SQL-DBA-Concepts-PocketGuide

Sl	Description		SQLTalk-Comment/Solution
106	Calculates average stalls per read, per write, and per total input/output		Tech target Microsoft https://technet.microsoft.com/en-us/library/jj643251.aspx -- Calculates average stalls per read, per write, and per total input/output -- for each database file. SELECT DB_NAME(database_id) AS [Database Name] , file_id , io_stall_read_ms , num_of_reads , CAST(io_stall_read_ms / (1.0 + num_of_reads) AS NUMERIC(10, 1)) AS [avg_read_stall_ms] , io_stall_write_ms , num_of_writes , CAST(io_stall_write_ms / (1.0 + num_of_writes) AS NUMERIC(10, 1)) AS [avg_write_stall_ms] , io_stall_read_ms + io_stall_write_ms AS [io_stalls] , num_of_reads + num_of_writes AS [total_io] , CAST((io_stall_read_ms + io_stall_write_ms) / (1.0 + num_of_reads + num_of_writes) AS NUMERIC(10,1)) AS [avg_io_stall_ms] FROM sys.dm_io_virtual_file_stats(NULL, NULL)ORDER BY avg_I -- Look at pending I/O requests by file SELECT DB_NAME(mf.database_id) AS [Database] , mf.physical_name ,r.io_pending , r.io_pending_ms_ticks , r.io_type , fs.num_of_reads , fs.num_of_writesFROM sys.dm_io_pending_io_requests AS r INNER JOIN sys.dm_io_virtual_file_stats(NULL, NULL) AS fs ON r.io_handle = fs.file_handle INNER JOIN sys.master_files AS mf ON fs.database_id = mf.database_id AND fs.file_id = mf.file_idORDER BY r.io_pending , r.io_pending_ms_ticks DESC ;

107	Log contents		select * from fn_dblog(NULL,NULL)
108	Query to get the table size		;with cte as (SELECT sch.name as SchemaName, t.name as TableName, SUM (s.used_page_count) as used_pages_count, SUM (CASE WHEN (i.index_id < 2) THEN (in_row_data_page_count + lob_used_page_count + row_overflow_used_page_count) ELSE lob_used_page_count + row_overflow_used_page_count END) as pages FROM sys.dm_db_partition_stats AS s JOIN sys.tables AS t ON s.object_id = t.object_id JOIN sys.indexes AS i ON i.[object_id] = t.[object_id] AND s.index_id = i.index_id JOIN sys.schemas AS sch ON sch.schema_id = t.schema_id GROUP BY sch.name,t.name) select cte.SchemaName, cte.TableName, cast((cte.pages * 8.)/1024 as decimal(10,3)) as TableSizeInMB,

			cast(((CASE WHEN cte.used_pages_count > cte.pages THEN cte.used_pages_count - cte.pages ELSE 0 END) * 8./1024) as decimal(10,3)) as IndexSizeInMB from cte order by 2 desc
109	Last stats update		SELECT DISTINCT OBJECT_NAME(s.[object_id]) AS TableName, c.name AS ColumnName, s.name AS StatName, s.auto_created, s.user_created, s.no_recompute, s.[object_id], s.stats_id, sc.stats_column_id, sc.column_id, STATS_DATE(s.[object_id], s.stats_id) AS LastUpdated FROM sys.stats s JOIN sys.stats_columns sc ON sc.[object_id] = s.[object_id] AND sc.stats_id = s.stats_id

			JOIN sys.columns c ON c.[object_id] = sc.[object_id] AND c.column_id = sc.column_id JOIN sys.partitions par ON par.[object_id] = s.[object_id] JOIN sys.objects obj ON par.[object_id] = obj.[object_id] WHERE OBJECTPROPERTY(s.OBJECT_ID,'IsUserTable') = 1 AND (s.auto_created = 1 OR s.user_created = 1);
110	Stats last update		select a.id as 'ObjectID', isnull(a.name,'Heap') as 'IndexName', b.name as 'TableName', stats_date (id,indid) as stats_last_updated_time from sys.sysindexes as a inner join sys.objects as b on a.id = b.object_id where b.type = 'U'
111	, the query plans , total workertime, sps,reads/writes etc---		------------------------------ -------- ---------------------- ---------------- ------------- ----- ------------------------------ ------- SELECT top 200 OBJECT_NAME(qt.objectid), qs.total_worker_time CPU, qs.last_worker_time Last_CPU, qs.last_execution_time, qs.execution_count, qs.total_logical_reads, qs.total_physical_reads, qp.query_plan

