Table Conversion - Frequently Asked Questions

Questions

Basics

- What is table conversion? When table conversion happens?
- How to check whether table conversion will be triggered or not before editing a table field?
- The conversion process for cluster table is the same as that for transparent table?
- Can I convert cluster table to be transparent?
- What will happen if I unlock the table from conversion in SE14? Can I unlock a table from conversion?
- How to check table entries during BSEG conversion?
- How to do SPDD adjustment for BSEG during upgrade (or EHP installation)?
- What kind of changes on cluster tables such as BSEG can trigger table conversion?

Trouble Shootings

- We noticed that a new table QCMBSEG was created after updating support package stacks. Why this table exists in our database and to solve this issue?
- Dump CONVT_NO_NUMBER occurred during table conversion in step 5. How to continue the conversion?
- Dump BCD_FIELD_OVERFLOW occurred during table conversion in step 5. How to continue the conversion?
- Table conversion for table BSEG took very long time during upgrade. How to do?

Answers

Basics

What is table conversion? When table conversion happens?
If you change ABAP dictionary definition in SE11 (or if ABAP dictionary definition is changed by transport), database structure of the table is adjusted to the change in the ABAP Dictionary during activation.

The database structure of a table can be adjusted to its changed ABAP Dictionary definition in three ways:
- By deleting and recreating the database table
- If the table does not contain any data, the existing table is deleted in the database and recreated.
  - By changing the database catalog (ALTER TABLE)
- If there is data in the table, the system tries to change the structure using ALTER TABLE.
  - By converting the table
- If the database system cannot execute the structural change with ALTER TABLE, the table is converted.

So generally when there is data in table and database system cannot execute the structural change with ALTER TABLE to update the database table, table conversion is triggered.

How to check whether table conversion will be triggered or not before editing a table field?
You can use report RUTATCHK to check whether table conversion will be triggered or not. Execute report RUTATCHK in transaction SE38, specify the data type of “before change” and “after change” and continue, if the change lead to table conversion, the result will show with “Conversion”. Otherwise, the corresponding SQL statement will be displayed instead. For details, you can refer to KBA 2465903 - Error DT064 “Structure change at field level (convert table ...)”.

The conversion process for cluster table is the same as that for transparent table?
NO. They are different.

Generally conversion process for transparent table TAB is as below:

Step 1: Set lock to against further structural changes
Step 2: Rename table TAB to QCMTAB in the database, delete all indexes, generate nametab for table QCMTAB

Step 3: Empty

Step 4: Activate table TAB, generate TAB nametab in new structure

Step 5: Create table QCM8TAB with the new structure, create primary index TAB~0 on table QCM8TAB, load data from QCMTAB to QCM8TAB, drop QCMTAB

Step 6: Rename table QCM8TAB to TAB in the database

Generally conversion process for cluster table TAB is as below:

Step 1: Set lock to against further structural changes

Step 2: Create transparent table QCMTAB with its new structure, load the relevant data from the corresponding table cluster to QCMTAB

Step 3: Delete the relevant old data from the corresponding table cluster

Step 4: Activate table TAB to generate runtime object in new structure

Step 5: Load the relevant data from QCMTAB to TAB

Step 6: Drop QCMTAB

As you can see, there is no table QCM8TAB created during cluster table conversion.

In the conversion for transparent table, table QCMTAB is renamed from original table with the old structure. But in the conversion for cluster table, table QCMTAB is newly created with the new structure.

**Can I convert cluster table to be transparent table?**

It is possible for you to convert cluster table to transparent table. But it is recommended to convert all cluster tables in the same table cluster to be transparent, not only part of them.

The standard table conversion via T-cd:SE14 cannot convert an entire table cluster and make all contained cluster tables transparent. Instead, the change has to be done cluster table by cluster table. While a single cluster table is being converted, it must not be updated because basically the entire conversion is downtime. The problem is that there is no mechanism during a cluster table conversion that prevents update accesses during the conversion. If the cluster table is updated by the application during the conversion, the result is undefined, i.e. the change might be present in the converted table or it might be missing. And because there are huge data contained in cluster tables, pay attention to make sure there are enough space on database.

If you already converted cluster tables into transparent tables, pay attention not to convert them back into cluster tables during next upgrade. It is very important that you should not skip SPDD adjustment for these tables. Otherwise these table will be converted back to cluster table during upgrade. It is suggested to double check the inactive version of the tables during upgrade in the shadow system before starting the activation phase. During upgrade, the active version in the shadow system will show these tables as cluster tables, and inactive version as transparent ones. The inactive version is the one that will be active after the activation phase. In case the inactive version is not OK or does not exist, it has to be adjusted or created by changing the table in the shadow system in transaction SE11.

**What will happen if I unlock a table from conversion in SE14? Can I unlock a table from conversion?**

The following will happen if you unlock table TAB from conversion:

1. Delete corresponding data from TCNV and TCNVDATA
2. Write SYSLOG (T-cd:SM21) message
3. Write related database protocol log in internal tables

Table unlock from conversion will lead to dictionary object inconsistencies, so normally it is not recommended to unlock a table from conversion. But sometimes when you noticed that the changes that you di to the table was wrong which leaded to the conversion, and you want to return table definition back to original one. For example, if conversion stopped at step5 (data load from QCMTAB to QCM8TAB), you can unlock the conversion, delete QCM8 table on DB level, rename the QCM table to its original name, reconstruct the runtime object, set the table definition via SE11 to its old state, and then activate the table.

You can refer to the below link for more details:

http://wiki.scn.sap.com/wiki/x/347zEg

You should perform the changes very carefully, otherwise it will lead to data loss.

**How to check table entries during BSEG conversion?**
Table QCMBSEG will be generated during cluster table BSEG conversion. But QCMBSEG has no dictionary definition in SE11. If you try to display table content for QCMBSEG via SE16, you will get error “Table QCMBSEG is not active in the dictionary”.

You can display table contents of QCMBSEG via SE17.

If you just want to know how many entries in table QCMBSEG, you can use report NROWS.

### How to do SPDD adjustment for BSEG during upgrade (or EHP installation)?

Cluster table BSEG contains financial accounting documents which usually contain a lot of data records, generally the conversion on table BSEG takes an extremely long time (sometimes several days) or terminates. SAP does not deliver any changes that would result in a conversion of the table BSEG. Unfortunately we still can find some conversion issues that happened on table BSEG.

Possible reasons for BSEG conversion are:

1. Customer-specific fields are deleted due to “reset to original” in SPDD adjustment.
   - If you do not perform adjustment to BSEG in SPDD, the SAP standard version will be taken automatically, and customer-specific fields will be deleted.

2. Program bug. See note 1283197.

You can refer to note 24864 for details.

Please do not trigger conversion for table BSEG during SPDD adjustment.

In ACT_UPG phase, you are requested to perform SPDD adjustment on shadow instance. If you find cluster table BSEG in SPDD list, please DO NOT just reset to original for it. If you click the yellow traffic light for BSEG, generally an “Adjustment proposal” will pump up automatically. If you click the “accept proposal” button, all customer-specific fields will remain, and generally BSEG conversion will not be triggered. You can check the inactive version via SE11 > “Active <->Inactive” for table BSEG after performing SPDD adjustment on it, to make sure that customer-specific fields remain in table BSEG on shadow instance.

If BSEG conversion already started, always it is too late to stop it (for example if step2 already finished). So it makes sense to ensure that BSEG conversion will not be triggered beforehand. You can set a breakpoint to let SAPup or SAPehpi stop just after finishing phase PARDIST_SHD, and check the entry for BSEG in table DDXTT on shadow instance. If there is no entry with MODEFLAG=U for BSEG in table DDXTT, you can ensure that conversion will not happen on BSEG afterwards.

