

PARAMETER STYLE DB2SQL - UDFs ???

A Natural UDF receives (passes???) parameters similar to **PARAMETER STYLE DB2SQL** for **Natural** stored procedures as described in the relevant section.

This section covers the following topics:

- [Passing DB2SQL???](#)
 - Defining UDF Parameters
-

Passing DB2SQL???

The input parameters are **passed** in the sequence in which they are defined in the CREATE FUNCTION statements. **Following the input parameters**, the result parameter(s) are passed. (**The result parameter(s) are passed after the input parameters???**) For each parameter, a **NULL** indicator is passed. In addition, the following parameters are **passed**:

- The SQLSTATE to be returned to DB2,
- The qualified name of the **Natural** (???) UDF,
- The specific name of the **Natural** (???) UDF,
- The SQL diagnostic string to be returned to DB2.

The parameters are defined in the Natural parameter data area (PDA) DB2SQL_P supplied in the Natural system library SYSDB2.

If the feature SCRATCHPAD *nnn* is specified additionally in the CREATE FUNCTION statement, **the SCRATCHPAD storage parameter is passed to the Natural (???) UDF**.

Use the following definitions:

```
01 SCRATCHPAD A(4+nnn)
01 REDEFINES (ohne S???) SCRATCHPAD
02 SCRATCHPAD_LENGTH(I4)
02
```

Redefine the SCRATCHPAD in the **Natural** (???) UDF according to your requirements.

The first four bytes of the **SCRATCHPAD** area contain a length (???) field. Before initially invoking the UDF with an SQL statement, DB2 resets the **SCRATCHPAD** area to x'00' and puts into the first four bytes the specified size *nnn* of the SCRATCHPAD as integer value.

Thereafter, DB2 does not reinitialize the SCRATCHPAD between the invocations of the UDF **for the invoking SQL statement???** Instead, after returning from the UDF, the contents of the SCRATCHPAD is **preserved** and restored at the next invocation of the UDF.

If the FINAL CALL option is specified for a scalar UDF, or if the UDF is a table UDF, the CALL TYPE parameter is passed. The CALL TYPE parameter is an integer indicating the type of call DB2 performs on the UDF. See the DB2 SQL GUIDE for details on the **parameter** (???) values provided.

If the option DBINFO is used additionally, the DBINFO data (???) structure is passed to the **Natural** (???) stored procedure. The DBINFO structure is described by the Natural PDA DBINFO_P that is supplied in the Natural system library SYSDB2.

Defining UDF Parameters

Define the UDF parameters in Natural as shown in the example program below:

```

DEFINE DATA PARAMETER
01 PI1    /* first input parameter
01 PI2

01 PIn ... /* last input parameter
01 RS1  /* first result parameter

01 RSn /* last result parameter
01 N_PI1 (I2) /* first NULL indicator
01 N_PI2 (I2)

01 N_Pin (I2)
01 N_RS1 (I2)

01 N_RSn (I2) /* last NULL indicator
PARAMETER USING DB2SQL_P /* function, specific, sqlstate, diagnose
PARAMETER
01 SCRATCHPAD A(4+nnn) /* only if SCRATCHPAD nnn is specified
  01 REDEFINES SCRATCHPAD
02 SCRATCHPAD_LENGTH (I4)
02
01 CALL_TYPE (I4) /* --- only if FINAL CALL specified or table UDF

PARAMETER USING DBINFO_P /* ---- only if DBINFO is specified
LOCAL

END-DEFINE

```

The **NDB server stub** that executes the **Natural UDF** determines the name of the subprogram and the library to be invoked from the qualified and specific names of the UDF. The SCHEMA name is used as library name and the **function** name is used as subprogram name.

The subprogram **ROUTINEN** (supplied in the Natural system library SYSDB2) is used to access the DB2 catalog in order to determine the format of the user parameters of the **Natural UDF**. After the formats have been determined, they are stored in the Natural buffer pool. During subsequent invocations of the **Natural UDF**, the formats are retrieved from the Natural buffer pool. This requires that at least READS SQL DATA is specified for a **Natural UDF**.

If a Natural runtime error occurs during the execution of a Natural UDF with **PARAMETER STYLE DB2SQL**, SQLSTATE is set to 38N99 and a diagnostic string contains the text of the Natural error message.

If an error occurs in the NDB server stub during the execution of a Natural UDF using **PARAMETER STYLE DB2SQL**, the SQLSTATE is set to 38S99 and a diagnostic string contains the text of the error message.

If the application wants to raise an error condition during the execution of a Natural UDF, the SQLSTATE parameter must be set to a value other than 00000. See the DB2 SQL Guide for specifications of user errors in the SQLSTATE parameter.

Additionally, a text for analysing errors can be placed in the DIAGNOSE parameter.

If a table (???) Natural UDF wants to signal to DB2 that it has found no row to return, '02000' (Kommata erforderlich???) should be returned in the SQLSTATE parameter.

For a Natural UDF that contains the attributes DISALLOW PARALLEL and FINAL CALL, the NDB server stub retains the Natural session allocated earlier. This Natural session will then be reused by all subsequent UDF invocations until Natural encounters the final call.