simple talk

Se

Pop Rivett's SQL Server FAQ No.5: Pop on the Audit Trail

03 November 2006
by Pop Rivett

Pop does SQL Server auditing

Good morning Pop, I see the turnips are doing well

Well, God bless you and keep you Miss, they do well, but we could do with a drop of rain. But do tell me, Miss why the long face?

Oh Pop…such a dreadful thing! I've just been checking through the accounts of the Womens' Institute and someone has hacked in and removed everything we had in savings for the new Village Hall!

Lorks a'mercy, missy…but at least your audit trail will show up how it happened and who it was that did it?

*What audit-tail?! [**Subsiding into sobs**] We haven't got such a thing. The vicar's son put the system together as a college project. Frankly, it's lucky that the tables even have primary keys.*

Well, when IT people talk of audit trails, they refer to keeping an auditable account of who changed the data in a table, when, from what and to what. Basically, an audit trail is a way of tracking the details of all changes to data in certain database tables – usually in order to check the integrity of financial transactions. It provides a record of who has accessed a database and what operations he or she has performed during a given period of time.

When data is financial, this is often built into the system. Records are never updated or deleted, merely superseded, or 'terminated', so that databases grow to become a history of all financial transactions. The application programmers are allowed access to the database only through stored procedures.

If you don't have the luxury of a properly-designed financial system, as you don't by the sounds of it, then you need to devise a means of logging all the transactions that you want to monitor. You might try looking in the database transaction log but the problem with this is that it's an indiscriminate record of all transactions, whether data-feeds, cosmetic changes or data modifications – so it's hard to use it to monitor and track down particular financial business transactions.

So what's to be done then, Pop?!

Well, by my reckoning, an audit trigger is the way to go for you. Take a look at this…you simply create one of these triggers for every table that you need to monitor. This trigger will provide an 'audit trail' of all changes made to a table. In the line **select @TableName = 'trigtest'** you simply swap **trigrest** for the name of your table, and then the trigger will monitor all changes to that table.

It will place a record of these changes in a table called **Audit**. It records in **Audit** details of all inserted, deleted, changed columns and the old/new values of the fdata that was altered in the table that the trigger protects. Anyway, here is the code…notice that it will put out an error message if there is no primary key on the table as the routine would then be unable to identify the affected rows…so it's lucky the vicar's son got that right at least!

```
-- Set up the tables
-- Firstly, we create the audit table.
-- There will only need to be one of these in a database


IF NOT EXISTS
      (SELECT * FROM sysobjects WHERE id = OBJECT_ID(N'[dbo].[Audit]')
              AND OBJECTPROPERTY(id, N'IsUserTable') = 1)
      CREATE TABLE Audit
              (Type CHAR(1),
              TableName VARCHAR(128),
              PK VARCHAR(1000),
              FieldName VARCHAR(128),
              OldValue VARCHAR(1000),
              NewValue VARCHAR(1000),
              UpdateDate datetime,
              UserName VARCHAR(128))
  GO

-- now we will illustrate the use of this tool
-- by creating a dummy test table called TrigTest.
```

```sql
IF EXISTS
    (SELECT * FROM sysobjects WHERE id = OBJECT_ID(N'[dbo].[trigtest]')
                AND OBJECTPROPERTY(id, N'IsUserTable') = 1)
DROP TABLE [dbo].[trigtest]
GO
CREATE TABLE trigtest
        (i INT NOT NULL,
         j INT NOT NULL,
         s VARCHAR(10),
         t VARCHAR(10))
GO

--note that for this system to work there must be a primary key
--to the table but then a table without a primary key
--isn't really a table is it?
ALTER TABLE trigtest ADD CONSTRAINT pk PRIMARY KEY (i, j)
GO

--and now create the trigger itself. This has to be created for every
-table you want to monitor

CREATE TRIGGER tr_trigtest ON trigtest FOR INSERT, UPDATE, DELETE
AS

DECLARE @bit INT ,
        @field INT ,
        @maxfield INT ,
        @char INT ,
        @fieldname VARCHAR(128) ,
        @TableName VARCHAR(128) ,
        @PKCols VARCHAR(1000) ,
        @sql VARCHAR(2000),
        @UpdateDate VARCHAR(21) ,
        @UserName VARCHAR(128) ,
        @Type CHAR(1) ,
        @PKSelect VARCHAR(1000)


--You will need to change @TableName to match the table to be audited
SELECT @TableName = 'trigtest'

-- date and user
SELECT          @UserName = SYSTEM_USER ,
        @UpdateDate = CONVERT(VARCHAR(8), GETDATE(), 112)
                + ' ' + CONVERT(VARCHAR(12), GETDATE(), 114)

-- Action
IF EXISTS (SELECT * FROM inserted)
        IF EXISTS (SELECT * FROM deleted)
                SELECT @Type = 'U'
```

```sql
        ELSE
                SELECT @Type = 'I'
ELSE
        SELECT @Type = 'D'

-- get list of columns
SELECT * INTO #ins FROM inserted
SELECT * INTO #del FROM deleted

-- Get primary key columns for full outer join
SELECT @PKCols = COALESCE(@PKCols + ' and', ' on')
                + ' i.' + c.COLUMN_NAME + ' = d.' + c.COLUMN_NAME
        FROM    INFORMATION_SCHEMA.TABLE_CONSTRAINTS pk ,

                INFORMATION_SCHEMA.KEY_COLUMN_USAGE c
        WHERE   pk.TABLE_NAME = @TableName
        AND     CONSTRAINT_TYPE = 'PRIMARY KEY'
        AND     c.TABLE_NAME = pk.TABLE_NAME
        AND     c.CONSTRAINT_NAME = pk.CONSTRAINT_NAME

-- Get primary key select for insert
SELECT @PKSelect = COALESCE(@PKSelect+'+','')
        + '''<' + COLUMN_NAME
        + '=''+convert(varchar(100),
coalesce(i.' + COLUMN_NAME +',d.' + COLUMN_NAME + '))+''>'''
        FROM    INFORMATION_SCHEMA.TABLE_CONSTRAINTS pk ,
                INFORMATION_SCHEMA.KEY_COLUMN_USAGE c
        WHERE   pk.TABLE_NAME = @TableName
        AND     CONSTRAINT_TYPE = 'PRIMARY KEY'
        AND     c.TABLE_NAME = pk.TABLE_NAME
        AND     c.CONSTRAINT_NAME = pk.CONSTRAINT_NAME

IF @PKCols IS NULL
BEGIN
        RAISERROR('no PK on table %s', 16, -1, @TableName)
        RETURN
END

SELECT          @field = 0,
        @maxfield = MAX(ORDINAL_POSITION)
        FROM INFORMATION_SCHEMA.COLUMNS WHERE TABLE_NAME = @TableName
WHILE @field < @maxfield
BEGIN
        SELECT @field = MIN(ORDINAL_POSITION)
                FROM INFORMATION_SCHEMA.COLUMNS
                WHERE TABLE_NAME = @TableName
                AND ORDINAL_POSITION > @field
        SELECT @bit = (@field - 1 )% 8 + 1
        SELECT @bit = POWER(2,@bit - 1)
        SELECT @char = (((@field - 1) / 8) + 1
```

```sql
        IF SUBSTRING(COLUMNS_UPDATED(),@char, 1) & @bit > 0
                                    OR @Type IN ('I','D')
        BEGIN
                SELECT @fieldname = COLUMN_NAME
                        FROM INFORMATION_SCHEMA.COLUMNS
                        WHERE TABLE_NAME = @TableName
                        AND ORDINAL_POSITION = @field
                SELECT @sql = '
insert Audit (    Type,
                TableName,
                PK,
                FieldName,
                OldValue,
                NewValue,
                UpdateDate,
                UserName)
select ''' + @Type + ''',''
        + @TableName + ''',' + @PKSelect
        + ',''' + @fieldname + ''''
        + ',convert(varchar(1000),d.' + @fieldname + ')'
        + ',convert(varchar(1000),i.' + @fieldname + ')'
        + ',''' + @UpdateDate + ''''
        + ',''' + @UserName + ''''
        + ' from #ins i full outer join #del d'
        + @PKCols
        + ' where i.' + @fieldname + ' <> d.' + @fieldname
        + ' or (i.' + @fieldname + ' is null and  d.'
                                + @fieldname
                                + ' is not null)'
        + ' or (i.' + @fieldname + ' is not null and  d.'
                                + @fieldname
                                + ' is null)'
                EXEC (@sql)
        END
END

GO


-------------------------------------------------------

--now we can test the trigger out

INSERT trigtest SELECT 1,1,'hi', 'bye'

INSERT trigtest SELECT 2,2,'hi', 'bye'

INSERT trigtest SELECT 3,3,'hi', 'bye'

SELECT * FROM Audit
```

```sql
SELECT * FROM trigtest

UPDATE trigtest SET s = 'hibye' WHERE i <> 1

UPDATE trigtest SET s = 'bye' WHERE i = 1

UPDATE trigtest SET s = 'bye' WHERE i = 1

UPDATE trigtest SET t = 'hi' WHERE i = 1

SELECT * FROM Audit

SELECT * FROM trigtest

DELETE trigtest

SELECT * FROM Audit

SELECT * FROM trigtest

GO

DROP TABLE Audit

GO

DROP TABLE trigtest

GO
```

