Technical Specification Form

PROJECT: SAP TRANSFER POSTING

SAP Technical Specification Form

|  |  |  |  |
| --- | --- | --- | --- |
| User Story ID # | 9000RUPESH | Module | PP |
| Date | 17-07-2025 | SL User |  |
| Consultant |  | Developer | Debidutta swain |
| Priority | ☐Urgent ☐High ☒Medium ☐Low | | |
| Application Type | ☐Report ☐Form ☐Interface ☒ODATA ☐Others | | |
| Execute Model | ☐Batch ☐ Foreground ☐Manual ☐Quarterly ☐Monthly  ☐Weekly ☐ Daily ☒Real time | | |
| Output Type | ☐Message ☐Dialog Screen ☐ABAP List ☒List View ☐ALV Grid | | |
| Remarks | SAP TRANSFER POSTING | | |

**Technical specification content**

1. Business Requirement
2. Development Tools
3. Development Objects

* Function Module
* DDIC Structure
* SEGW – OData Project.

1. Business API / OData API with Payload
2. Positive and Negative Testing Screenshot
3. **Business Requirement:**

Develop an OData POST API to perform Transfer posting from one storage location to another storage location from Outside SAP Environment .

**FS Details**

**Run the BAPI: BAPI\_GOODSMVT\_CREATE**

**GOODSMVT\_HEADER = PSTNG\_DATE = “System date”**

**DOC\_DATE= “System Date”**

**GOODSMVT\_CODE = “04”**

**Tables:**

**GOODSMVT\_ITEM:**

**MATERIAL = "MATNR "**

**PLANT = "WERKS"**

**STGE\_LOC = " Fetch Issue storage location"**

**BATCH = " CHARG "**

**MOVE\_TYPE = "Constant= 311"**

**ENTRY\_QNT = " Fetch Transfer stock QTY "**

**ENTRY\_UOM = " Fetch Transfer stock UOM "**

**MOVE\_MAT = " MATNR "**

**MOVE\_PLANT = "WERKS"**

**MOVE\_STLOC = " Fetch Receiving location"**

**IN CASE OF REVERSAL**

**STGE\_LOC = " Fetch Receiving location"**

**MOVE\_STLOC = " Fetch Issue storage location "**

**2.Development Tools :**

For development these are the following tool used :

* **OData Project**
* Created OData project with SEGW Tcode. Here I have used IMPORT from **DDIC structure** for Data modeling and Redefine the Create Entities Method to Achieve our Business Requirement .
* For Service Registration used Tcode : /O/IWFND/MAINT\_SERVICE.
* To Check the final data at Gateway Client used Tcode :/O/IWFND/GW\_CLIENT.

