

# **SDN Community Contribution**

**(This is not an official SAP document.)**

## **Disclaimer & Liability Notice**

This document may discuss sample coding or other information that does not include SAP official interfaces and therefore is not supported by SAP. Changes made based on this information are not supported and can be overwritten during an upgrade.

SAP will not be held liable for any damages caused by using or misusing the information, code or methods suggested in this document, and anyone using these methods does so at his/her own risk.

SAP offers no guarantees and assumes no responsibility or liability of any type with respect to the content of this technical article or code sample, including any liability resulting from incompatibility between the content within this document and the materials and services offered by SAP. You agree that you will not hold, or seek to hold, SAP responsible or liable with respect to the content of this document.

## Applies To:

Business Data Toolset (BDT) - SAP R/3 4.7

## Summary

Business Data Toolset is a central tool for maintaining master data and simple transaction data. This document introduces Business Data Toolset and explains the advantages of using Business Data Toolset, then illustrates these advantages by adding custom fields to GMGRANT transaction using Business Data Toolset.

**By:** Hari Krishna Nalluri

**Company:** Intelligroup Inc.

**Date:** Nov 16, 2005

## Table of Contents

Applies To: .....	2
Summary.....	2
Table of Contents.....	2
What is BDT? .....	3
Advantages .....	3
Extensibility: .....	3
Configurability .....	3
Divisibility .....	3
Development.....	4
Author Bio .....	39

## What is BDT?

The Business data toolset is a central control tool for maintaining SAP dialog programs. It is called as BDT in short. If an application transaction is BDT enabled, we can add new custom fields, hide standard fields, move standard fields from one screen to another, add new menu options, and add custom validations all without modifying the underlying transaction.

BDT is used at SAP for maintaining application transaction. Now SAP extended this tool for use by development partners and customers.

## Advantages

Business data toolset originated from the business partner project. The demands of more extensibility, configurability and divisibility required for data entry screens gave birth to the Business data toolset. These three demands, extensibility, configurability and divisibility are the advantages of the Business data toolset as explained below -

**Extensibility:** Though SAP incorporated an exhaustive list of attributes required in each business application, in each SAP implementation cycle we come across requirements where customers want to add more and more attributes specific to their business. Since these attributes are so specific to their business only, it's impossible for SAP to collect and implement all these attributes. This is more so in case of ramp-up applications like grants management where it takes few implementations to get a complete list of attributes. So what is required is a possibility to extend the standard business application without modifying any SAP standard objects. This could mean adding new fields, hiding existing fields, adding new sections, rearranging the sections as per user convenience all without modifying any standard SAP objects. The old fashioned user exits, BADIs and screen variants help us in fulfilling this requirement partially, but they don't always give the flexibility required. This is where business data toolset can help. Using business data toolset various parts of the dialog program can be extended which includes –

- Screen layout
- Screen sequence
- Program logic
- Menu
- Search help
- Field grouping
- Authorization checks

## Configurability

For lot of customers, the entire list of screen fields provided by SAP may not be useful. These unnecessary data fields can be suppressed but user still have to go through screen after screen on which only one or two fields are relevant. Switching screens often slows down data entry considerably. So what is required is option for configurability so that user can move the more important fields to initial screens, group the fields as per the business and hide the fields not required .BDT provides this flexibility to tailor entry screens to their individual needs and keep the number of screens to a minimum. For ex, the fields most commonly used by the user can be moved to the first screen and the less useful fields can be moved to the later screens.

## Divisibility

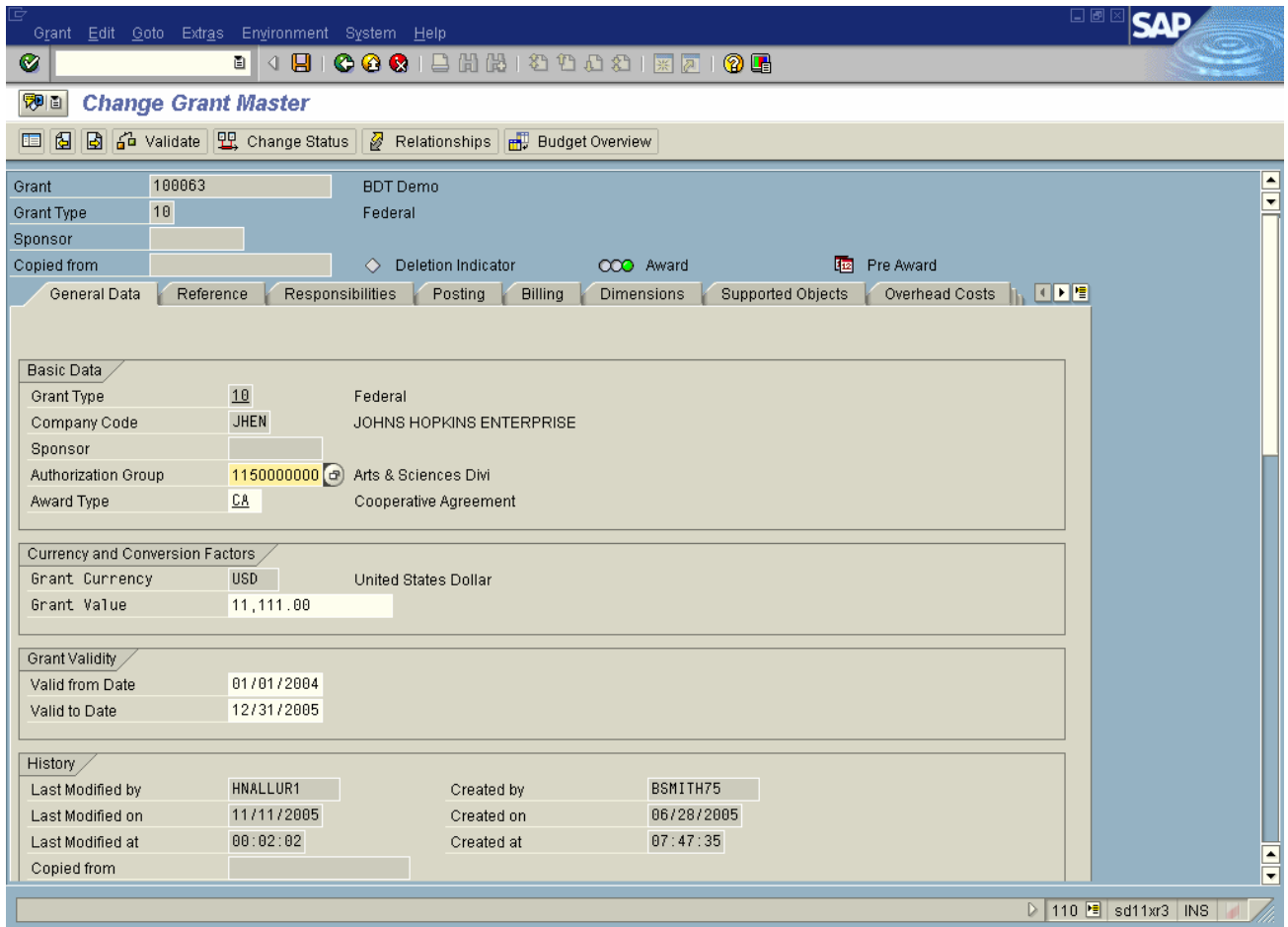
For large business objects like business partner, the total number of attributes might be in hundreds. It is impossible to include all these attributes in each type of maintenance; it would result in too many screens. So it is beneficial if the maintenance is broken into small parts based on some criteria. For ex, each business partner can have different roles like customer, vendor, sponsor etc and each of these roles result into different set of attributes. So if the maintenance is broken down into small parts based on roles, it will be easier to maintain. Classifying the information this way is called divisibility and BDT helps in achieving divisibility.

## Development

All the advantages mentioned above are achieved by storing the screen layout information in control tables. Screen layout consists of following entities:

1. **Field Group** - Field grouping is a way to group list of fields. Fields that co-exists all the time can be grouped into one Field group. Display attributes like hiding fields or making fields optional or mandatory or display only is set at field group level for ease of maintenance. For ex, if a transaction has two fields named start date and end date, its likely that these two fields always co-exists i.e. either both of the fields get displayed or both of them hidden but never one field is displayed and other is hidden so these two fields can be created under same field group.
2. **View** – One or more field groups constitute a view. All fields that are displayed and checked together are created as view. For ex, if a view two field groups included, all the fields from the two field groups always get displayed together. One field group can be hidden and another can be displayed (display attribute is configurable at field group level) but if both are displayed, they will be displayed together i.e. you can't split fields in a view and display some in one screen and another in another screen. Technically a view is nothing but a subscreen and all the fields in the included field groups are added to this subscreen through screen painter.
3. **Section** - One or more views can be grouped into a section. The BDT automatically puts a frame around the section in display. The advantage with having section is views that make sense together can be grouped into one section and they will be displayed together. If at later stage this is not required, the views can be displayed in separate block by just moving the view from the existing section to separate section.
4. **Screen** - One or more sections can be grouped into a screen. Screen is the biggest unit in the screen layout. Each screen will be displayed as a tab in the final transaction.

All the grouping information like field groups in views, views in sections, sections in screens is stored in control tables and maintenance transactions are provided to change this information. When the application transaction gets triggered, BDT queries these control tables and displays according to the configured data in the control tables.



In the above screen, each tab is defined as a screen in screen layout for ex, General Data as one screen, Reference as one screen etc. All the sections with frames like Basic Data, Currency, and Grant Validity etc are created as sections. Within Basic Data section, there is only one view (though multiple is possible). This view has five field groups, one for Grant Type and Grant Type description, and another for Company Code and company code description etc.

Consider the following changes:

- Hide Company Code and Description (Set at field group)
- Move Grant Validity to Basic Data section – Move the view to Basic data section
- Move Currency Section to Posting tab – Move Section to Posting Screen
- Add new set of fields in the General data tab - Create new field group, view, section and add to general data tab
- Hide the Responsibilities tab – Hide the screen from screen sequence

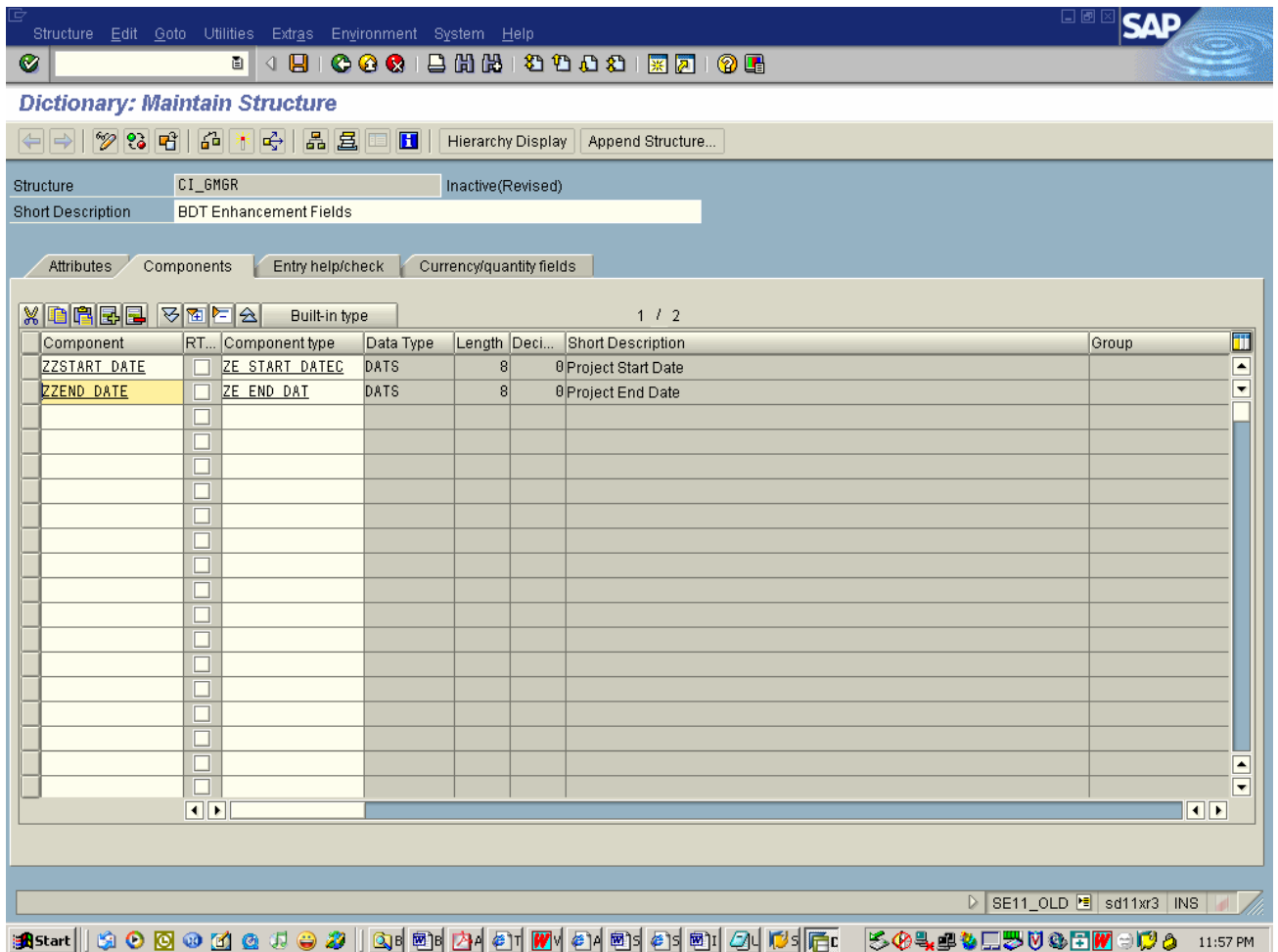
All these changes and much more are possible through BDT without modifying any standard program and by just adjusting the screen layout and there by control tables.

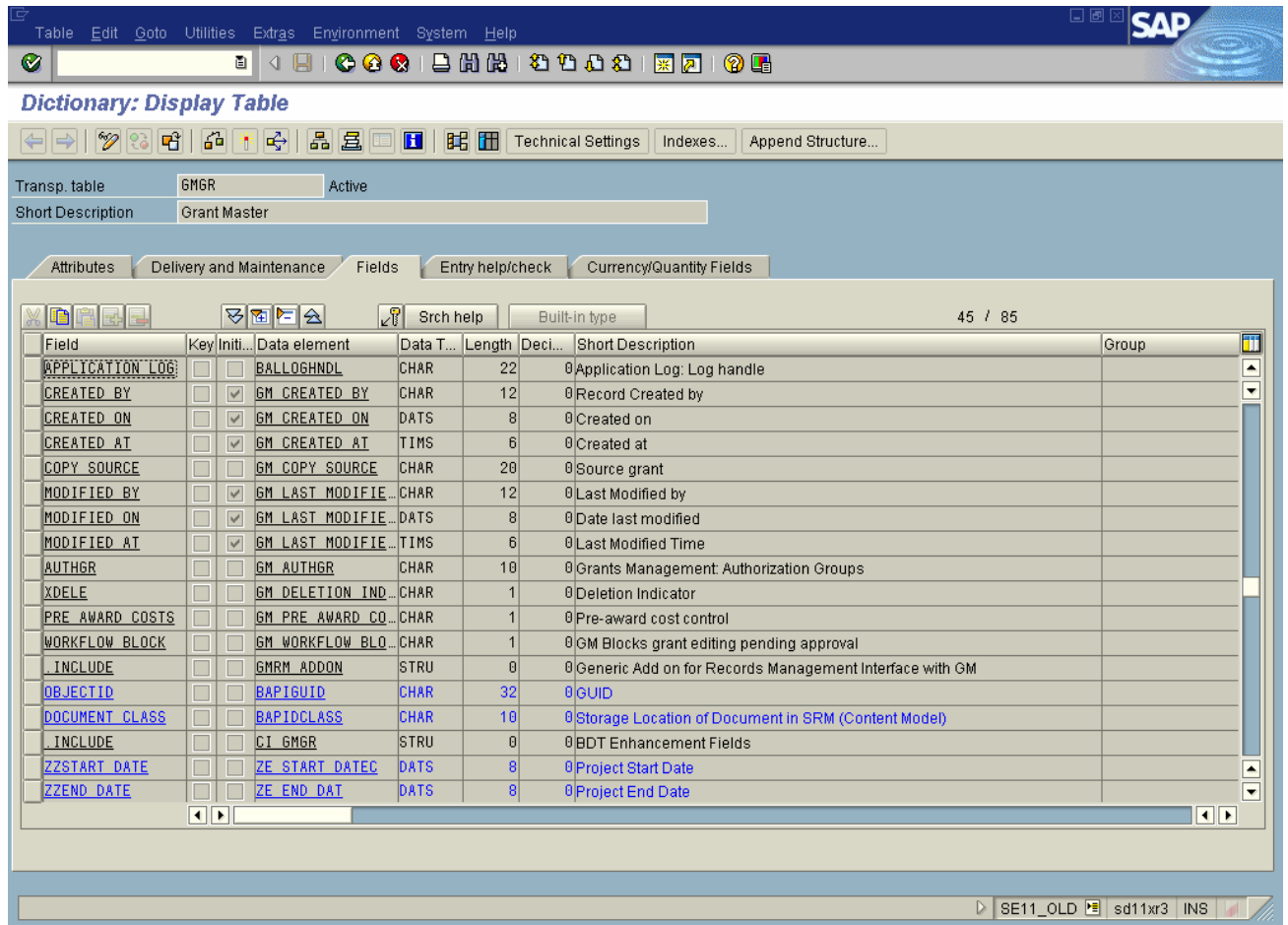
This section first explains how to add a custom screen with custom fields to GMGRANT transaction. Then it explains how to move these custom fields from custom screen to one of the SAP standard screens. We are going to add two custom date fields named Project Start date and Project End Date.

