 2014 at 5:56 am

Thanks Ashwin Dutt R !

Like (0)

Prabaharan Asokan

September 25, 2014 at 10:28 am

Hi,

Very useful blog.Saved it as a PDF. Thanks.

Prabaharan

Like (0)

Syam Babu

September 25, 2014 at 5:24 pm

Hi Chandra,

Great Work.

and also Covered gateway performance with this Blog

Thanks,

Syam

Like (0)

Chandrashekhar Mahajan | Post author

September 26, 2014 at 6:45 am

Thanks for your comments Syam Babu !

Yes the gateway performance is one of the important point when its comes to framework vs data provider $expand.

Regards,

Chandra

Like (0)

Andre Fischer

September 25, 2014 at 6:56 pm

Hi Chandrashekhar,

very well explained !

I will show the same in the upcoming d-code events in my session DEV360 .

Best Regards,

Andre

Like (0)

Chandrashekhar Mahajan | Post author

September 26, 2014 at 6:53 am

Thanks Andre Fischer ! Your appreciation means a lot to me!

I am very much excited and honored that my blog will be useful for your session DEV360

Regards,

Chandra

Like (0)

Arindam Samanta

September 26, 2014 at 10:00 am

This is great job you have done once again. It is a buatufull staff. Thanks a lot for this. All the very best for future work!

Like (0)

Vijay Vegesana

September 29, 2014 at 5:27 pm

Thanks Chandra, great blog..

Like (0)

Former Member

October 2, 2014 at 1:00 pm

Nice work Chandra!

it would be good if you could expand (no pun intended) on the use case for GET_EXPANDED_ENTITYSET and GET_EXPANDED_ENTITY.

As you have already pointed out, $expand can process an expansion request by looping through the access methods. It would be great to have some clarity of why you would do these methods as an alternative. The SAP Help sort of explains it but it's not easily understood until you have done a bit of expanding practice.

On a best practice note, I recommend using the technical request object to find keys and navigation values. I'm not 100% sure but the technical request seems to have been added to assist in OData compliance.

The 'direct' keys etc. exposed in the method signatures aren't quite the same. For one thing technical element names are always in uppercase even if the model element name is not. It's also easier to pass one technical object reference on to other methods than all of those signature parameters.

Regards

Ron.

P.S. it’s also nice to see that you didn’t generate your service from the BAPI interfaces – phew!

Like (0)

Chandrashekhar Mahajan | Post author

October 7, 2014 at 9:06 am

Hi Ron,

Thanks for your comments!

> *Ron Sargeant wrote:*
>
> *it would be good if you could expand (no pun intended) on the use case for GET_EXPANDED_ENTITYSET and GET_EXPANDED_ENTITY.*
>
> *As you have already pointed out, $expand can process an expansion request by looping through the access methods. It would be great to have some clarity of why you would do these methods as an alternative. The SAP Help sort of explains it but it’s not easily understood until you have done a bit of expanding practice.*

Are you referring to framework expand vs data provider expand? I already explained those points with example. Please let me know which use case you want me to explore more.

> *Ron Sargeant wrote:*
>
> *On a best practice note, I recommend using the technical request object to find keys and navigation values. I'm not 100% sure but the technical request seems to have been added to assist in OData compliance.*
>
> *The 'direct' keys etc. exposed in the method signatures aren't quite the same. For one thing technical element names are always in uppercase even if the model element name is not. It's also easier to pass one technical object reference on to other methods than all of those signature parameters.*

Yes I agree that using technical request object to get the keys and navigation value, filter value etc is best way. and in my earlier blog, Let's code CRUDQ and Function Import operations in OData service! I explained it with alternative way to get these values instead of method signature parameter.

Regards,

Chandra

Like (0)

Former Member

October 7, 2014 at 9:42 am

Hi Chandra,

I agree that you have shown that there are alternatives, but I wouldn’t say they were explained as a result of just being shown.

“Are you referring to framework expand vs data provider expand?”, Yes and no, in my understanding, there is just $expand and how the framework implements the request using it. The consumer can’t ask for a different means of expanding. A properly provided request can expand the feed with no special coding, yet there is an implementation stub for expanding feeds. I think you have shown “if you choose to build an expand method, you can do it like this”, but not explained why.

Regarding use of technical object, it can be confusing for readers to see an author using different coding methods, especially when that difference is not the topic of the article.