**3.Development Objects :**

* **Function Module:**

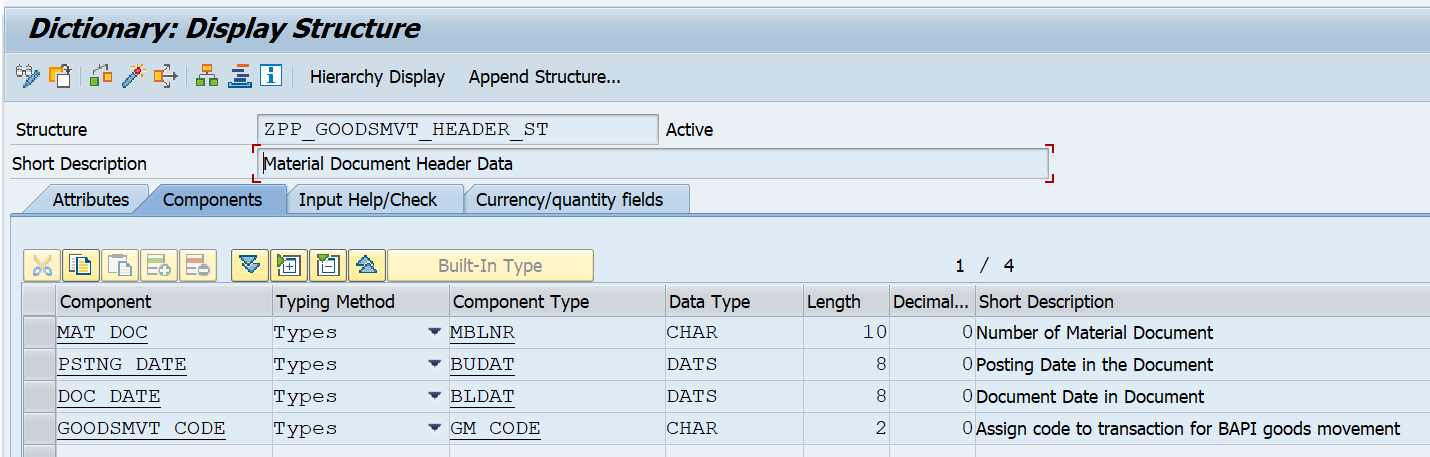
BAPI Name: BAPI\_GOODSMVT\_CREATE

Description : Post Goods Movements with MB\_CREATE\_GOODS\_MOVEMENT

* **DDIC Structure:**

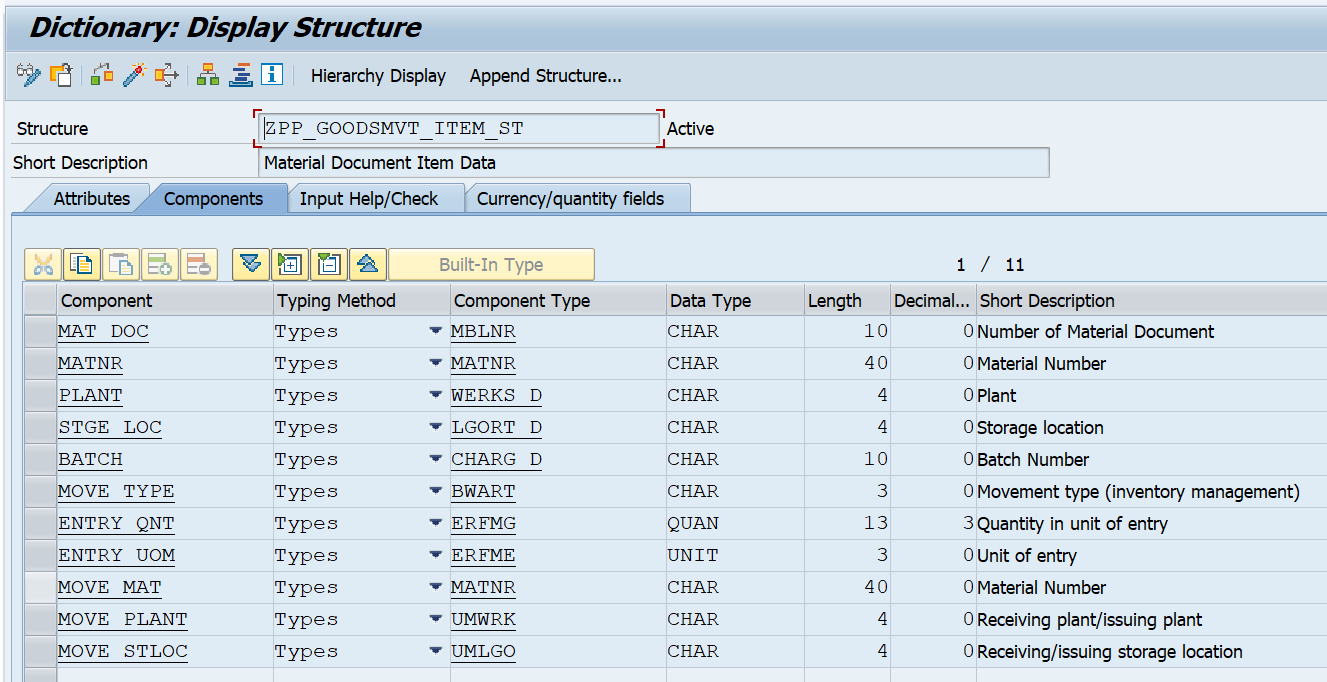
**Header**

Structure Name: ZPP\_GOODSMVT\_HEADER\_ST  
Description: Structure: Material Document Header Data



**Item**

Structure Name: ZPP\_GOODSMVT\_ITEM\_ST  
Description: Structure: Material Document Item Data



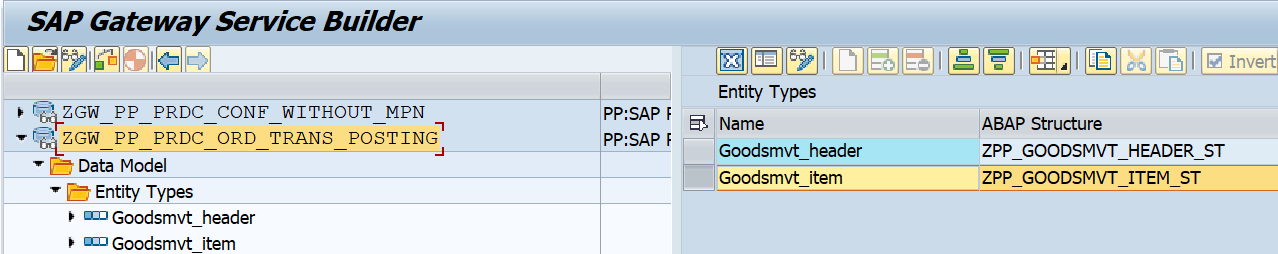
* **SEGW – OData Project:**

**Project Name:** ZGW\_PP\_PRDC\_ORD\_TRANS\_POSTING

**Service Name:** ZGW\_PP\_PRDC\_ORD\_TRANS\_POSTING\_SRV

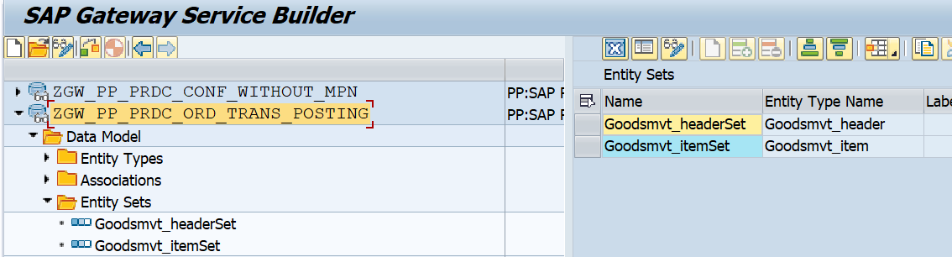
**Entities Type:**

* Goodsmvt\_header
* Goodsmvt\_item



**Entity sets:**

* Goodsmvt\_headerSet
* Goodsmvt\_itemSet



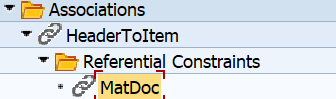
**Association:**

Basically Association is done in ODATA to create the relationship in between header and item Entity .

For more idea about how to create association please [click here](https://saplearners.com/association-and-navigation-in-sap-netweaver-gateway/) and check the step from 1 to 5.

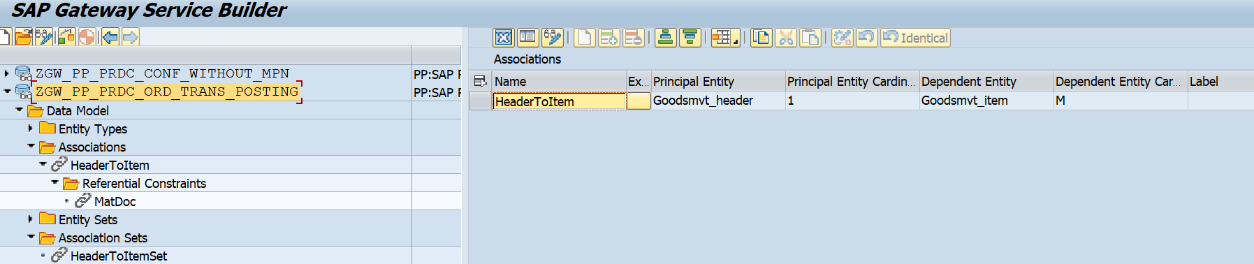
Association Name: HeaderToItem

Referential Constraints : MatDoc

Referential Constraints means based on Which Relation is Established in between Header and Item Entity Set.

