**Pooled and Cluster Tables**

**Use**

Table pools (pools) and table clusters (clusters) are special types of tables in the ABAP Dictionary. The data from several different tables can be stored together in a table pool or table cluster. Tables assigned to a table pool or table cluster are indicated as pooled tables or cluster tables.

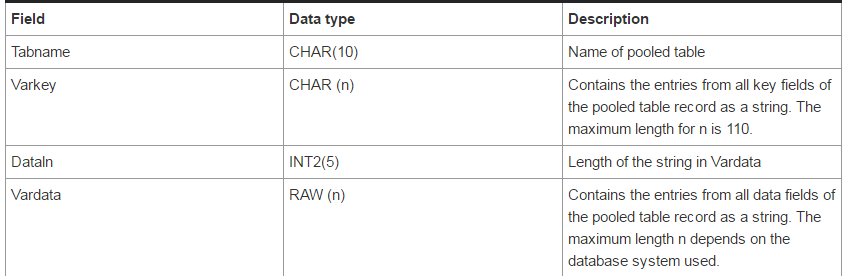
You must use a table pool or table cluster exclusively for storing internal control information (screen sequences, program parameters, temporary data, continuous texts such as documentation). Data of commercial relevance is stored in transparent tables.

**Features**

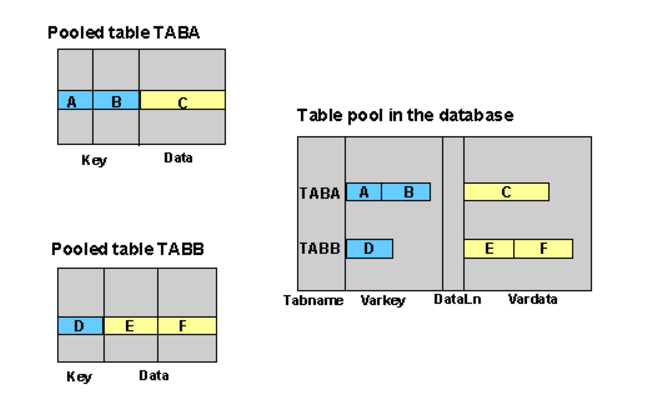
**Table Pools**

A table pool corresponds to a table in the database in which all records from the pooled tables assigned to the table pool are stored.

The definition of a pool consists of two key fields (Tabname and Varkey) and a long argument field (Vardata). A pool has the following structure:



If you save a pooled table record, the system stores it in the corresponding table pool. The name of the pooled table is written to the Tabname field. The contents of all key fields of the pooled table are written as a string to the Varkey field and the contents of all data fields as a string to the Vardata field. The database interface enters the length of the string stored in Vardata in the Dataln field. In the following figure you can see the structure of a table pool. The figure presents an example of a table pool with two assigned pooled tables



##### Constraints

Due to the structure of a table pool, there are certain restrictions on the pooled tables assigned to it:

●  The name of a pooled table cannot exceed 10 characters.

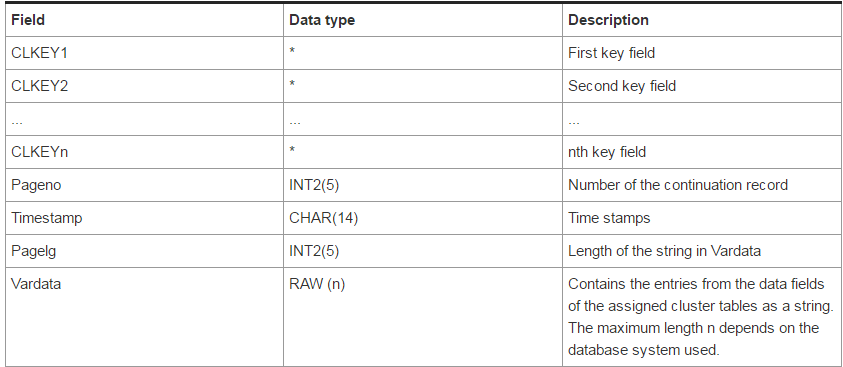
●  Since Varkey is a character field, all key fields of a pooled table must have character data types (for example, CHAR, NUMC, CLNT).

●  The total length of all key fields or all data fields of a pooled table must not exceed the length of the Varkey or Vardata field of the corresponding pool.

#### Table Clusters

Several logical data records from different cluster tables can be stored together in one physical record in a table cluster.

A cluster key consists of a series of freely definable key fields and a field (Pageno) for distinguishing continuation records. A cluster also contains a long field (Vardata) that contains the contents of the data fields of the cluster tables for this key. If the data does not fit into the long field, the system creates continuation records. Furthermore, control information about the structure of the data string is written at the beginning of the Vardata field. A table cluster has the following structure



The records of all cluster tables with the same key are stored under one key in the assigned table cluster. The values of the key fields are stored in the corresponding key fields of the table cluster. The values of all data fields of the assigned cluster tables are written as a string to the Vardata field of the table cluster.

Besides the actual data values, the data string contains information about the structure of the data and which table it comes from. If the string exceeds the maximum length of the Vardata field, the system writes a continuation record with the same key values. The continuation records for a key are distinguished by their value in the Pageno field. The actual length of the string in the Vardata field is stored in the Pagelg field by the database interface.

In the following figure, you can see the structure of a table cluster. The figure presents an example of a table cluster which contains two cluster tables.