In my earlier blogs I used the signature tables because there was no technical context that I recall in earlier versions. Now I always use it and don’t revert to “old style”. I try to keep true to ‘real code’ in examples, even if it takes longer. I use custom code patterns in the editor to make this easier to regulate.

Regards

Ron.

Like (0)

Former Member

December 8, 2014 at 5:32 pm

Hi Chandra,

its a great blog with good details.

I too have same question as Ron and want to understand difference,why and when we need to go for framework expand vs data provider expand.

Regards

Yugandhar Reddy

Like (0)

David Price

October 24, 2018 at 7:40 pm

I apologize for answering an old question, but think I can answer that… in some case reading the entire multi-level result set at one time will be much faster than looping over GET_ENTITYSET, e.g. when you have a large number of salesorders each with a large number of lines which in turn have their own expandable schedule lines… you don't want to go back to the database for every line of the order, you want to suck all those out at once.

That said, the framework has worked for every situation I've run into across several years, and "data provider expand" hasn't been useful to me yet. I'm toying with trying to adapt data provider expand into parallel processing for a problem where the result set takes too long to process in series, but I think the preferred solution in my case is parallel queries from the Web side.

Like (0)

' Pavan ' Golesar

July 22, 2015 at 10:11 am

Hello Chandra,

Very good helpful blog.. Its my problem solver in many cases.. i refer to it many times...It touches down almost all major aspects that I feel I are very helpful for learner like me..

Secondly, As mentioned earlier in this blog by you about the

> *Coding*
>
> *In this section, we will redefine methods in DPC_EXT class. Please note that code provided in this blog is in*
>
> *simplest form.* ***You may need to consider proper error handling and other best practices while writing the code.***

can you please share any document to refer to for the same.

Thanks & Best Regards,

Pavan Golesar

Like (0)

Former Member

November 6, 2014 at 8:57 am

Thanks Chandra, Great Blog!

Like (0)

Prashanth Kumar

December 4, 2014 at 11:17 am

Hi Chandra,

I have one doubt regarding the asociation, take one example i have one root entity and two or more child entities under the root entity.

But I don’t want to get the parent as well as two or more child details under single roof i.e. by using get_expanded_entity. Instead i need to handle the navigation dynamically means if i want parent and first child details or parent and second child details individually, this can be possible by using the related url. In this case my parent has two separate navigations based on navigation provided in the url i need to respond accordingly.

This can be possible if i will get the navigation name in get_expanded_entity method. But i am not getting the Navigation Name inside the method.

Is there any way to get the navigation name inside the get_expanded_entity, so that i can navigate between parent and 1st child or parent and 2nd child.

Please suggest.

Thanks in advance.

Regards,

Prashanth Kumar B.

Like (0)

Erik Hoven

January 21, 2015 at 10:35 am

Great work !!!!

Like (0)

Former Member

February 12, 2015 at 2:28 pm

Hi Chandra,

Great blog .

I am facing issue in my Gateway service . My line Item data is displayed 1st and than Header data . My requirement is to get Headerdata 1st . I am using expand statement for expanding line item.

/sap/opu/odata/sap/Z_GW_ECC_PO_SRV/PoHeaderSet/?$expand=PoLineitmSet

Association , Navigation exist

Am i doing something wrong?

Thanks,

Anju

Like (0)

Hemendra Sabharwal

April 14, 2015 at 10:02 am

Nice blog Chandra, thanks for sharing.

Warm Regards

Hemendra

Like (0)

Former Member

May 11, 2015 at 4:33 pm

Nice piece of info. Thanks for sharing!!!

Regards,

Krishna Chauhan

Like (0)

Former Member

December 11, 2015 at 9:56 am

Hi,

Thank you for sharing.

I have a question, as you said adopting framework or Data provider Expand will depend on the performance. So What I need to make it depending on a specific scenario where both logic did exist. For some I want that the Framework handle the Expand and other I want that the GW uses the Data provider implmentation.

How can I achieve that?

Best Regards

H.JAIDI

Like (0)

Former Member

March 2, 2016 at 2:54 pm

Hi,

I think there is a mistake in the example above. The cardinality from item to product is not 1:1 but N:1.

1:1 would mean each product is related to exactly one order item.

N:1 means certain order items may point to the same product.