**Association set:**

Name: HeaderToItemSet

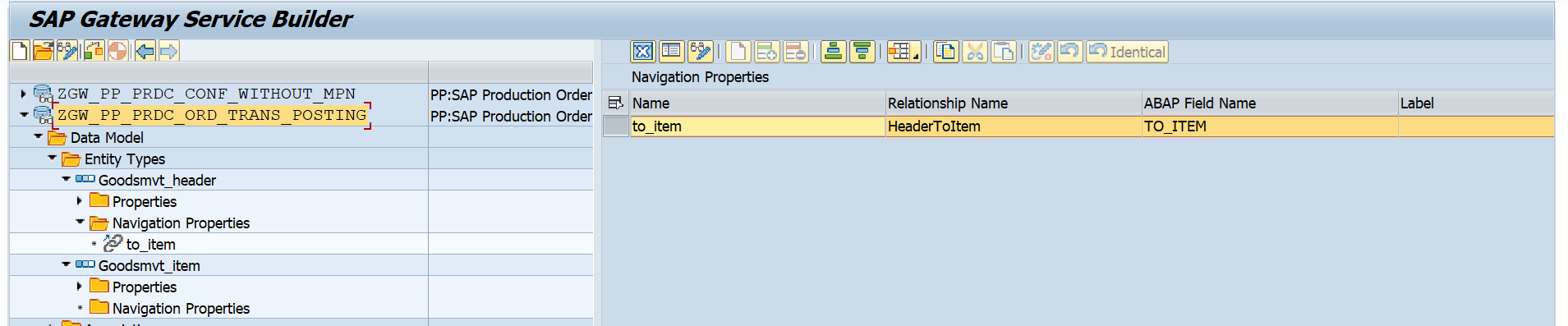


**Navigation Properties:**

When we do Parent child relation in SEGW Odata project, one Navigation property also got created at the time of Create Association and this navigation property will help us to:

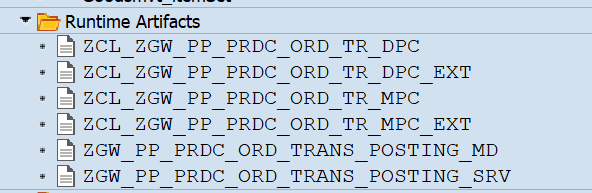
* navigate from Header details to Item details at the time of GET operation
* Carries item details with header details at the time of POST operation.

Also, it is needed at the time of deep entity modeling at MPC\_EXT Class, otherwise it will not Carrey the item detail from outside sap system. Always navigation Property comes under Header Entity after successfully creation of Association.



**Runtime Artifacts**

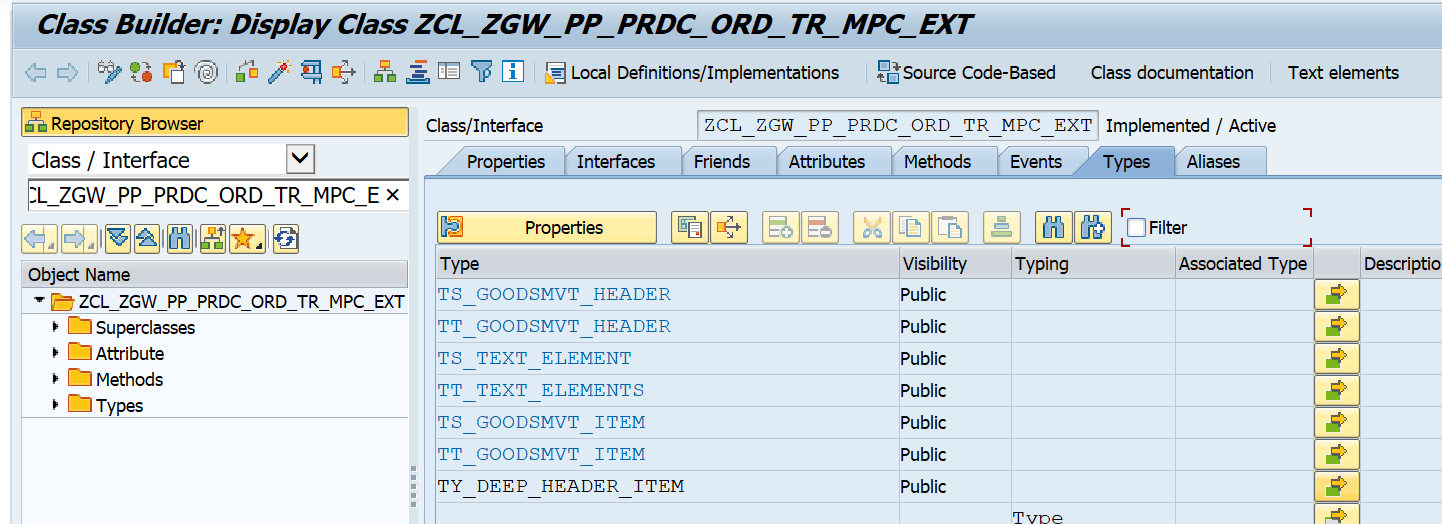
Here 6 Runtime Artifices to created after Generating the Project.



Here the operation can be handled through “CREATE\_DEEP\_ENTITY”. so, for that I had followed the following steps:

**step: 1**

* Go to Runtime Artifacts node, open the ZCL\_ZGW\_PP\_PRDC\_ORD\_TR\_MPC\_EXT class in **ABAP Workbench** & click on the **Types** tab.
* Click on Change button for edit.
* Click on the **Direct Type Entity** Button. Then create deep entity & activate.



**Business Logic for Type declaration in MPC\_EXT.**

class ZCL\_ZGW\_PP\_PRDC\_ORD\_TR\_MPC\_EXT definition  
  public  
  inheriting from ZCL\_ZGW\_PP\_PRDC\_ORD\_TR\_MPC  
  create public .  
  
public section.  
  
  types:  
   BEGIN OF TY\_DEEP\_HEADER\_ITEM ,  
    MAT\_DOC       Type  MBLNR,  
    PSTNG\_DATE    Type  BUDAT,  
    DOC\_DATE      Type  BLDAT,  
    GOODSMVT\_CODE Type  GM\_CODE,  
*\*   Navigation property name should be used otherwise empty records will be shown In all API operations*  
    to\_item       TYPE  TABLE OF TS\_GOODSMVT\_ITEM WITH DEFAULT KEY,  
   END OF TY\_DEEP\_HEADER\_ITEM .  
  
  methods DEFINE  
    redefinition .

**Step: 2**

**Redefine** the **DEFINE** method of the ZCL\_ZGW\_PP\_PRDC\_ORD\_TR\_MPC\_EXT and Enter the below code.

**Business Logic Under Define Method.**

method DEFINE.  
    SUPER->DEFINE( ).  
    DATA:lo\_entity\_type  TYPE REF TO /iwbep/if\_mgw\_odata\_entity\_typ.  
  
    LO\_ENTITY\_TYPE = MODEL->GET\_ENTITY\_TYPE( IV\_ENTITY\_NAME = 'Goodsmvt\_header' ).  
    LO\_ENTITY\_TYPE->BIND\_STRUCTURE(  
      exporting  
        IV\_STRUCTURE\_NAME   = 'ZCL\_ZGW\_PP\_PRDC\_ORD\_TR\_MPC\_EXT=>TY\_DEEP\_HEADER\_ITEM' ).  
  endmethod.

