**[sql server - procedure to drop system-versioned temporal tables - Stack Overflow](https://stackoverflow.com/questions/55522278/procedure-to-drop-system-versioned-temporal-tables)**

I'm looking for a procedure to drop system-versioned temporal tables, ideally without using dynamic SQL. I've looked through the Microsoft documentation and figured out how to get the autogenerated history table name, but I only know a little about cursors and even less about dynamic SQL.

[You can't just drop a temporal table](https://blogs.msdn.microsoft.com/sql_pfe_blog/2017/01/19/sql-2016-temporal-tables-how-do-you-drop-a-temporal-table/). You must first disable versioning, which will cause the history table to become an ordinary table. Then you can drop both the temporal table and its corresponding history table.

ALTER TABLE [dbo].[TemporalTest] SET ( SYSTEM\_VERSIONING = OFF )

GO

DROP TABLE [dbo].[TemporalTest]

GO

DROP TABLE [dbo].[TemporalTestHistory]

GO

I am using temporal tables with autogenerated history tables, so I don't know their names. However, the [Microsoft docs](https://dataedo.com/kb/query/sql-server/list-temporal-tables-in-database) provide information about how to list the history tables, so I have a way of getting those names.

select schema\_name(t.schema\_id) as temporal\_table\_schema,

t.name as temporal\_table\_name,

schema\_name(h.schema\_id) as history\_table\_schema,

h.name as history\_table\_name,

case when t.history\_retention\_period = -1

then 'INFINITE'

else cast(t.history\_retention\_period as varchar) + ' ' +

t.history\_retention\_period\_unit\_desc + 'S'

end as retention\_period

from sys.tables t

left outer join sys.tables h

on t.history\_table\_id = h.object\_id

where t.temporal\_type = 2

order by temporal\_table\_schema, temporal\_table\_name

I was hoping I could use a subquery with the DROP statement, e.g. DROP TABLE (SELECT '#t'). This throws a syntax error.

I'm looking for a stored procedure which takes two parameters: the name of the table to be dropped and whether the drop should take place if there's any data in the table (eg, must ROWCOUNT=0).

* [**sql-server**](https://stackoverflow.com/questions/tagged/sql-server)
* [**t-sql**](https://stackoverflow.com/questions/tagged/t-sql)
* [**temporal-tables**](https://stackoverflow.com/questions/tagged/temporal-tables)

[Share](https://stackoverflow.com/q/55522278)

[Improve this question](https://stackoverflow.com/posts/55522278/edit)

Follow

[edited Feb 9 at 21:29](https://stackoverflow.com/posts/55522278/revisions)

[[](https://stackoverflow.com/users/2756409/tylerh)](https://stackoverflow.com/users/2756409/tylerh)

[TylerH](https://stackoverflow.com/users/2756409/tylerh)

**21.2k**7474 gold badges7979 silver badges110110 bronze badges

asked Apr 4, 2019 at 17:56

[[](https://stackoverflow.com/users/9777873/frank)](https://stackoverflow.com/users/9777873/frank)

[Frank](https://stackoverflow.com/users/9777873/frank)

**328**11 gold badge22 silver badges1010 bronze badges

[Add a comment](https://stackoverflow.com/questions/55522278/procedure-to-drop-system-versioned-temporal-tables)

4 Answers

Sorted by:

                                              Highest score (default)                                                                   Trending (recent votes count more)                                                                   Date modified (newest first)                                                                   Date created (oldest first)

**12**

You can't issue DDL against data (like the output of a subquery), or pass an entity name as a variable to a statement; you need dynamic SQL.

