NUMERIC and DECIMAL

The NUMERIC data type specifies a numeric column where the value has a fixed decimal point, such as for currency data. NUMERIC(18) is a 64-bit integer value in SQL dialect 3 and is almost infinite. Since SQL dialect 3 numeric and decimal data types are stored as INTEGERS of the respective size.

SQL dialect 1 offers NUMERIC(15).

Syntax:

```
NUMERIC(precision, scale);
```

or

```
DECIMAL(precision, scale);
```

PRECISION refers to the total number of digits, and SCALE refers to the number of digits to the right of the decimal point. Both numbers can be from 1 to 18 (SQL dialect 1: 1-15), but SCALE must be less than or equal to PRECISION.

It is better to define NUMERIC always at its maximum length, as in this case, the 32 bit INTEGER value is used. Otherwise a 16 bit value is used internally, for example with NUMERIC(4,2), and this is not always transformed back correctly by the client program environments (an older BDE version could, for example, transform Euro 12.40 with NUMERIC(4,2) into Euro 1,240).

Firebird/InterBase® supports a number of options for specifying or not specifying PRECISION and SCALE:

- If neither PRECISION nor SCALE are specified, Firebird/InterBase® defines the column as INTEGER instead of NUMERIC and stores only the integer portion of the value.
- When using SQL dialect 1, if just PRECISION is specified, Firebird/InterBase® converts the column to a SMALLINT, INTEGER or DOUBLE PRECISION data type, based on the number of significant digits being stored.

In SQL dialect 3, if just PRECISION is specified, Firebird/InterBase® converts the column to a SMALLINT, INTEGER or INT64 data type, based on the number of significant digits being stored.

It is important to distinguish between the two dialects, because since INT64 is an INTEGER data type, and DOUBLE PRECISION is not, you will occasionally have rounding errors in SQL dialect 1, but not in SQL dialect 3 or later.

The NUMERIC data type should only be used for fields that are later to be used as part of a calculation.

Firebird/InterBase® converts the columns as follows:

Definition	**Data type Created**
Decimal(1)-Decimal(4)	Small Integer
Decimal(5)-Decimal(9)	Integer
Decimal(10)-Decimal(18)	Int (64)

Note that if a DECIMAL(5) data type is specified, it is actually possible to store a value as high as a DECIMAL(9) because Firebird/InterBase® uses the smallest available data type to hold the value. For a DECIMAL(5) column, this is an INTEGER, which can hold a value as high as a DECIMAL(9).

Enhancement in precision of calculations with NUMERIC/DECIMAL (Firebird 4.0)

Supported in IBExpert since version 2017.12.03.

Source: https://github.com/FirebirdSQL/firebird/blob/master/doc/sql.extensions/README.data_types

Function

Maximum precision of NUMERIC and DECIMAL data types is increased to 34 digits.

Author Alex Peshkoff <peshkoff@mail.ru>

Syntax rules

```
  NUMERIC ( P {, N} )
  DECIMAL ( P {, N} )
      where P is precision (P <= 34, was limited prior with 18 digits) and N
is optional number
      of digits after decimal separator (as before).
```

Storage

128-bit, format according to IEEE 754.

Example(s)

1. DECLARE VARIABLE VAR1 DECIMAL(25);
2. CREATE TABLE TABLE1 (FIELD1 NUMERIC(34, 17));

Note(s)

Numerics with precision less than 19 digits use SMALLINT, INTEGER, BIGINT or DOUBLE PRECISION as base datatype depending upon number of digits and dialect. When precision is between 19 and 34 digits DECFLOAT(34) is used for it. Actual precision is always increased to 34 digits. For complex calculations such digits are casted (internally, in trivial way) to DECFLOAT(34) and the result of various math (log, exp, etc.) and aggregate functions using high precision numeric argument is DECFLOAT(34).

NUMERIC-SORT

Firebird 2.5 introduced NUMERIC-SORT for Unicode collations only.

Format & usage

```
NUMERIC-SORT={0 | 1}
```

The default, 0, sorts numerals in alphabetical order. For example:

```
1
10
100
2
20
```

1 sorts numerals in numerical order. For example:

```
1
2
10
20
100
```

Example

```
create collation unicode_num for utf8
from unicode 'NUMERIC-SORT=1';
```

From:
https://www.ibexpert.com/docu/ - **IBExpert**

Permanent link:
https://www.ibexpert.com/docu/doku.php?id=01-documentation:01-13-miscellaneous:glossary:decimal

Last update: **2023/08/14 16:13**