Of course in both cases an order item always can point only to one product.

So let's assume you like to select all order items for a given product:

.../ProductSet('12345')/OrderItems

Of course when you don't provide a navigation property for product that links to the order items, then the cardinality is not important. At least so far I noticed no difference then.

Like (0)

Former Member

March 3, 2016 at 11:54 am

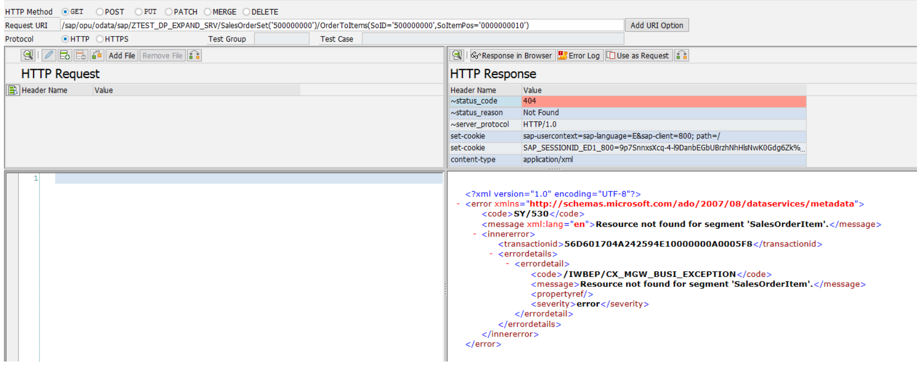
Hi Chandra,

This is a awesome reference material to start learning SAP Gateway. Thanks a lot and Appreciate your efforts here.

One small q I have to you: Please let me know your comments, whenever you get a chance.

When I am trying to Debug Framework Expand with URI: /sap/opu/odata/sap/ZTEST_DP_EXPAND_SRV/SalesOrderSet('500000000')/OrderToItems(SoId='500000000',SoItemPos='0000000010') I am getting error – "Resource not found for segment 'SalesOrderItem'". Please find attached snap shot

> *For below URI to work*
>
> *[/sap/opu/odata/sap/ZTEST_DP_EXPAND_SRV/SalesOrderSet('500000000')/OrderToItems(SoId='0500000000',SoItemPos='0000000010')]*
>
> *We need to implement logic with navigation property keys.*
>
> *Alternatively we can read as/sap/opu/odata/sap/ZTEST_DP_EXPAND_SRV/SalesOrderItemSet(SoId='0500000000',SoItemPos='0000000010')*



HTTP Method ● GET ○ POST ○ PUT ○ PATCH ○ MERGE ○ DELETE

Request URI /sap/opu/odata/sap/ZTEST_DP_EXPAND_SRV/SalesOrderSet('500000000')/OrderToItems(SoID='500000000',SoItemPos='0000000010')

Protocol ● HTTP ○ HTTPS Test Group Test Case

Header Name	Value
~status_code	404
~status_reason	Not Found
~server_protocol	HTTP/1.0
set-cookie	sap-usercontext=sap-language=E&sap-client=800; path=/
set-cookie	SAP_SESSIONID_ED1_800=9p7SnnxsXcq-4-l9DanbEGbUBrzhNhHlsNwK0Gdg6Zk%...
content-type	application/xml

```
  <?xml version="1.0" encoding="UTF-8"?>
- <error xmlns="http://schemas.microsoft.com/ado/2007/08/dataservices/metadata">
    <code>SY/530</code>
    <message xml:lang="en">Resource not found for segment 'SalesOrderItem'.</message>
  - <innererror>
      <transactionid>56D601704A242594E10000000A0005F8</transactionid>
    - <errordetails>
      - <errordetail>
          <code>/IWBEP/CX_MGW_BUSI_EXCEPTION</code>
          <message>Resource not found for segment 'SalesOrderItem'.</message>
          <propertyref/>
          <severity>error</severity>
        </errordetail>
      </errordetails>
    </innererror>
  </error>
```

But when I use alternative URI which is mentioned just below above URI, that works.

Can we not use URI which we are getting error?

Thanks a lot in advance.

