Adaptive Server Anywhere で DATEDIFF と DATEADD ファンクションに対する日付範囲の制限を拡張する

このマニュアルでは、特定タイプの日時値の格納と返却を行うための DATEDIFF **および** DATEADD **関数の制限と、データ・オーバフローまたはデータ・トランケーション・エラーが発生する場合に使用できる代替のユーザ関数について説明します。**

背景

DATEDIFF **と** DATEADD **関数を使用する他の** RDBMS (**たとえば、**Microsoft SQL Server**、**Adaptive Server Enterprise**、**Microsoft Access **など**) **と同様に、**Adaptive Server Anywhere **では、これら** 2 **つの関数は整数データ型を使用して時間間隔を格納または返却します。その結果、**2 **つの制限が生じます。**

最初の制限には、以下の関数によって返される間隔値が関係します。
DATEDIFF (date-part, date-expression1, date-expression2)

比較的小さな時間単位を date-part **パラメータで使用した場合は、**date-expression1 **と** date-expression2 **の差が以下の間隔を超えるとオーバフロー・エラーが発生します。**

- **ミリセカンド** (milliseconds) **で** 24 **日**
- **秒**(seconds)**で** 68 **年**
- **分**(minutes)**で** 4083 **年**

2 **番目の制限には、符号付き整数になるように関数** DATEADD (date-part, numeric-expression, date-expression) **の** numeric-expression **パラメータがトランケートされるケースが関係します。トランケーションの結果として、小さな時間単位を** numeric-expression **パラメータに使用した場合は、**numeric-expression **値が符号付き整数の範囲を超えていると、返される値は正しくありません。**

解決策

これらの制限を克服するには、ユーザ関数を使用します。ユーザ関数では、計算を分割できます。それにより、より大きな時間単位を使用して各部分を計算できるようになります。以下のユーザ関数例では、BIGINT **データ範囲の値を使用して、サポートされる日付範囲の拡張を可能にしています。**

4083 年を超えない日付範囲

以下のユーザ関数では、4083 **年までの日付範囲 (一般的な用途には十分すぎる範囲) を使用できます。**

DATEDIFF (second, starttime, endtime) **の代わりに、以下を使用できます。**

```
CREATE FUNCTION datediff64s ( IN starttime DATETIME, IN endtime DATETIME )
RETURNS BIGINT
BEGIN
    RETURN CAST ( DATEDIFF ( minute, starttime, endtime ) AS BIGINT )
    *60 + DATEDIFF ( second, DATEADD ( minute, DATEDIFF ( minute,
```

```
    starttime, endtime ), starttime ), endtime )
END
```

DATEDIFF (millisecond, starttime, endtime) **の代わりに、以下を使用できます。**

```
CREATE FUNCTION datediff64ms ( IN starttime DATETIME, IN endtime DATETIME )
RETURNS BIGINT
BEGIN
    RETURN CAST ( DATEDIFF ( minute, starttime, endtime ) AS
    BIGINT)*60000 + DATEDIFF ( millisecond, DATEADD ( minute,
    DATEDIFF ( minute, starttime, endtime ), starttime ), endtime )
END
```

DATEADD (second, difference, starttime) **の代わりに、以下を使用できます。**

```
CREATE FUNCTION dateadd64s ( IN difference BIGINT, IN starttime DATETIME )
RETURNS DATETIME
BEGIN
    RETURN DATEADD ( second, MOD ( difference, 60 ), DATEADD (
    minute, difference/60, starttime ) )
END
```

DATEADD (millisecond, difference, starttime) **の代わりに、以下を使用できます。**

```
CREATE FUNCTION dateadd64ms ( IN difference BIGINT, IN starttime DATETIME )
RETURNS DATETIME
BEGIN
    RETURN DATEADD ( millisecond, MOD ( difference, 60000), DATEADD (
    minute, difference/60000, starttime ) )
END
```

4083 年を超える日付範囲

4083 **年を超える範囲の場合は、必要なビルド** (9.0.2 build 3044 **以上、**9.0.1 build 1994 **以上、**8.0.3 build 5220 **以上、**8.0.2 build 4521 **以上**) **へのアップグレードによって極端な** DATEDIFF **および** DATEADD **の計算問題に対処すると、以下のユーザ関数を使用できるようになります。**

DATEDIFF (second, starttime, endtime) **の代わりに、以下を使用できます。**

```
CREATE FUNCTION datediff64s ( starttime DATETIME, endtime DATETIME )
RETURNS BIGINT
BEGIN
    RETURN CAST ( DATEDIFF ( day, starttime, endtime ) AS
    BIGINT)*86400 + DATEDIFF ( second, DATEADD ( day, DATEDIFF (
day, starttime, endtime ), starttime ), endtime )
END
```

DATEDIFF (millisecond, starttime, endtime) **の代わりに、以下を使用できます。**

```
CREATE FUNCTION datediff64ms ( starttime DATETIME, IN endtime DATETIME )
RETURNS BIGINT
BEGIN
    RETURN CAST ( DATEDIFF ( day, starttime, endtime ) AS
    BIGINT)*86400000 + DATEDIFF ( millisecond, DATEADD ( day,
DATEDIFF ( day, starttime, endtime ), starttime ), endtime )
END
```

DATEADD (second, difference, starttime) **の代わりに、以下を使用できます。**

```
CREATE FUNCTION dateadd64s ( IN difference BIGINT, IN starttime DATETIME )
RETURNS DATETIME
BEGIN
    RETURN DATEADD ( second, MOD ( difference, 86400), DATEADD (
    day, difference/86400, starttime ) )
END
```

DATEADD (millisecond, difference, starttime) **の代わりに、以下を使用できます。**

```
CREATE FUNCTION dateadd64ms ( IN difference BIGINT, IN starttime DATETIME )
RETURNS DATETIME
BEGIN
    RETURN DATEADD ( millisecond, MOD ( difference, 86400000 ),
    DATEADD ( day, difference/86400000, starttime ) )
END
```

『Adaptive Server Anywhere SQL User's Guide**』に記載されているように、このユーザ関数では完全な範囲の日時値** (7911-01-01 **まで**) **を使用できます。この範囲を超える関数の精度は、**Adaptive Server Anywhere **によってサポートされる範囲を超えているため保証できません。**