Why Pop, you're a life saver!

That's quite all right, missy. Just don't forget that there is, of course, an overhead to this technique. Also, bear in mind the potential problems of using the @@Identity variable when you have triggers on tables with **IDENTITY** columns.

Thank this author by sharing: G+1 in f y 0

This article has been viewed 77576 times.

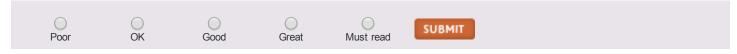**Author profile:** Pop Rivett

Pop spent his formative years working in assembler on IBM Series/1 but retrained in VB when that went out of fashion. He soon realised how little relational database expertise existed in most companies and so started to spend most of his time working on that. He now sticks to architecture, release control / IT processes, SQL Server, DTS, SSIS, and access methods in VB/ASP/.NET/Crystal Reports/reporting services. He has been involved with SQL Server since the old days of v4.2 to v2005. He tries to stay away from anything presentation oriented (see www.mindsdoor.net). Theoretically he is semi-retired but seems to keep being offered potentially interesting work.

Search for other articles by Pop Rivett

Rate this article: Avg rating: ★★★★☆ from a total of 89 votes.

○ ○ ○ ○ ○ **SUBMIT**
Poor OK Good Great Must read

Have Your Say

Do you have an opinion on this article? Then **add your comment below**:

You must be logged in to post to this forum

Click here to log in.

Subject:	**The source**
Posted by:	*Phil Factor* (view profile)
Posted on:	*Friday, November 3, 2006 at 2:46 PM*
Message:	I tried to copy the source and paste it into the query analyser to try it out, but it all came out in one long line! It works fine copying it from Firefox, so it seems to be an IE thing. I suspect that the article editor has mangled the HTML again. In case anyone else hits the problem, I've put the source HERE as a text file.
	Well I eventually tried it out and it seems to work devilish well. Hurrah for Pop!

Subject:	**re: source code**
Posted by:	*Tony Davis* (view profile)
Posted on:	*Sunday, November 5, 2006 at 6:25 AM*
Message:	I've now included the source as a zip file - just click the "Code Download" link in the bubble to the right of the article title.
	Tony (Simple-Talk Ed)

Subject:	**Aggregate columns**
Posted by:	*Anonymous* (not signed in)
Posted on:	*Monday, November 6, 2006 at 4:36 AM*
Message:	This script is sweet, and with code generators a breeze to implement widely.
	I was thinking about a small app to visualize the audit table... the audit data would be much easier to analyze if the deleted and inserted rows where compiled into the OldValue and NewValue records, getting one record in audit for each Insert, Update or Delete.
	Or even the totality of the record before and after...
	Alas, my SQL belt is a dirty shade of white at best, any tips on how to do that?

Subject: article

Posted by:	*Anonymous* (not signed in)
Posted on:	*Thursday, November 9, 2006 at 4:47 AM*
Message:	This is a great article. We have tried something similar to this type of trigger for a different purpose. When we had to transfer only the modified data from one server to another. Due to bandwidth available being low, we used this method to check what has changed , took only that and transferred

Subject:	**Funny**
Posted by:	*Anonymous* (not signed in)
Posted on:	*Tuesday, November 21, 2006 at 7:00 AM*
Message:	quote: but then a table without a primary key isn't really a table is it?
	So the audit table isn't really a table I suppose?

Subject:	**It worked and now it doesn't**
Posted by:	*KC* (not signed in)
Posted on:	*Tuesday, December 26, 2006 at 5:16 PM*
Message:	Hi.. hopefully someone here has experienced this before.
	OK I downloaded the code and tested it against my own database on SQL Server 2000. It worked fine.
	Now I've imposed the trigger on my live table at my company and it doesn't work. It keeps telling me the PK does not exist on the table. Of course it does. I'm using an identical table to the one that I used in my test. What am I doing wrong?
	Thanks.

Subject:	**Strange familiarity**
Posted by:	*Anonymous* (not signed in)
Posted on:	*Tuesday, January 9, 2007 at 4:11 PM*
Message:	<i>>What audit-tail?! [Subsiding into sobs] We
	> haven't got such a thing. The vicar's son put the
	> system together as a college project. Frankly,
	> it's lucky that the tables even have primary keys.</i>
	sounds like the place where i work.

