Error occurs at "Activate inactive runtime objects" phase

Image that table ZEMPLOYEE exist in both development system and quality system.

In development system, you added a new field PHONENO into ZEMPLOYEE, and then activated it.

You created a transport request which included the new structure of table ZEMPLOYEE, and imported this request into the quality system.

However table ZEMPLOYEE was problem in the quality system. The runtime object was consistent with dictionary definition, but the database object was inconsistent with the runtime object (Field PHONENO was defined in the database object, but not in the dictionary definition or the runtime object. Such problems may happen if you changed the table directly in database level, but not from SE11):

So when the request was imported into the quality system, error occurred in "Activate inactive runtime objects" phase:
“ABAP Dictionary Distribution” phase always compares the current runtime object with the new structure, and then determine what operation should be done to the database object.

In this example, as there was no field PHONENO in current runtime object, so DDL statement “ALTER TABLE ZEMPLOYEE ADD (PHONENO VARCHAR2 (000033) DEFAULT '00000000000' NOT NULL)” was generated in “ABAP Dictionary Distribution” phase and was executed in “Activate inactive runtime objects” phase. But there was already a field named PHONENO defined in database table ZEMPLOYEE, so the DDL statement failed to be executed in this phase with error “column being added already exists”.

In this phase, both the database object and the runtime object are to be adjusted to the new structure. The database object is adjusted firstly. Only after database object is adjusted successfully, can the runtime object to be re-generated.

In this example, as the database object failed to be adjusted, the re-generation of the new runtime object could not be executed, either. As the result, the dictionary definition was on the new structure, but the runtime object and the database object remained unchanged.

If you performed consistency check on runtime object, the result would be the following:

![Check: Active runtime object for ZEMPLOYEE is Inconsistent](image)

You can not resolve the inconsistency by activating the table directly in SE11. Activation will fail with the same error.

What you should do is to firstly reconstruct the table, then activate the table using transaction SE11. Activation in SE11 will do all the transport jobs(activation, distribution, conversion, activate inactive runtime object) to the end. For details, please refer to [Inconsistency occurs on database object check](#).