**AIF system architecture generic design:**



**Proxy based AIF design with XML engine:**

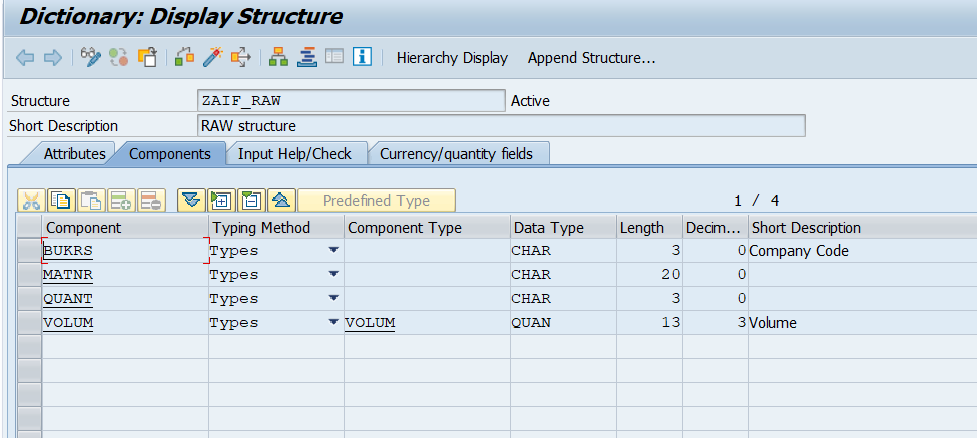
**Business Requirement:** Split the load from proxy into rational and logical loads and call AIF interface for each of the sub-loads.

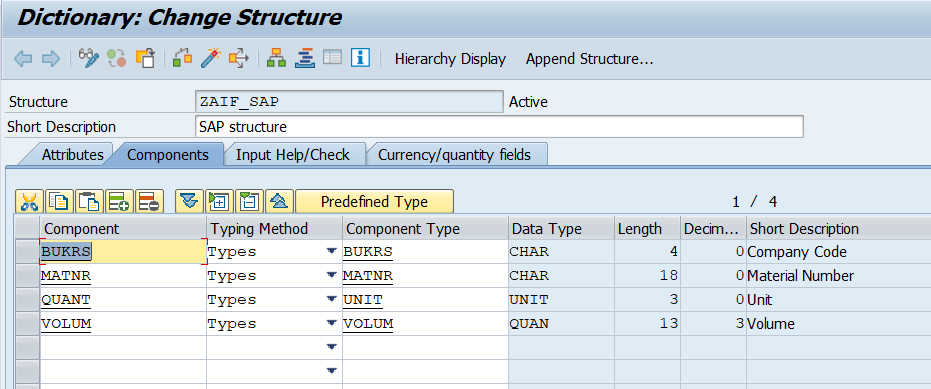
**Technical Design:** This requirement can be achieved only by calling a AIF interface with XML engine from proxy method.