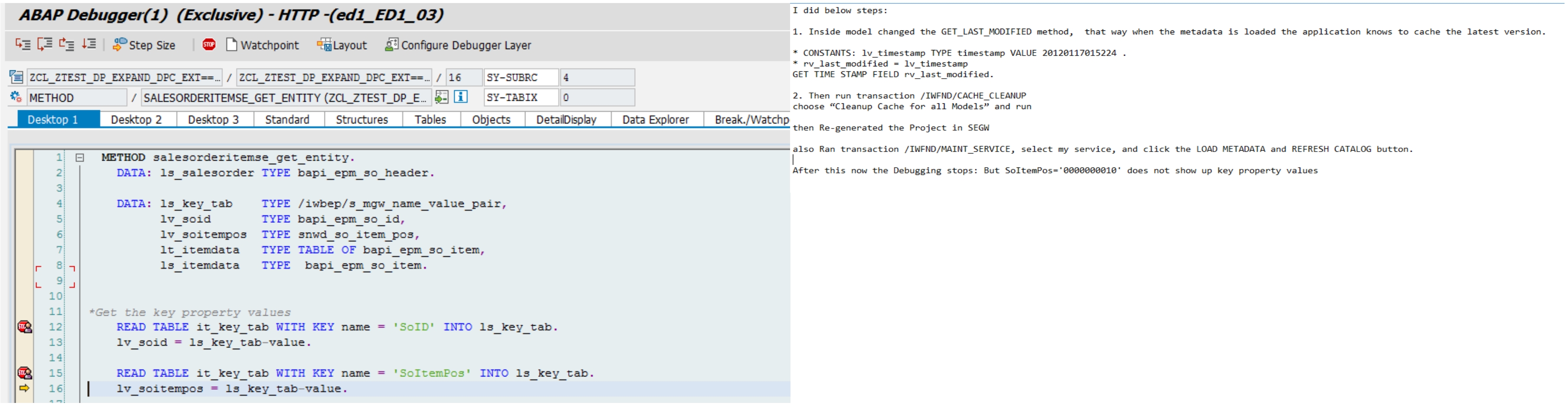
Regards,

Madhu M V

Like (0)

Former Member

March 3, 2016 at 12:31 pm



ZCL_ZTEST_DP_EXPAND_DPC_EXT==... / ZCL_ZTEST_DP_EXPAND_DPC_EXT==... / 16 SY-SUBRC 4
METHOD / SALESORDERITEMSE_GET_ENTITY (ZCL_ZTEST_DP_E... SY-TABIX 0

```
METHOD salesorderitemse_get_entity.
  DATA: ls_salesorder TYPE bapi_epm_so_header.

  DATA: ls_key_tab    TYPE /iwbep/s_mgw_name_value_pair,
        lv_soid       TYPE bapi_epm_so_id,
        lv_soitempos  TYPE snwd_so_item_pos,
        lt_itemdata   TYPE TABLE OF bapi_epm_so_item,
        ls_itemdata   TYPE  bapi_epm_so_item.


*Get the key property values
  READ TABLE it_key_tab WITH KEY name = 'SoID' INTO ls_key_tab.
  lv_soid = ls_key_tab-value.

  READ TABLE it_key_tab WITH KEY name = 'SoItemPos' INTO ls_key_tab.
  lv_soitempos = ls_key_tab-value.
```

I did below steps:

1. Inside model changed the GET_LAST_MODIFIED method, that way when the metadata is loaded the application knows to cache the latest version.

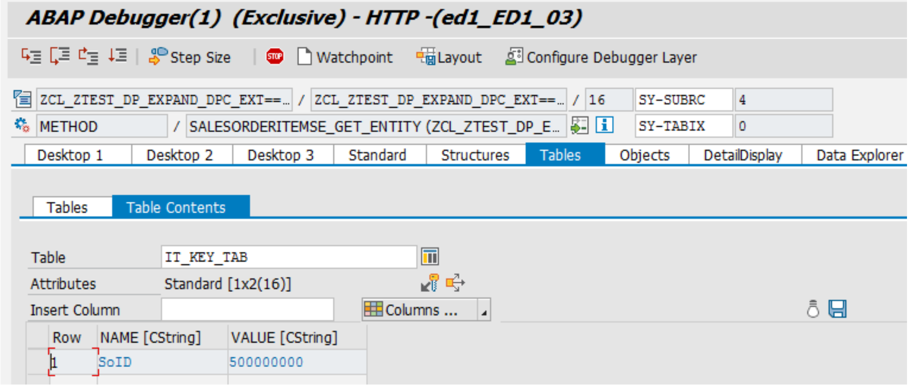
```
* CONSTANTS: lv_timestamp TYPE timestamp VALUE 20120117015224 .
* rv_last_modified = lv_timestamp
GET TIME STAMP FIELD rv_last_modified.
```

2. Then run transaction /IWFND/CACHE_CLEANUP
choose “Cleanup Cache for all Models” and run

then Re-generated the Project in SEGW

also Ran transaction /IWFND/MAINT_SERVICE, select my service, and click the LOAD METADATA and REFRESH CATALOG button.