			FROM sys.dm_exec_query_stats qs CROSS APPLY sys.dm_exec_sql_text(qs.sql_ handle) qt CROSS APPLY sys.dm_exec_query_plan(qs. plan_handle) qp where query_plan is not NULL ORDER BY qs.total_worker_time DESC -- CPU -- ORDER BY last_elapsed_time_in_S DESC -- CPU -- ORDER BY qs.total_logical_reads DESC -- logical reads -- ORDER BY qs.total_logical_writes DESC -- logical writes ------------------------------ -------- ---------------------- ---------------- -------------- -----
112	-------- To get only the query plans , CPU , exec count , last CPU ------- -------		------------------------------ SELECT qt.text, qs.total_worker_time CPU, qs.last_worker_time, qp.query_plan, qs.last_execution_time, qs.execution_count FROM sys.dm_exec_query_stats qs CROSS APPLY sys.dm_exec_sql_text(qs.sql_ handle) qt CROSS APPLY sys.dm_exec_query_plan(qs. plan_handle) qp where qt.text like '%USP_CREATEDACCOUNTS_COUNT_ BY_RETAILER%' -- and qs.execution_count > 10 ORDER BY qs.total_worker_time DESC -- CPU -- ORDER BY last_elapsed_time_in_S DESC -- CPU -- ORDER BY qs.total_logical_reads DESC -- logical reads -- ORDER BY qs.total_logical_writes DESC -- logical writes

113	to get top SP's , pl, worker time,reads etc		SELECT SUBSTRING(qt.TEXT, (qs.statement_start_offset/2)+ 1, ((CASE qs.statement_end_offset WHEN -1 THEN DATALENGTH(qt.TEXT) ELSE qs.statement_end_offset END - qs.statement_start_offset)/2)+ 1), qs.execution_count, qs.total_logical_reads, qs.last_logical_reads, qs.total_logical_writes, qs.last_logical_writes, qs.total_physical_reads, qs.last_physical_reads, qt.dbid, qs.total_worker_time CPU, qs.last_worker_time, qs.total_elapsed_time/1000000 total_elapsed_time_in_S, qs.last_elapsed_time/1000000 last_elapsed_time_in_S, qs.last_execution_time, qp.query_plan FROM sys.dm_exec_query_stats qs CROSS APPLY sys.dm_exec_sql_text(qs.sql_ handle) qt CROSS APPLY sys.dm_exec_query_plan(qs. plan_handle) qp where qs.execution_count > 10 ORDER BY last_elapsed_time_in_S DESC -- CPU -- ORDER BY qs.total_worker_time DESC - - CPU -- ORDER BY qs.total_logical_reads DESC -- logical reads -- ORDER BY qs.total_logical_writes DESC -- logical writes
114	Last stats		select STATS_DATE(OBJECT_ID , index_id) --as stats-updated from sys.indexes

115	Important DB admin queries		Blocks - bottlenecks ---------- SELECT dm_qp.query_plan, dm_es.program_name, dm_ws.wait_duration_ms, dm_ws.session_ID, dm_ws.blocking_session_id, dm_ws.wait_type, dm_es.status, --dm_t.TEXT, dm_es.cpu_time, dm_es.memory_usage, dm_es.logical_reads, dm_es.total_elapsed_time, DB_NAME(dm_r.database_id) DatabaseName, -- Optional columns dm_r.wait_resource, dm_es.login_name, dm_r.command FROM sys.dm_os_waiting_tasks dm_ws INNER JOIN sys.dm_exec_requests dm_r ON dm_ws.session_id = dm_r.session_id INNER JOIN sys.dm_exec_sessions dm_es ON dm_es.session_id = dm_r.session_id CROSS APPLY sys.dm_exec_sql_text (dm_r.sql_handle) dm_t CROSS APPLY sys.dm_exec_query_plan (dm_r.plan_handle) dm_qp WHERE dm_es.is_user_process = 1 GO --------------------------Disk Bottlenecks---------------------------- ---------- select * -- database_id,file_id,io_stall,io_pending _ms_ticks,scheduler_address from sys.dm_io_virtual_file_stats(NULL,NULL) IOVFS, sys.dm_io_pending_io_requests ioprior where iovfs.file_handle=ioprior.io_handle -- --------------------------------

```
select * from sys.dm_exec_requests -
-  where command like '%backup%'
select * from sys.dm_exec_sessions
where session_id = 1403
select * from sys.dm_exec_connections
where session_id = 1403
DBCC INPUTBUFFER(89)
 sp_who2

 SP_lock


---------------------- To check
sessions from management studio -------
--------

SELECT
dm_ws.session_ID,
dm_es.program_name,
dm_r.last_wait_type
FROM sys.dm_os_waiting_tasks dm_ws
INNER JOIN sys.dm_exec_requests dm_r ON
dm_ws.session_id = dm_r.session_id
INNER JOIN sys.dm_exec_sessions dm_es
ON dm_es.session_id = dm_r.session_id
CROSS APPLY sys.dm_exec_sql_text
(dm_r.sql_handle) dm_t
CROSS APPLY sys.dm_exec_query_plan
(dm_r.plan_handle) dm_qp
WHERE dm_es.is_user_process = 1  and
program_name like '%Microsoft%'
GO

---------------------- Summary of the
blocks ---------------
SELECT
db.name DBName,
tl.request_session_id,
wt.blocking_session_id,
OBJECT_NAME(p.OBJECT_ID)
BlockedObjectName,
tl.resource_type,
h1.TEXT AS RequestingText,
h2.TEXT AS BlockingTest,
tl.request_mode
FROM sys.dm_tran_locks AS tl
INNER JOIN sys.databases db ON
db.database_id =
tl.resource_database_id
INNER JOIN sys.dm_os_waiting_tasks AS
```

```
wt ON tl.lock_owner_address =
wt.resource_address
INNER JOIN sys.partitions AS p ON
p.hobt_id =
tl.resource_associated_entity_id
INNER JOIN sys.dm_exec_connections ec1
ON ec1.session_id =
tl.request_session_id
INNER JOIN sys.dm_exec_connections ec2
ON ec2.session_id =
wt.blocking_session_id
CROSS APPLY
sys.dm_exec_sql_text(ec1.most_recent_sq
l_handle) AS h1
CROSS APPLY
sys.dm_exec_sql_text(ec2.most_recent_sq
l_handle) AS h2
 GO
----------------------------------------
-----------

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

Buffer Usage
------------------


-- find out how big buffer pool is and
determine percentage used by each
database
DECLARE @total_buffer INT;
SELECT @total_buffer =
cntr_value   FROM
sys.dm_os_performance_counters
WHERE RTRIM([object_name]) LIKE
'%Buffer Manager'   AND counter_name =
'Total Pages';
;WITH src
AS(   SELECT        database_id,
db_buffer_pages = COUNT_BIG(*)
FROM
sys.dm_os_buffer_descriptors       --
WHERE database_id BETWEEN 5 AND
32766
GROUP BY database_id)SELECT   [db_name]
= CASE [database_id] WHEN
32767        THEN 'Resource
DB'        ELSE DB_NAME([database_id])
```

```
END,   db_buffer_pages,   db_buffer_MB
= db_buffer_pages /
128,   db_buffer_percent =
CONVERT(DECIMAL(6,3),        db_buffer_
pages * 100.0 / @total_buffer)
FROM src
ORDER BY db_buffer_MB DESC;
--then drill down into memory used by
objects in database of your choice
USE DBNAME;
WITH src AS(   SELECT       [Object] =
o.name,       [Type] =
o.type_desc,       [Index] =
COALESCE(i.name,
''),       [Index_Type] =
i.type_desc,       p.[object_id],
p.index_id,       au.allocation_unit_id

FROM       sys.partitions AS p   INNER
JOIN       sys.allocation_units AS
au       ON p.hobt_id =
au.container_id   INNER
JOIN       sys.objects AS o       ON
p.[object_id] = o.[object_id]   INNER
JOIN       sys.indexes AS i       ON
o.[object_id] = i.[object_id]       AND
p.index_id =
i.index_id   WHERE       au.[type] IN
(1,2,3)       AND o.is_ms_shipped = 0)
SELECT   src.[Object],   src.[Type],
src.[Index],   src.Index_Type,   buffer
_pages =
COUNT_BIG(b.page_id),   buffer_mb =
COUNT_BIG(b.page_id) / 128
FROM   src
INNER
JOIN   sys.dm_os_buffer_descriptors AS
b
 ON src.allocation_unit_id =
b.allocation_unit_id
WHERE   b.database_id = DB_ID()
GROUP
BY   src.[Object],   src.[Type],   src.