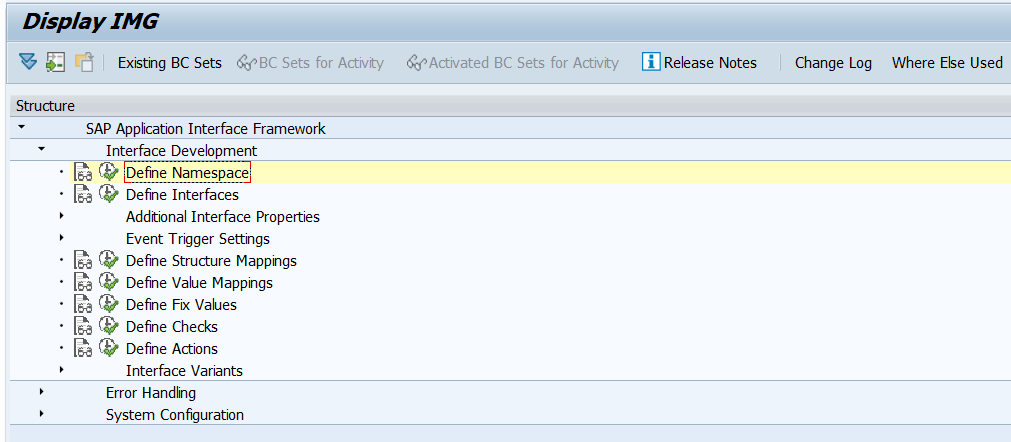
**Customizations:**

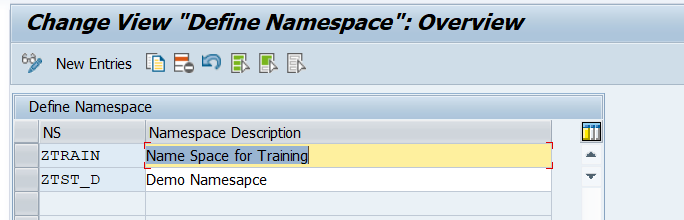
1. **Create RAW and SAP Structures.**



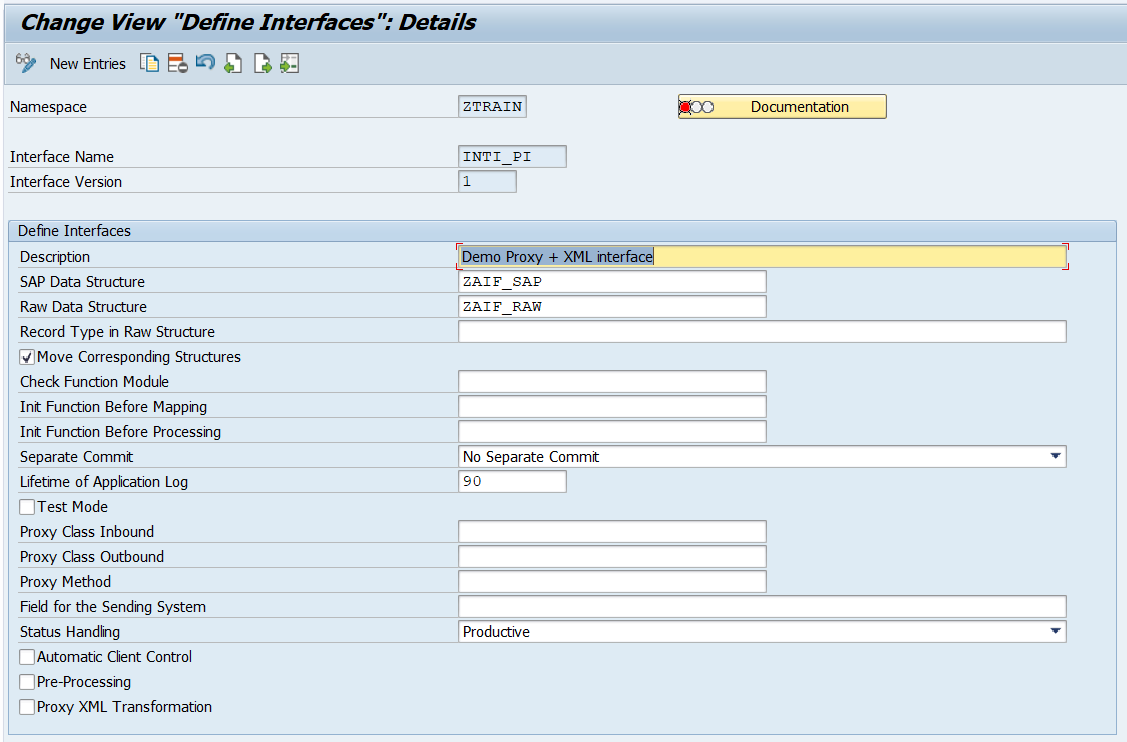


1. **Define Namespace:**



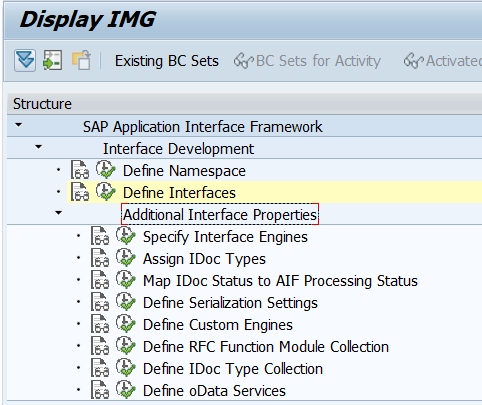


1. **Define Interface.**

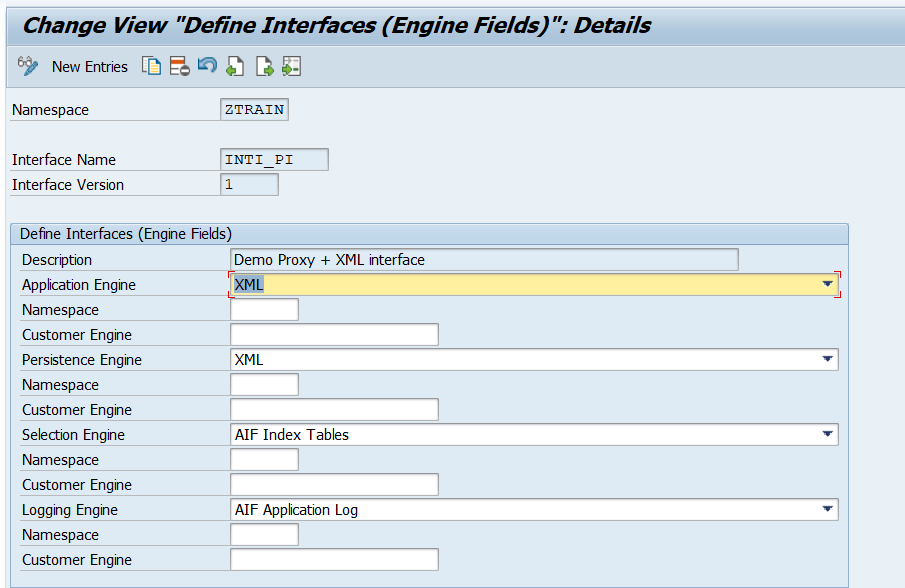


**Note: Here, we have assigned RAW and SAP structures to the respective place.**

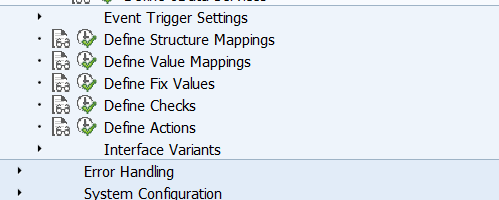