**Step : 3**

Go to the **ZCL\_ZGW\_PP\_PRDC\_ORD\_TR\_DPC\_EXT** class & redefine the method:  **CREATE\_DEEP\_ENTIT**y.

**Business Logic CREATE\_DEEP\_ENTITY method.**

method /IWBEP/IF\_MGW\_APPL\_SRV\_RUNTIME~CREATE\_DEEP\_ENTITY.  
  
IF IV\_ENTITY\_SET\_NAME = 'Goodsmvt\_headerSet'.  
*"--> Constant Declaration.*  
CONSTANTS: LC\_GM\_CODE   TYPE GM\_CODE VALUE '04',  
           LC\_MOVE\_TYPE TYPE BWART   VALUE '311'.  
  
*"--> Data Object Declaration.*  
DATA: LS\_DEEP\_ENTITY TYPE ZCL\_ZGW\_PP\_PRDC\_ORD\_TR\_MPC\_EXT=>TY\_DEEP\_HEADER\_ITEM,  
      ls\_header      TYPE ZPP\_GOODSMVT\_HEADER\_ST,  
      lt\_item        TYPE TABLE OF ZCL\_ZGW\_PP\_PRDC\_ORD\_TR\_MPC\_EXT=>TS\_GOODSMVT\_ITEM.  
  
DATA:LS\_GOODSMVT\_HEADER  TYPE BAPI2017\_GM\_HEAD\_01,  
     LS\_GOODSMVT\_CODE    TYPE BAPI2017\_GM\_CODE,  
     LS\_GOODSMVT\_ITEM    TYPE BAPI2017\_GM\_ITEM\_CREATE,  
     LT\_GOODSMVT\_ITEM    TYPE TABLE OF BAPI2017\_GM\_ITEM\_CREATE,  
     LS\_RETURN           TYPE BAPIRET2,  
     LT\_RETURN           TYPE TABLE OF BAPIRET2,  
     LS\_GOODSMVT\_HEADRET TYPE BAPI2017\_GM\_HEAD\_RET,  
     LV\_MATERIALDOCUMENT TYPE BAPI2017\_GM\_HEAD\_RET-MAT\_DOC,  
     LV\_MATDOCUMENTYEAR  TYPE BAPI2017\_GM\_HEAD\_RET-DOC\_YEAR.  
  
DATA: lo\_message         TYPE REF TO /iwbep/if\_message\_container,  
      LV\_MESSAGE TYPE BAPI\_MSG.  
  
*"--> Get the Requested Payload Details.*  
 CLEAR:LS\_GOODSMVT\_HEADER,LS\_GOODSMVT\_CODE,LS\_GOODSMVT\_ITEM,LS\_DEEP\_ENTITY,LS\_HEADER,LS\_RETURN,  
       LS\_GOODSMVT\_HEADRET,LV\_MATERIALDOCUMENT,LV\_MATDOCUMENTYEAR,LV\_MESSAGE,lo\_message.  
 REFRESH:LT\_GOODSMVT\_ITEM,LT\_ITEM,LT\_RETURN.  
  
 *"--> Prepare the Response message object.*  
  LO\_MESSAGE = MO\_CONTEXT->GET\_MESSAGE\_CONTAINER( ).  
  
  TRY.  
   IO\_DATA\_PROVIDER->READ\_ENTRY\_DATA(  
    importing  
     ES\_DATA = LS\_DEEP\_ENTITY  
     ).  
    catch /IWBEP/CX\_MGW\_TECH\_EXCEPTION. *" Technical Exception*  
 ENDTRY.  
  
 IF LS\_DEEP\_ENTITY is NOT INITIAL.  
  LS\_HEADER = CORRESPONDING #( LS\_DEEP\_ENTITY ).  
  lt\_item   = LS\_DEEP\_ENTITY-TO\_ITEM.  
 ENDIF.  
  
 *"--> Preparing data for BAPI .*  
 IF LS\_HEADER IS NOT INITIAL.  
   LS\_GOODSMVT\_HEADER-PSTNG\_DATE = LS\_HEADER-PSTNG\_DATE.  
   LS\_GOODSMVT\_HEADER-DOC\_DATE   = LS\_HEADER-DOC\_DATE.  
   LS\_GOODSMVT\_CODE-GM\_CODE      = LC\_GM\_CODE.  
  
   IF LT\_ITEM IS NOT INITIAL.  
     LOOP AT LT\_ITEM INTO DATA(LS\_ITEM).  
  
       LS\_GOODSMVT\_ITEM = CORRESPONDING #( LS\_ITEM ).  
       LS\_GOODSMVT\_ITEM-MOVE\_TYPE = LC\_MOVE\_TYPE.  
       LS\_GOODSMVT\_ITEM-MATERIAL  = LS\_ITEM-MATNR.  
  
       APPEND LS\_GOODSMVT\_ITEM TO LT\_GOODSMVT\_ITEM.  
       CLEAR: LS\_GOODSMVT\_ITEM,LS\_ITEM.  
     ENDLOOP.  
   ENDIF.  
 ENDIF.  
  
 *"-->Passing Data to BAPI for create record in sap System.*  
  CALL FUNCTION 'BAPI\_GOODSMVT\_CREATE'  
    EXPORTING  
      GOODSMVT\_HEADER              = LS\_GOODSMVT\_HEADER  
      GOODSMVT\_CODE                = LS\_GOODSMVT\_CODE  
    IMPORTING  
     GOODSMVT\_HEADRET              = LS\_GOODSMVT\_HEADRET  
     MATERIALDOCUMENT              = LV\_MATERIALDOCUMENT  
     MATDOCUMENTYEAR               = LV\_MATDOCUMENTYEAR  
    TABLES  
      GOODSMVT\_ITEM                = LT\_GOODSMVT\_ITEM  
      RETURN                       = LT\_RETURN  
       .  
  
  IF LS\_GOODSMVT\_HEADRET IS NOT INITIAL.  
     CALL FUNCTION 'BAPI\_TRANSACTION\_COMMIT'  
      EXPORTING  
        WAIT          = ABAP\_TRUE  
      IMPORTING  
        RETURN        = LS\_RETURN  
        .  
    IF LS\_RETURN IS INITIAL.  
      *"--> Creted Material document number obly added in the Header Entity.*  
      LS\_DEEP\_ENTITY-MAT\_DOC = LV\_MATERIALDOCUMENT.  
      LOOP AT  LS\_DEEP\_ENTITY-TO\_ITEM ASSIGNING FIELD-SYMBOL(<FS\_DATA>).  
        <FS\_DATA>-MAT\_DOC = LV\_MATERIALDOCUMENT.  
      ENDLOOP.  
  