After this now the Debugging stops: But SoItemPos='0000000010' does not show up key property values



ABAP Debugger(1) (Exclusive) - HTTP -(ed1_ED1_03)

ZCL_ZTEST_DP_EXPAND_DPC_EXT==... / ZCL_ZTEST_DP_EXPAND_DPC_EXT==...	/ 16	SY-SUBRC	4
METHOD / SALESORDERITEMSE_GET_ENTITY (ZCL_ZTEST_DP_E...		SY-TABIX	0

Table IT_KEY_TAB

Attributes Standard [1x2(16)]

Insert Column

Row	NAME [CString]	VALUE [CString]
1	SoID	500000000

Like (0)

Former Member

March 3, 2016 at 1:08 pm

Now I got it – I need to use IT_NAVIGATION_PATH to get both SoID and SoItemPos..

Like (0)

Former Member

August 2, 2016 at 3:26 pm

Hi Chandra,

Very nice and useful blog. I have bookmarked this link for my future reference.

I have a question around expand entity. The expand entity works for Get, however when I do a POST, it is not working. How do we create the underlying association entities ?

How to create association entities during POST. The URI with expand against the parent entity is not allowing me to create the expand entity record. Can you please advise?

Thanks,

Chakram Govindarajan

Like (0)

Former Member

September 17, 2016 at 7:14 pm

Hi ChandraSekar Mahajan,

Good article. I have a doubt. for entity SalesOrder DDIC structure bapi_epm_so_header is used. What are the DDIC structures used for Product and SalesOrder.

Thanks

Ram

Like (0)

Srawan Kumar Dubbakka

March 15, 2017 at 11:42 am

The DDIC structures for Product: BPM_EPM_PRODUCT_HEADER and for

SalesOrderItem: BAPI_EPM_SO_ITEM

SalesOrder: BAPI_EPM_SO_HEADER

Like (0)

Former Member

November 7, 2016 at 3:10 pm

Hello ,

Can you help me with my question : https://answers.sap.com/questions/52301/how-can-i-consume-service-odata.html

I follow their tutorials that have helped me to develop many of the applications I've ever done

Thanks very much
Best Regards
Cristina Rodrigues

Like (0)

Former Member

November 7, 2016 at 3:11 pm

Hello ,

Can you help me with my question : https://answers.sap.com/questions/52301/how-can-i-consume-service-odata.html

I follow their tutorials that have helped me to develop many of the applications I've ever done.

Best Regards
Cristina Rodrigues

Like (0)

Klaus Reimann

January 24, 2017 at 2:32 pm

Hello ***Chandrashekhar Mahajan***, hello dear community,

first i would like to thank ***Chandrashekhar Mahajan*** for the very useful blogpost!

My question: What about associations for n:m cardinalities like forinstance CRHD-OBJID -> HRP1001-OBJID and HRP1001-OTYPE = ‘A’ -> HRP1001-SOBID and HRP1001-SCLAS = ‘P’ -> PA0002-PERNR ?

What is best practice in this kind of case?

An Entity HRP1001 and 2 associations? Or is there another way?

As the EDM doesn’t support list- or table-types.

Best regards, Klaus

Like (0)

Former Member

April 21, 2017 at 11:59 am

Thank You so much Chandrashekhar Mahajan

Like (0)

Former Member

December 21, 2017 at 9:52 am

Hello sir,

I have been following your blogs for a good number of months, they have been very helpful to me and i have been practising SAP UI5 Odata. I am trying to create Odata for three entities using association and navigation property which can be explained as A to B & A to C. I have implemented get_entity and get_entitySet methods for all the entities. I m stucked in get_expanded_entity and get_expanded_entitySet methods. please help.

Like (0)