1. **Assign additional properties:**

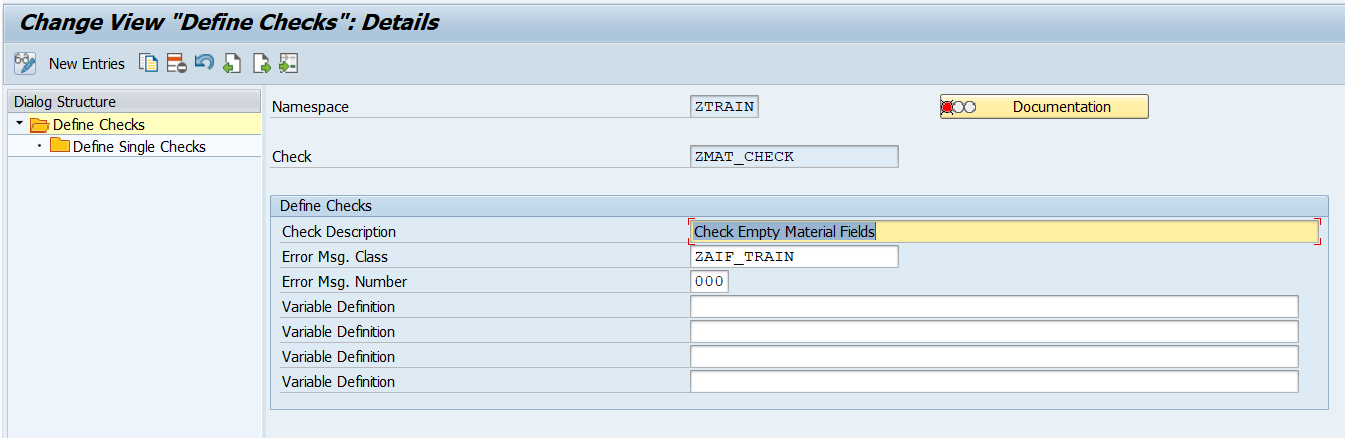


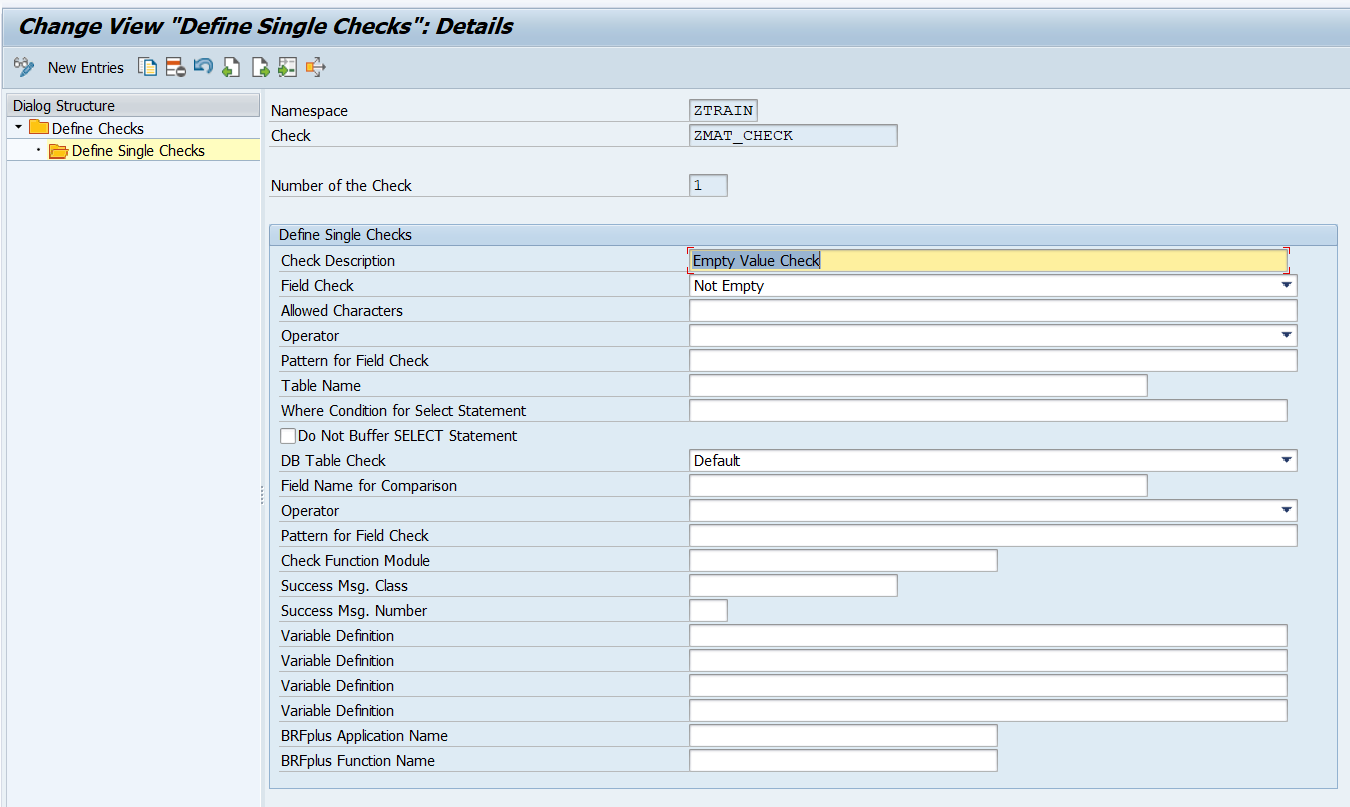
**Here, we specify interface engines like below.**



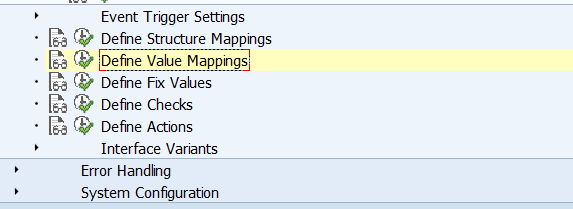
1. **Define Checks: In this section we create checks on RAW data or SAP data. Basically data validation is performed using checks. I have created one check to show the missing material number.**

