

## 同期の頻度を高める方法について

dbmsync をスケジュール実行する場合、処理の間隔は 1 分間に固定され、秒数による指定はできません。ただし、"EVERY:00:00" というスケジュールを指定することで、直前の同期が完了次第、新しい同期が開始されます。この指定で実行頻度が高くなりすぎた場合は、dbmsync フック・プロシージャ・コール sp\_hook\_dbmsync\_delay を作成して、各同期処理の間に休止時間を強制的に設定できます。

```
CREATE SYNCHRONIZATION SUBSCRIPTION TO "p1" FOR "rem1"  
    TYPE 'TCPIP' ADDRESS 'host=localhost'  
    OPTION SV='v1',SCH='EVERY:00:00';
```

```
CREATE PROCEDURE sp_hook_dbmsync_delay()  
BEGIN  
    UPDATE #hook_dict  
        SET value = '20'  
    WHERE name = 'maximum accumulated delay';  
    UPDATE #hook_dict  
        SET value = '5'  
    WHERE name = 'delay duration';  
END;
```

sp\_hook\_dbmsync\_delay プロシージャの使用時には、次のスケジュール時刻よりも早く同期を実行する必要がある場合、同期の開始タイミングをプログラムすることも可能です。

sp\_hook\_dbmsync\_delay プロシージャで遅延時間の値をゼロに設定した場合、dbmsync による

同期は遅延を伴わずに実行されます。たとえば、"SynchNow" という名前のテーブルを作成し、"SynchNow" テーブルにローが存在する限り、sp\_hook\_dbmsync\_delay プロシージャで遅延時間 ('delay duration') をゼロに設定するという方法が考えられます。この方法では、"SynchNow" テーブルにローを挿入することで、アプリケーションから同期を強制的に実行できます。

この方法は、1 分よりも短い間隔で同期が発生する場合にも使用できることに注意してください。下記の例では、30 分間隔で同期を実行していますが、sp\_hook\_dbmsync\_delay プロシージャは 1 分間隔で実行され、同期を強制的に実行するかどうかの判定が行われます。この方法は、必要なタイミングで簡単に同期を強制的に実行できるという点で、スケジュールを 'EVERY:00:30' に設定する方法よりも有効です。

```
CREATE SYNCHRONIZATION SUBSCRIPTION TO "p1" FOR "rem1"
```

```
    TYPE 'TCPIP' ADDRESS 'host=localhost'
```

```
    OPTION SV='v1',SCH='EVERY:00:00';
```

```
CREATE PROCEDURE sp_hook_dbmsync_delay()
```

```
BEGIN
```

```
    IF EXISTS ( SELECT 1 FROM SynchNow ) THEN
```

```
        UPDATE #hook_dict
```

```
            SET value = '0'
```

```
        WHERE name = 'delay duration';
```

```
    DELETE FROM SynchNow;
```

```
ELSE
```

```
    UPDATE #hook_dict
```

```
        SET value = '60'
```

```
WHERE name = 'delay duration';
```

```
END IF;
```

```
UPDATE #hook_dict
```

```
SET value = '1800'
```

```
WHERE name = 'maximum accumulated delay';
```

```
END;
```

-以上-

.