      COPY\_DATA\_TO\_REF(  
        exporting  
          IS\_DATA = LS\_DEEP\_ENTITY  
        changing  
          CR\_DATA = ER\_DEEP\_ENTITY  
      ).  
  
     LV\_MESSAGE = |Material Document '{ LV\_MATERIALDOCUMENT }' Cretaed Successfully.|.  
     LO\_MESSAGE->ADD\_MESSAGE\_TEXT\_ONLY(  
           exporting  
             IV\_MSG\_TYPE               =   /iwbep/cl\_cos\_logger=>SUCCESS         *" Message Type*  
             IV\_MSG\_TEXT               =   LV\_MESSAGE                            *" MESSAGE TEXT*  
             IV\_ADD\_TO\_RESPONSE\_HEADER =   ABAP\_TRUE                             *" Add the message to the header*  
         ).  
    ENDIF.  
  else.  
    IF LT\_RETURN IS NOT INITIAL.  
      lo\_message->add\_messages\_from\_bapi(  
          EXPORTING  
           it\_bapi\_messages          = LT\_RETURN  
           ).  
         RAISE EXCEPTION TYPE /iwbep/cx\_mgw\_busi\_exception  
         EXPORTING  
          textid            = /iwbep/cx\_mgw\_busi\_exception=>BUSINESS\_ERROR  
          message\_container = lo\_message.  
    ENDIF.  
  ENDIF.  
ENDIF.  
endmethod.

4.Business API / OData API with Payload

API: /sap/opu/odata/sap/ZGW\_PP\_PRDC\_ORD\_TRANS\_POSTING\_SRV/Goodsmvt\_headerSet

**Payload:**

{

"MatDoc": "",

"PstngDate": "\/Date(1752537600000)\/",

"DocDate": "\/Date(1752537600000)\/",

"GoodsmvtCode": "04",

"to\_item":{

"results" : [

{

"MatDoc": "",

"Matnr": "0001-0002SW-A01Y",

"Plant": "1001",

"StgeLoc": "A001",

"Batch": "0013671547",

"MoveType": "311",

"EntryQnt": "1",

"EntryUom": "NO",

"MoveMat": "0001-0002SW-A01Y",

"MovePlant": "1001",

"MoveStloc": "M011"

},

{

"MatDoc": "",

"Matnr": "0001-0002SW-A022Y",

"Plant": "1001",

"StgeLoc": "A001",

"Batch": "0013671546",

"MoveType": "311",

"EntryQnt": "1",

"EntryUom": "PAC",

"MoveMat": "0001-0002SW-A022Y",

"MovePlant": "1001",

"MoveStloc": "M011"

}

]

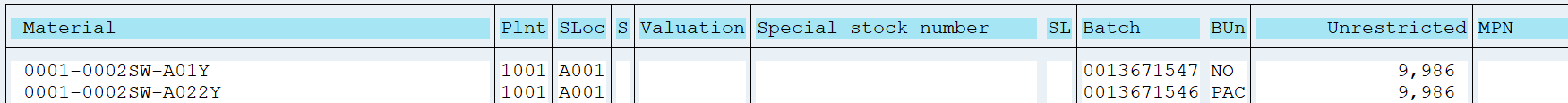
}

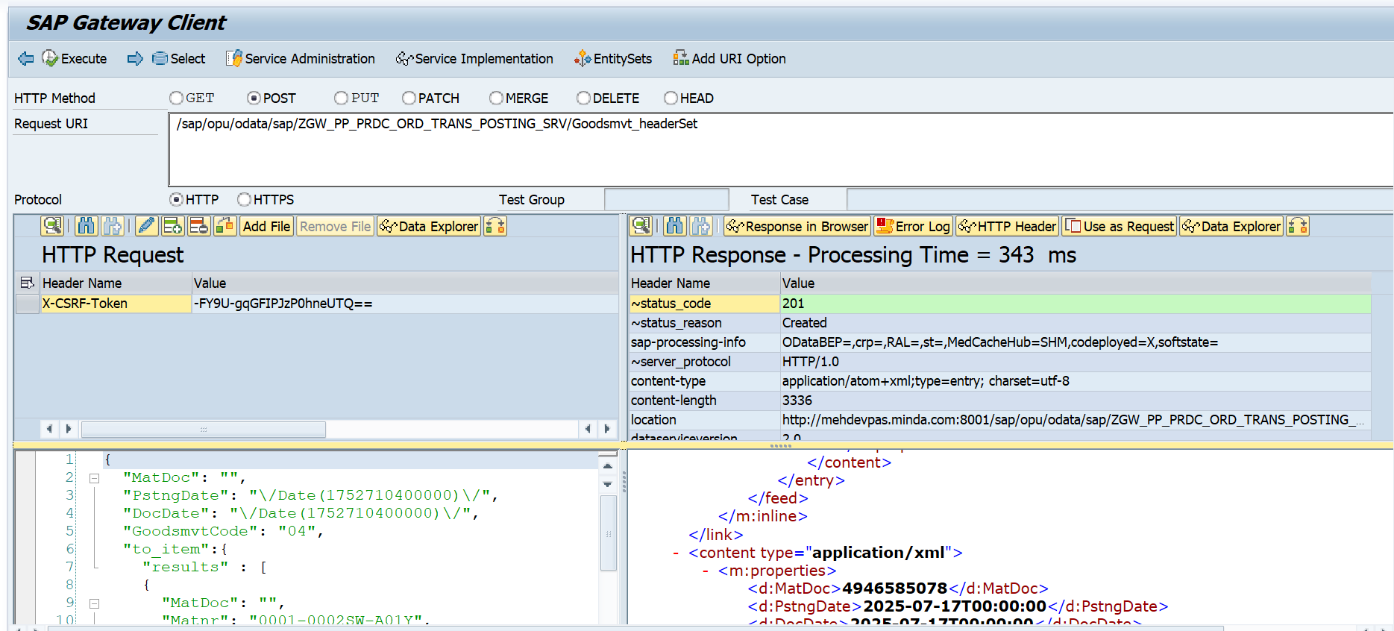
}

**5.Positive and Negative Testing Screenshot :**

**Positive Testing Screen:**

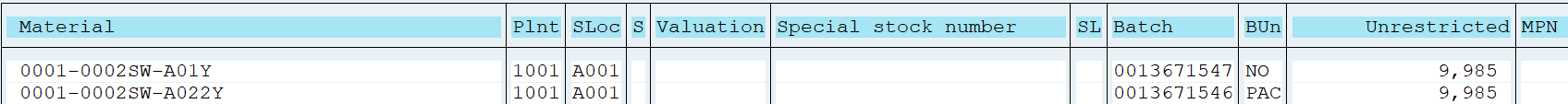
Before Transfer Posting the material (Mentioned in Payload) quantity at **MB52 Tcode.**