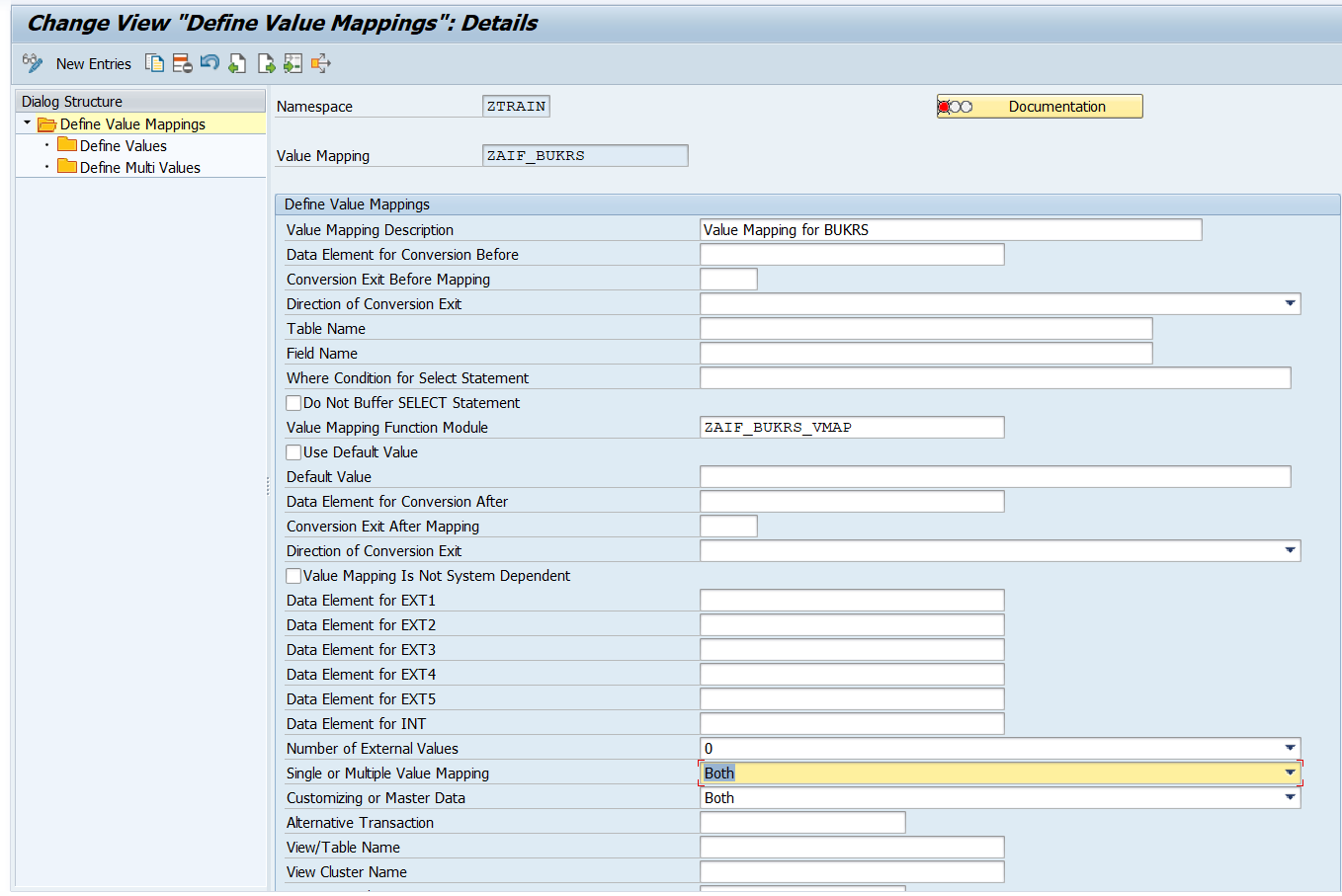






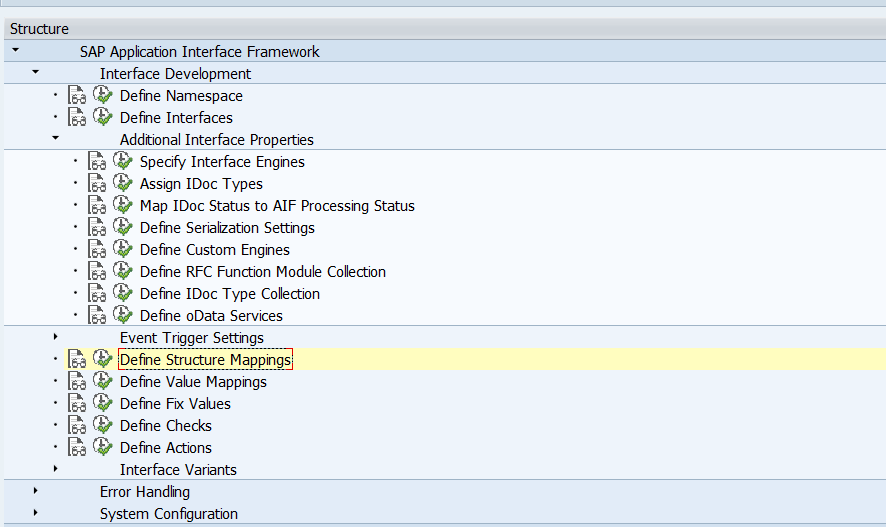
1. **Define Value mapping: This feature is used to map non sap values to SAP values either via conversion table or fixed values as per configuration.**

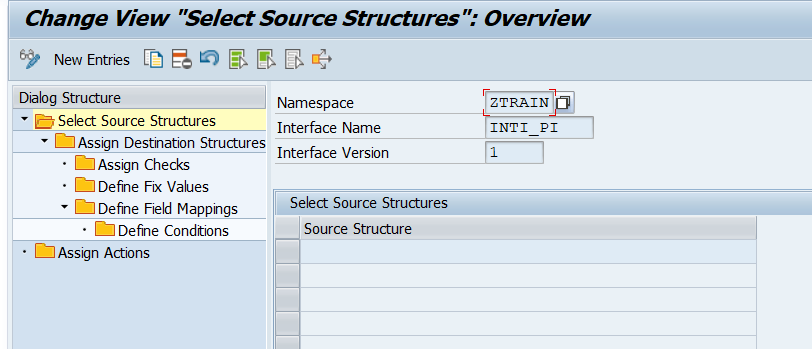


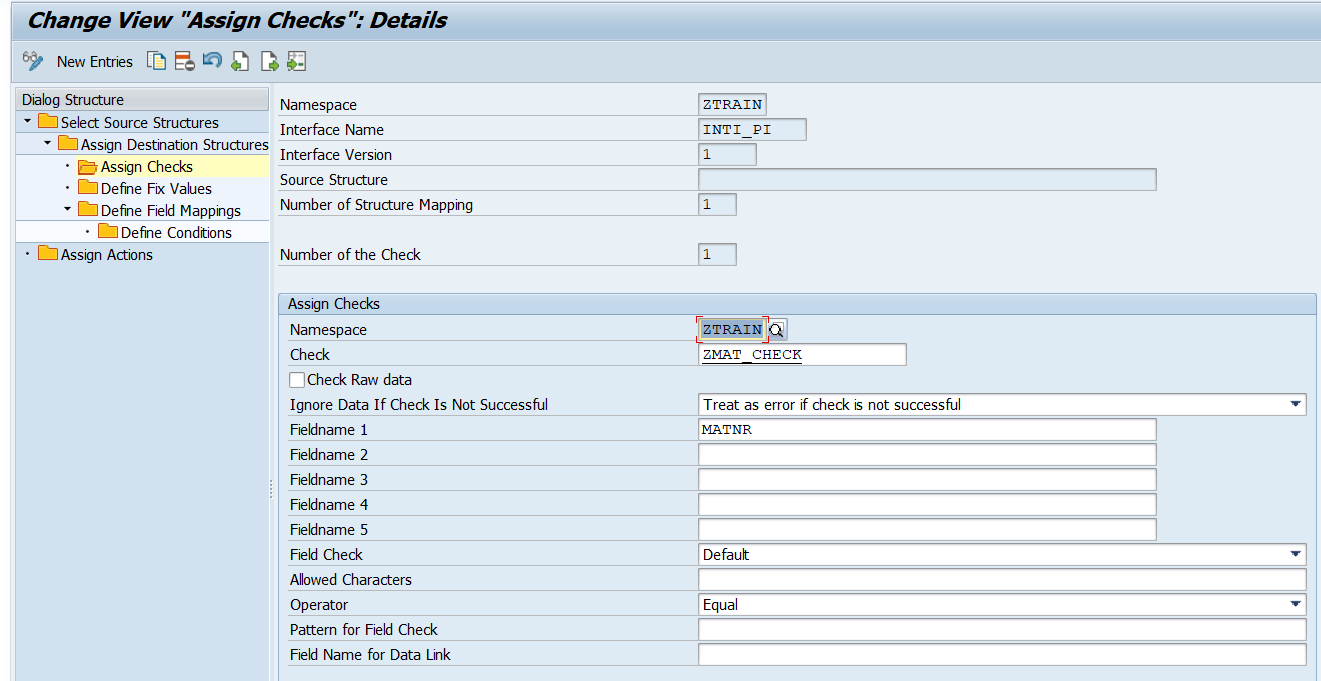


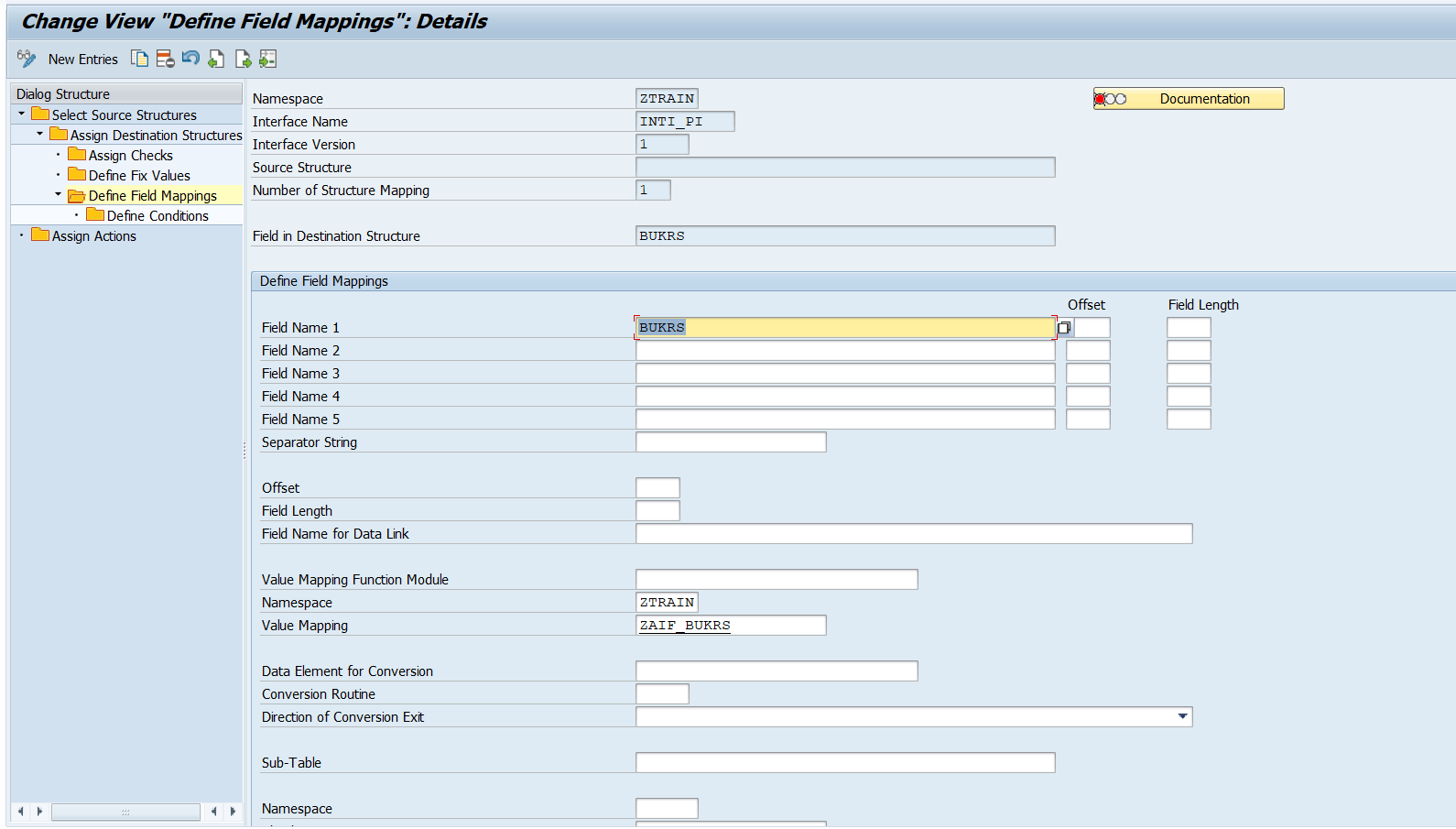
**Here, we have used a function module to perform the value mapping.**

1. **Define structure mapping: We map RAW and SAP structure value in this section. Also we assign value mapping and Checks in this section.**