Transaction used in the development

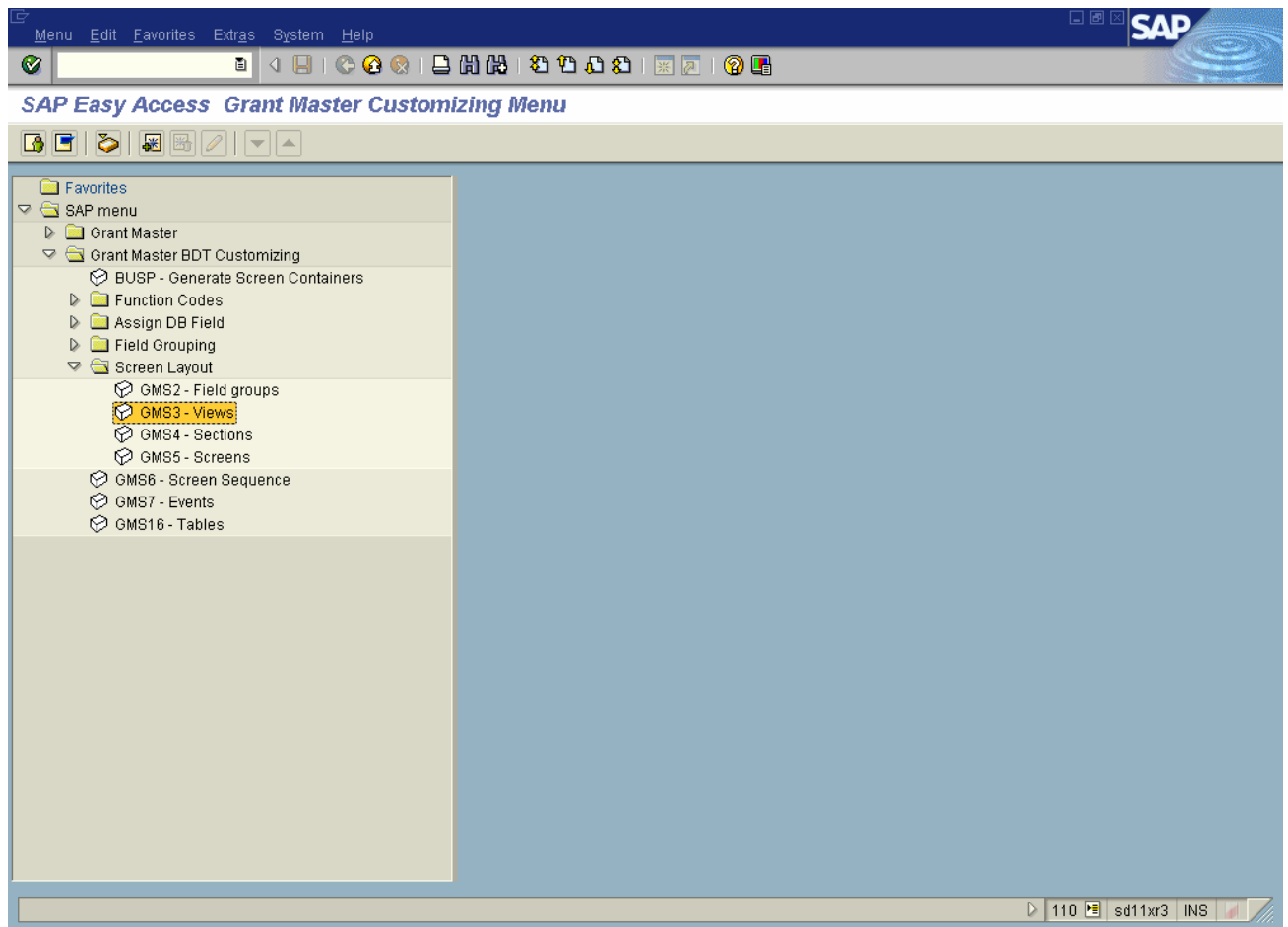
1. GRANT\_CUSTOMIZE (Area Menu) - For each business application supported by BDT, there will be an area menu which will have the list of transactions needed for the BDT development. For GMGRANT transaction, the area menu is GRANT\_CUSTOMIZE.
2. GMGRANT – This is the grant master transaction.

1) First step is to create the two custom date fields in the database. The master table used for storing grants is GMGR. A common include named CI\_GMGR is provided by SAP to add custom fields to the GMGR table. This is included in the master table GMGR so any additions to CI\_GMGR will get included in the GMGR table as well. Add two fields to CI\_GMGR and activate. Activation will adjust the GMGR table and the GMGR table will have our new 2 custom fields.



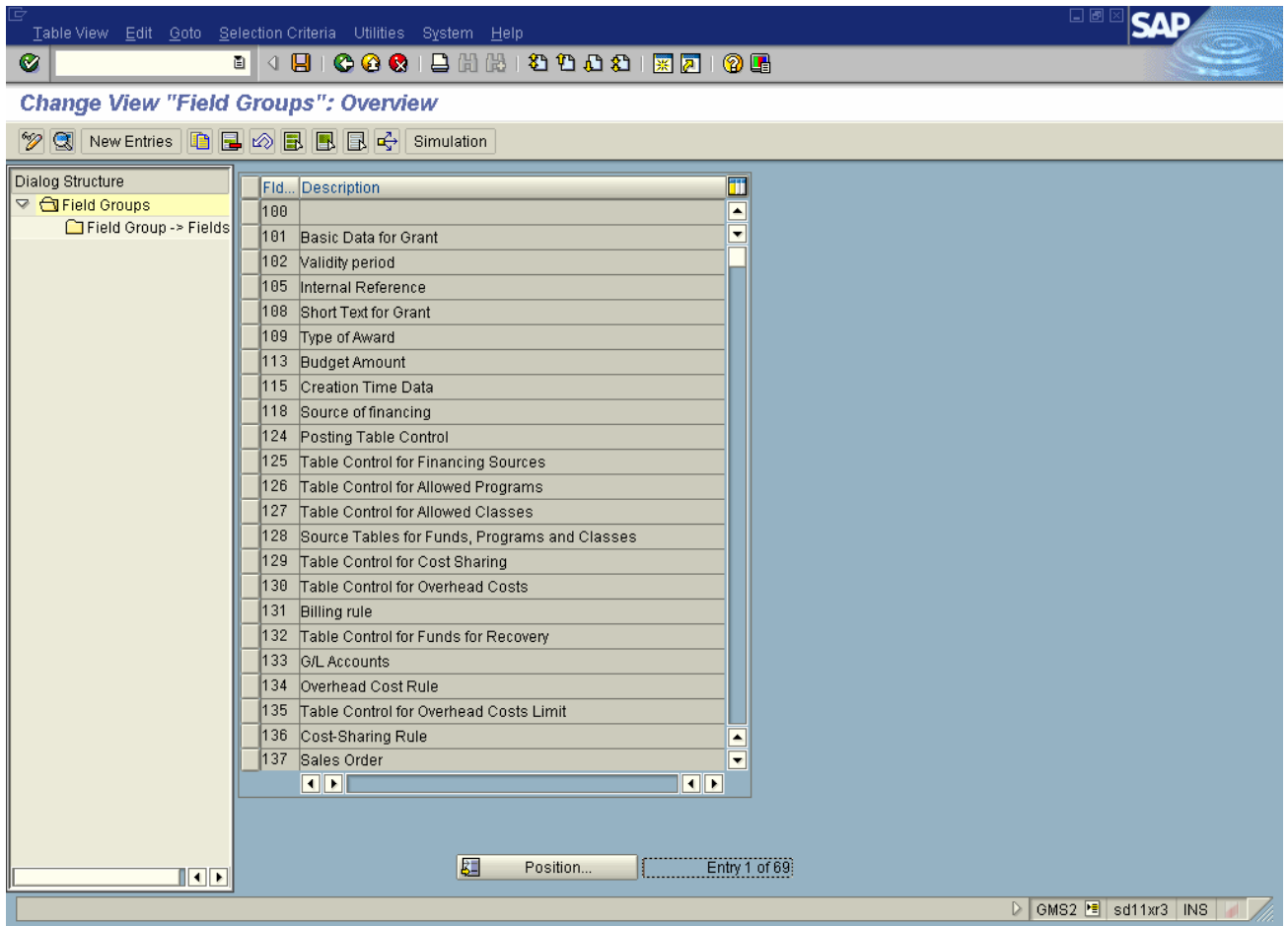


2) Execute the area Menu GRANT\_CUSTOMIZE. This will display the list of all the transaction needed for GMGRANT BDT development. Four transactions under Screen layout folder (see the screenshot below) one each for field group, view, section and screen are the most important transactions for BDT development.

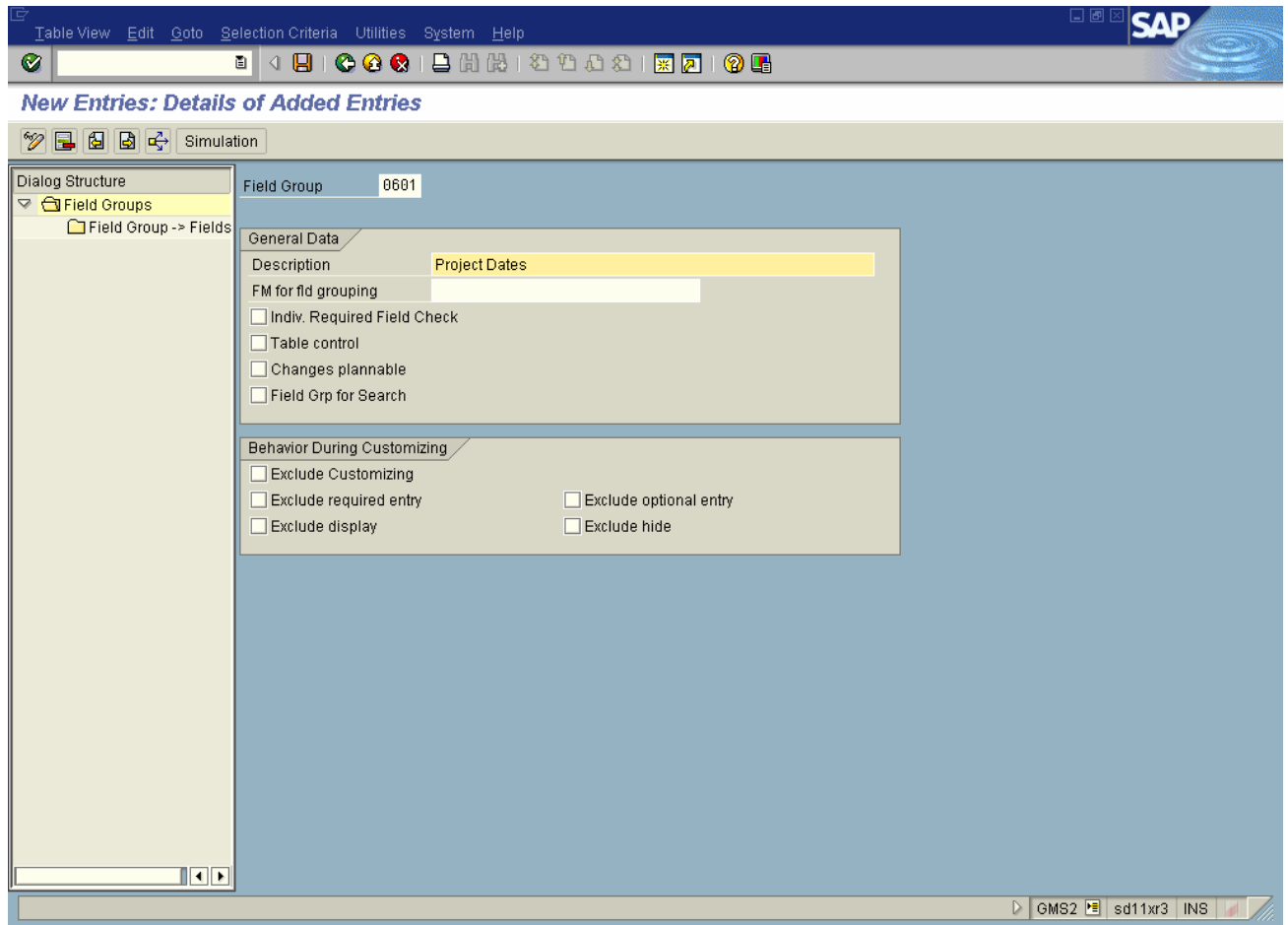


3) Next step is to create a field group for the custom fields. Field group is a number and Field group range 600-749 is available for customer. To create field group, execute transaction GMS2 (Grant Master BDT Customizing-> Screen layout -> 'Field groups'). This will display the list of existing field groups.

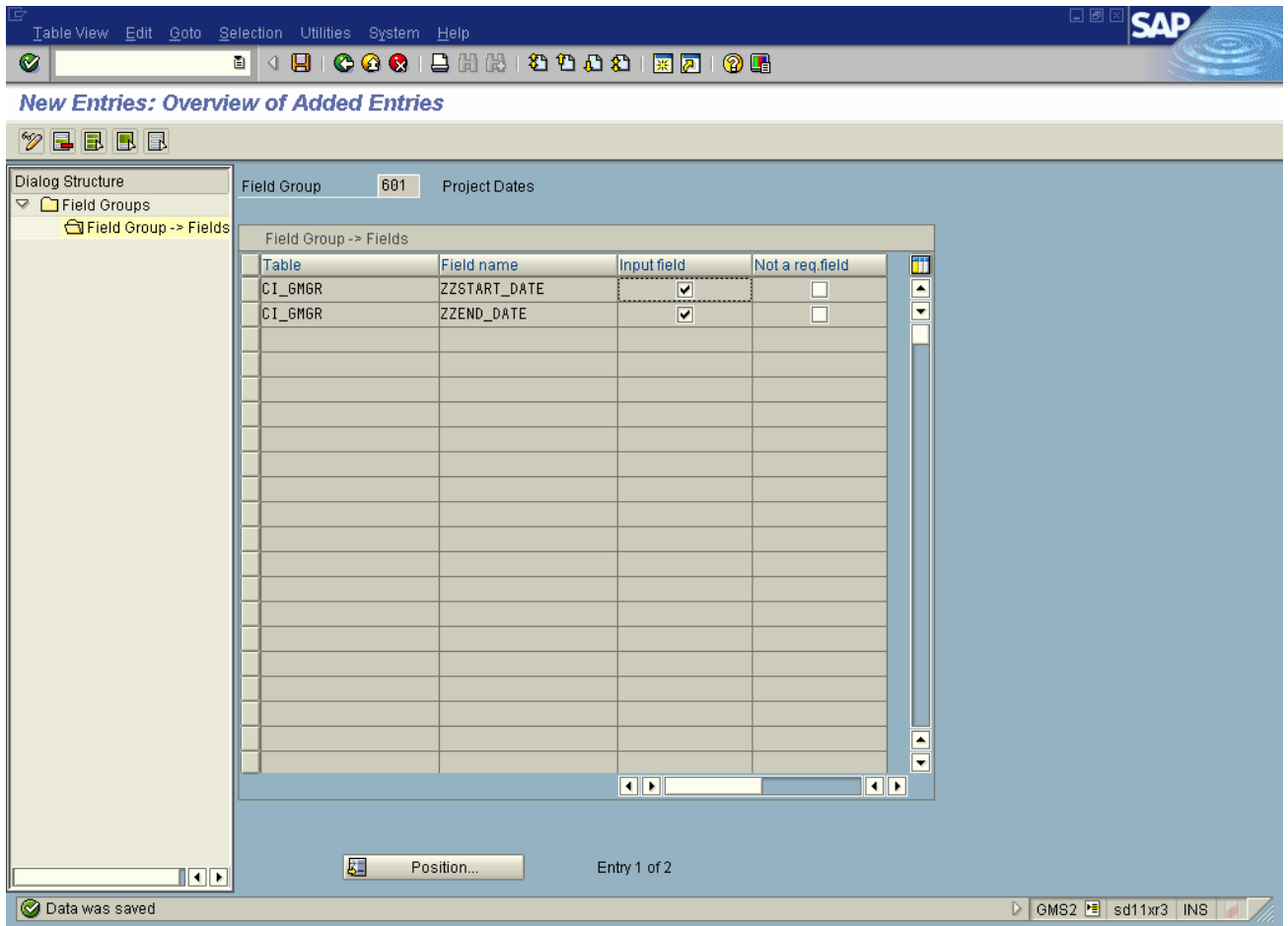




Click New Entries button. This will open a blank form in right frame for creation of new field group. Enter 0601 as the field group and Project Dates as the description. Other fields can be left blank.

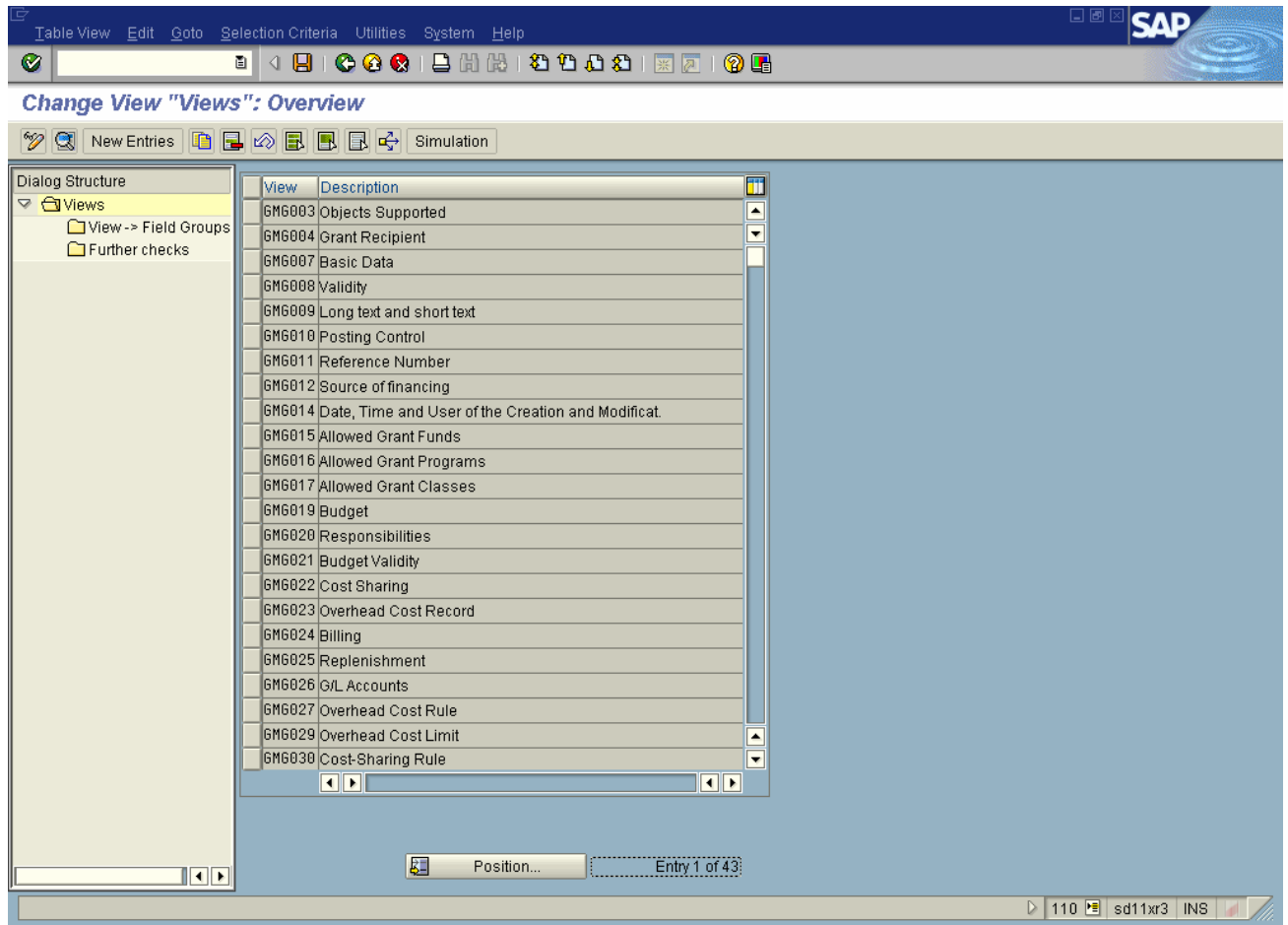


Save Data entered and hit the 'Field Group-> Fields' folder option in the left frame. This will open up a new blank screen in the right frame to enter the list of fields that form the field group. Enter our two custom date field names here-



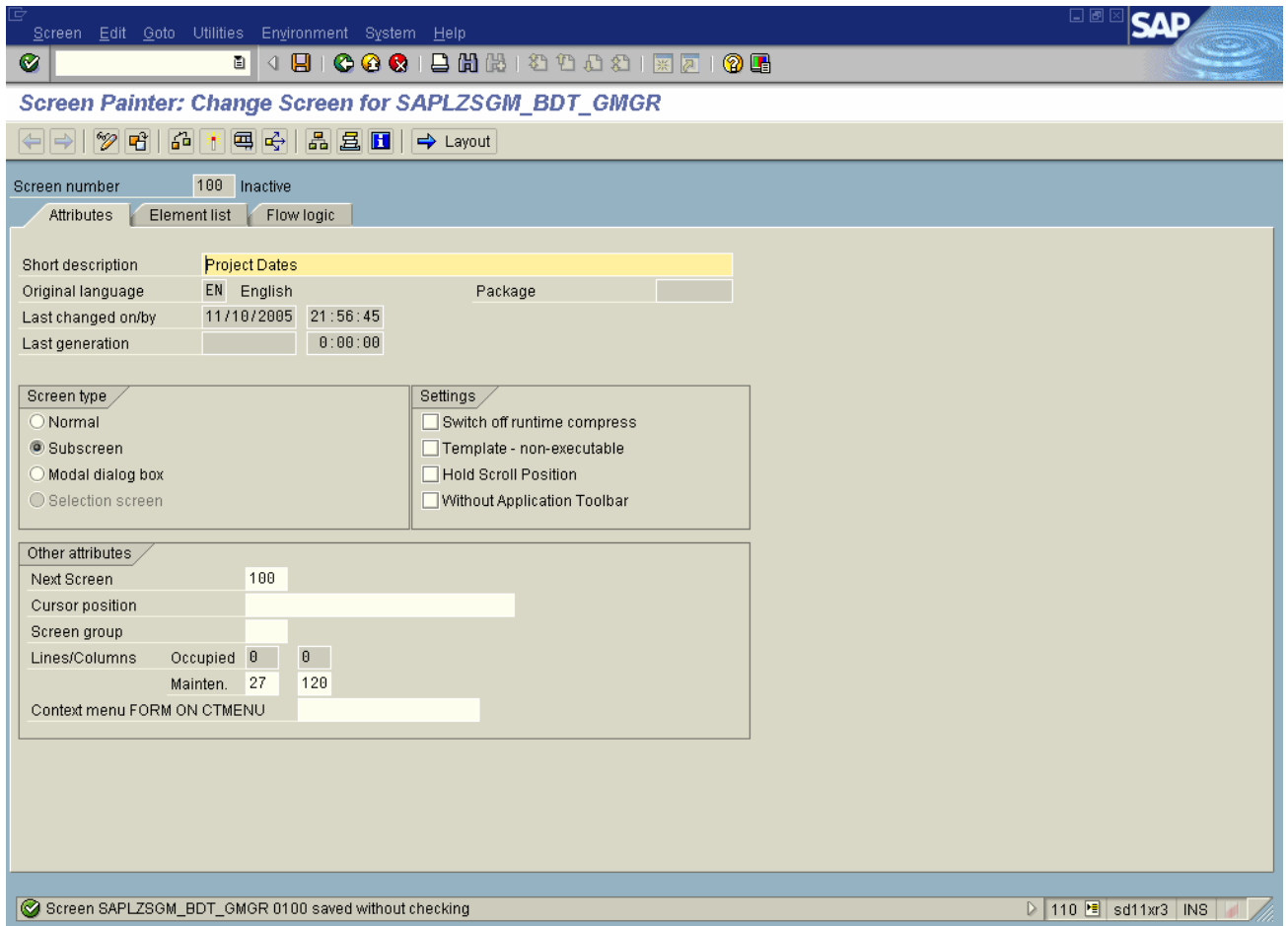
Hit save button. Warning message “Field does not exist on the assigned subscreen” is thrown, ignore this and proceed to save. This message disappears once these fields are added to the view. Once saved, return back to the original BDT customizing area menu.

4) Next step is to create a view. To create view execute transaction GMS3(Grant Master BDT Customizing-> Screen layout -> 'Views'). This will display the list of existing standard views.

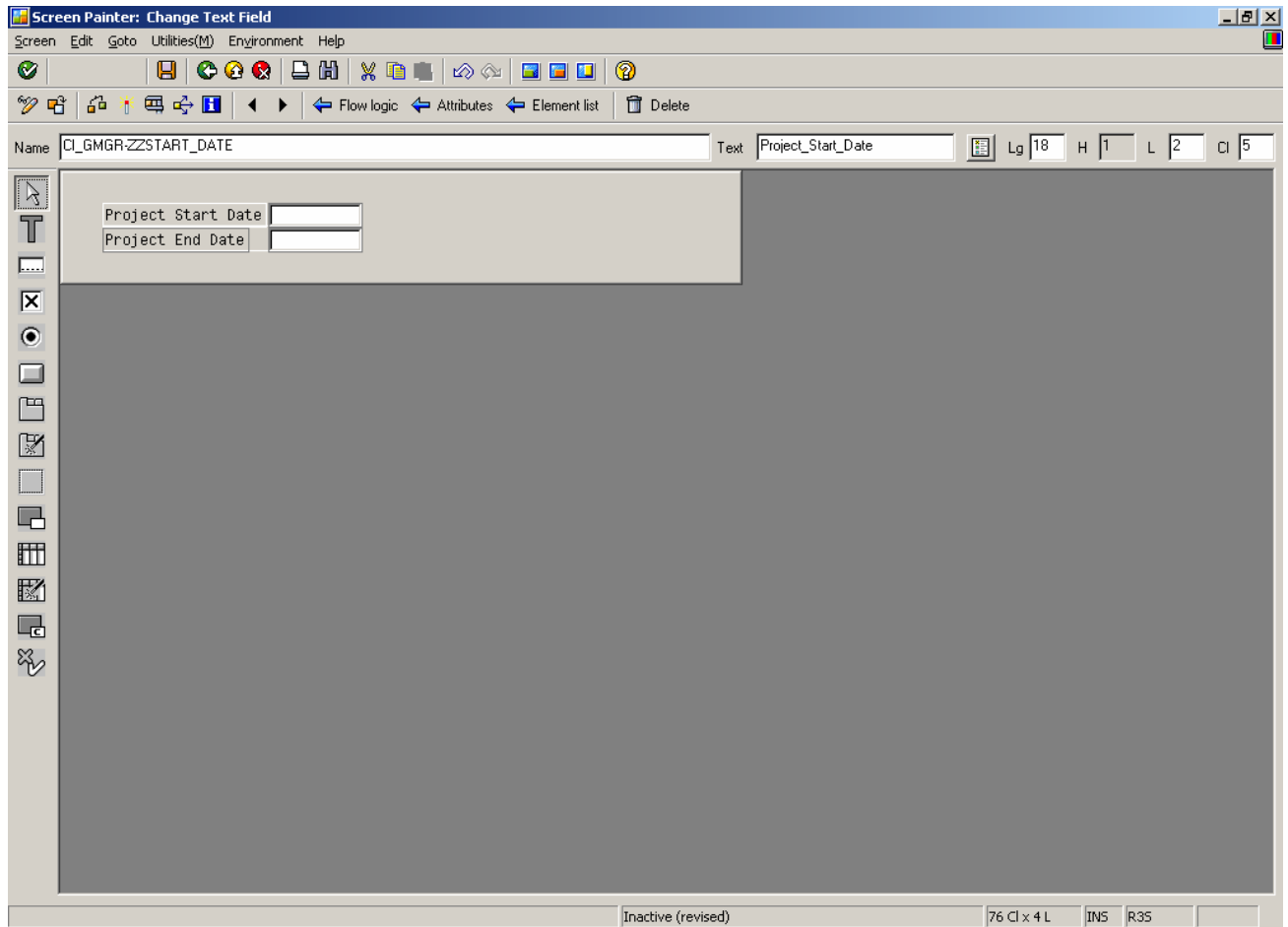


Hit the new entries button. This will open a blank form in right frame for creation of new view. Enter ZGMGPD as the view name, Project Dates as the description, GMG as application, Differentiation type as 0, Data set as GRANT1. Select Entry view check box.

As shown below in the screen, the next section of information to be entered is Subscreen. For this, open a separate session and create a custom function group. For our example, let's call it as ZSGM\_BDT\_GMGR. Go to SE51 and create a screen for this function group. This screen should be of type SUBSCREEN.

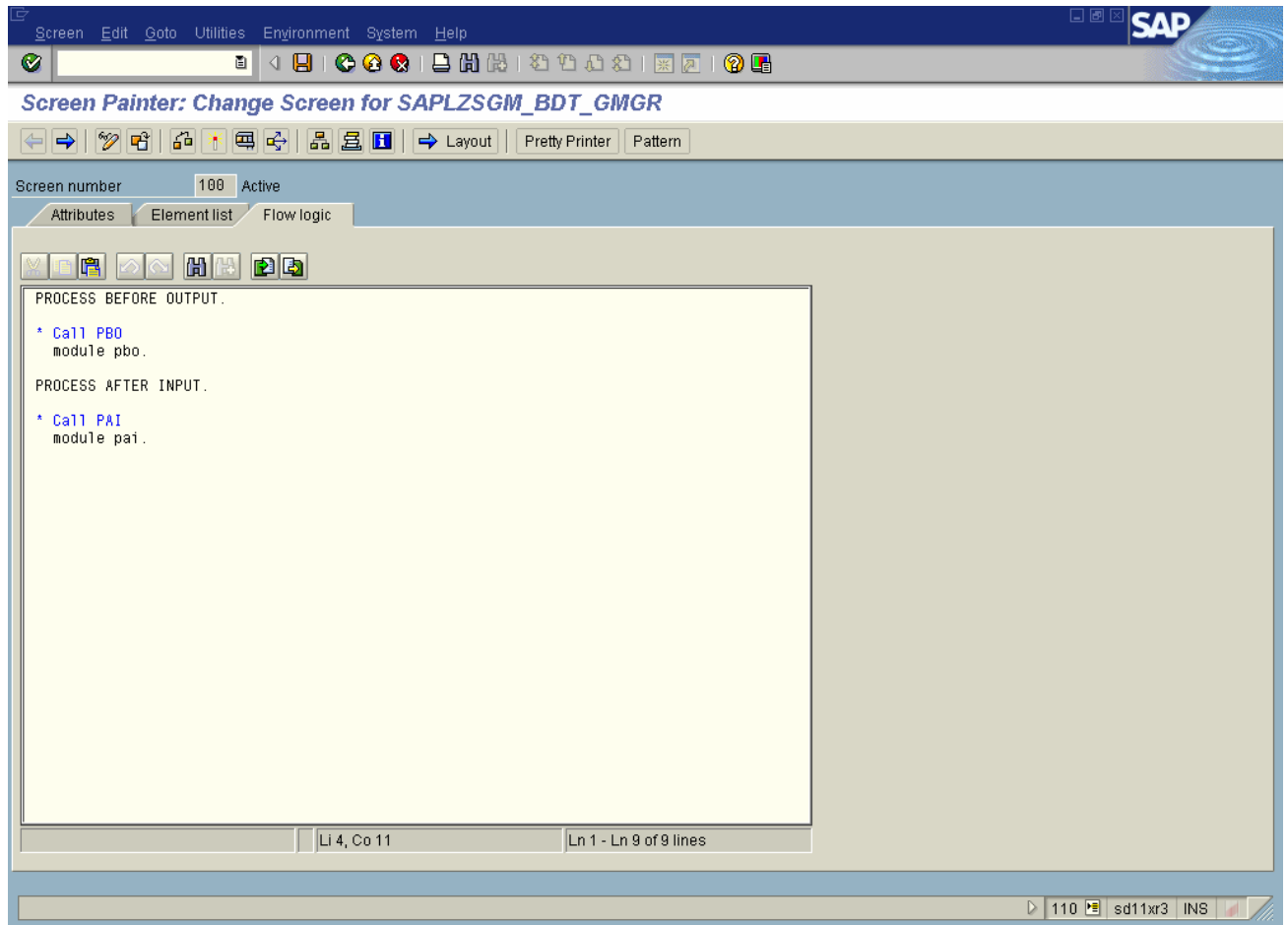


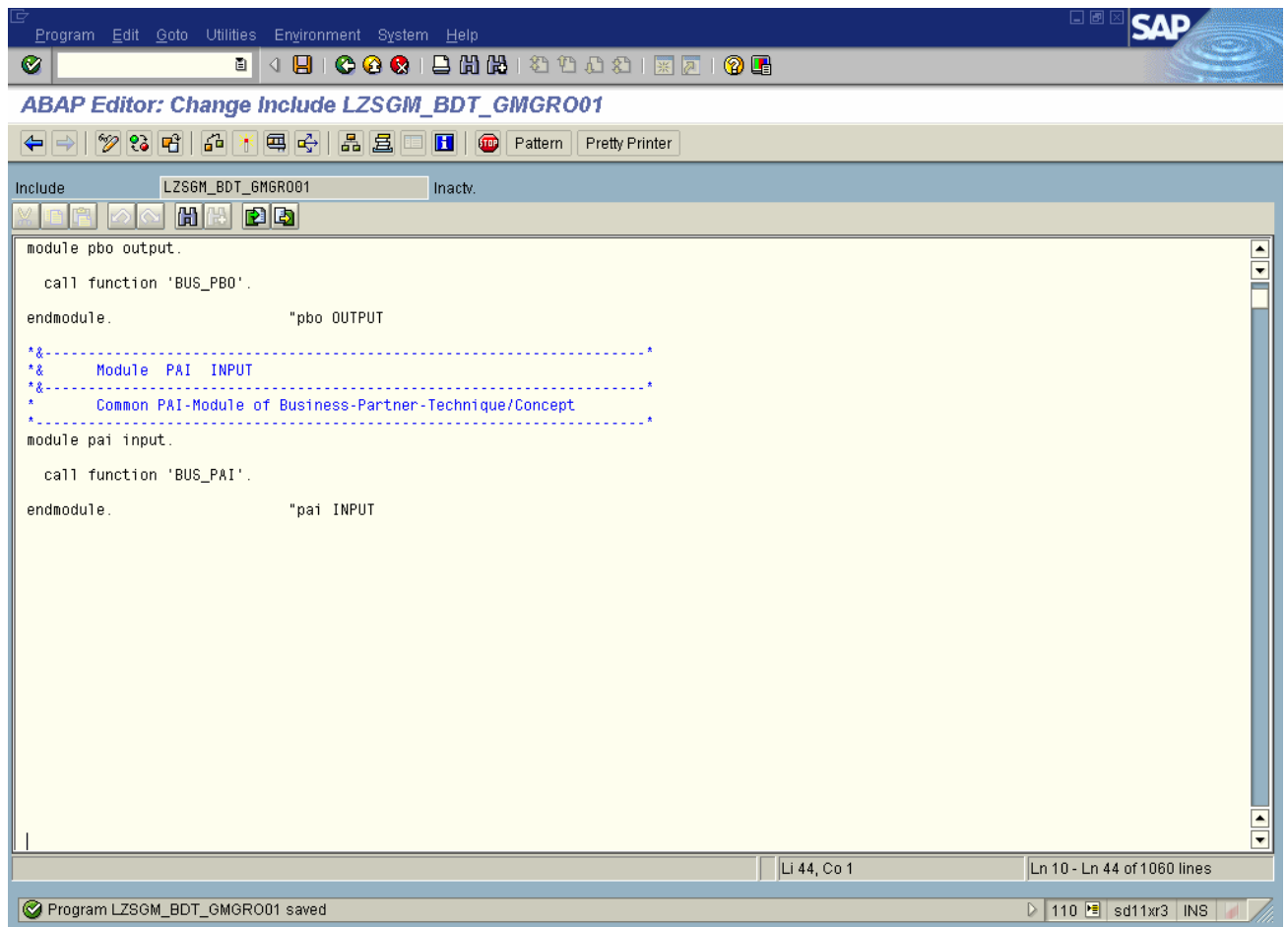
Now open the screen layout, Select Dictionary Structure CI\_GMGR and add our new custom Project Date fields to the screen.



Save the Screen. Return back to the flow logic section. Create one PBO module and one PAI module. In PBO module, call function module 'BUS\_PBO' and in PAI module call function module 'BUS\_PAI'.

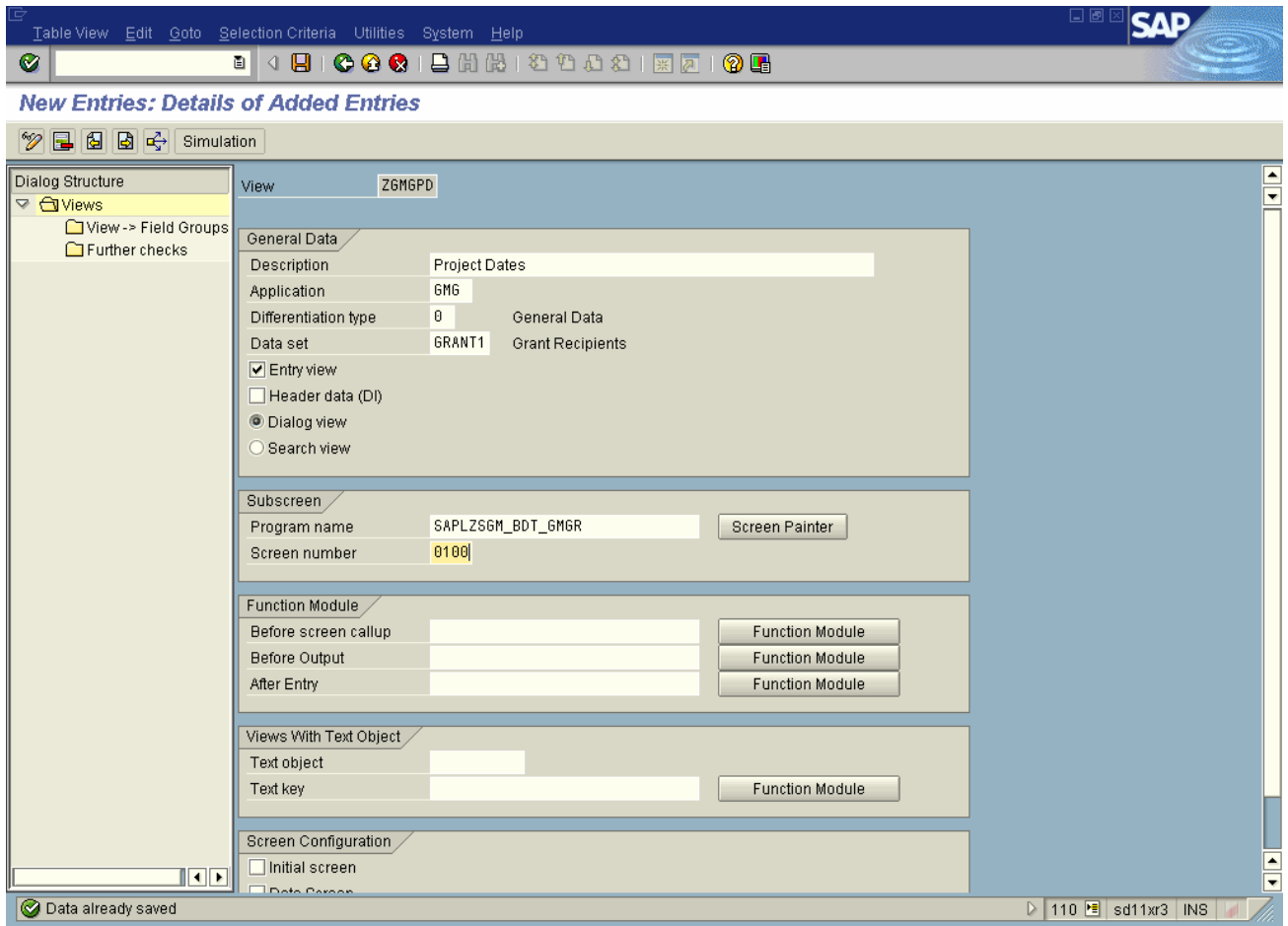
Other than calls to Function Modules 'BUS\_PBO' and 'BUS\_PAI', no other logic should be placed in the PBO and PAI modules. Data extraction, field checks should be carried out in a separate function modules.





Now get back to the view creation screen and enter the screen number we have created now.





Next section of information to be entered is Function Modules. There are 3 function modules, one for Before screen call up, second one for Before output and third one for After Entry. Any PBO and PAI logic that usually placed in the Flow logic section of Screen is to be placed in these function modules instead. BDT wants the interface and the Program Logic to be different is to enable DIRECT INPUT. "One of the goals for development with the BDT is a strict separation between interface and program logic. This ensures that all checks can be carried out even if you are using direct input and an external interface."

Before Output Function module can be used to put the PBO logic and After Entry function module can be used to put in the PAI checks. "Before screen callup" is useful mostly when the view contains table controls or step loops and not needed for this example.

Let's create a function modules named Z\_PROJECT\_DATES\_PBO and Z\_PROJECT\_DATES\_PA I as PBO and PAI function modules respectively. These don't need any parameters. BDT calls these function modules during the PBO, PAI events.

Now Go toTOP module of our function group and declare the following fields -

Tables : CI\_GMGR.

Data : wa\_gmgr type gmgr, wa\_gmgr\_old type gmgr.

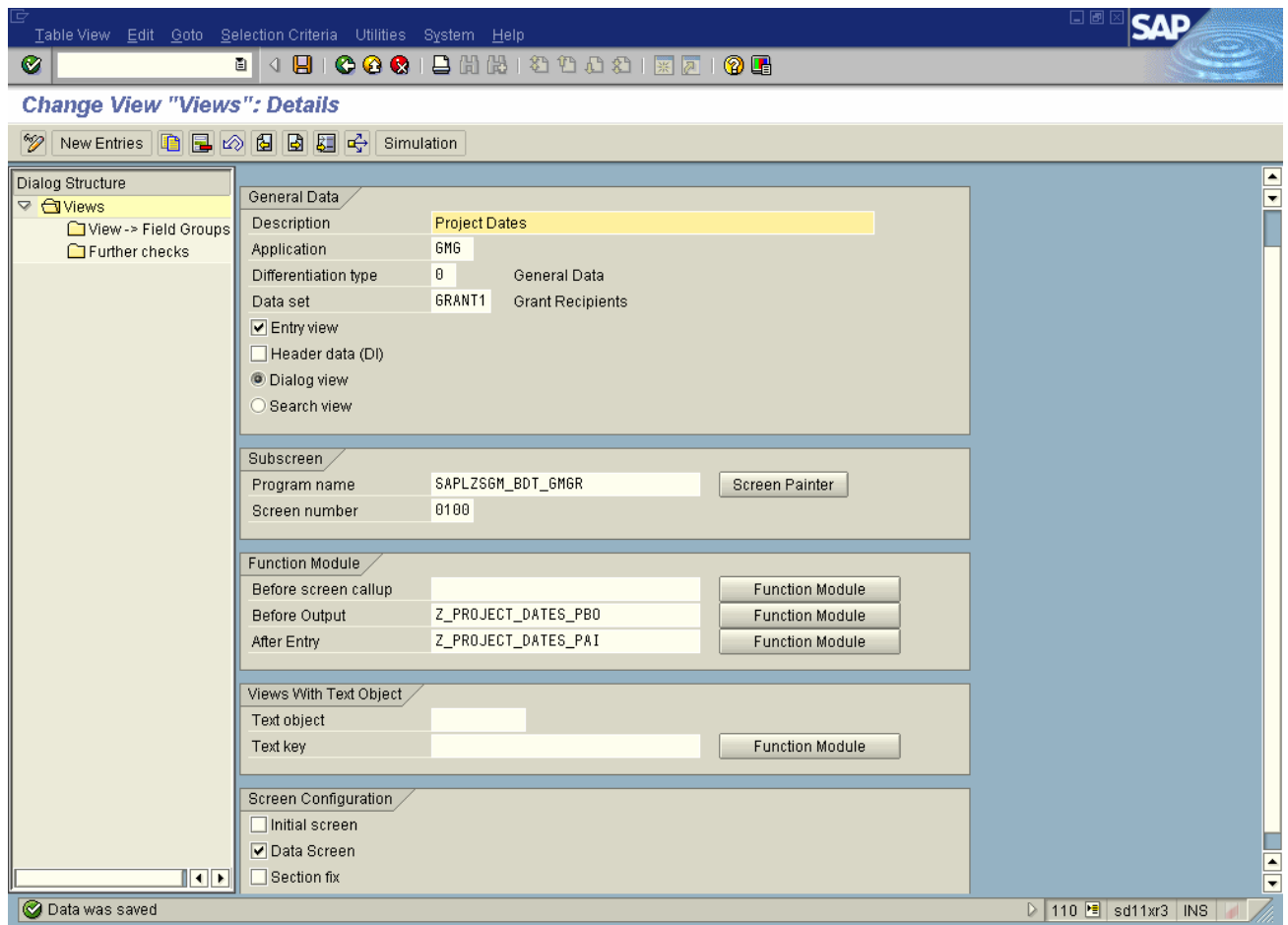
With in the source code of PBO Function module **Z\_PROJECT\_DATES\_PBO**, put following piece of code

```
CALL FUNCTION 'GMG_GMGT_GMGR_GET'
  IMPORTING
    e_gmgr = wa_gmgr.
MOVE-CORRESPONDING wa_gmgr TO ci_gmgr..
```

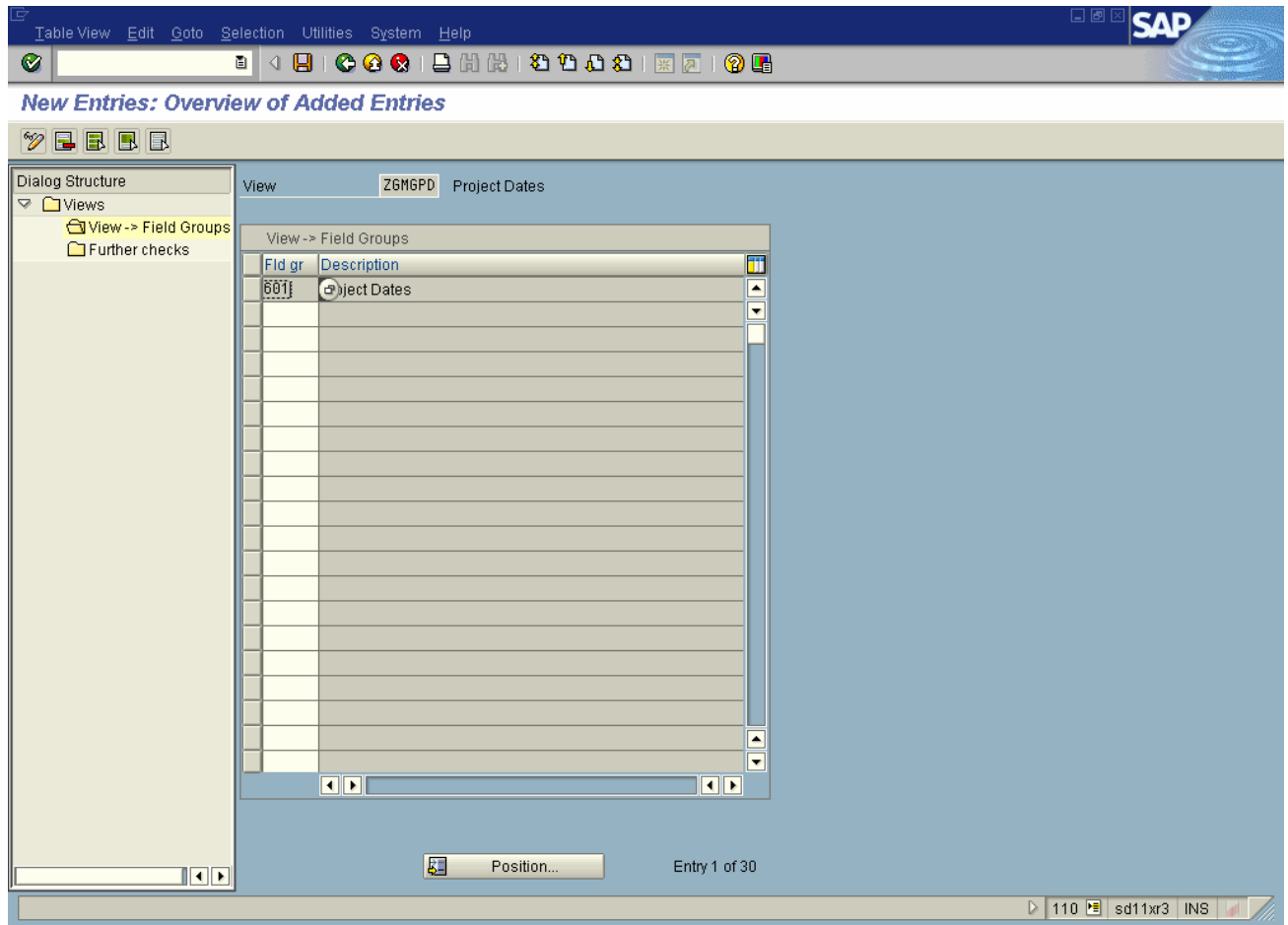
With in the source code of PAI function module **Z\_PROJECT\_DATES\_PAI**, put following piece of code –

```
MOVE-CORRESPONDING ci_gmgr TO wa_gmgr.
CALL FUNCTION 'GMG_GMGT_GMGR_COLLECT'
  EXPORTING
    i_subname = 'CI_GMGR'
    i_gmgr = wa_gmgr.
```

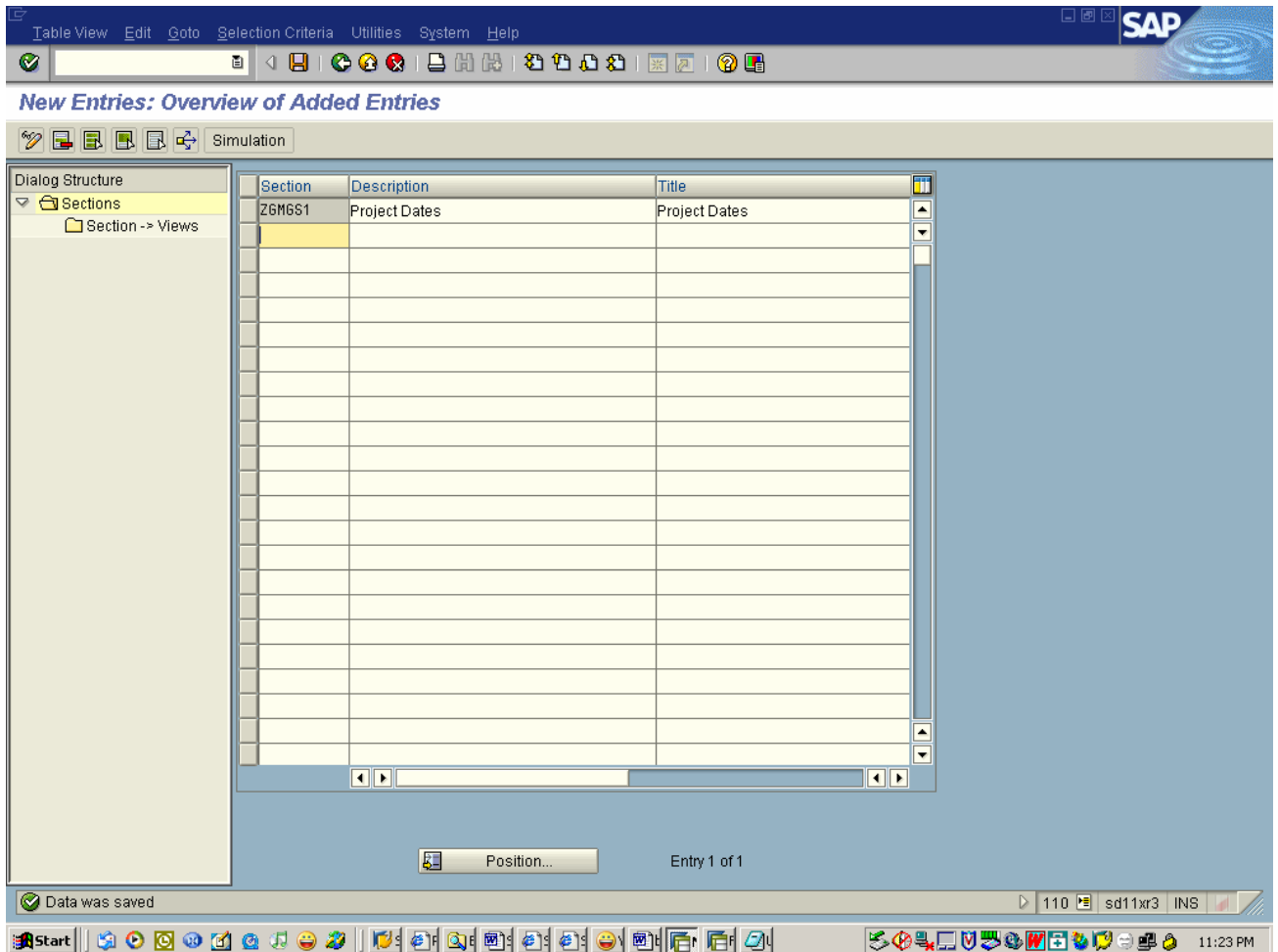
GMG\_GMGT\_GMGR\_GET and GMG\_GMGT\_GMGR\_COLLECT are the read and collect function modules of GMGR table. For the tables in application objects that are supported by BDT, there will be usually a Read and Collect function module. Read function module is used to read the current object data( in this case, the grant that is being processed). Collect function module is used to collect the current object data. So in this PBO Function module, Read function module GMG\_GMGT\_GMGR\_GET is called and it returns the current data of the GMGR structure for the grant being processed. When user makes changes and hits enter, PAI function module gets called and transfers the user changes to the main program using the GMG\_GMGT\_GMGR\_COLLECT. When the screen displayed again, the recent version collected by GMG\_GMGT\_GMGR\_COLLECT will be returned by the GMG\_GMGT\_GMGR\_GET FM. Now we have the PBO and PAI function modules, enter these in the View definition as shown below. Set the Data screen check box and save the view.



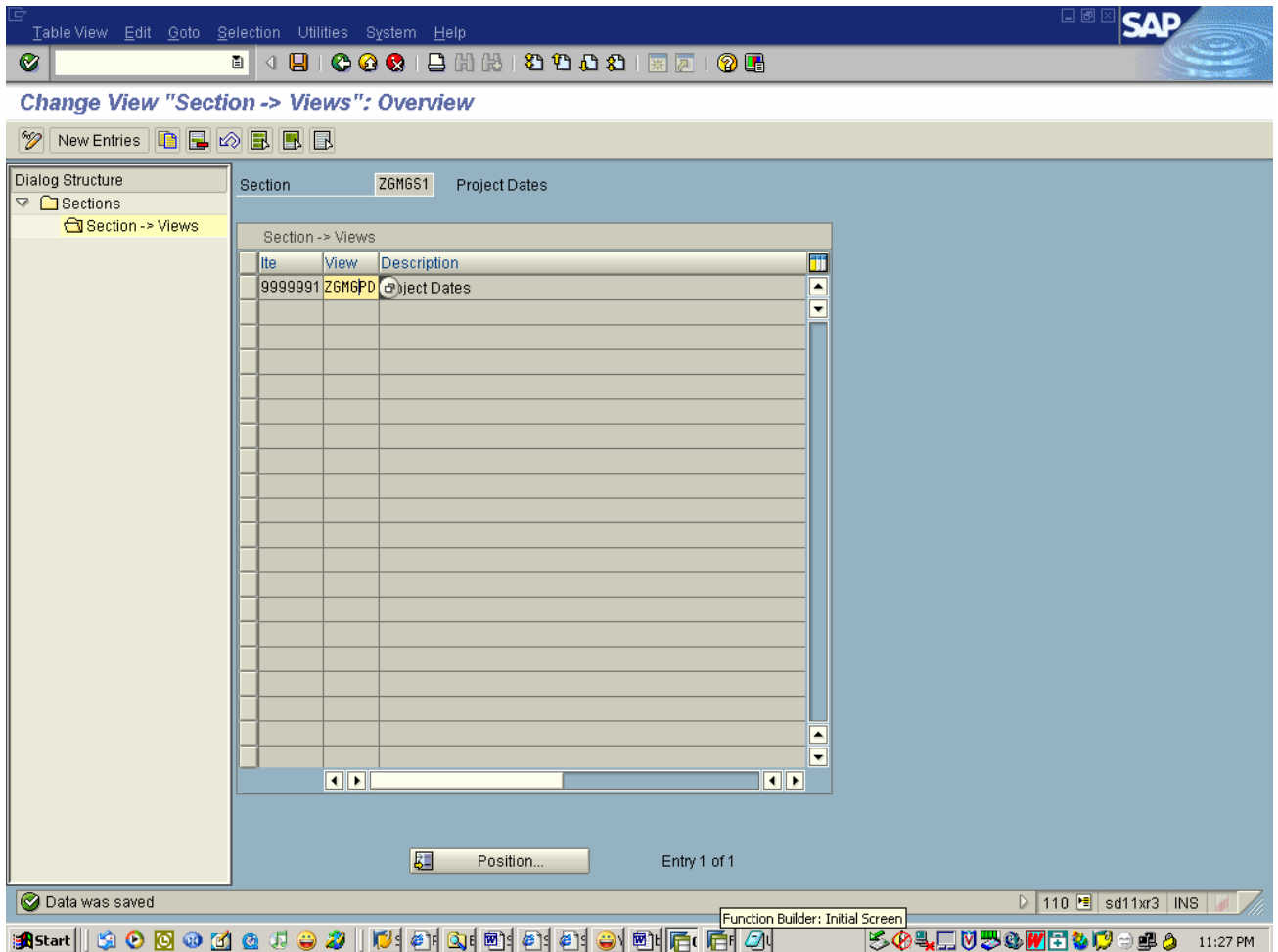
Now the view information successfully entered, we need to connect this view with the field group created in the prior step. So click the View->Field Groups option in the left frame. This opens up a data entry screen in the right frame. Enter the field group "0601" we created in the first step and save. This finishes the creation of view.



7) Next step is to create a section. Create section named ZGMGS1 using the transaction GMS4(Grant Master BDT Customizing-> Screen layout -> 'Sections'). Enter Project Dates as description and Title. Value entered in title column here will be used as the frame heading put around the section.

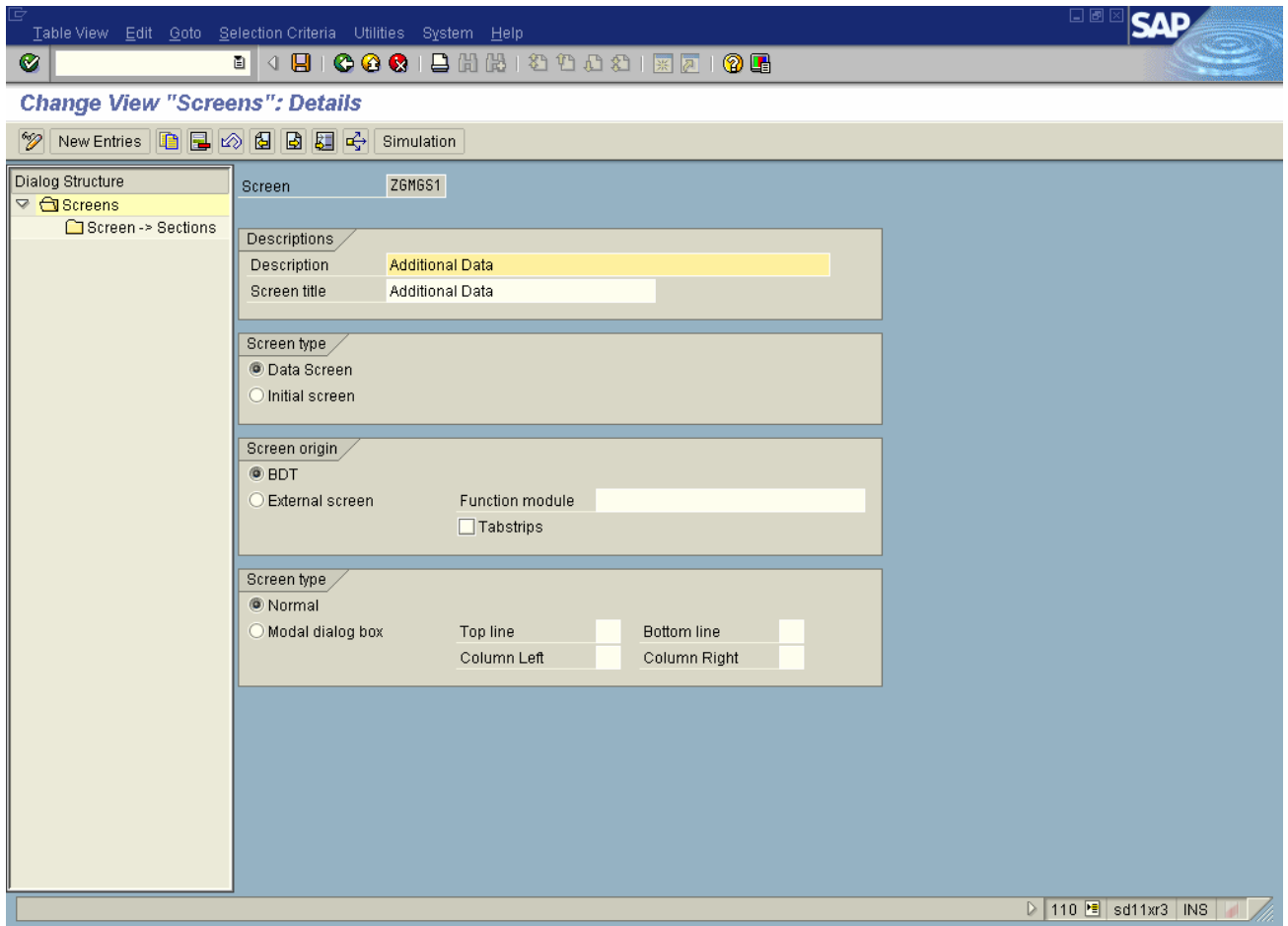


Once section is saved, hit the Section->Views button in the left frame. Data entry screen opens up in the right frame. Enter the view ZGMGPD in there along with Item number as 999991. This number is significant if there is more than one view in the section. If more than one view exists in the section, then the views are displayed as per the item number.

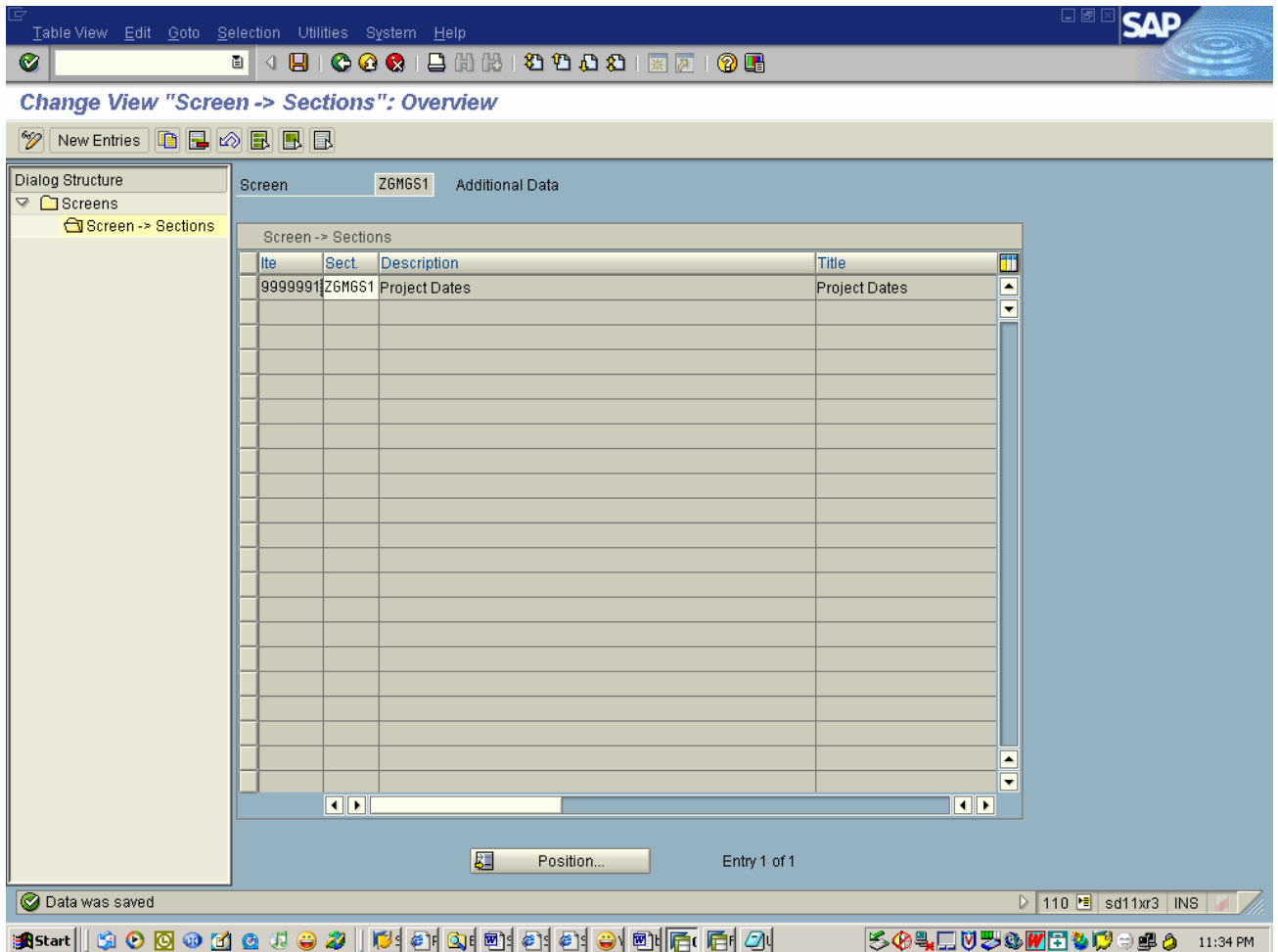


Save the section and come out of the transaction. This finishes the creation of section.

7) Next step is to create a screen. Create screen named ZGMGS1 (this name can be any thing) using the transaction GMS5 (Grant Master BDT Customizing -> Screen layout -> 'Screens'). Enter Additional data as description and Title. Value entered in title here will be used as the tab heading in the final transaction display.



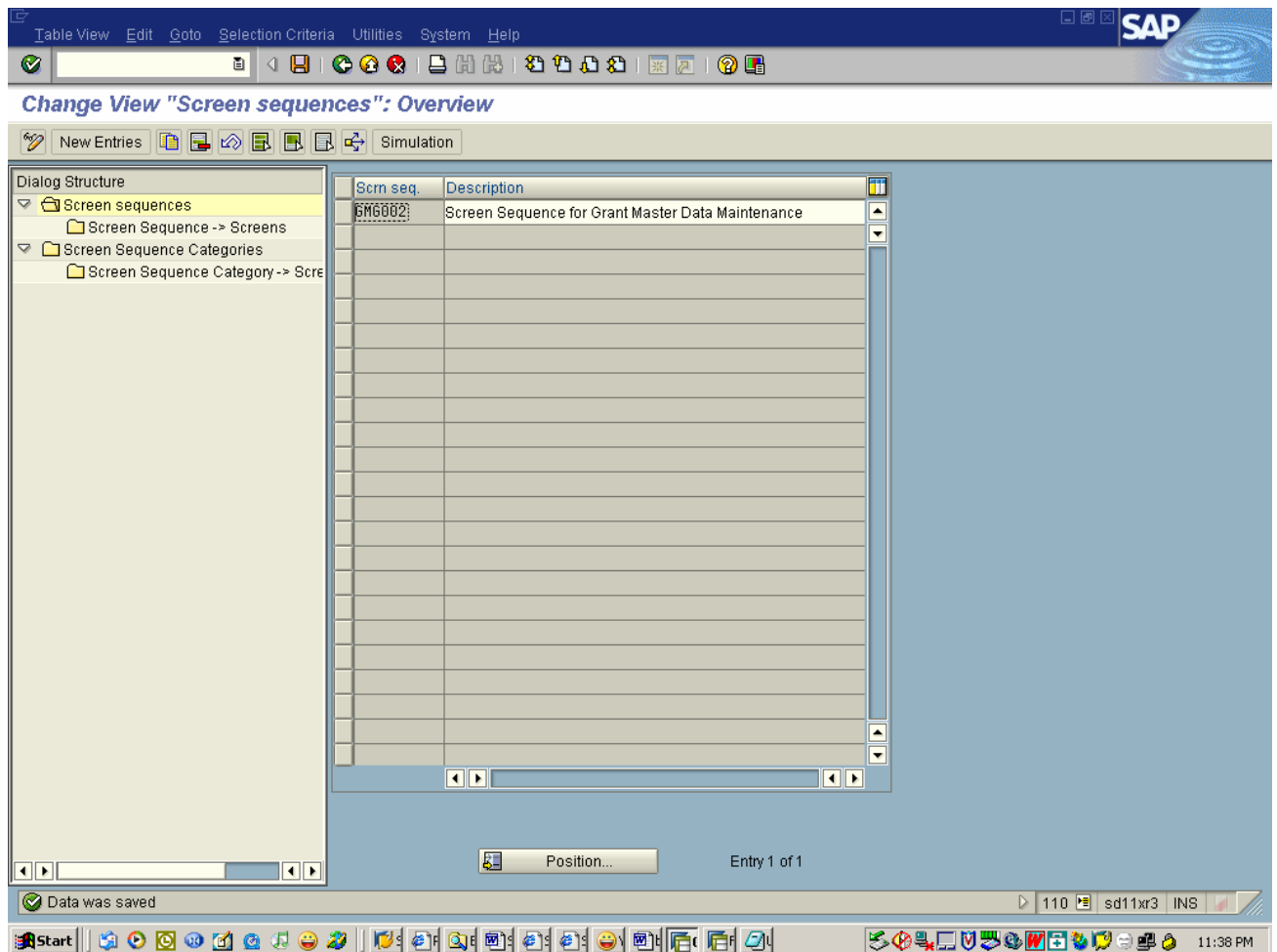
Now the screen is created, associate the section created in the prior screen to this screen. For this click the Screen->Sections button on the left frame. This opens up the Data entry screen in the right frame. Enter the Item Number and Section Name as entered in the view section. The Item number has same significance as in view case.



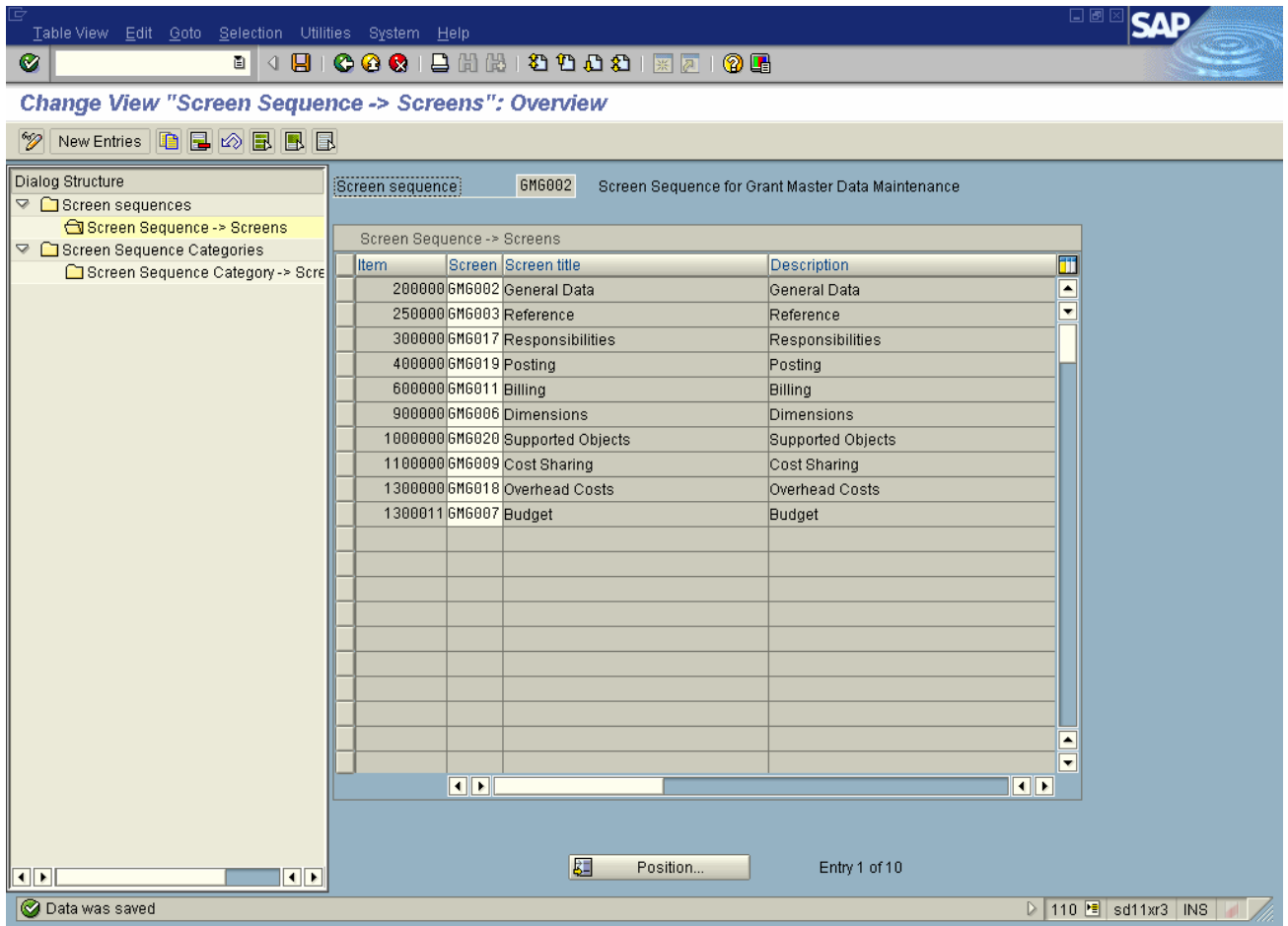
Save the screen and come out of the transaction. This finishes the screen layout design.

7) Now return the BDT customizing screen and select the screen sequence transaction GMS6(Grant Master BDT Customizing-> Screen Sequence). This is the place where we include the custom screen in the GMGRANT transaction.





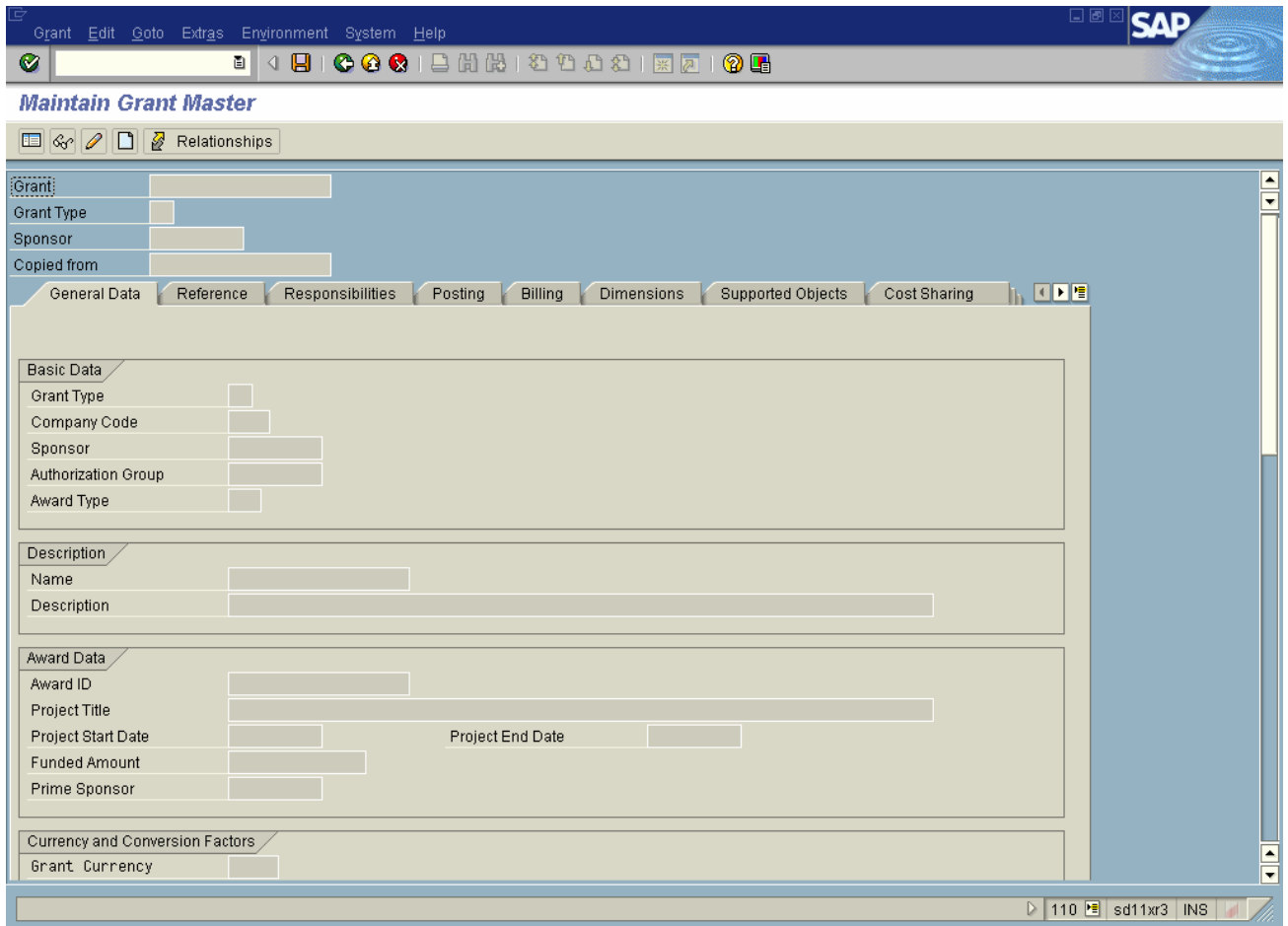
A default screen sequence will be defined already, select that and Hit the screen Sequence->Screens button on the left frame. This will display the list of existing screens in the GMGRANT transaction.



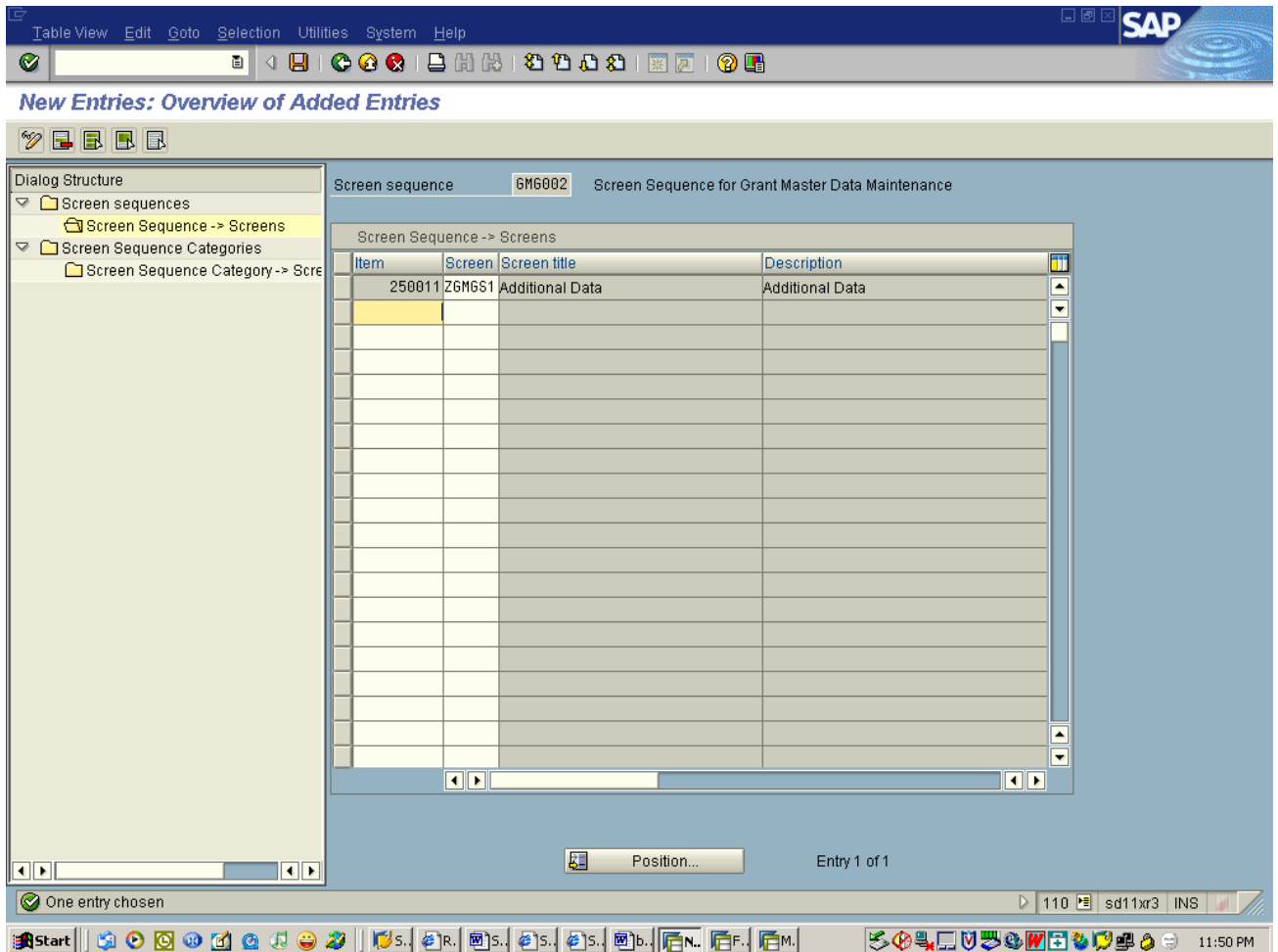
The screenshot shows the SAP Business Data Toolset for Dialog Programs interface. The main window displays a table titled "Screen Sequence -> Screens" for the screen sequence "GM6002". The table lists 10 screens with their respective item numbers, screen numbers, titles, and descriptions. The status bar at the bottom indicates "Entry 1 of 10" and "Data was saved".

Item	Screen	Screen title	Description
200000	GM6002	General Data	General Data
250000	GM6003	Reference	Reference
300000	GM6017	Responsibilities	Responsibilities
400000	GM6019	Posting	Posting
600000	GM6011	Billing	Billing
900000	GM6006	Dimensions	Dimensions
1000000	GM6020	Supported Objects	Supported Objects
1100000	GM6009	Cost Sharing	Cost Sharing
1300000	GM6018	Overhead Costs	Overhead Costs
1300011	GM6007	Budget	Budget

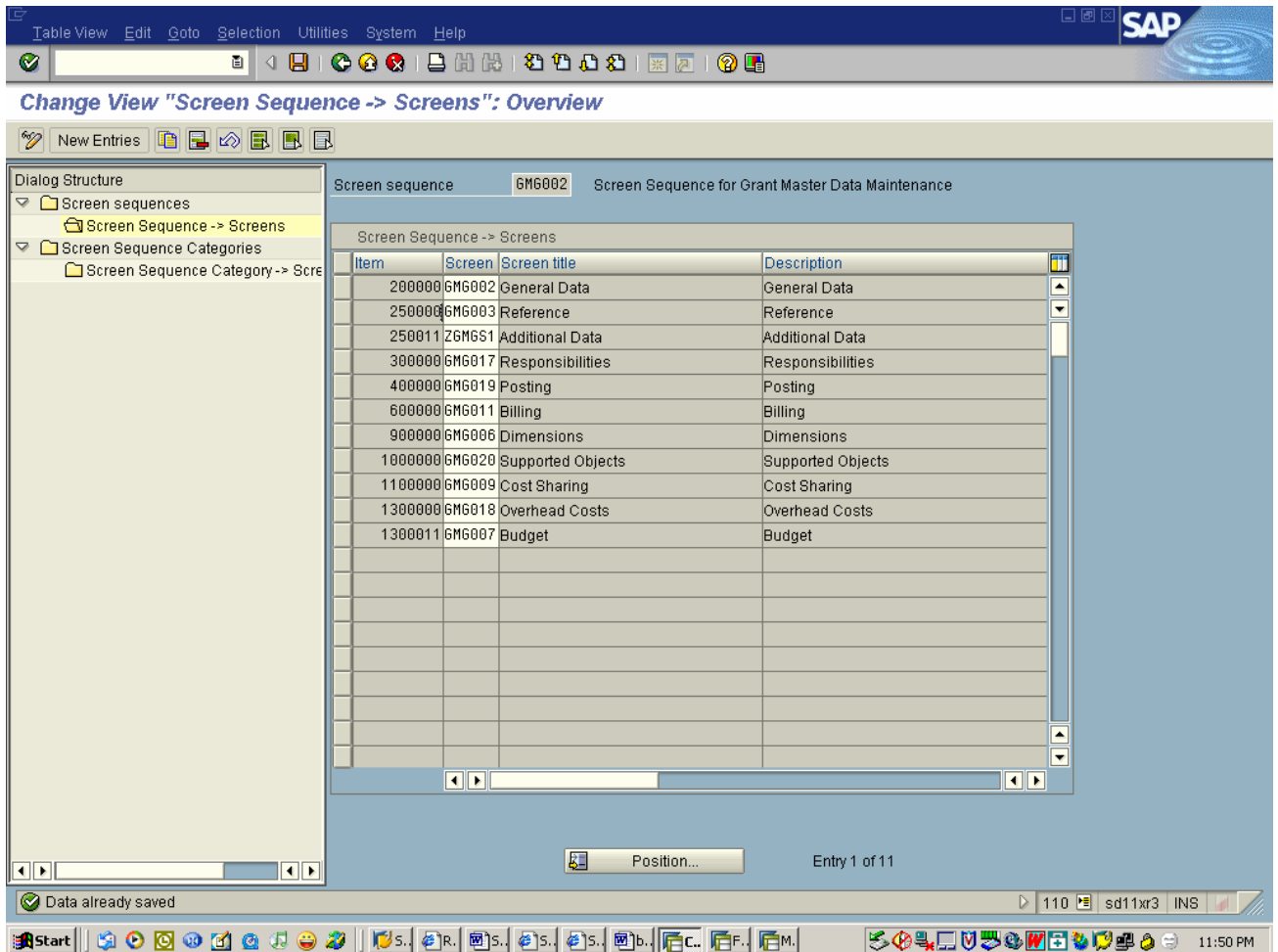
There are total of 10 screens in this table and each screen in this table corresponds one to one to the tabs on GMGRANT transaction as shown below –



Now add our custom screen to the screen sequence. As in the case of Views, Sections and Screens, Screen sequence also contains two columns, one for Item Number and Another for Screen Number. Item number determines the order of screens displayed in the final layout. Let's add our custom screen after 2<sup>nd</sup> Tab "**Reference**". For this Item Number should be more than the Item Number of Reference tab (250000) and less than Item number of Responsibilities tab (300000). So create the custom entry with Item number 250011. Hit the new entries button and add new entry.

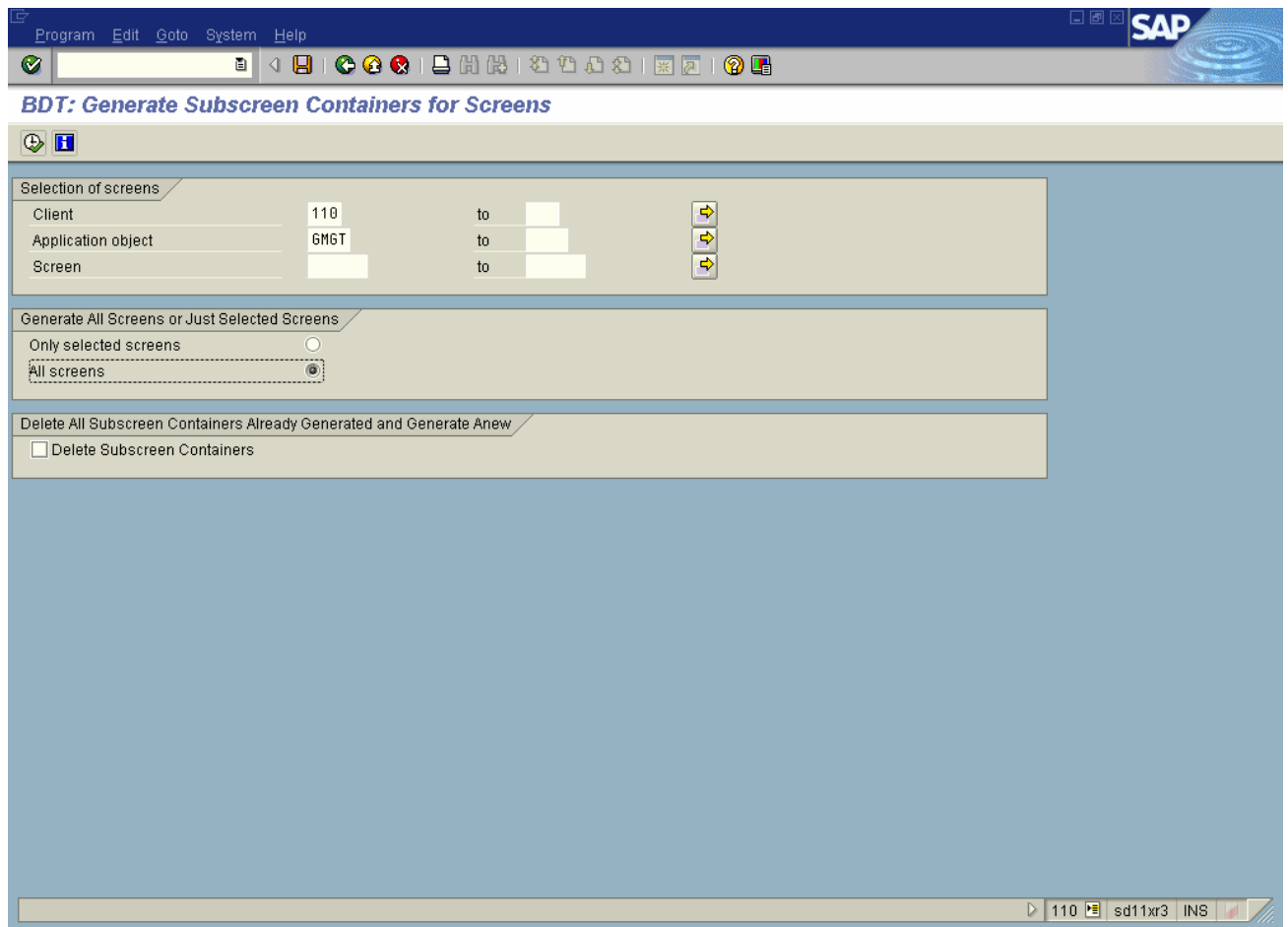


Save the entry and Hit Back button. This shows the new custom screen in the screen sequence and its placed after Reference tab.



