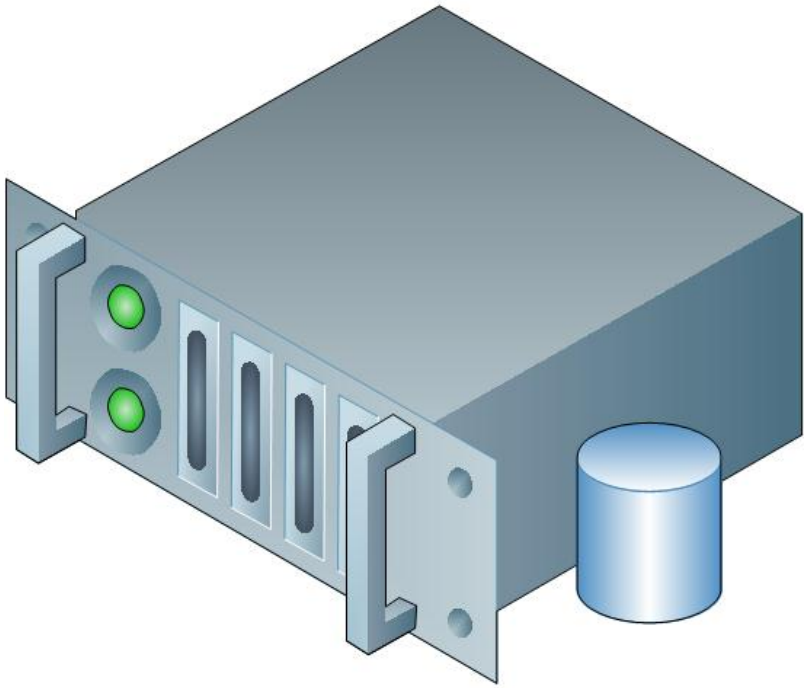
SOLID QUALITY MENTORS

Getting a Handle on TempDB

Andrew J. Kelly SQL MVP
akelly@solidq.com



Overview

- What is TempDB
- TempDB Objects
- Space & I/O Issues
- Contention within TempDB

TempDB

- What is TempDB
- How and when is it created
  - Recreated each time SQL Server starts
  - Modeled after the model db
  - File size set back to default during rebuild

TempDB Objects

- Internal
  - To store intermediate runs for sort.
  - To store intermediate results for hash joins and hash aggregates.
  - To store XML variables or other large object (LOB) data type variables. The LOB data type includes all of the large object types: **text**, **image**, **ntext**, **varchar(max)**, **varbinary(max)**, and all others.
  - By queries that need a spool to store intermediate results.
  - By keyset cursors to store the keys.
  - By static cursors to store a query result.
  - By Service Broker to store messages in transit.
  - By triggers to store data for internal processing + Inserted & Deleted.
  - Internal objects are also used by any feature that uses these operations.

TempDB Objects

- Version stores (*Row Versioning*)
  - Two Types: ONLINE and Common
  - Used for:
    - Snapshot isolation
    - Triggers
    - MARS
    - Online index build
- User objects
    - Query
    - Triggers
    - Temporary tables, table variables, and table-valued functions

TempDB Objects

- Other processes or objects using tempdb
  - DBCC CHECKDB
  - LOB parameters
  - Cursors
  - Service Broker and event notification
  - XML and LOB variables
  - Query notifications
  - Database mail
  - Index creation (Sort in Tempdb)
  - User-defined functions

What to monitor

- Space issues
- I/O issues
- DML contention
- DDL contention

Space issues

- Sysmon counters
  - Database: Log File(s) Size(KB
  - Database: Log File(s) Used (KB)
  - Transactions: Free Space in tempdb (KB)
  - Transactions: Version Store Size (KB)
  - Transactions: Version Generation Rate (KB/s)
  - Transactions: Version Cleanup Rate (KB/s)

DMV's

  - sys.dm_db_file_space_usage
  - sys.dm_db_session_file_usage
  - sys.dm_db_session_space_usage
  - sys.dm_db_task_space_usage

I/O issues

- Use normal IO troubleshooting methods and monitoring to find TempDB I/O bottlenecks
- Don’t assume TempDB activity is cached
- Sys.dm_db_virtual_file_stats is great for determining TempDB physical I/O usage
- Avoid Raid 5 for heavy TempDB usage, preferably RAID 10

I/O issues

- Spread TempDB across as many spindles as feasible or potentially place on its own disk array
- Separate the Log file from the data files
- See Troubleshooting Performance Problems in SQL Server 2008 for details:
  - http://msdn.microsoft.com/en-us/library/dd672789(SQL.100).aspx

TempDB Caching

- There have been several improvements to TempDB in 2005 & 2008 to aide in temporary object reuse and minimize contention

- http://technet.microsoft.com/en-us/library/cc966545.aspx

NOTE: Don't explicitly DROP temp tables

Contention

- Contention can be caused on internal allocation maps (PFS – Page 1, GAM – Page 2, SGAM – Page 3)
- Creating more than one data file can help spread the load across multiple's of these pages
  - It's not an I/O issue so each file does not have to be on a separate array
- May require use of trace flag –T1118

Contention

- Sysmon Counters
  - Access Methods::Worktables Created/sec
  - Access Methods::Workfiles Created/sec
  - