



ホーム | 製品 | サービス & ソリューション | サポート & ダウンロード | マイアカウント



DB2 Developer Domain > 製品別技術情報 > DB2いろはがるた >

DB2いろはがるた



第39回 「ゆ」 - UDF、誰もが作れるSQL関数

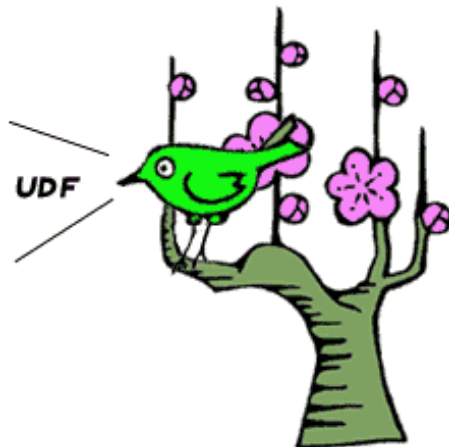


執筆者

春野 さくら

「DB2いろはがるた」を執筆するために参上した、なぞの女性。日本の古典文学を愛する。

今年もはや如月にはいり、そろそろ、梅の花の便りがあるころになりました。今回はUDF(ユーザー定義関数)のお話です。



いろいろなベンダーのRDBMS製品は、予め組み込まれている多数のSQL関数を提供しています。ユーザーはSQL文の中でSQL関数を使うことによって、とても簡単にデータを加工することができます。DB2ではさまざまなタイプの関数が用意されていますが、おおきくわけて3種類あります。

・スカラー関数

行関数とも呼ばれ、結果表の行それぞれに対しての処理結果を提供します。スカラー関数は式の利用が許される場所であればどこでも使用できます。

・列関数

ベクトル関数とも呼ばれ、行のグループを処理して結果を提供します。

・表関数

フラットファイルのデータでも、あたかも表であるかのようにSQL文で値をとりだすことができます。SQL文のFROM文節で使用します。

例えば、よく使われる代表的なSQL関数として、列のある部分だけを取り出すSUBSTR関数、列の全データの合計を算出するSUM関数、行数を算出するCOUNT関数などがあります。DB2 UDBではその他にも、ROUND関数、YEAR関数など実に100以上の関数を提供しています。次の例では、SUBSTR関数はwphone列の最初の3文字の文字列を戻します。

```
SELECT Iname, SUBSTR(wphone, 1,3) FROM db2cert.candidate
```



ところで、DB2 UDBではこのようなSQL関数を、ユーザーが独自に作成することができます。これをユーザー定義関数(UDF)と呼びます。

動的ロード可能ライブラリーとして作成(OLE DB外部関数、ソース関数を除く)し、CREATE FUNCTION文でDB2に登録します。ユーザーはUDFを、Visual Basic, Java,C++, C言語、SQLプロシージャー言語などで作成することができます。同じ名前を持つが異なるパラメーターを持つUDFも作成することができます。これは、多重定義関数と呼ばれます。UDFでは、既存のUDB組み込み関数、または、既に作成されたUDFを基にして別のUDFを作成することもできます。

また、UDFは非分離資源または分離資源として実行できます。非分離とは、UDFがデータベース・エージェントと同じプロセスで実行されることで、プロセス通信のオーバーヘッドが少ないので分離資源として実行されたUDFよりパフォーマンスが良いといわれています。

ただし、バグのある非分離資源として実行されたUDFは、UDBの制御ブロックを破壊する可能性があるため、本番業務で実行する前には、あらかじめ十分なテストが必要です。

分離資源では、UDFはデータベース・エージェントと別のプロセスで実行されます。

DB2のUDFには5つのタイプがあります。

1. 外部スカラー関数

一つのスカラー値を戻します。プログラミングが必要です。

2. 外部表関数

一つの表を戻します。プログラミングが必要です。

3. OLE DB外部表関数

OLE DBプロバイダーからデータへアクセスし、一つの表を戻します (Windows版のみ) この関数はプログラミングが不要で、CREATE FUNCTION文の実行だけで作成できます。

4. ソース関数

UDBに組み込まれた関数、あるいは、既に作成、登録されたUDFを基にした関数です。この関数はプログラミングが不要で、CREATE FUNCTION文の実行だけで作成できます。次の例では、UDBに組み込まれたAVG関数を利用して、CANADIAN_DOLLARデータ・タイプ用のAVG関数を作成しています。

```
CREATE FUNCTION AVG (CANADIAN_DOLLAR)
  RETURNS CANADIAN_DOLLAR
  SOURCE SYSIBM.AVG(DECIMAL(9,2))
```

5. SQL スカラー、表、行 関数

SQL RETURN ステートメントだけを使用して定義されます。SQL文を含み、スカラー値、行、または表のいずれかを戻すことができます。

```

CREATE FUNCTION tan (x double) RETURNS double LANGUAGE SQL
CONTAINS SQL NO EXTERNAL ACTION DETERMINISTIC
RETURN sin(x) / cos(x)
CREATE FUNCTION fromperson (p person)
RETURNS ROW (name varchar(10), firstname varchar(10))
LANGUAGE SQL CONTAINS SQL NO EXTERNAL ACTION DETERMINISTIC
RETURN VALUES (p..name, p..firstname)
CREATE FUNCTION deptemployees (deptno char(3))
RETURNS TABLE (empno char(6), lastname varchar(15),
firstname varchar(12))
LANGUAGE SQL READS SQL DATA NO EXTERNAL ACTION DETERMINISTIC
RETURN
SELECT EMPNO, LASTNAME, FIRSTNAME FROM EMPLOYEE
WHERE EMPLOYEE.WORKDEPT = DEPTEMPLOYEES.DEPTNO

```

ところで、UDFを使うと非常に便利な場合があります。賢明な読者の方はすでにご想像したと思いますが、UDFは他のRDBMS製品からDB2へプログラムを移行するときに使うととても便利です。例えば、Oracle製品でのSINH関数はDB2では同じ名前の組み込み関数を提供していません。そういった場合、DB2側でSINH関数をUDFで予め作成しておくことによって、ソース・プログラムの変更することなく、実行することができるようになります。DB2側でUDFを下の例のように書くことができます。

```

CREATE FUNCTION SINH (X Double)
RETURNS Double
SPECIFIC SINHDouble
LANGUAGE SQL
CONTAINS SQL
NO EXTERNAL ACTION
DETERMINISTIC
RETURN
(exp(X)-exp(-X)) / 2;

```

DB2では前述したように、いろいろな言語でUDFを作成することができるので、ここはエンジニアにとって腕のみせどころといえるかもしれませんね。

[↑ 上に戻る](#)