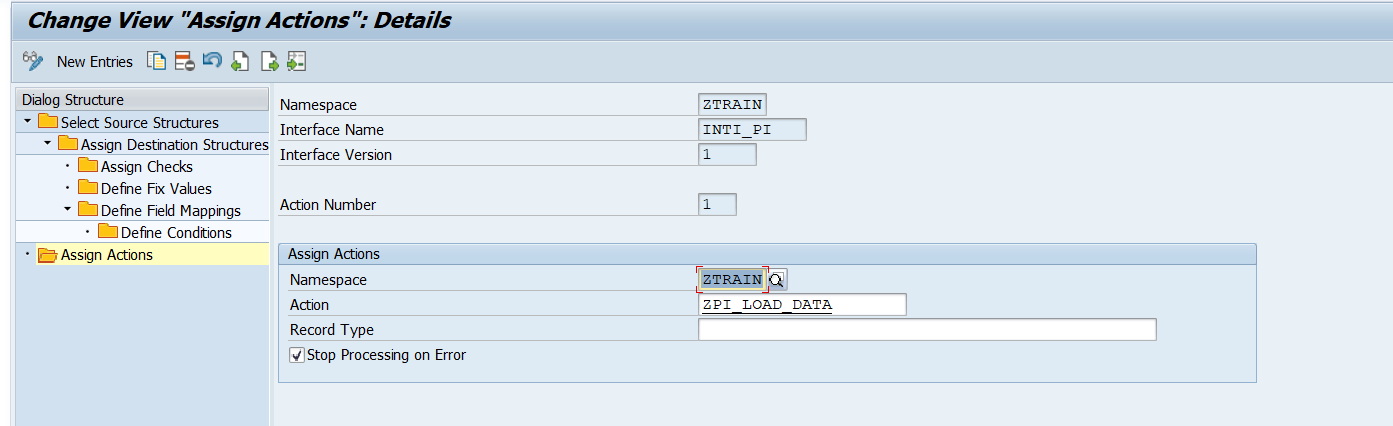


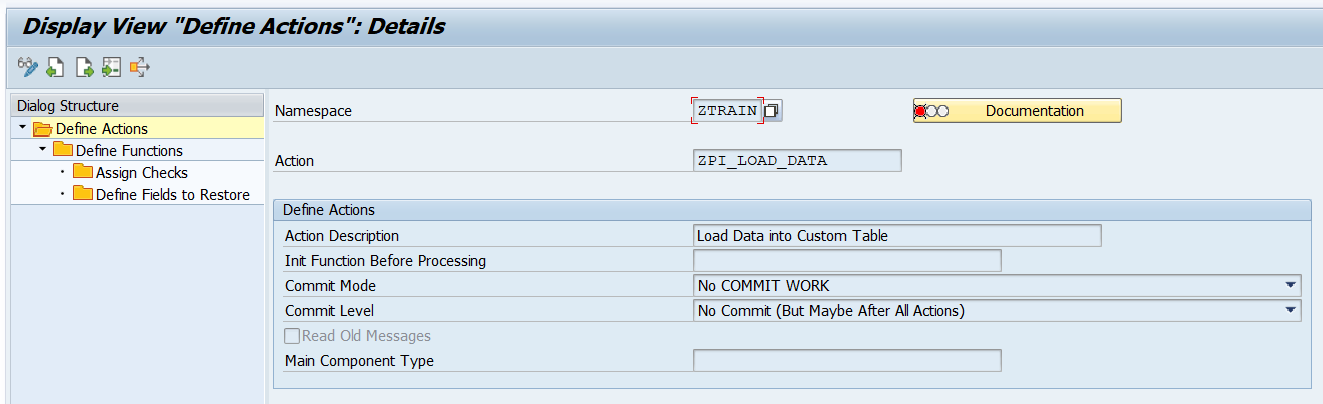


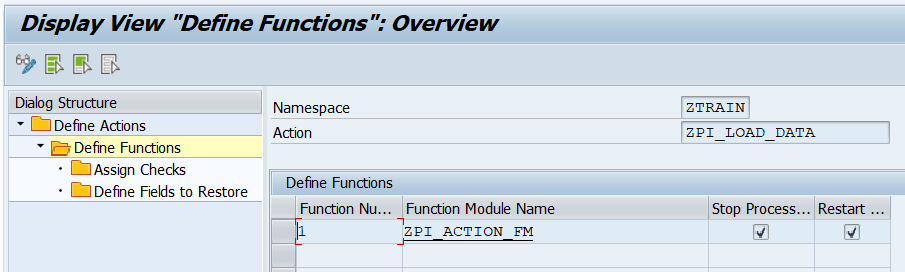




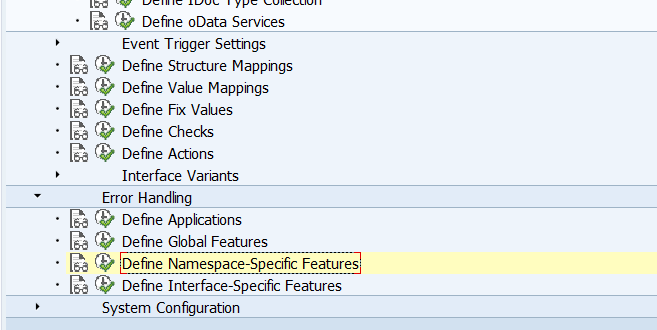
1. **Assign Action: AIF data processing logic is written inside action function module.**