Subject:	**Who's ripping off who?**
Posted by:	*Anonymous* (not signed in)
Posted on:	*Monday, January 15, 2007 at 11:07 AM*
Message:	http://www.nigelrivett.net/SQLTriggers/AuditTrailTrigger.html

Subject:	**Re: Who's ripping off who?**
Posted by:	*Tony Davis* (view profile)
Posted on:	*Wednesday, January 17, 2007 at 5:40 AM*
Message:	
	Nobody is ripping off anybody ;). We've done a deal with Nigel that allows up to use the "notes" he has on his website and turn them into Simple-Talk articles.
	Cheers,
	Tony (Simple-Talk Ed.)

Subject:	**Re: Who's ripping off who?**

Posted by:	*Phil Factor* (view profile)
Posted on:	*Friday, January 26, 2007 at 2:49 AM*
Message:	Nigel Rivett is author of
	http://www.nigelrivett.net/SQLTriggers/AuditTrailTrigger.html
	and co-author of this article. The material is, obviously, used with
	permission.

Subject:	**Auditing different Username**
Posted by:	*Anonymous* (not signed in)
Posted on:	*Monday, March 19, 2007 at 5:14 AM*
Message:	Hi,
	the code works fine but I need to audit not the system_user (it's
	always the same, the one in the connection string) but the username
	used to log in to the system.
	How can i do ??
	Many thanks

Subject:	**How to apply audit trail trigger to all tables**
Posted by:	*Anonymous* (not signed in)
Posted on:	*Tuesday, March 20, 2007 at 6:06 AM*
Message:	Ok, the trigger works.
	How can i apply it on all table of my database without creating N
	"CREATE TRIGGER" statement ? Is it possible to apply the trigger
	on all tables in "one shot" ?
	Thanks

Subject:	**How do create one or more table in auditrail in sqlserver**
Posted by:	*Anonymous* (not signed in)
Posted on:	*Tuesday, March 20, 2007 at 9:23 AM*
Message:	Its very useful part of this me.i am working in sofrware line so i want
	to create in the audittrail,so i want to this part,so please consider my .

Subject:	**Re: How to apply audit trail trigger to all tables**
Posted by:	*Phil Factor* (view profile)
Posted on:	*Tuesday, March 20, 2007 at 5:35 PM*
Message:	No, it is a shame but a trigger is 'owned' by a particular table. A
	trigger cannot be made to fire on actions to other tables.

Subject:	**Audittrail**
Posted by:	*Anonymous* (not signed in)
Posted on:	*Monday, March 26, 2007 at 4:40 AM*
Message:	The above statement the have no insert,delete,update responds to
	the database.
	place verify the statement.

Subject:	**Performace**
Posted by:	*Nona* (not signed in)
Posted on:	*Wednesday, June 27, 2007 at 11:46 AM*
Message:	Wondering what is the performance hit for this SP.

Subject:	**Composite primary keys?**
Posted by:	*Anonymous* (not signed in)
Posted on:	*Monday, July 16, 2007 at 3:28 AM*
Message:	Great article works fine! However doesn't seem to work with tables
	with composite primary keys. Is there any way to modify the code for
	this to work or should this work anyway?

Subject: **nTEXT, TEXT, IMAGE Issue**
Posted by: *a4400* (not signed in)
Posted on: *Thursday, August 23, 2007 at 8:49 AM*
Message: The script is practically useless , when it comes to auditing table with
ntext, text and image type columns.

MS SQL Server do not allow acces to the ntext, text and image
columns inside a BEFORE/AFTER trigger.

The only solution is to JOIN the inserted with the original table.

For the deleted table i don't see any solution for now. may you have
ideas.

Subject: **Similar**
**Posted
by:**   *Cornjerker* (not signed in)
**Posted
on:**   *Sunday, January 27, 2008 at 8:27 AM*
Message:I did something similar in summer of '05.

http://groups.google.com/group/microsoft.public.sqlserver.programming/msg/555e9e02ad464cff?
dmode=source

I have since modified it quite a bit over the years to handle added and deleted records. We had
a situation where we needed to track usernames but we were using a common username for
database access. So you can add the username to the "Application Name" setting of the
connection string. With SQL Server 2005 I've changed varchar(8000) to varchar(max).

Subject: **Current code**
Posted by: *Cornjerker* (not signed in)
Posted on: *Sunday, January 27, 2008 at 8:35 AM*
Message: I decided to post my current version. One issue I recently hit was a
user had created a table with about 20 varchar(max) fields. Running
update queries smacked my server. I'll need to fix that somehow. I
must say though, that using this at the database level has saved me
countless hours in not have to develop this for every application I
write. Enjoy!

```
USE [YOUR_DATABASE_NAME_HERE]
GO
/****** Object: StoredProcedure [dbo].[sp_CreateLogTrigger_v5] Script
Date: 01/27/2008 08:16:56 ******/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
ALTER PROCEDURE [dbo].[sp_CreateLogTrigger_v5] (@tableName
varchar(50)) AS

DECLARE @fieldName varchar(50)
DECLARE @fieldType as varchar(15)
DECLARE @trigText varchar(max)
DECLARE @CR as varchar(2)

SET @trigText = ''
SET @CR = char(10)

-----DELETE CURRENT LOG TRIGGERS IF THEY EXIST -----
IF EXISTS (select * from dbo.sysobjects where id = object_id(N'[dbo].
```

```
[trig_Log_Update_' + @tableName + ']') and OBJECTPROPERTY(id,
N'IsTrigger') = 1)
EXEC ('DROP TRIGGER dbo.trig_Log_Update_' + @tableName)

IF EXISTS (select * from dbo.sysobjects where id = object_id(N'[dbo].
[trig_Log_Insert_' + @tableName + ']') and OBJECTPROPERTY(id,
N'IsTrigger') = 1)
EXEC ('DROP TRIGGER dbo.trig_Log_Insert_' + @tableName)

IF EXISTS (select * from dbo.sysobjects where id = object_id(N'[dbo].
[trig_Log_Delete_' + @tableName + ']') and OBJECTPROPERTY(id,
N'IsTrigger') = 1)
EXEC ('DROP TRIGGER dbo.trig_Log_Delete_' + @tableName)
----------------------------------------------------------------


-----CREATE TRIGGER FOR UPDATING RECORDS-----
DECLARE field_cursor CURSOR FOR
Select upper(column_name) as field, upper(data_type) as data_type
From INFORMATION_SCHEMA.COLUMNS
Where TABLE_SCHEMA = 'dbo' And TABLE_NAME = @tableName
Order By ORDINAL_POSITION

OPEN field_cursor

FETCH NEXT FROM field_cursor INTO @fieldName, @fieldType

WHILE @@FETCH_STATUS = 0
BEGIN
IF @fieldName <> 'pk'
BEGIN
SET @trigText = @trigText + 'INSERT INTO tLog (field_name,
old_value, new_value, app_name, uid, dt, table_name, row_pk) ' +
@CR
IF @fieldType='FLOAT'
SET @trigText = @trigText + 'SELECT ''' + @fieldName + ''',
CONVERT(DECIMAL(15,2), d.' + @fieldName + '),
CONVERT(DECIMAL(15,2), i.' + @fieldName + '), APP_NAME(),
SUSER_SNAME(), GETDATE(), '''+@tableName+''', d.pk ' + @CR
ELSE
SET @trigText = @trigText + 'SELECT ''' + @fieldName + ''', d.' +
@fieldName + ', i.' + @fieldName + ', APP_NAME(),
SUSER_SNAME(), GETDATE(), '''+@tableName+''', d.pk ' + @CR
SET @trigText = @trigText + 'FROM deleted d ' + @CR
SET @trigText = @trigText + 'JOIN inserted i on d.pk = i.pk ' + @CR
SET @trigText = @trigText + 'WHERE d.' + @fieldName + '<>i.' +
@fieldName + '' + @CR
SET @trigText = @trigText + 'OR (d.' + @fieldName + ' is null AND i.'
+ @fieldName + ' is not null) ' + @CR
SET @trigText = @trigText + 'OR (d.' + @fieldName + ' is not null
AND i.' + @fieldName + ' is null) ' + @CR + @CR
END

FETCH NEXT FROM field_cursor INTO @fieldName, @fieldType
END

CLOSE field_cursor
DEALLOCATE field_cursor

EXEC ('
```

```
CREATE TRIGGER [trig_Log_Update_' + @tableName + '] ON [' +
@tableName + ']
FOR UPDATE
AS

' + @trigText + '

')

----- CREATE TRIGGER FOR INSERTING RECORDS -----
SET @trigText = ''

DECLARE field_cursor_ins CURSOR FOR
Select upper(column_name) as field, upper(data_type) as data_type
From INFORMATION_SCHEMA.COLUMNS
Where TABLE_SCHEMA = 'dbo' And TABLE_NAME = @tableName
Order By ORDINAL_POSITION

OPEN field_cursor_ins

FETCH NEXT FROM field_cursor_ins INTO @fieldName, @fieldType

WHILE @@FETCH_STATUS = 0
BEGIN
IF @fieldType='FLOAT'
SET @trigText = @trigText + 'SET @newValue = (select
CONVERT(DECIMAL(15,2), ' + @fieldName + ') from inserted) ' +
@CR
ELSE
SET @trigText = @trigText + 'SET @newValue = (select ' +
@fieldName + ' from inserted) ' + @CR

SET @trigText = @trigText + 'INSERT INTO tLog (field_name,
old_value, new_value, app_name, uid, dt, table_name, row_pk)
values ('''+@fieldName+''', ''--new record--'', @newValue,
APP_NAME(), USER, GETDATE(), '''+@tableName+''', @row_pk) ' +
@CR + @CR

FETCH NEXT FROM field_cursor_ins INTO @fieldName, @fieldType
END

CLOSE field_cursor_ins
DEALLOCATE field_cursor_ins


EXEC ('

CREATE TRIGGER [trig_Log_Insert_' + @tableName + '] ON [' +
@tableName + ']
FOR INSERT
AS

DECLARE @newValue as varchar(max)
DECLARE @row_pk as int

SET @row_pk = (select pk from inserted)

' + @trigText + '

')
```

----- CREATE TRIGGER FOR DELETING RECORDS -----

EXEC ('

```
CREATE TRIGGER [trig_Log_Delete_' + @tableName + '] ON [' +
@tableName + ']
FOR DELETE
AS

DECLARE @row_pk as int
SET @row_pk = (select pk from deleted)
INSERT INTO tLog (new_value, app_name, uid, dt, table_name,
row_pk) values ("--record deleted--", APP_NAME(), USER,
GETDATE(), "'+@tableName+'", @row_pk)


')
```

Subject:	**nTEXT, TEXT, IMAGE Issue**
Posted by:	*AppDBA* (not signed in)
Posted on:	*Monday, February 18, 2008 at 1:14 AM*
Message:	Same issue as a4400, the following trigger code will not compile (as per BOL) if the table contains text, ntext or image data: SELECT * INTO #ins FROM INSERTED SELECT * INTO #del FROM DELETED I've tried workarounds e.g. building a temp copy of the base table excluding the offending columns but then running into scope issues referencing the INSERTED/DELETED virtual tables from an exec statement. Bah!

Subject:	**Web Access**
Posted by:	*Ed Dror* (not signed in)
Posted on:	*Thursday, March 6, 2008 at 5:36 PM*
Message:	It working fine on 2005 Except binary (Image) filed How to skip them? But what about accessing the data from the Internet? it show only one user all the time How do I integrate the ASP.NET security (Login user) with SQL data?
	Thanks, Ed Dror odeddror@cox.net

Subject:	**Web Access**
Posted by:	*Anonymous* (not signed in)
Posted on:	*Sunday, May 18, 2008 at 8:48 PM*
Message:	Ed Dror, still connecting to SQL Server with a fixed user..the "wanted" user name (the final user who changed the data) can be audited. To achieve that see the stored procedure "AuditDatabase_SetSessionInfo" in download are of AuditDatabase site (http://www.auditdatabase.com/Downloads.html).
	I think that with small modifications to "MGP Rivett" triggers you could integrate this stored procedure in your application to record the final user name
	Heber Silva

Subject:	**no line breaks? no word wrap**
Posted by:	*Anonymous* (not signed in)
Posted on:	*Monday, July 21, 2008 at 10:53 PM*
Message:	When the code is copied, just paste them into WordPad first.

From there, copy over to query window.

Subject:	**no line breaks? no word wrap**
Posted by:	*Anonymous* (not signed in)
Posted on:	*Tuesday, July 22, 2008 at 1:18 AM*
Message:	When the code is copied, just paste them into WordPad first.

From there, copy over to query window.

Subject:	**What does this code mean???**
Posted by:	*Anonymous* (not signed in)
Posted on:	*Tuesday, July 22, 2008 at 1:22 AM*
Message:	Coule anyboy be kind enough to explain me this? what is >>> COALESCE(@PKSelect+'+','') <<< ?