[Index],   src.Index_Type
ORDER BY   buffer_pages DESC;

Connections
-----------------
select * from sys.dm_exec_connections

Sessions
```

```
-----------------
select * from sys.dm_exec_sessions


Index Usage Stats
----------------------
SELECT
--      row_number() over(order by
user_seeks,user_lookups,user_scans),
--       [Database] = d.name,
      [Schema]= s.name,
      [Table]= o.name,
      [Index]= x.name,
      [Scans] = user_scans,
      [Seeks] = user_seeks,
      [Lookups] = user_lookups,
      [Last Scan] = last_user_scan,
      [Last Seek] = last_user_seek,
      [Last lookUp]= last_user_lookup,
      [System Scans] = system_scans
FROM  sys.dm_db_index_usage_stats u
INNER JOIN sys.sysdatabases d on
u.database_id = d.dbid
INNER JOIN sys.sysindexes x on
u.object_id = x.id  and u.index_id =
x.indid
INNER JOIN sys.objects o on u.object_id
= o.object_id
INNER JOIN sys.schemas s on s.schema_id
= o.schema_id
where x.name is not null and
u.database_id=14
order by 1 desc

Memory clerks , rings
-----------------------------

SELECT * FROM sys.dm_os_memory_clerks
ORDER BY (single_pages_kb +
multi_pages_kb + awe_allocated_kb) desc

select * from sys.dm_os_ring_buffers;


Top running
--------------------------
SELECT SUBSTRING(qt.TEXT,
(qs.statement_start_offset/2)+1,
((CASE qs.statement_end_offset
WHEN -1 THEN DATALENGTH(qt.TEXT)
ELSE qs.statement_end_offset
```

```
END - qs.statement_start_offset)/2)+1),
qs.execution_count,
qs.total_logical_reads,
qs.last_logical_reads,
qs.total_logical_writes,
qs.last_logical_writes,
qs.total_physical_reads,
qs.last_physical_reads,
qt.dbid,
qs.total_worker_time CPU,
qs.last_worker_time,
qs.total_elapsed_time/1000000
total_elapsed_time_in_S,
qs.last_elapsed_time/1000000
last_elapsed_time_in_S,
qs.last_execution_time,
qp.query_plan
FROM sys.dm_exec_query_stats qs
CROSS APPLY
sys.dm_exec_sql_text(qs.sql_handle) qt
CROSS APPLY
sys.dm_exec_query_plan(qs.plan_handle)
qp
where qs.execution_count > 10
ORDER BY
last_elapsed_time_in_S       DESC --
CPU
-- ORDER BY qs.total_worker_time DESC -
- CPU
-- ORDER BY qs.total_logical_reads DESC
-- logical reads
-- ORDER BY qs.total_logical_writes
DESC -- logical writes

--------------------------------------
To get only the query plans , CPU ,
exec count , last CPU  --------------

SELECT
qt.text,
qs.total_worker_time  CPU,
qs.last_worker_time,
qp.query_plan,
qs.last_execution_time,
qs.execution_count
FROM sys.dm_exec_query_stats qs
CROSS APPLY
sys.dm_exec_sql_text(qs.sql_handle) qt
CROSS APPLY
sys.dm_exec_query_plan(qs.plan_handle)
qp
```