While it is only possible to set breakpoint before a specified phase starts, using option “SAPup stop <Phasename>” or “SAPehpi stop <Phasename>” according to note 48184. As from Netweaver 700, you can use “Set breakpoint” option directly in SUM Gui screen to achieve it. You can find the phase name is the upgrade Gui screen, or you can check the file `/<DIR_PUT>/abap/htdocs/phaselist.xml` to find out what is the next phase after PARDIST_SHD, and set a breakpoint with this phase’s name. After phase PARDIST_SHD finished, you can log on the shadow instance and check the entry for BSEG in table DDXTT. If the MODEFLAG for table BSEG is A, not U, then it indicate that conversion is not triggered, and you can continue. But if you find the entry for BSEG with MODEFLAG=U in table DDXTT, please contact SAP for help.

### What kind of changes on cluster table such as BSEG can trigger table conversion?

Any change of a field’s type or length will trigger conversion of a cluster table, except adding new fields as of SAP_BASIS release 7.0.

Adding new fields into cluster table BSEG is also OK, will NOT trigger table conversion. New fields are always added at the end of a cluster record, it is just like adding new fields to transparent tables. During system update/upgrade, changes to cluster table BSEG delivered by support packages are always adding new fields. That is also the reason why we must adjust cluster table BSEG in transaction SPDD to have all customer fields remained. Because if you selected “reset to original” for BSEG, all customer fields would be deleted, which would trigger long-time table conversion on cluster table BSEG.

### Trouble Shootings

**We noticed that a new table QCMBSEG was created after updating support package stacks. Why this table exists in our database and to solve this issue?**

Sometimes even BSEG conversion finished completely, table QCMBSEG was not deleted and still existing in database. The reason is that if BSEG conversion stopped at step 5 and started again, table QCMBSEG will not be deleted even after step5 finished completely. It is SAP design, not a problem. In such a case, you can just delete table QCMBSEG via SE14 > Extras > Invalid Temp. Table > select table QCMBSEG > Delete.

**Dump CONVT_NO_NUMBER occurred during table conversion in step 5. How to continue the conversion?**

The reason is that one or some table fields were changed from Character type to Number type (which triggered table conversion), and the table contains data which are not compatible with this change. The data is mentioned in the dump file, you can find the sentence like “Unable to interpret “00” as a number”.

The solution is to update the table QCMTAB and change the data in the corresponding field to number type.
For example you can execute SQL statement like:

```sql
update QCMTAB set <Table_field>='000' where <table_field>='*00';
```

After that, continue conversion via SE14 > Continue Adjustment.

For details, please refer to the following link:

http://wiki.scn.sap.com/wiki/x/1YKHEg

**Dump BCD_FIELD_OVERFLOW occurred during table conversion in step 5. How to continue the conversion?**

The reason is that the length of one or some table fields was shrunk which triggered table conversion. Generally it was caused by wrong operation, so it is necessary to roll back the conversion.

For details, please refer to the following link:

http://wiki.scn.sap.com/wiki/x/8Yz_Eg

**Table conversion for table BSEG took very long time during upgrade. How to do?**

If BSEG conversion already started, check at which step it is running or terminated.

If conversion is running or terminated at step2, you can unlock the conversion as below:

1. In case that step2 is still running, cancel the active corresponding conversion background job via SM37 > “Cancel active job”.
2. Unlock BSEG conversion via SE14 > BSEG > ”Unlock Table”.
3. Delete table QCMBSEG via SE14 > QCMBSEG > ”Delete database table”.
4. Change BSEG definition in SE11 (like adding YY* or ZZ* customized fields) to avoid conversion.
5. Activate table BSEG in SE11.

If step 2 has already been completed, you have to continue the conversion. Generally the runtime will be extremely long. So if you find BSEG conversion is still running at step2, try to terminate it as soon as possible in order to prevent it from running into step3.