==

```
-- Get primary key select for insertSELECT @PKSelect =
COALESCE(@PKSelect+'+','') + '"<' + COLUMN_NAME +
'="'+convert(varchar(100),coalesce(i.' + COLUMN_NAME +',d.' +
COLUMN_NAME + '))+">"' FROM
INFORMATION_SCHEMA.TABLE_CONSTRAINTS pk ,
INFORMATION_SCHEMA.KEY_COLUMN_USAGE c WHERE
pk.TABLE_NAME = @TableName AND CONSTRAINT_TYPE =
'PRIMARY KEY' AND c.TABLE_NAME = pk.TABLE_NAME AND
c.CONSTRAINT_NAME = pk.CONSTRAINT_NAME
```

Subject:	**Database Audit Trail**
Posted by:	*John* (view profile)
Posted on:	*Wednesday, August 6, 2008 at 3:13 PM*
Message:	This article rocks!!!! If you write books I am buying. All code needs a massage, I expect this, you make it a no brainer. You are going to the top of my favorites.

Subject:	**Web Access**
Posted by:	*smcnerney* (view profile)
Posted on:	*Wednesday, October 15, 2008 at 2:09 PM*
Message:	I am trying to apply this audit trigger to my database and I am having the same problem with obtaining the username of the actual person logged into the Web Application. I am looking at the suggested code from the AuditDatabase_SetSessionInfo download but I am confused what this is doing and how I would integrate it into the trigger.

I am hoping this post is still active and someone by now may have had success in doing this and could give some pointers.

Subject:	**Auditing real username**
Posted by:	*manticore* (view profile)
Posted on:	*Friday, December 19, 2008 at 9:14 AM*
Message:	For what its worth I've successfully implemented (in ASP.NET) AuditDatabase_SetSessionInfo in order to get the real username audited.

Basically as far as I can tell you're looking at 2 methods, either add the AuditDatabase_SetSessionInfo SP code to your

insert/delete/update SP's and pass the
@ApplicationUser/@LogicalAddress variables to those SPs from your
application code.

Or

Use the same SQL connection object to run both the
AuditDatabase_SetSessionInfo SP and your database update logic.
Failure to use the same connection object means the session info is
stored in the wrong SQL session. In order to do this in ASP.Net
tableadapters I've handled the TableAdapter.Connection.StateChange
event. My handler sub creates a command object using the same
connection to run the AuditDatabase_SetSessionInfo SP. I've done it
this way as I can't see another (easy) way of hooking into tableadapter
connections when calling insert/delete/update methods.

Subject:	**Concatenating two of the fields**
Posted by:	*llwalker* (view profile)
Posted on:	*Tuesday, June 16, 2009 at 6:58 AM*
Message:	I love this script. It is great and a lifesaver.

I have been trying to adapt it to my needs and am running into a wall.
I need to modify the portion where the insert occurs to the audit table.
I would like to have the @fieldname and the two
convert(varchar(1000),d. and convert(varchar(1000),i. concatenated so
that they go into one field. The convert statements keep getting
entered as text instead of the real values.
This is my re-write on that insert statement.

```
INSERT INTO Name_log (date_time, log_type, sub_type, user_id, id,
log_text)
select '" + @UpdateDate + '"
+ ',"CHANGE"'
+ ',"CHANGE"'
+ ',"' + @UserName + '"
+ ',"' + @PKSelect + '"
+ ',"' + @fieldname + ' convert(varchar(1000),d.'+ ')'+ '>'+'
convert(varchar(1000),i.'+ ')"'
+ ' from #ins i full outer join #del d'
+ @PKCols
+ ' where i.' + @fieldname + ' <> d.' + @fieldname
+ ' or (i.' + @fieldname + ' is null and d.'
+ @fieldname
+ ' is not null)'
+ ' or (i.' + @fieldname + ' is not null and d.'
+ @fieldname
+ ' is null)'
```

I would appreciate any help. I've probably been looking at it too long
and am missing something very simple.
Thanks.

Subject:	**Create stored procedure**
Posted by:	*Justeena* (view profile)
Posted on:	*Wednesday, August 12, 2009 at 11:35 PM*
Message:	Can any one can send me how to create stored procedure for the following trigger where TableName as a parameter.So I can call this stored procedure only changing the table name.

```
CREATE TRIGGER tr_trigtest ON trigtest FOR INSERT, UPDATE,
DELETE
AS

DECLARE @bit INT ,
@field INT ,
@maxfield INT ,
@char INT ,
@fieldname VARCHAR(128) ,
@TableName VARCHAR(128) ,
@PKCols VARCHAR(1000) ,
@sql VARCHAR(2000),
@UpdateDate VARCHAR(21) ,
@UserName VARCHAR(128) ,
@Type CHAR(1) ,
@PKSelect VARCHAR(1000)


--You will need to change @TableName to match the table to be
audited
SELECT @TableName = 'trigtest'

-- date and user
SELECT @UserName = SYSTEM_USER ,
@UpdateDate = CONVERT(VARCHAR(8), GETDATE(), 112)
+ ' ' + CONVERT(VARCHAR(12), GETDATE(), 114)

-- Action
IF EXISTS (SELECT * FROM inserted)
IF EXISTS (SELECT * FROM deleted)
SELECT @Type = 'U'
ELSE
SELECT @Type = 'I'
ELSE
SELECT @Type = 'D'

-- get list of columns
SELECT * INTO #ins FROM inserted
SELECT * INTO #del FROM deleted

-- Get primary key columns for full outer join
SELECT @PKCols = COALESCE(@PKCols + ' and', ' on')
+ ' i.' + c.COLUMN_NAME + ' = d.' + c.COLUMN_NAME
FROM INFORMATION_SCHEMA.TABLE_CONSTRAINTS pk ,

INFORMATION_SCHEMA.KEY_COLUMN_USAGE c
WHERE pk.TABLE_NAME = @TableName
AND CONSTRAINT_TYPE = 'PRIMARY KEY'
AND c.TABLE_NAME = pk.TABLE_NAME
AND c.CONSTRAINT_NAME = pk.CONSTRAINT_NAME

-- Get primary key select for insert
SELECT @PKSelect = COALESCE(@PKSelect+'+',")
+ '''<' + COLUMN_NAME
+ '='+convert(varchar(100),
coalesce(i.' + COLUMN_NAME +',d.' + COLUMN_NAME + '))+'>'''
FROM INFORMATION_SCHEMA.TABLE_CONSTRAINTS pk ,
INFORMATION_SCHEMA.KEY_COLUMN_USAGE c
WHERE pk.TABLE_NAME = @TableName
AND CONSTRAINT_TYPE = 'PRIMARY KEY'
AND c.TABLE_NAME = pk.TABLE_NAME
```

```
               AND c.CONSTRAINT_NAME = pk.CONSTRAINT_NAME

               IF @PKCols IS NULL
               BEGIN
               RAISERROR('no PK on table %s', 16, -1, @TableName)
               RETURN
               END

               SELECT @field = 0,
               @maxfield = MAX(ORDINAL_POSITION)
               FROM INFORMATION_SCHEMA.COLUMNS WHERE
               TABLE_NAME = @TableName
               WHILE @field < @maxfield
               BEGIN
               SELECT @field = MIN(ORDINAL_POSITION)
               FROM INFORMATION_SCHEMA.COLUMNS
               WHERE TABLE_NAME = @TableName
               AND ORDINAL_POSITION > @field
               SELECT @bit = (@field - 1 )% 8 + 1
               SELECT @bit = POWER(2,@bit - 1)
               SELECT @char = ((@field - 1) / 8) + 1
               IF SUBSTRING(COLUMNS_UPDATED(),@char, 1) & @bit > 0 OR
               @Type IN ('I','D')
               BEGIN
               SELECT @fieldname = COLUMN_NAME
               FROM INFORMATION_SCHEMA.COLUMNS
               WHERE TABLE_NAME = @TableName
               AND ORDINAL_POSITION = @field
               SELECT @sql = '
               insert Audit ( Type,
               TableName,
               PK,
               FieldName,
               OldValue,
               NewValue,
               UpdateDate,
               UserName)
               select ''' + @Type + ''','''
               + @TableName + ''',' + @PKSelect
               + ',''' + @fieldname + ''''
               + ',convert(varchar(1000),d.' + @fieldname + ')'
               + ',convert(varchar(1000),i.' + @fieldname + ')'
               + ',''' + @UpdateDate + ''''
               + ',''' + @UserName + ''''
               + ' from #ins i full outer join #del d'
               + @PKCols
               + ' where i.' + @fieldname + ' <> d.' + @fieldname
               + ' or (i.' + @fieldname + ' is null and d.' + @fieldname + ' is not null)'
               + ' or (i.' + @fieldname + ' is not null and d.' + @fieldname + ' is null)'
               EXEC (@sql)
               END
               END
```

