Accessing an SQL Table

▶ **To be able to access an SQL table with a Natural program via Natural SQL Gateway**

1. Establish a connection to the ConnecX SQL Engine JDBC server.

2. Deploy a ConnecX SQL Engine data dictionary (CDD) containing the definition of the SQL table to be accessed.

   - If the table does not yet exists on the SQL database system, it can be created with a `CREATE TABLE` statement.

   - If the table already exists, the table definition can be imported from the SQL catalog into the CDD, using the ConnecX SQL Engine data dictionary manager. Keep in mind that the **Import** function of the data dictionary manager creates the table definition with the qualifier name `dbo`. This is usually undesired and can be easily reverted to the original qualifier by means of the **Change Owner** tool of the data dictionary manager.

3. Invoke the Natural utility `SYSDDM` and enter function code `Z` to create a Natural data definition module (DDM) describing the SQL table.

4. In the `NTDB` macro in the Natural parameter module `NATPARM`, define the DBID of the DDM as database type SQL.

5. In the `NDBID` macro in the `NDBPARM` module, define the DBID of the DDM as database type CNX. The SQL table to be accessed has to be defined in the CDD, see also the *ConnecX SQL Engine Data Dictionary* documentation.

6. Once you have defined a DDM for an SQL table, you can access the data stored in this table, using a Natural program.

   Natural SQL Gateway translates the statements of a Natural program into SQL statements.

   Natural SQL Gateway automatically provides for the preparation and execution of each statement in dynamic mode. Static execution is currently not supported. A statement is only prepared once (if possible) and can then be executed several times. For this purpose, Natural internally maintains a table of all prepared statements (see *Statement Table* in *Internal Handling of Dynamic Statements*).

   Almost the full range of possibilities offered by the Natural programming language can be used for the development of Natural applications which access SQL tables. For a number of Natural DML statements, however, there are certain restrictions and differences as far as their use with SQL is concerned; see *Using Natural DML Statements* as described in *Using Natural Statements and System Variables*. In the Natural *Statements* documentation, you can find notes on Natural usage with SQL in the descriptions of the statements concerned.

   Note:
   As there is no SQL equivalent to Adabas Internal Sequence Numbers (ISNs), any Natural features which use ISNs are not available when accessing SQL tables with Natural.

   In addition to the Natural DML statements, Natural provides SQL statements for SQL databases as discussed in *Using Natural SQL Statements*. In the section *SQL Statements* in the Natural *Statements* documentation, you can find detailed information on these statements.