CREATE PROCEDURE dbo.DropTemporalTable

@schema sysname = N'dbo',

@table sysname

AS

BEGIN

SET NOCOUNT ON;

DECLARE @sql nvarchar(max) = N'';

SELECT @sql += N'ALTER TABLE ' + src + N' SET (SYSTEM\_VERSIONING = OFF);

DROP TABLE ' + src + N';

DROP TABLE ' + hist + N';'

FROM

(

SELECT src = QUOTENAME(SCHEMA\_NAME(t.schema\_id))

+ N'.' + QUOTENAME(t.name),

hist = QUOTENAME(SCHEMA\_NAME(h.schema\_id))

+ N'.' + QUOTENAME(h.name)

FROM sys.tables AS t

INNER JOIN sys.tables AS h

ON t.history\_table\_id = h.[object\_id]

WHERE t.temporal\_type = 2

AND t.[schema\_id] = SCHEMA\_ID(@schema)

AND t.name = @table

) AS x;

EXEC sys.sp\_executesql @sql;

END

GO

[Share](https://stackoverflow.com/a/55523018)

[Improve this answer](https://stackoverflow.com/posts/55523018/edit)

Follow

[edited Apr 4, 2019 at 20:32](https://stackoverflow.com/posts/55523018/revisions)

answered Apr 4, 2019 at 18:47

[[](https://stackoverflow.com/users/61305/aaron-bertrand)](https://stackoverflow.com/users/61305/aaron-bertrand)

[Aaron Bertrand](https://stackoverflow.com/users/61305/aaron-bertrand)

**279k**3838 gold badges477477 silver badges505505 bronze badges

* This is awesome, thanks! I turned it into a procedure and parameterized it on @t. SELECT src = QUOTENAME(SCHEMA\_NAME(t.schema\_id)) + N'.' + QUOTENAME(t.name), hist = QUOTENAME(SCHEMA\_NAME(h.schema\_id)) + N'.' + QUOTENAME(h.name) FROM sys.tables AS t INNER JOIN sys.tables AS h ON t.history\_table\_id = h.[object\_id] WHERE t.temporal\_type = 2 AND t.name=@t

– [Frank](https://stackoverflow.com/users/9777873/frank)

[CommentedApr 4, 2019 at 19:58](https://stackoverflow.com/questions/55522278/procedure-to-drop-system-versioned-temporal-tables" \l "comment97752187_55523018)

[Add a comment](https://stackoverflow.com/questions/55522278/procedure-to-drop-system-versioned-temporal-tables)

**1**

I am using temporal tables with autogenerated history tables, so I don't know their names. However, the Microsoft docs provide information about how to list the history tables, so I have a way of getting those names.

I don't know if you know this but you can specify the name of the history table in several ways.

If you create your own history table:

ALTER TABLE [TemporalTable] SET (SYSTEM\_VERSIONING = ON (HISTORY\_TABLE = [dbo].[HistoryTable]))

Or if you want your history table created for you:

CREATE TABLE [dbo].[TemporalTable](

[Id] UNIQUEIDENTIFIER NOT NULL CONSTRAINT [DF\_TemporalTable\_Id] DEFAULT(NEWID()),

[SysStartTime] DATETIME2(2) GENERATED ALWAYS AS ROW START NOT NULL CONSTRAINT [DF\_TemporalTable\_SysStartTime] DEFAULT(SYSUTCDATETIME()),

[SysEndTime] DATETIME2(2) GENERATED ALWAYS AS ROW END NOT NULL CONSTRAINT [DF\_TemporalTable\_SysEndTime] DEFAULT(CONVERT(DATETIME2(2),'9999-12-31 23:59:59.99')),

CONSTRAINT [PK\_Product] PRIMARY KEY CLUSTERED

(

[Id] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON, FILLFACTOR = 100) ON [PRIMARY],

PERIOD FOR SYSTEM\_TIME ([SysStartTime], [SysEndTime])

) ON [PRIMARY]

WITH

(

SYSTEM\_VERSIONING = ON ( HISTORY\_TABLE = [dbo].[HistoryTable] )

)

Other than that I don't know of a way to create a stored procedure which takes a table name as a parameter and drops it without using dynamic SQL. Something like below will be necessary. Depending on if you want the stored procedure to look up the name of the history table linked to it you can add the query you mentioned in your post instead of a parameter.