Subject:	**Create stored procedure**
Posted by:	*Justeena* (view profile)
Posted on:	*Thursday, August 13, 2009 at 1:11 AM*
Message:	Can any one can send me how to create stored procedure for the following trigger where TableName as a parameter.So I can call this stored procedure only changing the table name.

```sql
CREATE TRIGGER tr_trigtest ON trigtest FOR INSERT, UPDATE,
DELETE
AS

DECLARE @bit INT ,
@field INT ,
@maxfield INT ,
@char INT ,
@fieldname VARCHAR(128) ,
@TableName VARCHAR(128) ,
@PKCols VARCHAR(1000) ,
@sql VARCHAR(2000),
@UpdateDate VARCHAR(21) ,
@UserName VARCHAR(128) ,
@Type CHAR(1) ,
@PKSelect VARCHAR(1000)


--You will need to change @TableName to match the table to be
audited
SELECT @TableName = 'trigtest'

-- date and user
SELECT @UserName = SYSTEM_USER ,
@UpdateDate = CONVERT(VARCHAR(8), GETDATE(), 112)
+ ' ' + CONVERT(VARCHAR(12), GETDATE(), 114)

-- Action
IF EXISTS (SELECT * FROM inserted)
IF EXISTS (SELECT * FROM deleted)
SELECT @Type = 'U'
ELSE
SELECT @Type = 'I'
ELSE
SELECT @Type = 'D'

-- get list of columns
SELECT * INTO #ins FROM inserted
SELECT * INTO #del FROM deleted

-- Get primary key columns for full outer join
SELECT @PKCols = COALESCE(@PKCols + ' and', ' on')
+ ' i.' + c.COLUMN_NAME + ' = d.' + c.COLUMN_NAME
FROM INFORMATION_SCHEMA.TABLE_CONSTRAINTS pk ,

INFORMATION_SCHEMA.KEY_COLUMN_USAGE c
WHERE pk.TABLE_NAME = @TableName
AND CONSTRAINT_TYPE = 'PRIMARY KEY'
AND c.TABLE_NAME = pk.TABLE_NAME
AND c.CONSTRAINT_NAME = pk.CONSTRAINT_NAME

-- Get primary key select for insert
SELECT @PKSelect = COALESCE(@PKSelect+'+','')
+ '''<' + COLUMN_NAME
+ '='+convert(varchar(100),
coalesce(i.' + COLUMN_NAME +',d.' + COLUMN_NAME + '))+'>'''
FROM INFORMATION_SCHEMA.TABLE_CONSTRAINTS pk ,
INFORMATION_SCHEMA.KEY_COLUMN_USAGE c
```

```
                WHERE pk.TABLE_NAME = @TableName
                AND CONSTRAINT_TYPE = 'PRIMARY KEY'
                AND c.TABLE_NAME = pk.TABLE_NAME
                AND c.CONSTRAINT_NAME = pk.CONSTRAINT_NAME

                IF @PKCols IS NULL
                BEGIN
                RAISERROR('no PK on table %s', 16, -1, @TableName)
                RETURN
                END

                SELECT @field = 0,
                @maxfield = MAX(ORDINAL_POSITION)
                FROM INFORMATION_SCHEMA.COLUMNS WHERE
                TABLE_NAME = @TableName
                WHILE @field < @maxfield
                BEGIN
                SELECT @field = MIN(ORDINAL_POSITION)
                FROM INFORMATION_SCHEMA.COLUMNS
                WHERE TABLE_NAME = @TableName
                AND ORDINAL_POSITION > @field
                SELECT @bit = (@field - 1 )% 8 + 1
                SELECT @bit = POWER(2,@bit - 1)
                SELECT @char = ((@field - 1) / 8) + 1
                IF SUBSTRING(COLUMNS_UPDATED(),@char, 1) & @bit > 0 OR
                @Type IN ('I','D')
                BEGIN
                SELECT @fieldname = COLUMN_NAME
                FROM INFORMATION_SCHEMA.COLUMNS
                WHERE TABLE_NAME = @TableName
                AND ORDINAL_POSITION = @field
                SELECT @sql = '
                insert Audit ( Type,
                TableName,
                PK,
                FieldName,
                OldValue,
                NewValue,
                UpdateDate,
                UserName)
                select ''' + @Type + ''','''
                + @TableName + ''',' + @PKSelect
                + ',''' + @fieldname + ''''
                + ',convert(varchar(1000),d.' + @fieldname + ')'
                + ',convert(varchar(1000),i.' + @fieldname + ')'
                + ',''' + @UpdateDate + ''''
                + ',''' + @UserName + ''''
                + ' from #ins i full outer join #del d'
                + @PKCols
                + ' where i.' + @fieldname + ' <> d.' + @fieldname
                + ' or (i.' + @fieldname + ' is null and d.' + @fieldname + ' is not null)'
                + ' or (i.' + @fieldname + ' is not null and d.' + @fieldname + ' is null)'
                EXEC (@sql)
                END
                END
```

Subject: **This SP misses some data.**
Posted by: *pra_1983* (view profile)
Posted on: *Friday, November 4, 2011 at 1:04 PM*

Message: This procedure misses some of the columns of modified data after a
                 column in the table is deleted.

                 my audit table is missing some of the columes that are modified after
                 I delete a column from middle which I am not using. Please suggest
                 what I am doing wrong.

Subject: **This SP misses data when you remove colums using SQL
                    Server 2008!**
Posted by: *elfrabo* (view profile)
Posted on: *Thursday, March 15, 2012 at 2:36 AM*
Message: If you use SQL Server 2008 and you remove some columns in the
                    middle of a table you will miss modifications on the columns to the
                    right of the removed columns!

                    In SQL Server 2008, the ORDINAL_POSITION column of the
                    INFORMATION_SCHEMA.COLUMNS view is not compatible with
                    the bit pattern of columns returned by COLUMNS_UPDATED. To
                    obtain a bit pattern compatible with COLUMNS_UPDATED, reference
                    the ColumnID property of the COLUMNPROPERTY system function
                    when you query the INFORMATION_SCHEMA.COLUMNS view (See
                    http://msdn.microsoft.com/en-us/library/ms186329.aspx)

                    We changed the script a bit to solve the problem: we added a new
                    variable (integer) containing the ID of the column

                    select @fieldname = COLUMN_NAME from
                    INFORMATION_SCHEMA.COLUMNS where TABLE_NAME =
                    @TableName and ORDINAL_POSITION = @field

                    select @fieldid = COLUMNPROPERTY(OBJECT_ID('dbo.' +
                    @TableName), @fieldname, 'ColumnID')

                    select @bit = (@fieldid - 1)% 8 + 1
                    select @bit = power(2,@bit - 1)
                    select @char = ((@fieldid - 1) / 8) + 1
                    if substring(COLUMNS_UPDATED(),@char, 1) & @bit > 0 or @Type
                    in ('I','D')

Subject: **Different schemas + application name**
Posted by: *dejandular* (view profile)
Posted on: *Friday, January 11, 2013 at 6:03 AM*
Message: Thank you Pop Rivet for original code and thank you elfrabo for sql
                    2008 update!

                    I have tweaked the code a little more. It now saves application name
                    and allows using different schemas, not only dbo. Also NOCOUNT
                    ON has been added to the trigger.

                    Here is the code for table and trigger:
                    USE [snipi]
                    GO

                    /****** Object: Table [dbo].[Audit] Script Date: 01/11/2013 13:54:53
                    ******/
                    SET ANSI_NULLS ON
                    GO

                    SET QUOTED_IDENTIFIER ON

```
GO

SET ANSI_PADDING ON
GO

CREATE TABLE [dbo].[Audit](
[Type] [char](1) NULL,
[TableSchema] [nchar](10) NULL,
[TableName] [varchar](128) NULL,
[PK] [varchar](1000) NULL,
[FieldName] [varchar](128) NULL,
[OldValue] [varchar](1000) NULL,
[NewValue] [varchar](1000) NULL,
[UpdateDate] [datetime] NULL,
[UserName] [varchar](128) NULL,
[Application] [varchar](500) NULL
) ON [PRIMARY]

GO

CREATE TRIGGER <triggername> ON <tablename> FOR INSERT,
UPDATE, DELETE
AS
SET NOCOUNT ON

DECLARE @bit INT ,
@field INT ,
@fieldId INT ,
@maxfield INT ,
@char INT ,
@fieldname VARCHAR(128) ,
@SchemaName VARCHAR(128) ,
@TableName VARCHAR(128) ,
@PKCols VARCHAR(1000) ,
@sql VARCHAR(2000),
@UpdateDate VARCHAR(21) ,
@UserName VARCHAR(128) ,
@Type CHAR(1) ,
@PKSelect VARCHAR(1000) ,
@FullTableName VARCHAR(256)


--You will need to change @TableName to match the table to be
audited
SELECT @TableName = <tablename>, @SchemaName =
<schemaname (usually dbo)>

SELECT @FullTableName = @SchemaName + '.' + @TableName

-- date and user
SELECT @UserName = SYSTEM_USER ,
@UpdateDate = CONVERT(VARCHAR(8), GETDATE(), 112)
+ ' ' + CONVERT(VARCHAR(12), GETDATE(), 114)

-- Action
IF EXISTS (SELECT * FROM inserted)
IF EXISTS (SELECT * FROM deleted)
SELECT @Type = 'U'
ELSE
SELECT @Type = 'I'
ELSE
```

```
SELECT @Type = 'D'

-- get list of columns
SELECT * INTO #ins FROM inserted
SELECT * INTO #del FROM deleted

-- Get primary key columns for full outer join
SELECT @PKCols = COALESCE(@PKCols + ' and', ' on')
+ ' i.' + c.COLUMN_NAME + ' = d.' + c.COLUMN_NAME
FROM INFORMATION_SCHEMA.TABLE_CONSTRAINTS pk ,

INFORMATION_SCHEMA.KEY_COLUMN_USAGE c
WHERE pk.TABLE_NAME = @TableName
AND pk.TABLE_SCHEMA = @SchemaName
AND CONSTRAINT_TYPE = 'PRIMARY KEY'
AND c.TABLE_NAME = pk.TABLE_NAME
AND c.CONSTRAINT_NAME = pk.CONSTRAINT_NAME
AND c.CONSTRAINT_SCHEMA = pk.TABLE_SCHEMA

-- Get primary key select for insert
SELECT @PKSelect = COALESCE(@PKSelect+'+',")
+ '"<' + COLUMN_NAME
+ '="'+convert(varchar(100),
coalesce(i.' + COLUMN_NAME +',d.' + COLUMN_NAME + '))+">"'
FROM INFORMATION_SCHEMA.TABLE_CONSTRAINTS pk ,
INFORMATION_SCHEMA.KEY_COLUMN_USAGE c
WHERE pk.TABLE_NAME = @TableName
AND pk.TABLE_SCHEMA = @SchemaName
AND CONSTRAINT_TYPE = 'PRIMARY KEY'
AND c.TABLE_NAME = pk.TABLE_NAME
AND c.CONSTRAINT_NAME = pk.CONSTRAINT_NAME
AND c.CONSTRAINT_SCHEMA = pk.TABLE_SCHEMA

IF @PKCols IS NULL
BEGIN
RAISERROR('no PK on table %s', 16, -1, @FullTableName)
RETURN
END

SELECT @field = 0,
@maxfield = MAX(ORDINAL_POSITION)
FROM INFORMATION_SCHEMA.COLUMNS WHERE
TABLE_NAME = @TableName AND TABLE_SCHEMA =
@SchemaName
WHILE @field < @maxfield
BEGIN
SELECT @fieldname = COLUMN_NAME from
INFORMATION_SCHEMA.COLUMNS
WHERE TABLE_NAME = @TableName
AND TABLE_SCHEMA = @SchemaName
AND ORDINAL_POSITION = @field

SELECT @fieldid =
COLUMNPROPERTY(OBJECT_ID(@FullTableName), @fieldname,
'ColumnID')


SELECT @field = MIN(ORDINAL_POSITION)
FROM INFORMATION_SCHEMA.COLUMNS
WHERE TABLE_NAME = @TableName
AND TABLE_SCHEMA = @SchemaName
```

```
AND ORDINAL_POSITION > @field
select @bit = (@fieldid - 1 )% 8 + 1
select @bit = power(2,@bit - 1)
select @char = ((@fieldid - 1) / 8) + 1
if substring(COLUMNS_UPDATED(),@char, 1) & @bit > 0 or @Type
in ('I','D')


BEGIN


SELECT @sql = '
insert Audit ( Type,
TableSchema,
TableName,
PK,
FieldName,
OldValue,
NewValue,
UpdateDate,
UserName,
Application)
select ''' + @Type + ''','
+ @SchemaName + ''', '
+ @TableName + ''',' + @PKSelect
+ ',''' + @fieldname + '''
+ ',convert(varchar(1000),d.' + @fieldname + ')'
+ ',convert(varchar(1000),i.' + @fieldname + ')'
+ ',''' + @UpdateDate + ''''
+ ',''' + @UserName + ''''
+ ',''' + REPLACE(APP_NAME(), '''', '''''') + ''''
+ ' from #ins i full outer join #del d'
+ @PKCols
+ ' where i.' + @fieldname + ' <> d.' + @fieldname
+ ' or (i.' + @fieldname + ' is null and d.'
+ @fieldname
+ ' is not null)'
+ ' or (i.' + @fieldname + ' is not null and d.'
+ @fieldname
+ ' is null)'
EXEC (@sql)
END
END


GO
```