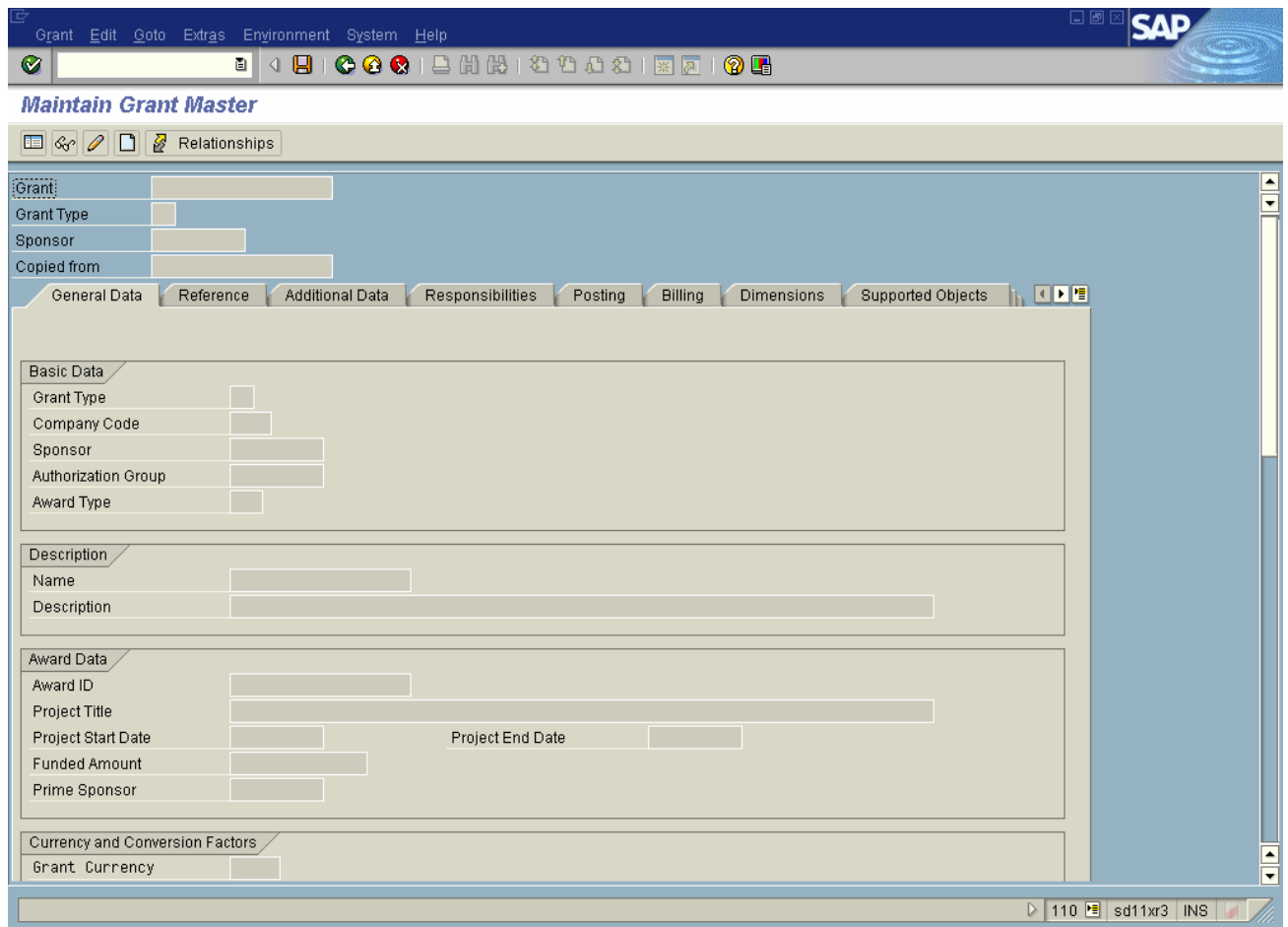
This finishes the screen design part. Now exit out of the screen sequence transaction.

8) Now return the BDT customizing screen and execute the transaction BUSP((Grant Master BDT Customizing->Generate Screen Containers). This program adjusts the subscreen containers assigned to the BDT screen.

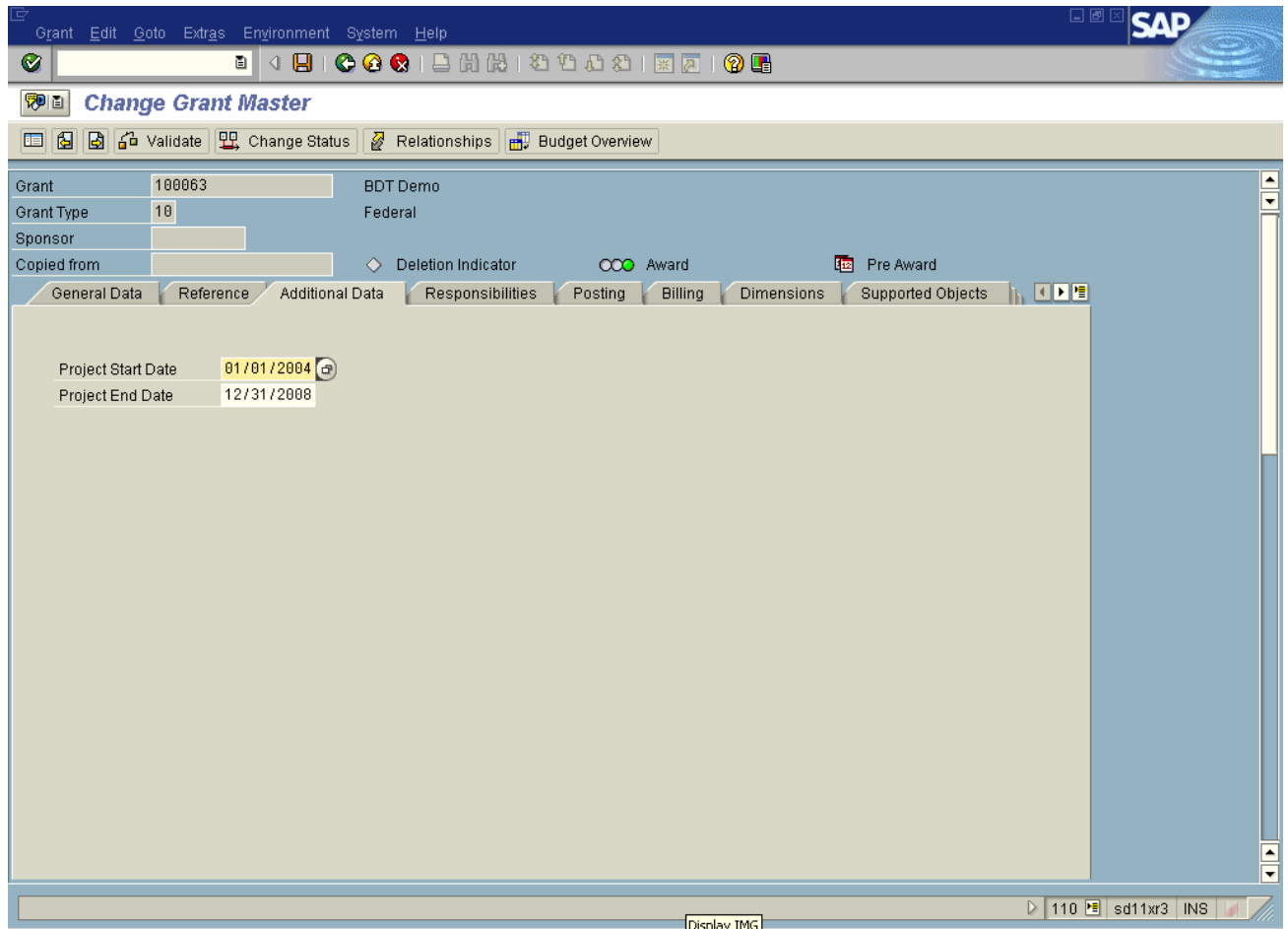


Select Application Object as GMGT, Check All screens radio button and execute the program. Once program run without any errors, we are done with the development and configuration of the screens.

9) Now execute the GMGRANT transaction. The “Additional Data” tab comes after the Reference tab.



If the Additional Data tab is selected, the two project date fields are displayed as shown below –

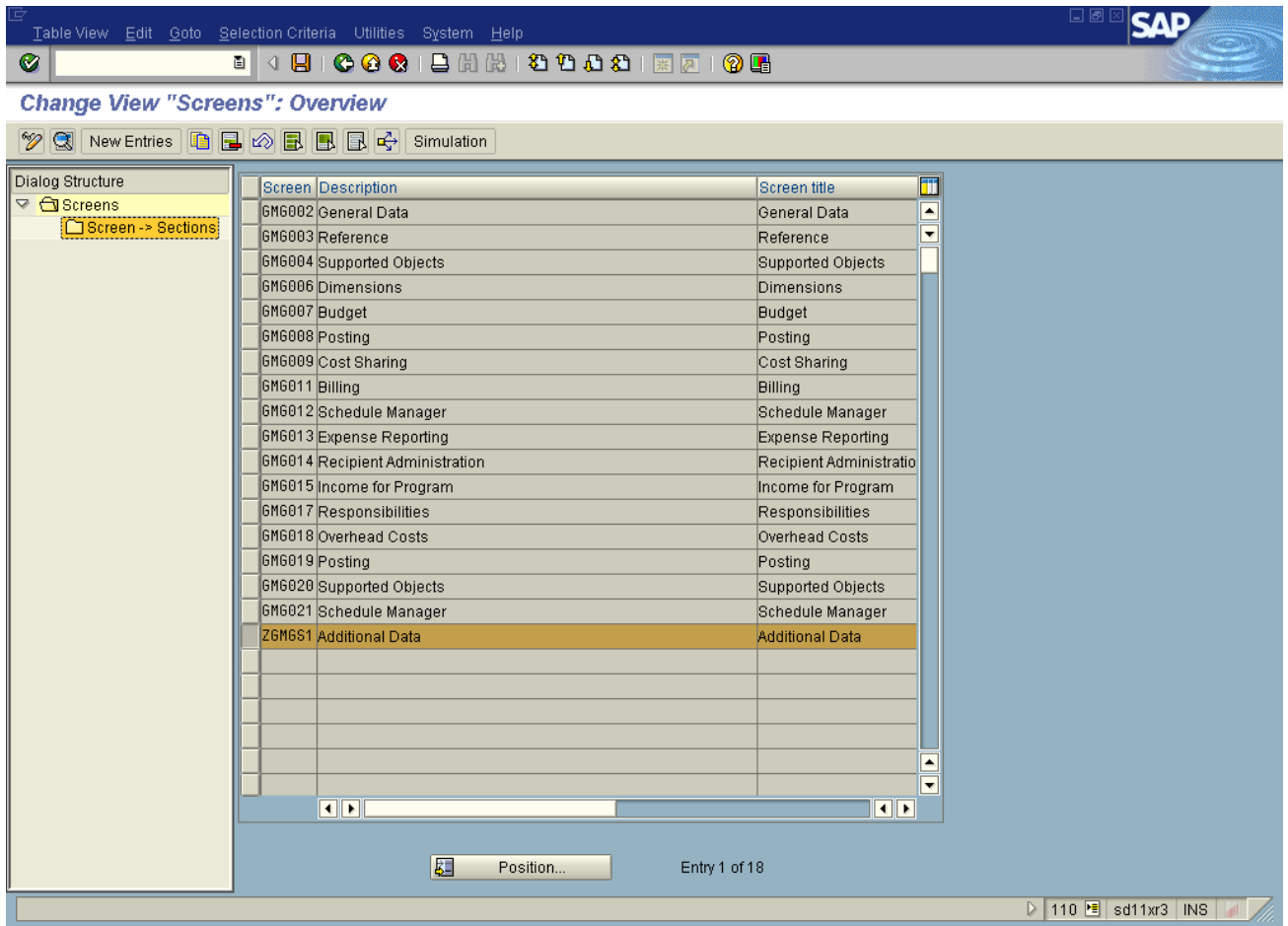


Adding custom fields is as simple as this. If the requirement is to hide Posting tab, all that needs to be done is removing the Posting Screen from the screen sequence.

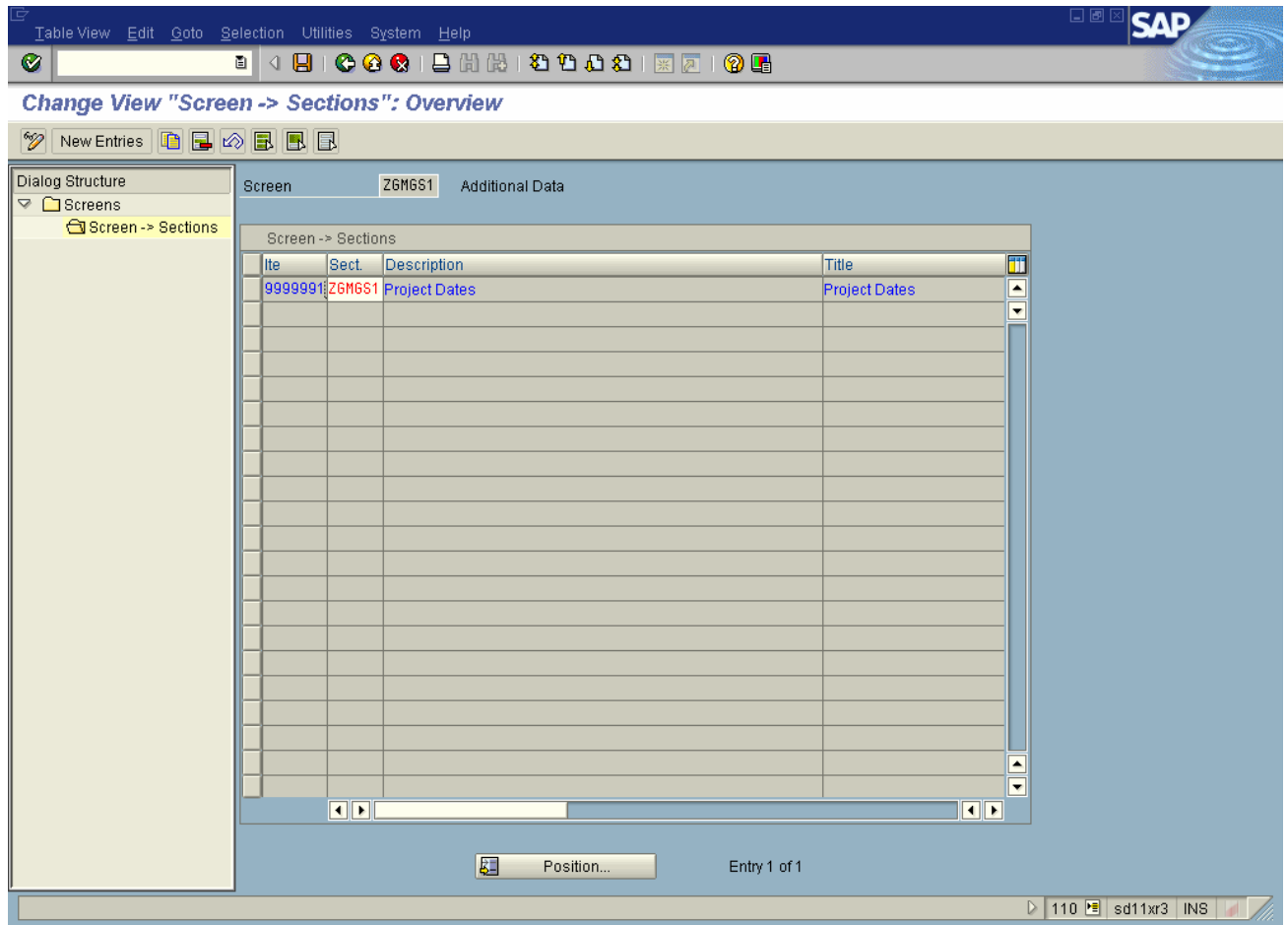
10) If desired output is to include these custom fields in the General data tab screen rather than in the custom tab, all that needs to be done is move the custom section ZSMGS1 from the custom screen to the General data screen. Since this information is maintained in control tables, BDT will adjust the display as per the configuration. This is explained below –

Return to BDT customizing area menu GRANT\_CUSTOMIZE and execute transaction GMS5(Grant Master BDT Customizing-> Screen layout -> 'Screens'). Here select the custom screen ZGMGS1 and Hit the Screen->Sections folder on the left frame.