CREATE PROCEDURE DeleteTemporalTable (@tableName NVARCHAR(MAX), @historyTableName) AS

BEGIN

EXEC ('

IF EXISTS (

SELECT

1

FROM

SYSOBJECTS

WHERE

id = OBJECT\_ID(N''[dbo].[' + @tableName + ']'') AND

OBJECTPROPERTY(id, N''IsTable'') = 1 AND

OBJECTPROPERTY(id, N''TableTemporalType'') = 2

)

BEGIN

ALTER TABLE [dbo].[' + @tableName + '] SET (SYSTEM\_VERSIONING = OFF)

END

DROP TABLE IF EXISTS [dbo].[' + @tableName + ']

DROP TABLE IF EXISTS [dbo].[' + @historyTableName + ']')

END

If you're just looking for a way to delete all tables, and you want something that works with temporal tables, this is a script I use:

DECLARE @temporalTableName varchar(max) = (SELECT TOP 1 [name] FROM sys.tables WHERE [temporal\_type\_desc] = 'SYSTEM\_VERSIONED\_TEMPORAL\_TABLE')

WHILE @temporalTableName IS NOT NULL

BEGIN

EXEC('ALTER TABLE [dbo].[' + @temporalTableName + '] SET (SYSTEM\_VERSIONING = OFF)')

SET @temporalTableName = (SELECT TOP 1 [name] FROM sys.tables WHERE [temporal\_type\_desc] = 'SYSTEM\_VERSIONED\_TEMPORAL\_TABLE')

END

DECLARE @tableName varchar(max) = (SELECT TOP 1 [name] FROM sys.tables)

WHILE @tableName IS NOT NULL

BEGIN

EXEC('DROP TABLE [dbo].[' + @tableName + ']')

SET @tableName = (SELECT TOP 1 [name] FROM sys.tables)

END

[Share](https://stackoverflow.com/a/55522956)

[Improve this answer](https://stackoverflow.com/posts/55522956/edit)

Follow

answered Apr 4, 2019 at 18:42

[[](https://stackoverflow.com/users/4290902/vqf5mg96cstt)](https://stackoverflow.com/users/4290902/vqf5mg96cstt)

[Vqf5mG96cSTT](https://stackoverflow.com/users/4290902/vqf5mg96cstt)

**2,841**33 gold badges2727 silver badges4444 bronze badges

* Very helpful insights, thanks! In this case I don't care about the name of the history table so prefer that it be autogenerated. As often happens with Microsoft products, I find myself surprised that they neglected some important basics.

– [Frank](https://stackoverflow.com/users/9777873/frank)

[CommentedApr 4, 2019 at 20:07](https://stackoverflow.com/questions/55522278/procedure-to-drop-system-versioned-temporal-tables" \l "comment97752423_55522956)

* @Frank The primary focus of T-SQL is to store and retrieve data, not to drop and create tables without knowing their names.

– [Aaron Bertrand](https://stackoverflow.com/users/61305/aaron-bertrand)

[CommentedApr 4, 2019 at 20:11](https://stackoverflow.com/questions/55522278/procedure-to-drop-system-versioned-temporal-tables" \l "comment97752552_55522956)

* I'm aware of the purpose of T-SQL. I'm also aware that Microsoft implemented temporal tables without extending DROP TABLE to play nicely, so I'm simply trying to script the ability to cleanly drop temporal tables with autogenerated history tables.

– [Frank](https://stackoverflow.com/users/9777873/frank)

[CommentedApr 4, 2019 at 20:19](https://stackoverflow.com/questions/55522278/procedure-to-drop-system-versioned-temporal-tables" \l "comment97752826_55522956)

* 1

My point was that the primary use case of T-SQL is not *dropping tables*, regardless of the introduction of temporal. Same happens with schema-bound objects, foreign keys, etc.

– [Aaron Bertrand](https://stackoverflow.com/users/61305/aaron-bertrand)

[CommentedApr 4, 2019 at 20:25](https://stackoverflow.com/questions/55522278/procedure-to-drop-system-versioned-temporal-tables" \l "comment97752984_55522956)

[Add a comment](https://stackoverflow.com/questions/55522278/procedure-to-drop-system-versioned-temporal-tables)

**1**

I've adapted Aaron's answer. Here's final solution:

CREATE OR ALTER PROCEDURE [dbo].[DropTemporalTable]

@t as nvarchar(max),

@s as nvarchar(max) = 'dbo'

-- See https://stackoverflow.com/questions/55522278/procedure-to-drop-system-versioned-temporal-tables/55524067#55524067

AS

BEGIN

-- TODO: check if src table is empty first

DECLARE @sql nvarchar(max) = N'';

