**Adaptative Files Autogrow**

**By** [**Samuel HERVOUET**](http://www.sqlservercentral.com/Authors/Scripts/Samuel_HERVOUET/200049/)**, 2016/01/07**

**FROM:** [**http://www.sqlservercentral.com/scripts/Files/127332/**](http://www.sqlservercentral.com/scripts/Files/127332/)

Minimum Autogrow is adjusted for all databases depending file size, 3 thresholds (lower, medium and upper limit in MB) and file type (datafiles, tempfiles).

Thresholds and file size are in MB

If autogrow is already set with higher value, nothing is done  
If autogrow is disable, nothing is done  
If autogrow is in percent, autogrow is set in MB

This script create the SP in master DB and create a job scheduled every 6 hours

How to use it  :

EXEC master.dbo.DBM\_Set\_File\_Growth

     @datafilesize\_lowerlimit

    ,@datafilesize\_upperlimit

    ,@datafile\_growth\_target\_lower

    ,@datafile\_growth\_target\_medium

    ,@datafile\_growth\_target\_upper

    ,@logfilesize\_lowerlimit

    ,@logfilesize\_upperlimit

    ,@logfile\_growth\_target\_lower

    ,@logfile\_growth\_target\_medium

    ,@logfile\_growth\_target\_upper

Examples :

EXEC master.dbo.DBM\_Set\_File\_Growth 400,2000,100,200,500,null,null,null,null,null  
     For Datafiles less than 400 MB, set 100 MB autogrow  
     For Datafiles between 400 MB and 2000 MB set 200 MB autogrow  
     For Datafiles greater than 2000 MB set 500 MB autogrow  
                          
EXEC master.dbo.DBM\_Set\_File\_Growth null,null,null,null,null,1000,5000,200,500,1000  
     For Logfiles less than 1000 MB, set 200 MB autogrow  
     For Logfiles between 1000 MB and 5000 MB set 500 MB autogrow  
     For Logfiles greater than 5000 MB set 1000 MB autogrow  
                          
EXEC master.dbo.DBM\_Set\_File\_Growth 400,2000,100,200,500,1000,5000,200,500,1000  
     Set autogrow for Datafiles AND Logfiles as explained in the 2 previous examples

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* PURPOSE : This procedure set minimal File growth values for every datafiles and logfiles (SQL 2005 and above)

\* DBM\_Set\_File\_Growth \_job is executed every 6h daily

\* Create DBM\_Set\_File\_Growth stored procedure and DBM\_Set\_File\_Growth\_job job.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

PRINT 'DBM\_Set\_File\_Grouwth SP creation'

USE DBM\_Toolbox

GO

IF EXISTS (SELECT \* FROM sys.objects WHERE object\_id = OBJECT\_ID(N'dbo.DBM\_Set\_File\_Growth') AND type in (N'P', N'PC'))

DROP PROCEDURE dbo.DBM\_Set\_File\_Growth

GO

CREATE PROCEDURE DBM\_Set\_File\_Growth

(

@datafilesize\_lowerlimit INT,

@datafilesize\_upperlimit INT,

@datafile\_growth\_target\_lower INT,

@datafile\_growth\_target\_medium INT,

@datafile\_growth\_target\_upper INT,

@logfilesize\_lowerlimit INT,

@logfilesize\_upperlimit INT,

@logfile\_growth\_target\_lower INT,

@logfile\_growth\_target\_medium INT,

@logfile\_growth\_target\_upper INT

)

AS

/\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Name : DBM\_Toolbox.dbo.DBM\_Set\_File\_Growth

Purpose : Set minimal autogrow values for datafiles and logfiles depending on the file size

If autogrow is already set with higher value, nothing is done

If autogrow is disable, nothing is done

If autogrow is in percent, autogrow is set in MB

Author : Samuel HERVOUET

Prerequisite : None

How to use it : EXEC DBM\_Toolbox.dbo.DBM\_Set\_File\_Growth @datafilesize\_lowerlimit

,@datafilesize\_upperlimit

,@datafile\_growth\_target\_lower

,@datafile\_growth\_target\_medium

,@datafile\_growth\_target\_upper

,@logfilesize\_lowerlimit

,@logfilesize\_upperlimit

,@logfile\_growth\_target\_lower

,@logfile\_growth\_target\_medium

,@logfile\_growth\_target\_upper

Examples : EXEC DBM\_Toolbox.dbo.DBM\_Set\_File\_Growth 400,2000,100,200,500,null,null,null,null,null