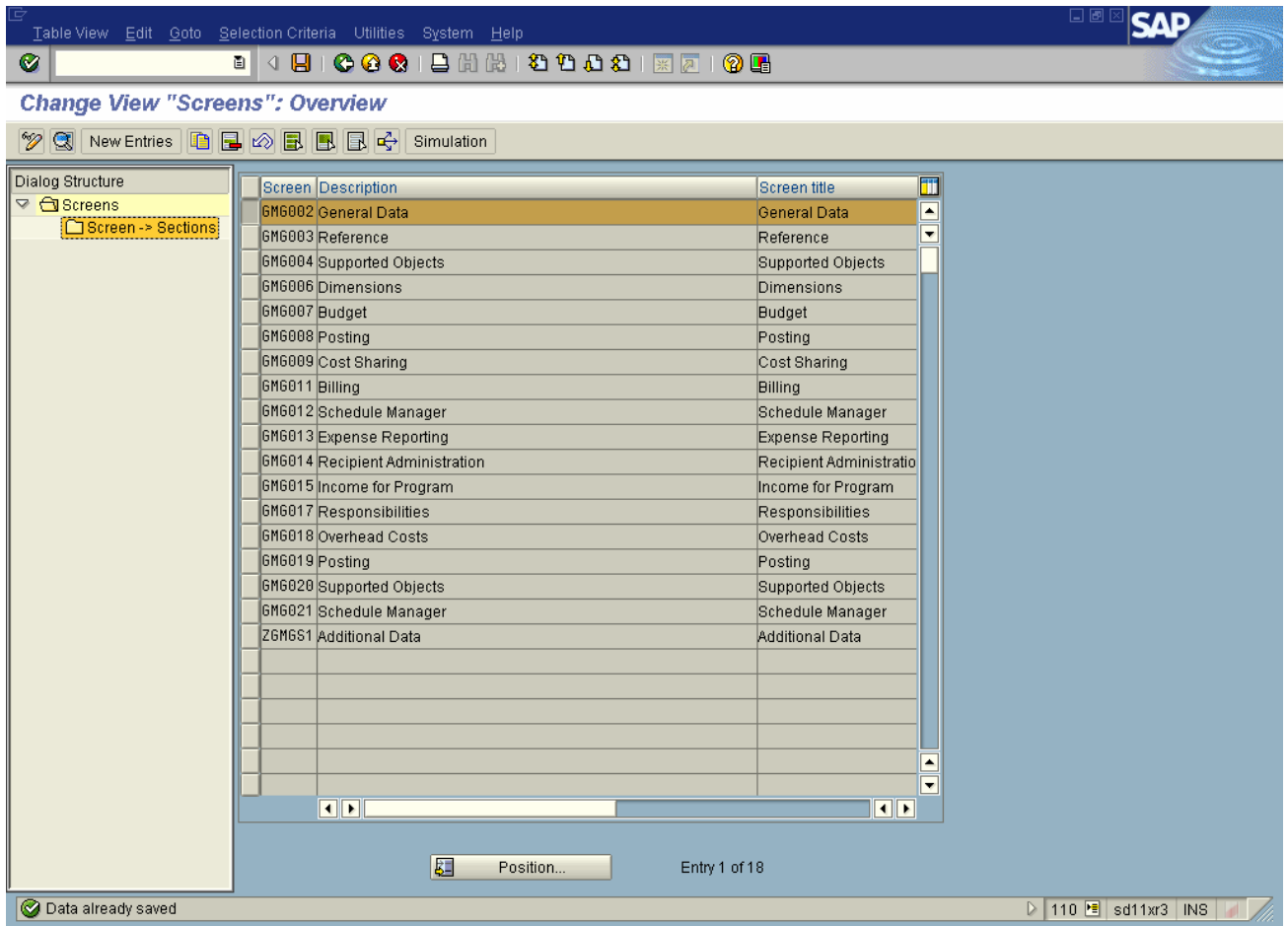


This displays sections information in ZGMGS1 screen as shown below –

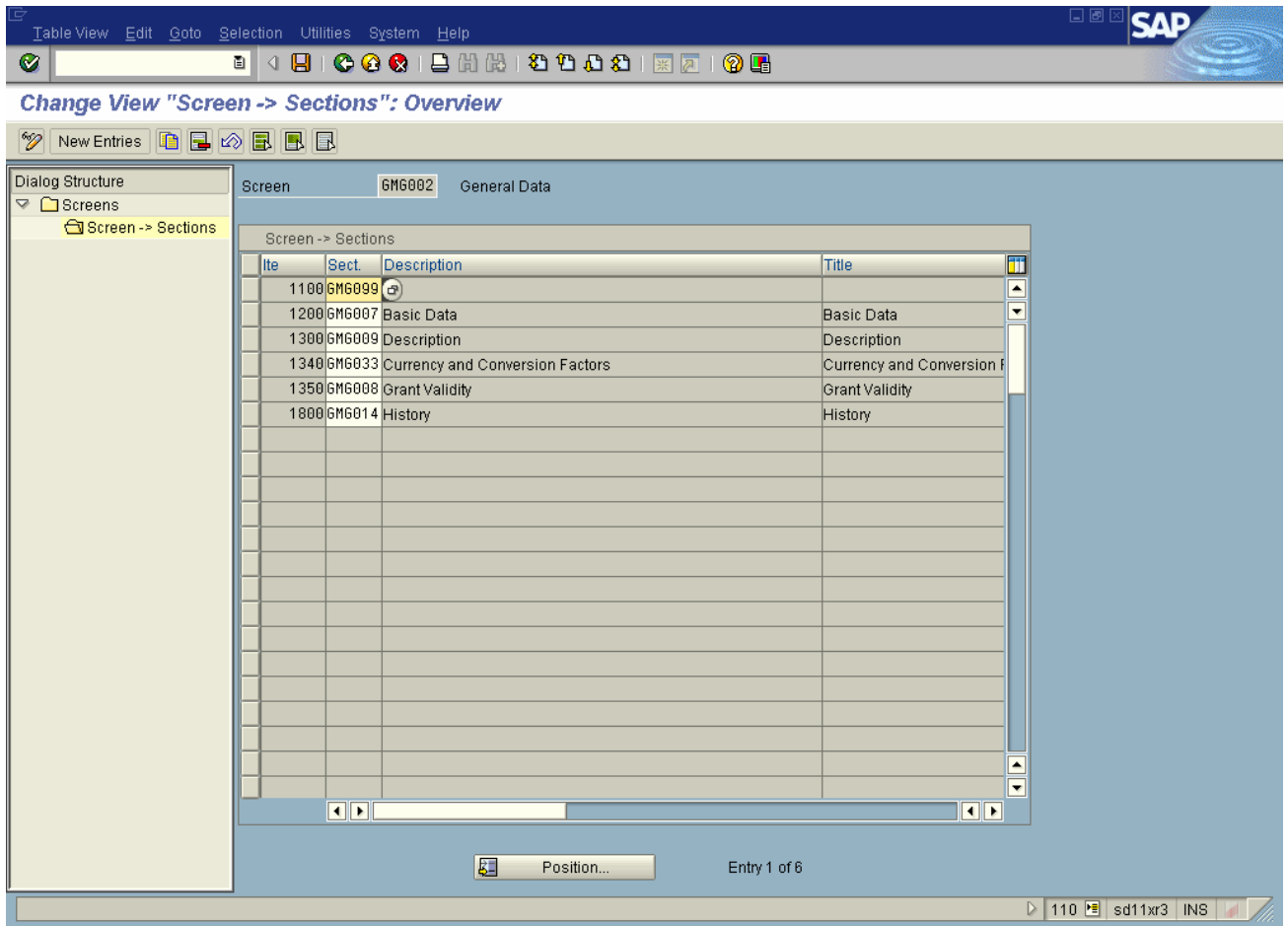


Here select the section in the right frame and Hit the Delete Row button and save the screen. This will remove the custom section ZGMGS1 from custom screen. Now Hit back button and come out of screen ZGMGS1.

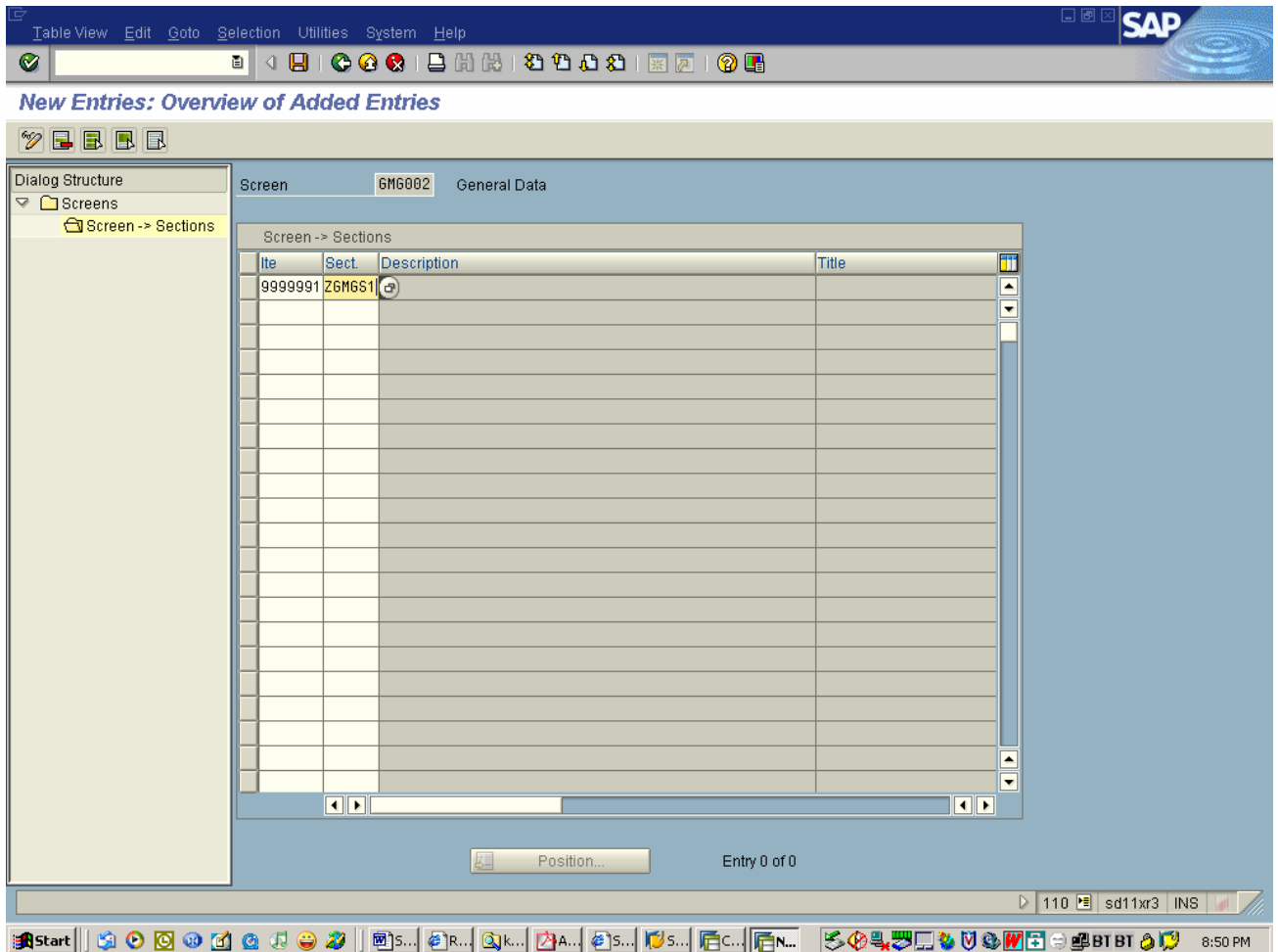
Since we want to include our custom fields in General Data tab, next step is to add the custom section to General Data Screen GMG02. For this select GMG02 screen and Hit the Screens->Section button on the left frame.



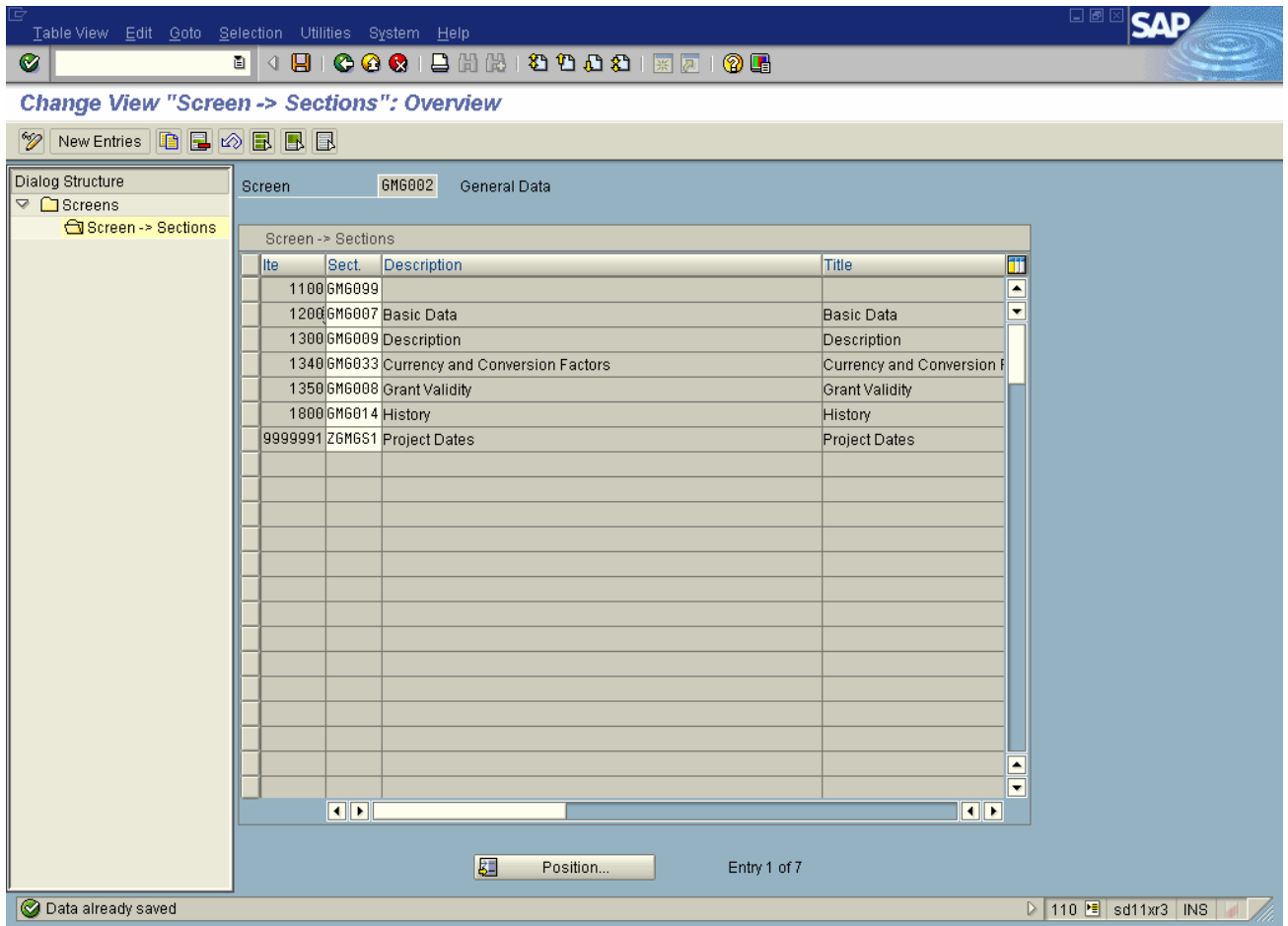
This will display the list of sections in General Data tab as shown below.



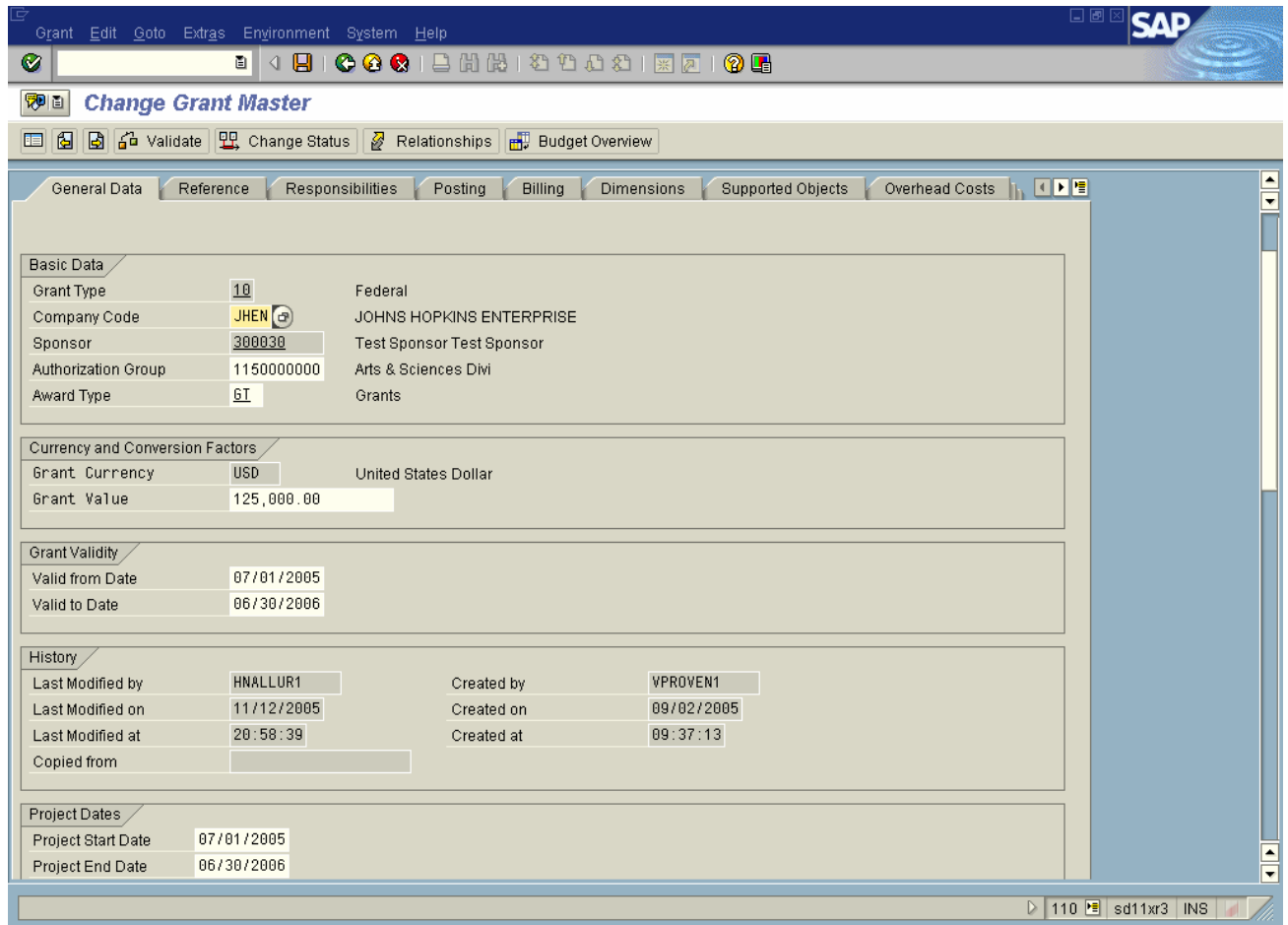
Hit the New Entries button. This will give us option to enter new sections in the right frame. Enter the custom section as shown below and save.



Now the General Data screen contains our custom section and the entries will look like below. Save the screen and return back to the area menu.



Execute transaction BUSP((Grant Master BDT Customizing->Generate Screen Containers) as done before. Now execute GMGRANT transaction again. This will display our custom fields in the General tab as shown below. Additional Tab disappears as there are no fields to display in the screen.



Just like we moved the custom section from one screen to another screen, same can be done for any standard sections as well. For example, History section can be moved from General data to Reference Tab by removing History section from General Data screen and adding it Reference Screen.

## Author Bio

Hari Krishna Nalluri is an SAP consultant working for Intelligroup, Inc. with a total industry experience of 5 years. He has recently been involved in Grants Management module implementation.