--Short snapshot of wait stats (uses WAITFOR DELAY '00:10:00')

IF EXISTS (SELECT \* FROM [tempdb].[sys].[objects]

WHERE [name] = N'##Waits1')

DROP TABLE [##Waits1];

IF EXISTS (SELECT \* FROM [tempdb].[sys].[objects]

WHERE [name] = N'##Waits2')

DROP TABLE [##Waits2];

GO

SELECT [wait\_type], [waiting\_tasks\_count], [wait\_time\_ms],

[max\_wait\_time\_ms], [signal\_wait\_time\_ms]

INTO ##Waits1

FROM sys.dm\_os\_wait\_stats;

GO

WAITFOR DELAY '00:10:00';

GO

SELECT [wait\_type], [waiting\_tasks\_count], [wait\_time\_ms],

[max\_wait\_time\_ms], [signal\_wait\_time\_ms]

INTO ##Waits2

FROM sys.dm\_os\_wait\_stats;

GO

WITH [DiffWaits] AS

(SELECT

-- Waits that weren't in the first snapshot

[ts2].[wait\_type],

[ts2].[wait\_time\_ms],

[ts2].[signal\_wait\_time\_ms],

[ts2].[waiting\_tasks\_count]

FROM [##Waits2] AS [ts2]

LEFT OUTER JOIN [##Waits1] AS [ts1]

ON [ts2].[wait\_type] = [ts1].[wait\_type]

WHERE [ts1].[wait\_type] IS NULL

AND [ts2].[wait\_time\_ms] > 0

UNION

SELECT

-- Diff of waits in both snapshots

[ts2].[wait\_type],

[ts2].[wait\_time\_ms] - [ts1].[wait\_time\_ms] AS [wait\_time\_ms],

[ts2].[signal\_wait\_time\_ms] - [ts1].[signal\_wait\_time\_ms] AS [signal\_wait\_time\_ms],

[ts2].[waiting\_tasks\_count] - [ts1].[waiting\_tasks\_count] AS [waiting\_tasks\_count]

FROM [##Waits2] AS [ts2]

LEFT OUTER JOIN [##Waits1] AS [ts1]

ON [ts2].[wait\_type] = [ts1].[wait\_type]

WHERE [ts1].[wait\_type] IS NOT NULL

AND [ts2].[waiting\_tasks\_count] - [ts1].[waiting\_tasks\_count] > 0

AND [ts2].[wait\_time\_ms] - [ts1].[wait\_time\_ms] > 0),

[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 [DiffWaits]

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'ONDEMAND\_TASK\_QUEUE',

N'PWAIT\_ALL\_COMPONENTS\_INITIALIZED',

N'QDS\_PERSIST\_TASK\_MAIN\_LOOP\_SLEEP',

N'QDS\_SHUTDOWN\_QUEUE',

N'QDS\_CLEANUP\_STALE\_QUERIES\_TASK\_MAIN\_LOOP\_SLEEP',

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\_SLEEP',

N'SQLTRACE\_WAIT\_ENTRIES', N'WAIT\_FOR\_RESULTS',

N'WAITFOR', N'WAITFOR\_TASKSHUTDOWN',

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')

)

SELECT

[W1].[wait\_type] AS [WaitType],

CAST ([W1].[WaitS] AS DECIMAL (16, 2)) AS [Wait\_S],

CAST ([W1].[ResourceS] AS DECIMAL (16, 2)) AS [Resource\_S],

CAST ([W1].[SignalS] AS DECIMAL (16, 2)) AS [Signal\_S],

[W1].[WaitCount] AS [WaitCount],

CAST ([W1].[Percentage] AS DECIMAL (5, 2)) AS [Percentage],

CAST (([W1].[WaitS] / [W1].[WaitCount]) AS DECIMAL (16, 4)) AS [AvgWait\_S],

CAST (([W1].[ResourceS] / [W1].[WaitCount]) AS DECIMAL (16, 4)) AS [AvgRes\_S],

CAST (([W1].[SignalS] / [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], [W1].[wait\_type], [W1].[WaitS],

[W1].[ResourceS], [W1].[SignalS], [W1].[WaitCount], [W1].[Percentage]

HAVING SUM ([W2].[Percentage]) - [W1].[Percentage] < 95; -- percentage threshold

GO

-- Cleanup

IF EXISTS (SELECT \* FROM [tempdb].[sys].[objects]

WHERE [name] = N'##Waits1')

DROP TABLE [##Waits1];

IF EXISTS (SELECT \* FROM [tempdb].[sys].[objects]

WHERE [name] = N'##Waits2')

DROP TABLE [##Waits2];

GO