For Datafiles less than 400 MB, set 100 MB autogrow

For Datafiles between 400 MB and 2000 MB set 200 MB autogrow

For Datafiles greater than 2000 MB set 500 MB autogrow

EXEC DBM\_Toolbox.dbo.DBM\_Set\_File\_Growth null,null,null,null,null,1000,5000,200,500,1000

For Logfiles less than 1000 MB, set 200 MB autogrow

For Logfiles between 1000 MB and 5000 MB set 500 MB autogrow

For Logfiles greater than 5000 MB set 1000 MB autogrow

EXEC DBM\_Toolbox.dbo.DBM\_Set\_File\_Growth 400,2000,100,200,500,1000,5000,200,500,1000

Set autogrow for Datafiles AND Logfiles as explained in the 2 previous examples

Version : 1.0

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*/

/\* -- For test only

DECLARE @datafilesize\_lowerlimit INT

DECLARE @datafilesize\_upperlimit INT

DECLARE @datafile\_growth\_target\_lower INT

DECLARE @datafile\_growth\_target\_medium INT

DECLARE @datafile\_growth\_target\_upper INT

-----------

DECLARE @logfilesize\_lowerlimit INT

DECLARE @logfilesize\_upperlimit INT

DECLARE @logfile\_growth\_target\_lower INT

DECLARE @logfile\_growth\_target\_medium INT

DECLARE @logfile\_growth\_target\_upper int

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

SET @datafilesize\_lowerlimit = 400

SET @datafilesize\_upperlimit = 2000

SET @datafile\_growth\_target\_lower = 100

SET @datafile\_growth\_target\_medium = 200

SET @datafile\_growth\_target\_upper = 500

----------

SET @logfilesize\_lowerlimit = 500

SET @logfilesize\_upperlimit = 1000

SET @logfile\_growth\_target\_lower = 200

SET @logfile\_growth\_target\_medium = 500

SET @logfile\_growth\_target\_upper = 1000

\*/

SET NOCOUNT ON

PRINT '\*\*\*\*\*\*\*\*\*\*\* Parameters \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*'

PRINT 'Datafiles size lower limit : ' + convert(varchar(20), @datafilesize\_lowerlimit) + ' MB'

PRINT 'Datafiles size upper limit : ' + convert(varchar(20), @datafilesize\_upperlimit) + ' MB'

PRINT 'Datafiles growth lower target : ' + convert(varchar(20), @datafile\_growth\_target\_lower) + ' MB'

PRINT 'Datafiles growth medium target : ' + convert(varchar(20), @datafile\_growth\_target\_medium) + ' MB'

PRINT 'Datafiles growth upper target : ' + convert(varchar(20), @datafile\_growth\_target\_upper) + ' MB'

PRINT ' '

PRINT 'Logfiles size lower limit : ' + convert(varchar(20), @logfilesize\_lowerlimit) + ' MB'

PRINT 'Logfiles size upper limit : ' + convert(varchar(20), @logfilesize\_upperlimit) + ' MB'

PRINT 'Logfiles growth lower target : ' + convert(varchar(20), @logfile\_growth\_target\_lower) + ' MB'

PRINT 'Logfiles growth medium target : ' + convert(varchar(20), @logfile\_growth\_target\_medium) + ' MB'

PRINT 'Logfiles growth upper target : ' + convert(varchar(20), @logfile\_growth\_target\_upper) + ' MB'

PRINT '\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*'

-- Retrieve files informations

DECLARE @sql VARCHAR(8000)

SET @sql=' USE [?]

SELECT ''?'' [Dbname]

,[name] [Filename]

,type\_desc [Type]

,physical\_name [FilePath]

,CONVERT(INT,[size]/128.0) [TotalSize\_MB]

,CONVERT(INT,FILEPROPERTY(name, ''SpaceUsed''))/128.0 AS [Space\_Used\_MB]

,CASE is\_percent\_growth

WHEN 1 THEN CONVERT(VARCHAR(5),growth)

ELSE CONVERT(VARCHAR(20),(growth/128))

END [Autogrow\_Value]

,CASE is\_percent\_growth

WHEN 1 THEN ''Pct''

ELSE ''MB''

END [Unit]

,CASE max\_size

WHEN -1 THEN CASE growth

WHEN 0 THEN CONVERT(VARCHAR(30),''Restricted'')

