

Performance question on triggers

Source:

<http://www.tech-archive.net/Archive/SQL-Server/microsoft.public.sqlserver.programming/2004-04/1596.html>

From: Bernd Maierhofer \((dato\) (*bernd.maierhofer_at_dato.at*)

Date: 04/07/04

Date: Wed, 7 Apr 2004 14:38:21 +0200

Hi and thanks for reading:

MSSQL 2000 Given a table

Table A

id integer

x,a,b,c float

x is calculated from the values a,b,c of the rows < id id is the primary key

Up to now I calculated x at runtime using a function:

```
select id,a,b,c,f_func(id) as x f_func(id) basically is a select sum(a)
where id < @id
```

But as the table grows, this is too slow. As changes to a,b,c usually occur for the newest records only, I wanted to switch to using a field x wich is only calculated when a,b,c for one of the preceeding rows is changed. Thus by using a trigger:

```
create trigger xx on a
after insert, update, delete as
```

```
set nocount off
```

```
IF ( (SELECT trigger_nestlevel( object_ID('lsmp') ) ) = 1 ) begin
```

```
  update a set x = f_func(id) where id in (select id from deleted)
```

```
  update a set x = f_func(id) where id in (select id from inserted)
```

```
end
```

```
set nocount on
```

```
end
```

The performance is very poor, as the trigger is fired recursively. One option is to disable recursive and/or nested triggers. Is it possible to issue a

microsoft.public.sqlserver.programming: Performance question on triggers

sp_configure 'nested_triggers','0' within the trigger? And would this inhibit firing the trigger each time, I update the table?

Any idea? Any different approach?

TIA
Bernd