Transfer posting Request :

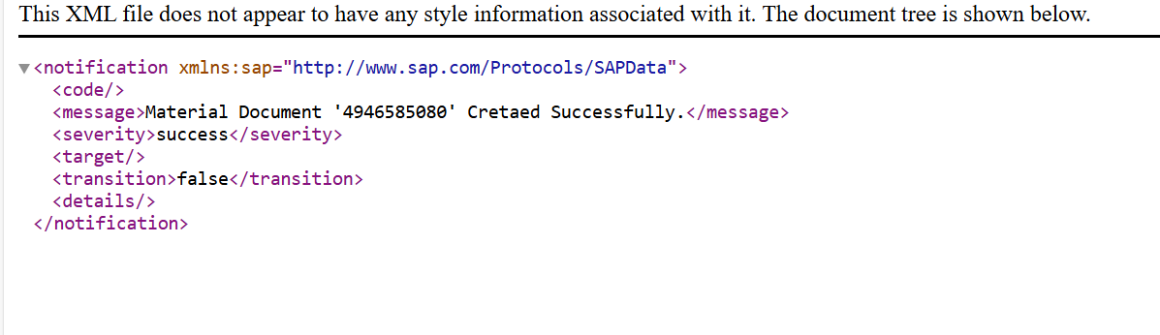
After Transfer Posting , the material (Mentioned in Payload) quantity at **MB52 Tcode.**

Here Only 1 quantity transferred from each Material.

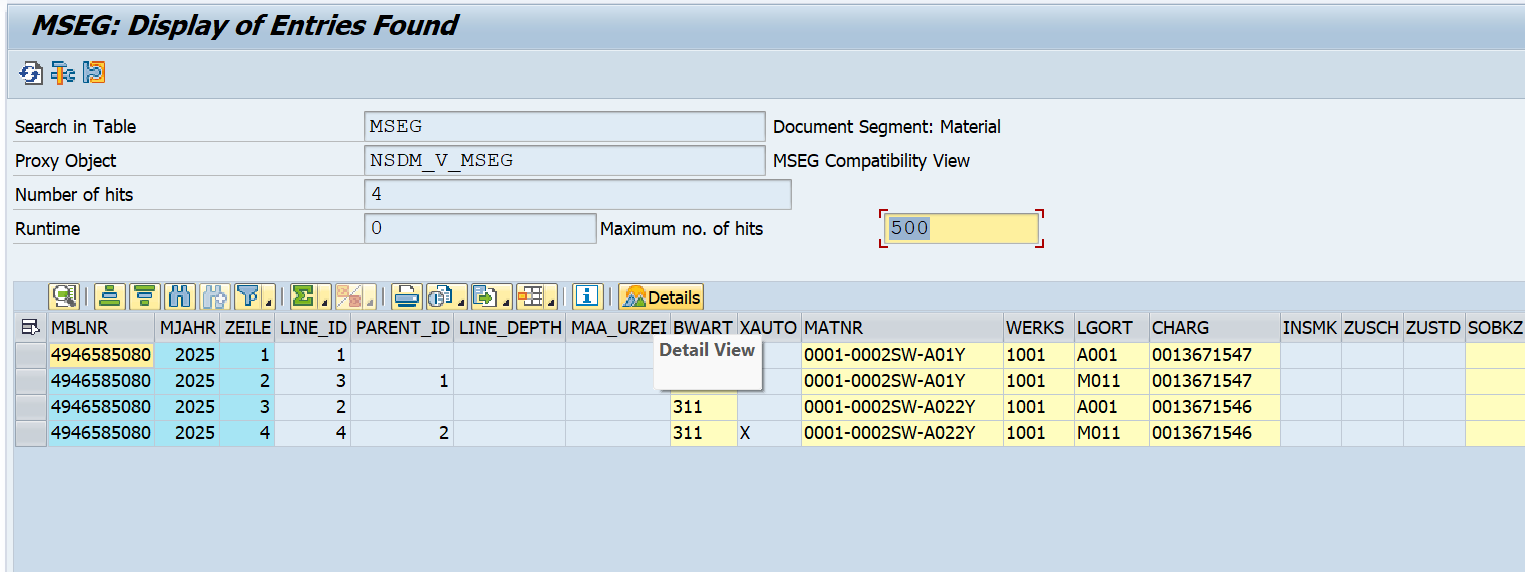
****

Success message from backend in Header Response With Created Material Document Number.

Message: Material Document '4946585080' Cretaed Successfully.



Checking Also at Database level:



**Negative Testing Screen.**

**CASE 1:**

If you trying to transfer material’s quantity which is grater the available quantity.

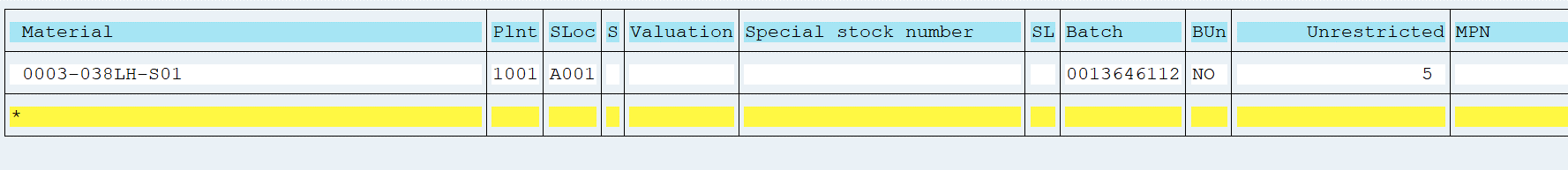
EX: material: 0003-038LH-S01

Plant : 1001

Storage Loc.: A001

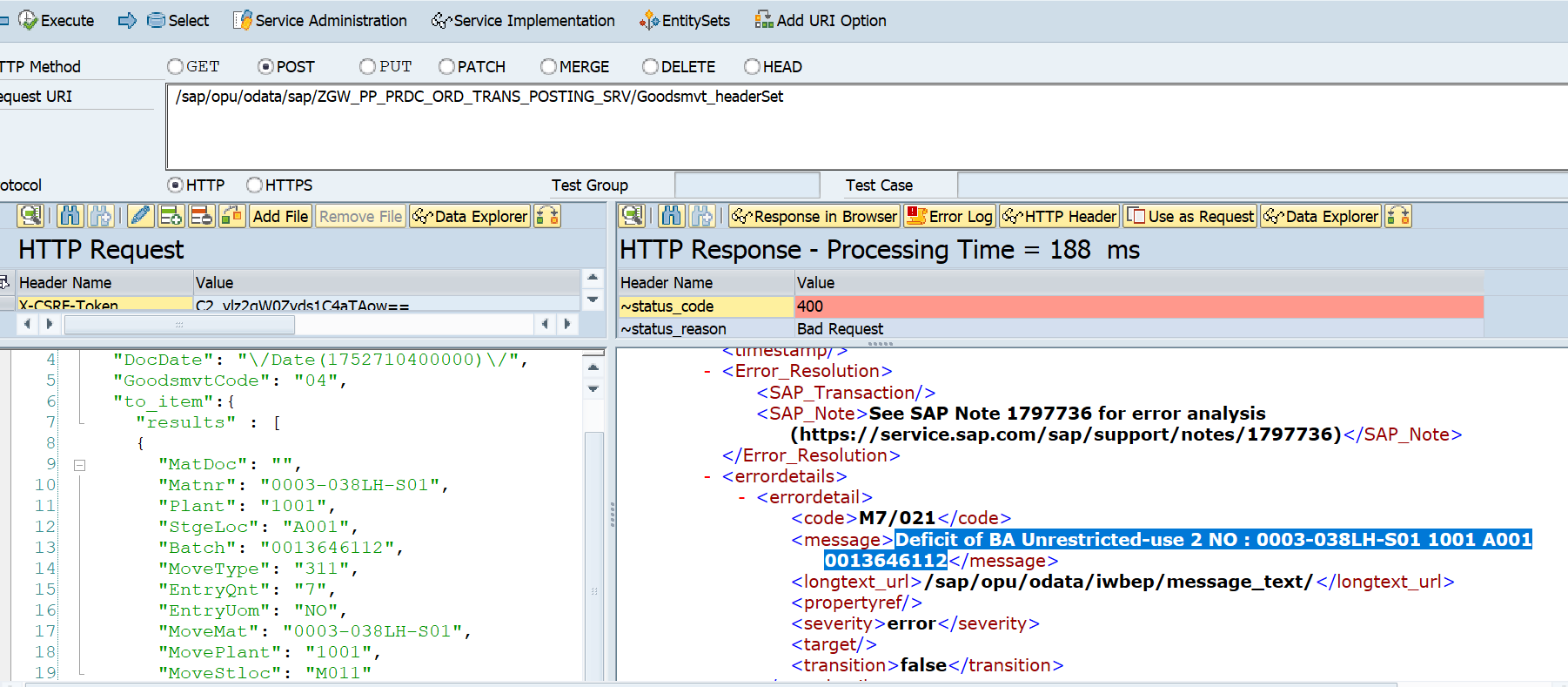
Batch: 0013646112

Available Unrestricted Stuck : 5



But here user if try to post a request of Grater then 5 (here I entered 7) ,then following error will through:

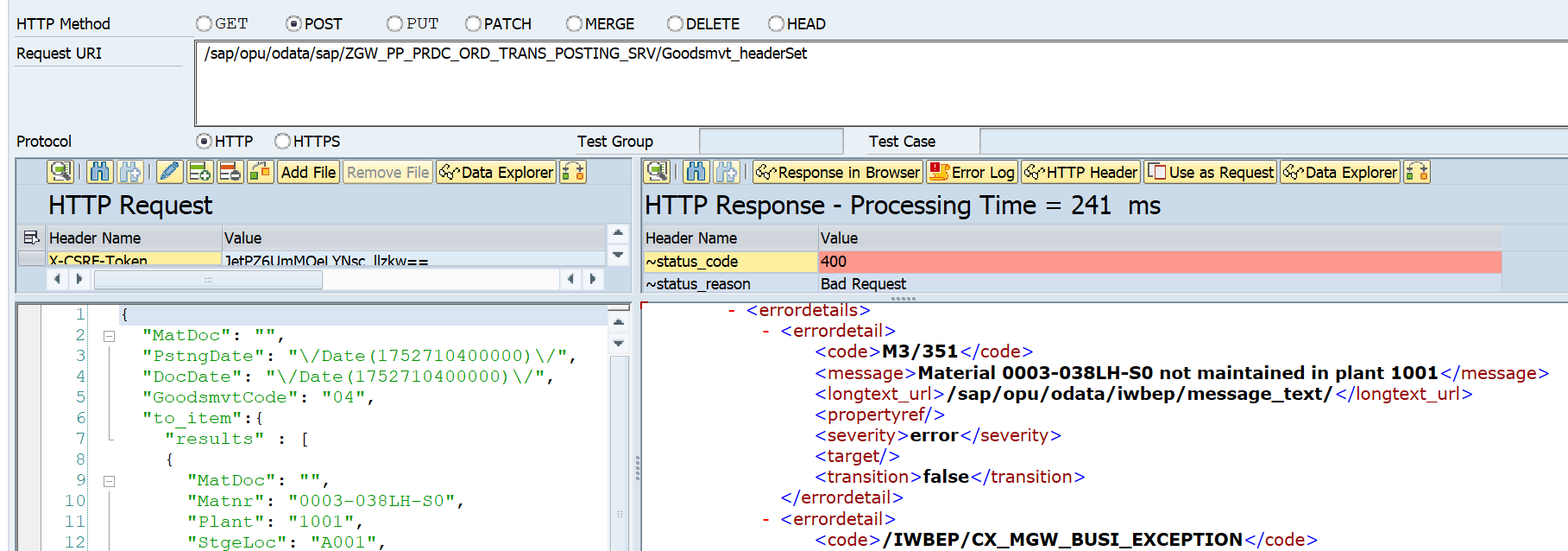
**Error** : M7/021 :Deficit of BA Unrestricted-use 2 NO : 0003-038LH-S01 1001 A001 0013646112.  
note : if such kind of error came that means, entered struck transfer quantity is grater then available quantity.



**Case 2**

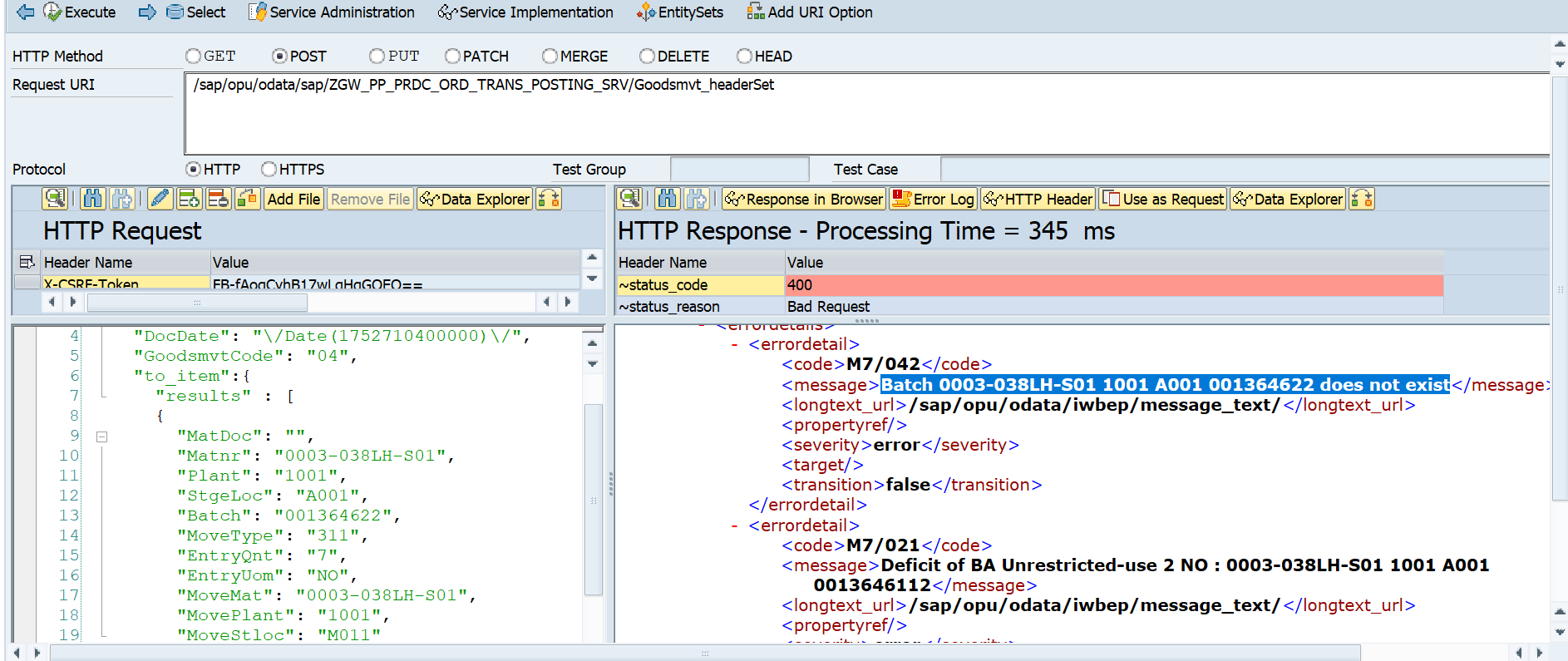
If some one entered wrong material Number.

Error : M3/351 : Material <Material Code> not maintained in plant 1001



**Case 3**

If Wrong Batch Number entered .

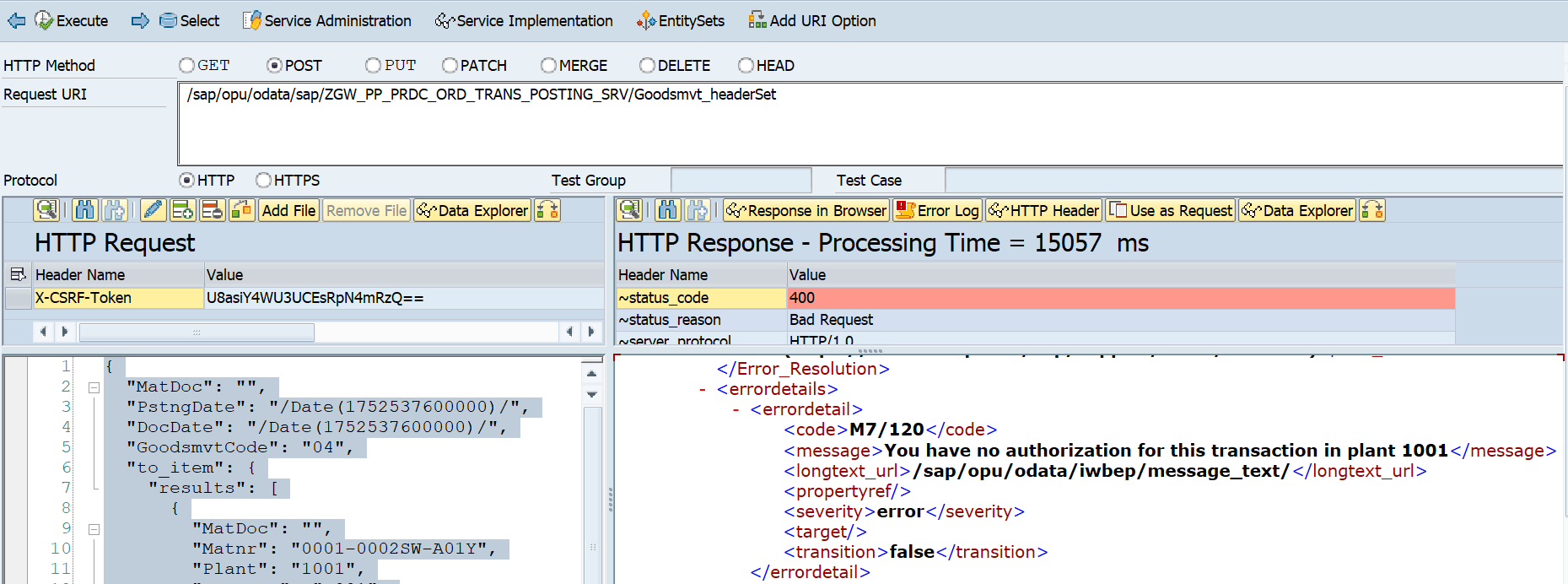
Error: M7/042 : Batch 0003-038LH-S01 1001 A001 001364622 does not exist

**Note :** like above, if any wrong data trying to pass, BAPI will handle those errors and BAPI error will raise as Business Exceptions in the Response body , even if you maintained 10 line item and error found in one line item then BAPI will not perform transfer posting other remaining correct data.

**Case 4**

If you are trying to do transfer posting to an unauthorized Plant.

Here I have used a SAP user That is “CAPPUSER3”, who only have the authorization of 3001 Plant, but here I am trying to do Transfer posting to Plant 1001, so in this case bellow error will arise with the first unauthorized plant if system encounter.  
  
if you see such kind of issue then pass the authorized plant in the item level.  
  
Error:M7/120: You have no authorization for this transaction in plant<Plant code>.



Postman