SELECT @sql += N'ALTER TABLE ' + src + N' SET (SYSTEM\_VERSIONING = OFF);

DROP TABLE ' + src + N';

DROP TABLE ' + hist + N';

'

FROM

(

SELECT src = QUOTENAME(SCHEMA\_NAME(t.schema\_id))

+ N'.' + QUOTENAME(t.name),

hist = QUOTENAME(SCHEMA\_NAME(h.schema\_id))

+ N'.' + QUOTENAME(h.name)

FROM sys.tables AS t

INNER JOIN sys.tables AS h

ON t.history\_table\_id = h.[object\_id]

WHERE t.temporal\_type = 2 AND t.name=@t AND t.schema\_id=SCHEMA\_ID(@s)

) AS x;

PRINT @sql;

--EXEC sys.sp\_executesql @sql;

END

[Share](https://stackoverflow.com/a/55524067)

[Improve this answer](https://stackoverflow.com/posts/55524067/edit)

Follow

[edited Apr 4, 2019 at 20:15](https://stackoverflow.com/posts/55524067/revisions)

answered Apr 4, 2019 at 20:04

[[](https://stackoverflow.com/users/9777873/frank)](https://stackoverflow.com/users/9777873/frank)

[Frank](https://stackoverflow.com/users/9777873/frank)

**328**11 gold badge22 silver badges1010 bronze badges

* To be fair, your question says you are deriving a list of table names because you don't know them, and want to execute the drop for all the tables in that set. Making a stored procedure that takes a single table name as a parameter means (a) you have to know the table name first, and (b) now you have to write code that calls the stored procedure for every table name in your set.

– [Aaron Bertrand](https://stackoverflow.com/users/61305/aaron-bertrand)

[CommentedApr 4, 2019 at 20:09](https://stackoverflow.com/questions/55522278/procedure-to-drop-system-versioned-temporal-tables" \l "comment97752495_55524067)

* This also assumes that you won't ever have tables in any schema other than the caller's default schema.

– [Aaron Bertrand](https://stackoverflow.com/users/61305/aaron-bertrand)

[CommentedApr 4, 2019 at 20:10](https://stackoverflow.com/questions/55522278/procedure-to-drop-system-versioned-temporal-tables" \l "comment97752520_55524067)

* I've fixed the schema issue at least to a degree, making it an optional parameter that defaults to 'dbo'. Sorry if my prompt was unclear: I meant that I didn't know the name of the **history** table because it was autogenerated. Really appreciate your help.

– [Frank](https://stackoverflow.com/users/9777873/frank)

[CommentedApr 4, 2019 at 20:16](https://stackoverflow.com/questions/55522278/procedure-to-drop-system-versioned-temporal-tables" \l "comment97752739_55524067)

[Add a comment](https://stackoverflow.com/questions/55522278/procedure-to-drop-system-versioned-temporal-tables)

**1**

/\*\*\*\*\*\* Object: Table [Intake].[MER\_SF\_Asset\_Temporal] Script Date: 04/05/2021 12:33:57 PM \*\*\*\*\*\*/

ALTER TABLE [Intake].[MER\_SF\_Asset\_Temporal] SET ( SYSTEM\_VERSIONING = OFF )

GO

/\*\*\*\*\*\* Object: Table [Intake].[MER\_SF\_Asset\_Temporal] Script Date: 04/05/2021 12:33:57 PM \*\*\*\*\*\*/

IF EXISTS (SELECT \* FROM sys.objects WHERE object\_id = OBJECT\_ID(N'[Intake].[MER\_SF\_Asset\_Temporal]') AND type in (N'U'))

DROP TABLE [Intake].[MER\_SF\_Asset\_Temporal]

GO

/\*\*\*\*\*\* Object: Table [Intake].[MERSFAssetTemporalHistory] Script Date: 04/05/2021 12:33:58 PM \*\*\*\*\*\*/

IF EXISTS (SELECT \* FROM sys.objects WHERE object\_id = OBJECT\_ID(N'[Intake].[MERSFAssetTemporalHistory]') AND type in (N'U'))

DROP TABLE [Intake].[MERSFAssetTemporalHistory]

GO