			where qt.text like '%USP_CREATEDACCOUNTS_COUNT_BY_RETAILER%' -- and qs.execution_count > 10 ORDER BY qs.total_worker_time DESC -- CPU -- ORDER BY last_elapsed_time_in_S DESC -- CPU -- ORDER BY qs.total_logical_reads DESC -- logical reads -- ORDER BY qs.total_logical_writes DESC -- logical writes ---------- ---------- ---------- ---------- - , the query plans , total workertime, sps,reads/writes etc--- SELECT top 100 OBJECT_NAME(qt.objectid), qs.total_worker_time CPU, qs.last_worker_time Last_CPU, qs.last_execution_time, qs.execution_count, qs.total_logical_reads, qs.total_physical_reads, qs.total_logical_writes, qp.query_plan FROM sys.dm_exec_query_stats qs CROSS APPLY sys.dm_exec_sql_text(qs.sql_handle) qt CROSS APPLY sys.dm_exec_query_plan(qs.plan_handle) qp where query_plan is not NULL ORDER BY qs.total_worker_time DESC -- CPU -- ORDER BY last_elapsed_time_in_S DESC -- CPU ORDER BY qs.total_logical_reads DESC -- logical reads -- ORDER BY qs.total_logical_writes DESC -- logical writes ---------- ---------- ----------

```
Wait Stats
--------------

select *
from sys.dm_os_wait_stats
-- where wait_type like 'PAGEIOLATCH%'
or wait_type like 'LCK_M%' or wait_type
like 'ASYNC_IO_COMPLETION%' or
wait_type like 'IO_COMPLETION%' or
wait_type like
'ACCESS_METHODS_DATASET_PARENT%'  or
wait_type like
'ACCESS_METHODS_SCAN_RANGE_GENERATOR%'
or wait_type like
'SOS_SCHEDULER_YIELD%' or wait_type
like 'LATCH%' or wait_type like
'CXPACKET%' or wait_type like
'EXECSYNC%' or wait_type like
'ASYNC_NETWORK_IO%'
ORDER BY wait_type asc;

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

WITH [Waits] AS
    (SELECT
        [wait_type],
        [wait_time_ms] / 1000.0 AS
[WaitS],
        ([wait_time_ms] -
[signal_wait_time_ms]) / 1000.0 AS
[ResourceS],
        [signal_wait_time_ms] / 1000.0
AS [SignalS],
        [waiting_tasks_count] AS
[WaitCount],
       100.0 * [wait_time_ms] / SUM
([wait_time_ms]) OVER() AS
[Percentage],
        ROW_NUMBER() OVER(ORDER BY
[wait_time_ms] DESC) AS [RowNum]
    FROM sys.dm_os_wait_stats
    WHERE [wait_type] NOT IN (
        N'BROKER_EVENTHANDLER',
N'BROKER_RECEIVE_WAITFOR',
        N'BROKER_TASK_STOP',
N'BROKER_TO_FLUSH',
        N'BROKER_TRANSMITTER',
N'CHECKPOINT_QUEUE',
        N'CHKPT', N'CLR_AUTO_EVENT',
```

```
        N'CLR_MANUAL_EVENT',
N'CLR_SEMAPHORE',

        N'DBMIRROR_DBM_EVENT',
N'DBMIRROR_EVENTS_QUEUE',
        N'DBMIRROR_WORKER_QUEUE',
N'DBMIRRORING_CMD',

        N'DIRTY_PAGE_POLL',
N'DISPATCHER_QUEUE_SEMAPHORE',
        N'EXECSYNC', N'FSAGENT',
        N'FT_IFTS_SCHEDULER_IDLE_WAIT',
N'FT_IFTSHC_MUTEX',

        N'HADR_CLUSAPI_CALL',
N'HADR_FILESTREAM_IOMGR_IOCOMPLETION',
        N'HADR_LOGCAPTURE_WAIT',
N'HADR_NOTIFICATION_DEQUEUE',
        N'HADR_TIMER_TASK',
N'HADR_WORK_QUEUE',

        N'KSOURCE_WAKEUP',
N'LAZYWRITER_SLEEP',
        N'LOGMGR_QUEUE',
N'MEMORY_ALLOCATION_EXT',
        N'ONDEMAND_TASK_QUEUE',
        N'PREEMPTIVE_XE_GETTARGETSTATE'
,
        N'PWAIT_ALL_COMPONENTS_INITIALI
ZED',
        N'PWAIT_DIRECTLOGCONSUMER_GETNE
XT',
        N'QDS_PERSIST_TASK_MAIN_LOOP_SL
EEP', N'QDS_ASYNC_QUEUE',
        N'QDS_CLEANUP_STALE_QUERIES_TAS
K_MAIN_LOOP_SLEEP',
        N'QDS_SHUTDOWN_QUEUE',
N'REDO_THREAD_PENDING_WORK',