ELSE CONVERT(VARCHAR(30),''Unlimited'') END

ELSE CONVERT(VARCHAR(25),max\_size/128)

END [Max\_Size]

FROM [?].sys.database\_files'

-- Create temp table to store Files informations

IF EXISTS(SELECT 1 FROM tempdb..sysobjects WHERE name='##AutogrowthDetails')

DROP TABLE ##AutogrowthDetails

CREATE TABLE ##AutogrowthDetails (

Dbname VARCHAR(128)

,Filename VARCHAR(128)

,Type VARCHAR(10)

,Filepath VARCHAR(2000)

,TotalSize\_MB INT

,Space\_Used\_MB INT

,Autogrow\_Value VARCHAR(15)

,Unit VARCHAR(15)

,Max\_Size VARCHAR(30)

)

INSERT INTO ##AutogrowthDetails EXEC sp\_msforeachdb @sql

DECLARE @dbname varchar(8000)

DECLARE @file varchar(8000)

DECLARE @filename varchar(8000)

DECLARE @type varchar(20)

DECLARE @totalsizemb int

DECLARE @autogrowthvalue varchar(20)

DECLARE @filegrowth\_target varchar(20)

DECLARE @sql2 varchar(8000)

-- Set file autogrow value depending on thresholds

DECLARE SetAutogrowthSize cursor for

SELECT dbname, filename, type, TotalSize\_MB, Autogrow\_Value

FROM ##AutogrowthDetails

WHERE dbname not in ('master','msdb','tempdb','model')

AND dbname IN (select name from sys.databases where state\_desc = 'ONLINE')

AND Autogrow\_Value <> 0

AND (

(

(TotalSize\_MB < @datafilesize\_lowerlimit AND Autogrow\_Value < @datafile\_growth\_target\_lower and type = 'ROWS')

OR

((TotalSize\_MB BETWEEN @datafilesize\_lowerlimit AND @datafilesize\_upperlimit) AND Autogrow\_Value < @datafile\_growth\_target\_medium AND type = 'ROWS')

OR

(TotalSize\_MB > @datafilesize\_upperlimit AND Autogrow\_Value < @datafile\_growth\_target\_upper AND type = 'ROWS')

)

OR

(

(TotalSize\_MB < @logfilesize\_lowerlimit AND Autogrow\_Value < @logfile\_growth\_target\_lower and type = 'LOG')

OR

((TotalSize\_MB BETWEEN @logfilesize\_lowerlimit AND @logfilesize\_upperlimit) AND Autogrow\_Value < @logfile\_growth\_target\_medium AND type = 'LOG')

OR

(TotalSize\_MB > @logfilesize\_upperlimit AND Autogrow\_Value < @logfile\_growth\_target\_upper AND type = 'LOG')

)

)

ORDER BY dbname, filename

OPEN SetAutogrowthSize

FETCH NEXT FROM SetAutogrowthSize INTO @dbname, @file, @type, @totalsizemb, @autogrowthvalue

WHILE @@FETCH\_STATUS = 0

BEGIN

IF @type = 'ROWS' AND (@totalsizemb < @datafilesize\_lowerlimit AND @autogrowthvalue < @datafile\_growth\_target\_lower)

SET @filegrowth\_target = @datafile\_growth\_target\_lower

IF @type = 'ROWS' AND ((@totalsizemb between @datafilesize\_lowerlimit AND @datafilesize\_upperlimit) AND @autogrowthvalue < @datafile\_growth\_target\_medium)

SET @filegrowth\_target = @datafile\_growth\_target\_medium

IF @type = 'ROWS' AND ((@totalsizemb = @datafilesize\_upperlimit OR @totalsizemb > @datafilesize\_upperlimit) AND @autogrowthvalue < @datafile\_growth\_target\_upper)

SET @filegrowth\_target = @datafile\_growth\_target\_upper

IF @type = 'LOG' AND (@totalsizemb < @logfilesize\_lowerlimit AND @autogrowthvalue < @logfile\_growth\_target\_lower)

SET @filegrowth\_target = @logfile\_growth\_target\_lower

IF @type = 'LOG' AND ((@totalsizemb between @logfilesize\_lowerlimit AND @logfilesize\_upperlimit) AND @autogrowthvalue < @logfile\_growth\_target\_medium)

SET @filegrowth\_target = @logfile\_growth\_target\_medium

IF @type = 'LOG' AND ((@totalsizemb = @logfilesize\_upperlimit OR @totalsizemb > @logfilesize\_upperlimit) AND @autogrowthvalue < @logfile\_growth\_target\_upper)

