

## Re: => Trigger to split Trailer Loads

**Source:**

<http://www.tech-archive.net/Archive/SQL-Server/microsoft.public.sqlserver.mseq/2004-04/0052.html>

---

**From:** Rhonda Fischer (*anonymous\_at\_discussions.microsoft.com*)

**Date:** 04/14/04

Date: Wed, 14 Apr 2004 02:56:06 -0700

Hello Hugo,

Thank you very much for your assistance, it was most helpful. It is a big problem and I'm trying to break it down.

I've started a solution that partially works, however have not as yet given consideration to deleting split orders.

I am using a Trigger and a cursor with a while loop, however the recursive behaviour of the cursor is causing me some problems, unless it's the algorithm I've used.

I can successfully copy an order from tblOrders to tblSplitOrders, duplicating order details and splitting the load into multiples of 26 and a final order with the remaining loads. However if there are any more than one table insert I have incorrect orders copied over.

\*\*\*\*\*

**WORKING EXAMPLE:**

\*\*\*\*\*

```
INSERT INTO tblOrders (DeliveryID, CollectionID, TempID, OrderDate, NumOfLoads)
VALUES ( 8, 8, 3, '2004-04-13', 106)
```

\*\*\*\*\*

My Trace Statements in the Trigger

Load values of first record into local variables

Enter Cursor

Enters LineSplit >0

Split load by multiples of 26 – loop and insert here

(1 row(s) affected)

Split load by multiples of 26 – loop and insert here

(1 row(s) affected)

Split load by multiples of 26 – loop and insert here

microsoft.public.sqlserver.mseq: Re: => Trigger to split Trailer Loads

(1 row(s) affected)

Split load by multiples of 26 – loop and insert here

(1 row(s) affected)

Insert Loads Remainder

(1 row(s) affected)

Loop to next record in the cursor

Close the Cursor

(1 row(s) affected)

\*\*\*\*\*

Data in tblSplitOrders

ID DeliveryID CollectionID TempID OrderDate NumOfLoads

111 8 8 3 13/04/2004 26

112 8 8 3 13/04/2004 26

113 8 8 3 13/04/2004 26

114 8 8 3 13/04/2004 26

115 8 8 3 13/04/2004 2

\*\*\*\*\*

And finally the Trigger:

```
CREATE TRIGGER trgLoadExceedTrlCapacity ON [tblOrders]
```

```
FOR INSERT
```

```
AS
```

```
    /* The purpose of this trigger is to ensure that no entries have in excess of 26 pallet loads */
```

```
    /* The Supermarket may order more than 26 loads of produce but the Trailer can only manage 26 loads */
```

```
    /* so extra lines are inserted into the new table tblSplitOrders with a maximum of 26 loads */
```

```
    Declare @intDeliveryID int, @intCollectionID int, @intTempID int, @dteOrderDate datetime,
```

```
            @intNumOfLoads int, @intLineSplit int, @intLoadsRemainder int, @intMaxLoads int,
```

```
    @LoadExcessCursor Cursor
```

```
    Set @intMaxLoads = 26
```

```
    Set @LoadExcessCursor = Cursor For
```

```
        Select DeliveryID, CollectionID, TempID, OrderDate, NumOfLoads,
```

```
            cast(round(NumOfLoads/@intMaxLoads, 0) as int) as LineSplit,
```

```
            NumOfLoads – (cast(round(NumOfLoads/@intMaxLoads, 0) as int) * @intMaxLoads) as
```

```
    LoadsRemainder
```

```
    From tblOrders
```

```
    Open @LoadExcessCursor
```

```
    /*Load values of first record into local variables */
```

```
    print 'Load values of first record into local variables'
```