Subject:	**Not working from front end**
Posted by:	*wanderingaround* (view profile)
Posted on:	*Tuesday, May 7, 2013 at 9:58 AM*
Message:	Thanks for the post... works great if I modify a record in Management studio but calls up the 'No PK on table' message when it's triggered from the front end... The table definitely has a PK (as mentioned works in SSMS on that table...)

Any ideas?? thanks, Mike

Subject:	**Additional Column**
Posted by:	*goldkitten* (view profile)
Posted on:	*Thursday, July 11, 2013 at 7:57 AM*
Message:	Thanks for the post....you came to my resue. I tried adding a column from the base table to be added to the Audit table but got this error:

Conversion failed when converting the varchar value to data type int.

Any assistant will be greatly appreciated.
Cheers,JW

Subject:	**No data on Audit Tables**
Posted by:	*AMCN* (view profile)
Posted on:	*Tuesday, December 30, 2014 at 9:27 AM*
Message:	I am using this trigger on my data warehouse and is working fine until I refresh my large dimension Tables and the Fact Table.

Anybody has experience when making large amount of Updates and Insert data is not inserted in the Audit table?

Subject:	**The fieldid issue, skipping certain fields and using the logged in username**
Posted by:	*wkelvinf* (view profile)
Posted on:	*Thursday, February 5, 2015 at 6:12 AM*
Message:	I used the updated script provided by the poster dejandular which was based on the code sinippet provided by the poster elfrabo. Thanks! However, dejandular script did not solve the problem of certain columns being excluded. I found out that it had to do with the ordering he used. I changed it to this before it worked

…
SELECT @field = MIN(ORDINAL_POSITION)
FROM INFORMATION_SCHEMA.COLUMNS
WHERE TABLE_NAME = @TableName
AND ORDINAL_POSITION > @field
AND DATA_TYPE NOT IN ('IMAGE', 'DATETIME')

SELECT @fieldname = COLUMN_NAME from
INFORMATION_SCHEMA.COLUMNS
WHERE TABLE_NAME = @TableName
AND ORDINAL_POSITION = @field
AND DATA_TYPE NOT IN ('IMAGE', 'DATETIME')

SELECT @fieldid =
COLUMNPROPERTY(OBJECT_ID('dbo.'+@TableName),
@fieldname, 'ColumnID')
select @bit = (@fieldid - 1)% 8 + 1
select @bit = power(2,@bit - 1)
select @char = ((@fieldid - 1) / 8) + 1
if substring(COLUMNS_UPDATED(),@char, 1) & @bit > 0 or @Type
in ('I','D')
…

Also, for those who would like to skip certain fields like update/modified date, image etc, for each of the statements that reference INFORMATION_SCHEMA.COLUMNS, add this: AND DATA_TYPE NOT IN ('IMAGE', 'DATATIME') as you can see in my script above.

Finally, as to the username, my application always sends the logged in user (ModifiedBy). So this is what I did to get the username:

-- date and user
--SELECT @UserName = SYSTEM_USER,

IF EXISTS (SELECT * FROM inserted)
SELECT @UserName = ModifiedBy FROM inserted

```
ELSE
SELECT @UserName = ModifiedBy FROM deleted
-- in case null or empty, set to system user
IF (@TableName IS NULL OR @TableName = ")
SELECT @UserName = SYSTEM_USER
```

Subject:	**Excluding computed columns**
Posted by:	*SQLWayne* (view profile)
Posted on:	*Friday, March 27, 2015 at 2:07 PM*
Message:	Building on wkelvinf's update of 5 Feb 2015, to filter computed columns from the audit, I will be modifying my implementation to exclude them. You can see what user tables have such, and the field names, with this code:

```
select so.name as TableName, sc.*
from syscolumns sc
join sysobjects so
on sc.id = so.id
where so.type = 'U'
and sc.iscomputed = 1
order by so.Name, sc.Name
```

I'll try and post my modified code when I'm done. And I LOVE the fact that code from nine years ago is still being discussed and updated for new versions of SQL Server!

Subject:	**The ordinal position fix in SQL 2014**
Posted by:	*SQLWayne* (view profile)
Posted on:	*Friday, March 27, 2015 at 3:01 PM*
Message:	is apparently not needed.

I created an empty 60 column table, deleted four fields from the middle via SSMS and when I queried the ordinal position again, they were correctly sequenced. I'm guessing it was a problem in 2008 and fixed in either R2 or 2012 or 2014. When I first tested this code, I included the fix provided by elfrabo and the trigger process went off to lala land, I put the original code back in and it worked fine.

```
select *
from INFORMATION_SCHEMA.COLUMNS
where TABLE_NAME = 'tempStudents'
```

Subject:	**Cascade error**
Posted by:	*Tissid* (view profile)
Posted on:	*Wednesday, January 27, 2016 at 1:24 PM*
Message:	Dear,

I love your solution, it's exactly what I was searching for but... It´s not working with cascade deletion , i got the error :
The Subquery returned more than one value

Any ideas ?

Thanks a lot