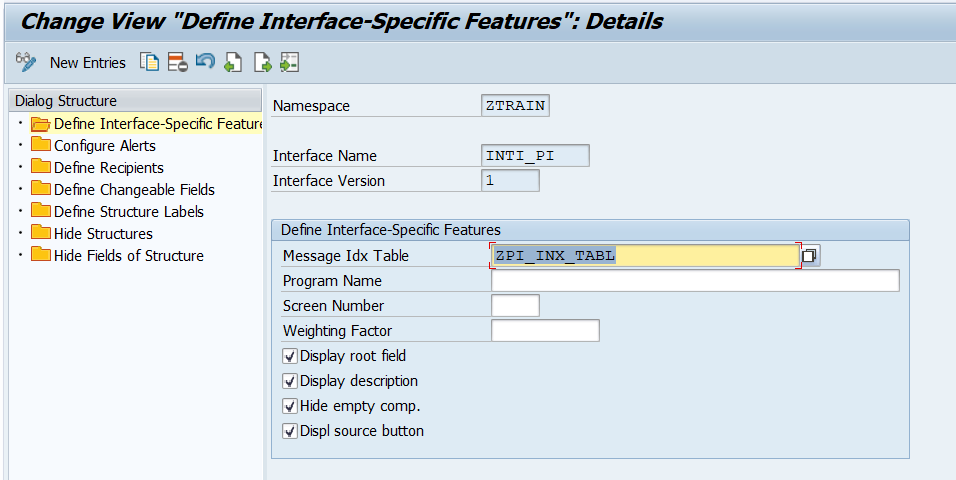




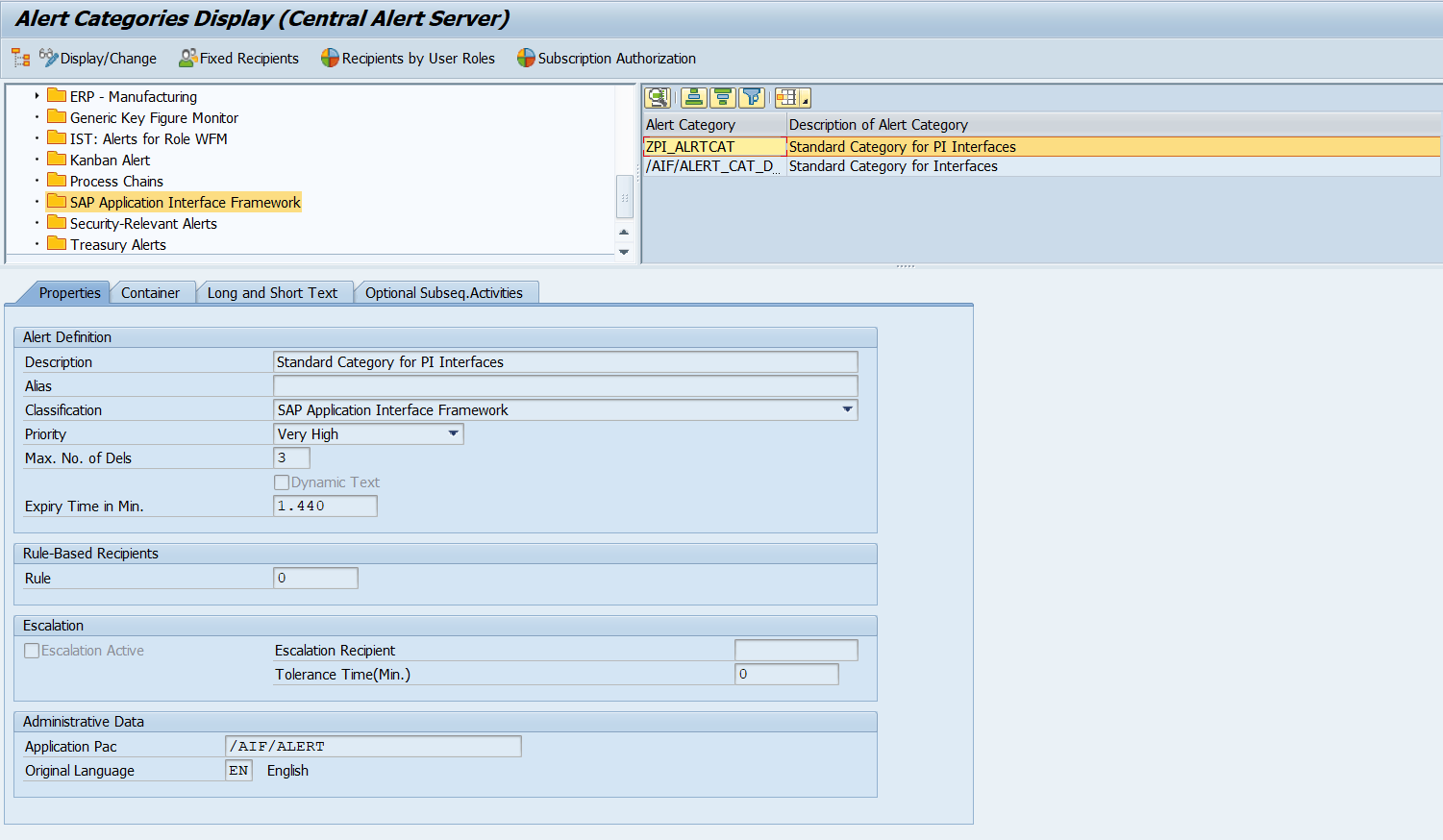


1. **Define index table: Ideally, for any custom AIF interface, we should create a separate index table to store data.**

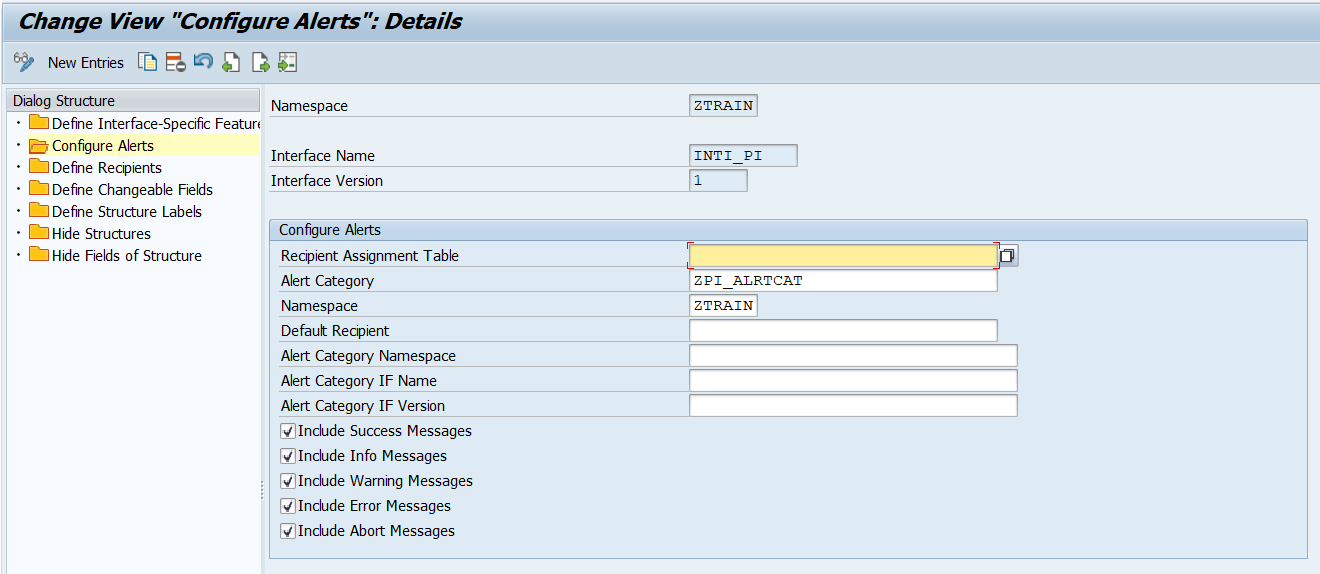




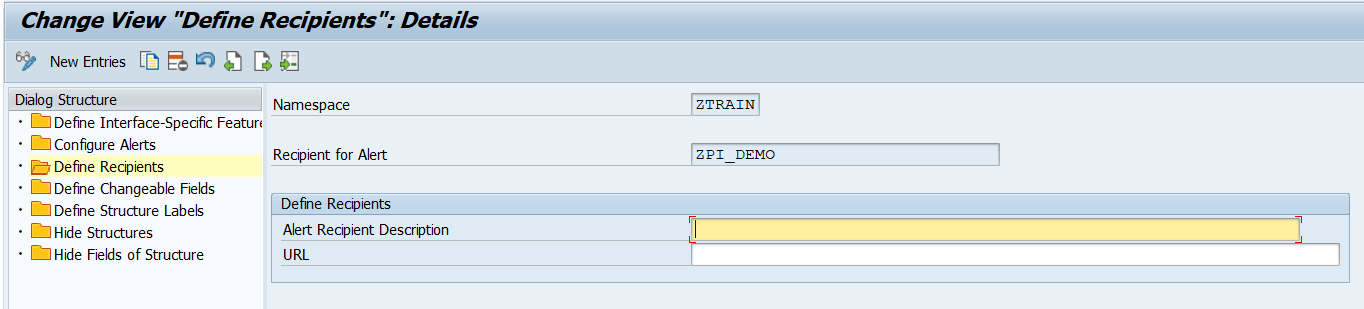
1. **Configure Alerts: In this section we create a alert category for the interface and assign to it.**
   1. **Transaction code /nalrtcatdef**