        N'REQUEST_FOR_DEADLOCK_SEARCH',
N'RESOURCE_QUEUE',
        N'SERVER_IDLE_CHECK',
N'SLEEP_BPOOL_FLUSH',
        N'SLEEP_DBSTARTUP',
N'SLEEP_DCOMSTARTUP',
        N'SLEEP_MASTERDBREADY',
N'SLEEP_MASTERMDREADY',
        N'SLEEP_MASTERUPGRADED',
N'SLEEP_MSDBSTARTUP',
        N'SLEEP_SYSTEMTASK',
N'SLEEP_TASK',
        N'SLEEP_TEMPDBSTARTUP',
```

```
N'SNI_HTTP_ACCEPT',
        N'SP_SERVER_DIAGNOSTICS_SLEEP',
N'SQLTRACE_BUFFER_FLUSH',
        N'SQLTRACE_INCREMENTAL_FLUSH_SL
EEP',
        N'SQLTRACE_WAIT_ENTRIES',
N'WAIT_FOR_RESULTS',
        N'WAITFOR',
N'WAITFOR_TASKSHUTDOWN',
        N'WAIT_XTP_RECOVERY',
        N'WAIT_XTP_HOST_WAIT',
N'WAIT_XTP_OFFLINE_CKPT_NEW_LOG',
        N'WAIT_XTP_CKPT_CLOSE',
N'XE_DISPATCHER_JOIN',
        N'XE_DISPATCHER_WAIT',
N'XE_TIMER_EVENT')
    AND [waiting_tasks_count] > 0
    )
SELECT
    MAX ([W1].[wait_type]) AS
[WaitType],
    CAST (MAX ([W1].[WaitS]) AS DECIMAL
(16,2)) AS [Wait_S],
    CAST (MAX ([W1].[ResourceS]) AS
DECIMAL (16,2)) AS [Resource_S],
    CAST (MAX ([W1].[SignalS]) AS
DECIMAL (16,2)) AS [Signal_S],
    MAX ([W1].[WaitCount]) AS
[WaitCount],
    CAST (MAX ([W1].[Percentage]) AS
DECIMAL (5,2)) AS [Percentage],
    CAST ((MAX ([W1].[WaitS]) / MAX
([W1].[WaitCount])) AS DECIMAL (16,4))
AS [AvgWait_S],
    CAST ((MAX ([W1].[ResourceS]) / MAX
([W1].[WaitCount])) AS DECIMAL (16,4))
AS [AvgRes_S],
    CAST ((MAX ([W1].[SignalS]) / MAX
([W1].[WaitCount])) AS DECIMAL (16,4))
AS [AvgSig_S]
FROM [Waits] AS [W1]
INNER JOIN [Waits] AS [W2]
    ON [W2].[RowNum] <= [W1].[RowNum]
GROUP BY [W1].[RowNum]
HAVING SUM ([W2].[Percentage]) - MAX(
[W1].[Percentage] ) < 95; -- percentage
threshold
GO
```

			Qry for Exec Count ----------

```
SELECT DB_NAME(st.dbid) DBName

     ,OBJECT_SCHEMA_NAME(st.objectid,db
id) SchemaName

     ,OBJECT_NAME(st.objectid,dbid)
StoredProcedure

     ,max(cp.usecounts) Execution_count

     ,sum(qs.total_worker_time)
total_cpu_time

     ,sum(qs.total_worker_time) /
(max(cp.usecounts) * 1.0)  avg_cpu_time

FROM sys.dm_exec_cached_plans cp join
sys.dm_exec_query_stats qs on
cp.plan_handle = qs.plan_handle

     CROSS APPLY
sys.dm_exec_sql_text(cp.plan_handle) st

where DB_NAME(st.dbid) is not null and
cp.objtype = 'proc'

and DB_NAME(st.dbid)=db_name(db_id())
AND OBJECT_NAME(st.objectid,dbid) in
('USP_CONFIGURABLE_ALERTTYPES_GET_BYACC
OUNTNO')

group by
DB_NAME(st.dbid),OBJECT_SCHEMA_NAME(obj
ectid,st.dbid),
OBJECT_NAME(objectid,st.dbid)


PLE
-------

SELECT *

FROM sys.dm_os_performance_counters

WHERE counter_name = 'Page life
expectancy'

AND OBJECT_NAME = 'SQLServer:Buffer
Manager'
```

			SP Mem Stats --------------- SELECT TEXT ,query_plan ,requested_memory_kb ,granted_memory_kb ,used_memory_kb FROM sys.dm_exec_query_memory_grants emg CROSS APPLY sys.dm_exec_sql_text(sql_handle) CROSS APPLY sys.dm_exec_query_plan(emg.plan_handle) ORDER BY emg.requested_memory_kb DESC