Re: => Trigger to split Trailer Loads

microsoft.public.sqlserver.mseq: Re: => Trigger to split Trailer Loads

```
Fetch Next From @LoadExcessCursor
Into @intDeliveryID, @intCollectionID, @intTempID, @dteOrderDate, @intNumOfLoads,
@intLineSplit, @intLoadsRemainder

While (@@FETCH_STATUS = 0)
begin
print 'Enter Cursor'
if @intLineSplit = 0
begin
print 'Enters LineSplit = 0'
/* NumOfLoads between 0-26 insert without splitting order lines*/
INSERT INTO tblSplitOrders (DeliveryID, CollectionID, TempID, OrderDate, NumOfLoads)
VALUES ( @intDeliveryID, @intCollectionID, @intTempID, @dteOrderDate,
@intNumOfLoads)
end
else
if @intLineSplit > 0
begin
print 'Enters LineSplit >0'
/* NumOfLoads greater than 26 - Split Loads to have a maximum of 26 Loads per record */
/* loop to Insert split loads */
While @intLineSplit >= 1
begin
print 'Split load by multiples of 26 - loop and insert here'
INSERT INTO tblSplitOrders (DeliveryID, CollectionID, TempID, OrderDate,
NumOfLoads)
VALUES ( @intDeliveryID, @intCollectionID, @intTempID, @dteOrderDate,
@intMaxLoads)
Set @intLineSplit = @intLineSplit - 1
end
/* insert last entry with NumOfLoads equal to LoadsRemainder */
print 'Insert Loads Remainder'
INSERT INTO tblSplitOrders (DeliveryID, CollectionID, TempID, OrderDate, NumOfLoads)
VALUES ( @intDeliveryID, @intCollectionID, @intTempID, @dteOrderDate,
@intLoadsRemainder)
end
print 'Loop to next record in the cursor'
Fetch Next From @LoadExcessCursor
Into @intDeliveryID, @intCollectionID, @intTempID, @dteOrderDate, @intNumOfLoads,
@intLineSplit, @intLoadsRemainder
end

print 'Close the Cursor'
Close @LoadExcessCursor
Deallocate @LoadExcessCursor
```

\*\*\*\*\*

NOT WORKING EXAMPLE - THREE INSERTS INTO tblOrders

\*\*\*\*\*

I then delete all data from the two tables and start again. Just to keep it a bit simpler I have not included any loads

greater than 26 to be split.

```
INSERT INTO tblOrders (DeliveryID, CollectionID, TempID, OrderDate, NumOfLoads)
VALUES ( 8, 8, 3, '2004-04-13', 2)
```

```
INSERT INTO tblOrders (DeliveryID, CollectionID, TempID, OrderDate, NumOfLoads)
VALUES ( 9, 9, 3, '2004-04-13', 5)
```

```
INSERT INTO tblOrders (DeliveryID, CollectionID, TempID, OrderDate, NumOfLoads)
VALUES ( 10, 9, 3, '2004-04-13', 9)
```

\*\*\*\*\*

Again my trace statements in the trigger

Load values of first record into local variables

Enter Cursor

Enters LineSplit = 0

(1 row(s) affected)

Loop to next record in the cursor

Close the Cursor

(1 row(s) affected)

Load values of first record into local variables

Enter Cursor

Enters LineSplit = 0

(1 row(s) affected)

Loop to next record in the cursor

Enter Cursor

Enters LineSplit = 0

(1 row(s) affected)

Loop to next record in the cursor

Close the Cursor

(1 row(s) affected)

Load values of first record into local variables

Enter Cursor

Enters LineSplit = 0

(1 row(s) affected)

Loop to next record in the cursor

Enter Cursor

Enters LineSplit = 0

(1 row(s) affected)

microsoft.public.sqlserver.mseq: Re: => Trigger to split Trailer Loads

Loop to next record in the cursor  
Enter Cursor  
Enters LineSplit = 0

(1 row(s) affected)

Loop to next record in the cursor  
Close the Cursor

(1 row(s) affected)

\*\*\*\*\*

And the output in my tblSplitOrders table

ID	DeliveryID	CollectionID	TempID	OrderDate	NumOfLoads
116	8	8	3	13/04/2004	2
117	8	8	3	13/04/2004	2
118	9	9	3	13/04/2004	5
119	8	8	3	13/04/2004	2
120	9	9	3	13/04/2004	5
121	10	9	3	13/04/2004	9

\*\*\*\*\*

I'm not sure why it doesn't work, I expected to copy over three entries of loads 2, 5 and 9. I see the path that the code has taken but can't understand why it exits the while loop and closes the cursor for some entries and not others.

Are you able to offer me some advice? I've been working hard on this problem and reading lots and your help would be very much appreciated.

Thank you kindly  
Rhonda