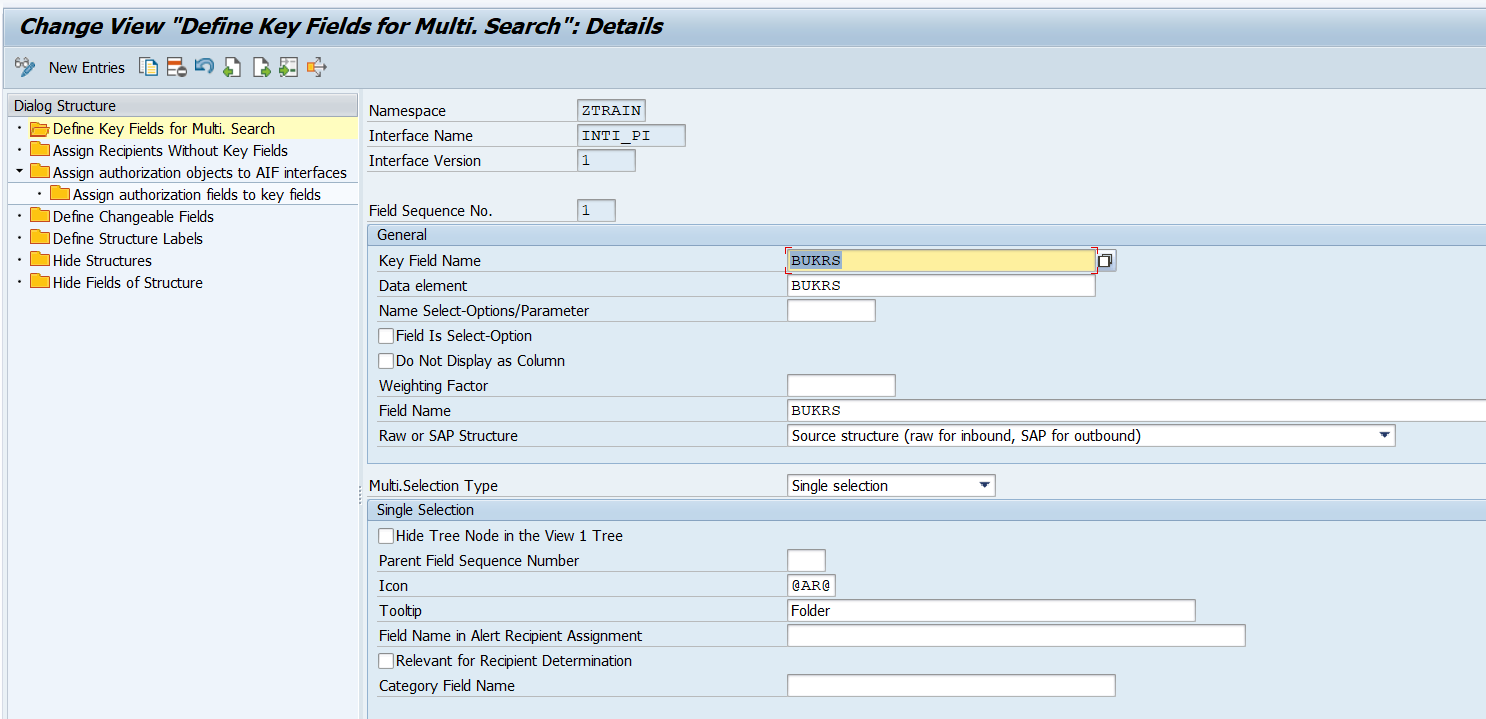
* 1. **Create new category and assign it to the interface.**



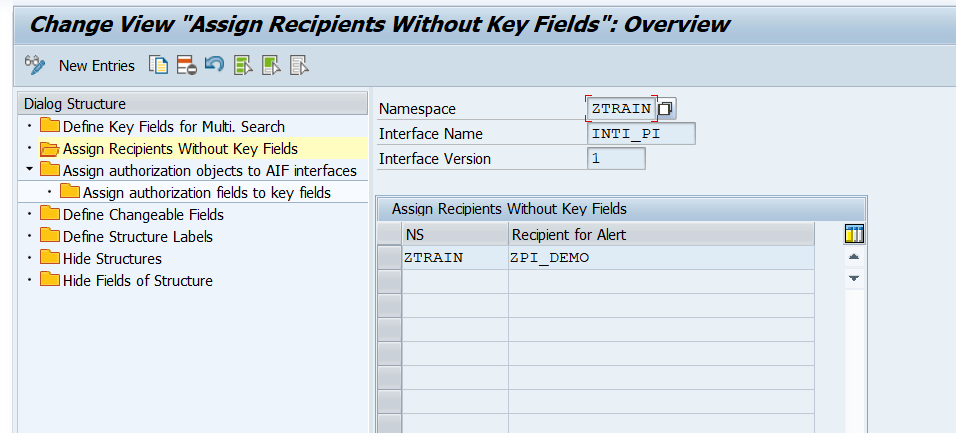
1. **Define Recipients: You need to complete this step if you want to error monitoring in interface monitor and to receive alerts.**



1. **Define Interface Specific features:**
   1. **Define Key fields for multi-search: Define key fields of your interface.**

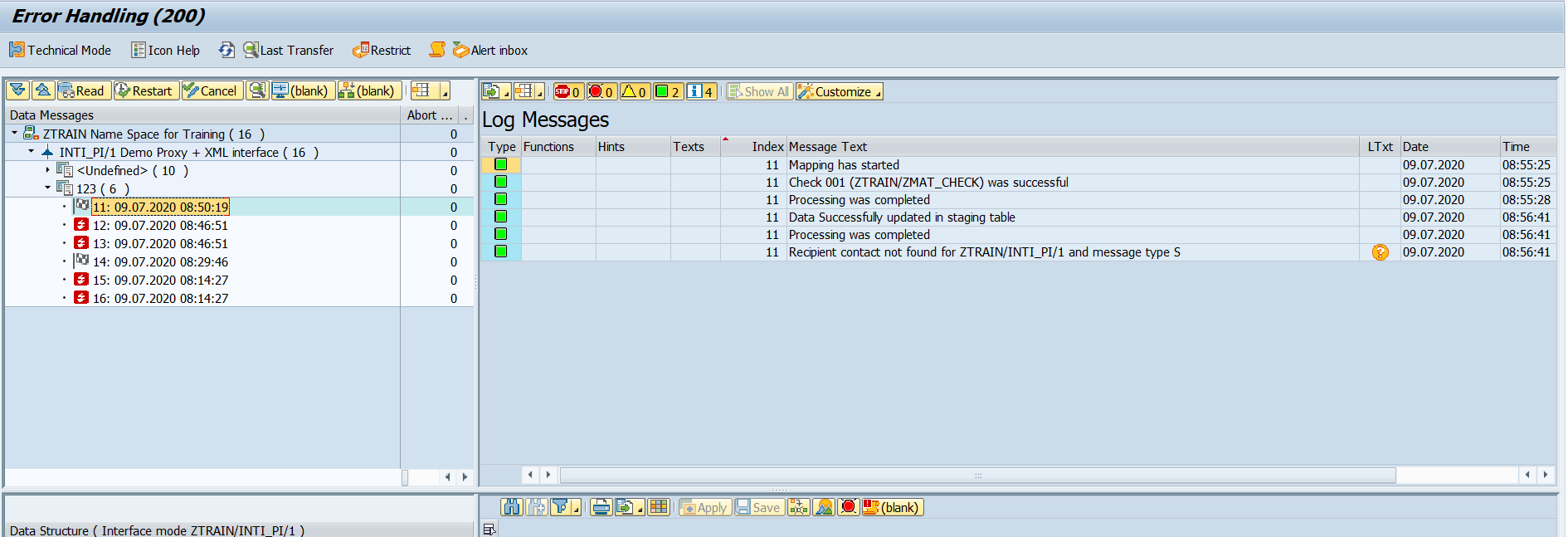


* 1. **Assign recipients without key fields.**



**Error Handling and monitoring**

**Transaction code /n/aif/err is used for Error monitoring.**



**Interface Monitor:**

**The transaction code for interface monitor is /AIF/IFMON.**