SET @filegrowth\_target = @logfile\_growth\_target\_upper

SET @sql2 = 'ALTER DATABASE ['+ @dbname + '] MODIFY FILE (NAME = '''+@file+''', FILEGROWTH =' +@filegrowth\_target + ')'

Print '\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*'

Print '| Database Name: ' + @dbname + ' | Logical File Name: ' + @file + ' | File Type: ' + @type + ' | Current Size: ' + convert(varchar(20), @totalsizemb) + ' MB | Current Growth increment: ' + @autogrowthvalue

Print '| Process will update filegrowth size from ' + convert(varchar(20), @autogrowthvalue) + ' to ' + convert(varchar(20), @filegrowth\_target) + ' MB'

Print '| Executing following ALTER command: '

Print '| '+ @sql2

Print '\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*'

Print ' '

exec (@sql2)

FETCH NEXT FROM SetAutogrowthSize INTO @dbname, @file, @type, @totalsizemb, @autogrowthvalue

END

CLOSE SetAutogrowthSize

DEALLOCATE SetAutogrowthSize

GO

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

-- Job creation (scheduled every 6h)

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

PRINT 'DBM\_Set\_File\_Growth\_job job creation'

USE [msdb]

GO

IF EXISTS (SELECT name FROM [msdb].[dbo].[sysjobs] WHERE name = 'DBM\_Set\_File\_Growth\_job')

BEGIN

PRINT 'Replace DBM\_Set\_File\_Growth\_job'

EXEC msdb.dbo.sp\_delete\_job @job\_name = N'DBM\_Set\_File\_Growth\_job' ;

END

GO

DECLARE @jobId BINARY(16)

EXEC msdb.dbo.sp\_add\_job @job\_name=N'DBM\_Set\_File\_Growth\_job',

@enabled=1,

@notify\_level\_eventlog=0,

@notify\_level\_email=2,

@notify\_level\_netsend=2,

@notify\_level\_page=2,

@delete\_level=0,

@category\_name=N'Database Maintenance',

@owner\_login\_name=N'sa', @job\_id = @jobId OUTPUT

select @jobId

GO

EXEC msdb.dbo.sp\_add\_jobserver @job\_name=N'DBM\_Set\_File\_Growth\_job', @server\_name = N'(LOCAL)'

GO

USE [msdb]

GO

EXEC msdb.dbo.sp\_add\_jobstep @job\_name=N'DBM\_Set\_File\_Growth\_job', @step\_name=N'DBM\_Set\_File\_Growth\_job - Step 1',

@step\_id=1,

@cmdexec\_success\_code=0,

@on\_success\_action=1,

@on\_fail\_action=2,

@retry\_attempts=0,

@retry\_interval=0,

@os\_run\_priority=0, @subsystem=N'TSQL',

@command=N'EXEC DBM\_Toolbox.dbo.DBM\_Set\_File\_Growth 400,2000,100,200,500,1000,5000,100,500,1000

GO',

@database\_name=N'DBM\_Toolbox',

@flags=0

GO

USE [msdb]

GO

EXEC msdb.dbo.sp\_update\_job @job\_name=N'DBM\_Set\_File\_Growth\_job',

@enabled=1,

@start\_step\_id=1,

@notify\_level\_eventlog=0,

@notify\_level\_email=2,

@notify\_level\_netsend=2,

@notify\_level\_page=2,

@delete\_level=0,

@description=N'DISTRIBUTED-DB',

@category\_name=N'Database Maintenance',

@owner\_login\_name=N'sa',

@notify\_email\_operator\_name=N'',

@notify\_netsend\_operator\_name=N'',

@notify\_page\_operator\_name=N''

GO

USE [msdb]

GO

DECLARE @schedule\_id int

EXEC msdb.dbo.sp\_add\_jobschedule @job\_name=N'DBM\_Set\_File\_Growth\_job', @name=N'Every6h',

@enabled=1,

@freq\_type=4,

@freq\_interval=1,

@freq\_subday\_type=8,

@freq\_subday\_interval=6,

@freq\_relative\_interval=0,

@freq\_recurrence\_factor=1,

@active\_start\_date=20131125,

@active\_end\_date=99991231,

@active\_start\_time=0,

@active\_end\_time=235959, @schedule\_id = @schedule\_id OUTPUT

select @schedule\_id

GO