Incremental Conversion of tables-- The concept behind it

Purpose

Incremental conversion of tables or properly called as ICNV is actually helpful to reduce the conversion of long tables that have higher amount of data. In this format the tables are converted before hand the downtime starts and the system will have less business down hours. The purpose of this wiki is to tell people what actually happens during the internal conversion of the table here.

States of an incremental conversion:

- Initialization
  
  The prerequisites for the data transfer are prepared. The table can not be used during this phase.

- Data transfer
  
  The data are moved (and converted) from the table with the old structure into the table with the new structure.

- Switch to the new structure
  
  The new table definition is activated. The table can not be used during this phase.

Initialization: Start

The table TAB was modified in the ABAP Dictionary. The length of last field of this table was changed from 60 to 30. Thus a conversion is necessary.

The table has two indexes which exists on the database.

The table was marked for incremental conversion.

Initialization: Create QCM-Table
In the first step of the initialization a table QCMTAB with the revised structure (where field 3 has length 30) is created. The two indexes are also created on the database under different names (suffix X). The table QCMTAB is empty.

**Initialization: Rename old DB-Table**

- The table TAB (old structure) is now renamed on the database to QCM1TAB.
- A field (Indicator field) is added to this table. This is possible by executing a database statement (ALTER TABLE ... ADD FIELD)

**Initialization: Create Triggers**

- The table TAB (old structure) is now renamed on the database to QCM1TAB.
- A field (Indicator field) is added to this table. This is possible by executing a database statement (ALTER TABLE ... ADD FIELD)
A update trigger and a delete trigger are created on QCM1TAB.

**Initialization: Create DB-View**

A view TAB is created on the table QCM1TAB. This view is simply a projection on the fields of QCM1TAB without the indicator field. Thus, this view reflects the old table structure.

The initialization is now finished. The table TAB was replaced during this phase by a view TAB with the same structure. Application programs can access the data now by this view! Thus, the changes made during initialization are not visible to application programs using the table TAB. They simply work in the same way as before!

**Data Transfer: Start**
Now the data transfer starts.
The records in the table QCM1TAB are copied into the table QCMTAB. This is done with the ABAP command MOVE-CORRESPONDING. Thus in our example the first 30 places of FIELD 3 are copied into the new structure.
You can find information about the way in which data are converted in the documentation of the ABAP command MOVE.

Each record which was copied to QCMTAB is marked with the value C in the indicator field

Working on the tables during data transfer

The application programs can use the data during the data transfer. Thus, it is possible that application programs change data in the table TAB. This change is done via the view in the table QCM1TAB. The problem is now to mark these changed data sets and to bring the changes also to the table QCMTAB.
Data Transfer: Updating records

If a record in table QCM1TAB is updated the update trigger fires. The value C in the indicator field (which indicates that this record was already copied to QCMTAB) is set to U. Thus, the data transfer can find all records which are updated after they are copied to the new table by their values in the indicator field.

Data Transfer: Deleting records

If a record in QCMTAB is deleted by the application program, the delete trigger fires. The delete trigger simply deletes the corresponding record in the table QCMTAB.

Data transfer: Inserting records
If an application program inserts a new record into the table QCM1TAB nothing must be done. The record is simply written to the table with SPACE as value in the indicator flag. The value SPACE shows that this record is not yet copied to the table QCMTAB.

Switch to new structure: Start

You have to decide when the switch to the new structure can be made.

The switch first deletes the view TAB on the database.

Then a last data transfer is started.

Active version of TAB

Revised version of TAB

Delete DB-View

Switch to new structure: Last data transfer

During the last data transfer all remaining records of the table QCM1TAB (records which are not yet copied or changed after they are copied to the table QCMTAB) are transferred to the table QCMTAB.

These records are identified by the values SPACE or U in the indicator field.
Switch to new structure: Activation

After the data are successfully copied to the table with the new structure the activation in the ABAP dictionary takes place. The revised version of the runtime object overwrites the old active version. Thus the new structure becomes now active.

Switch to new structure: Renamed table

The table QCMTAB is renamed to TAB
Switch to new structure: Deletion

The table QCM1TAB is no longer necessary and thus deleted on the database.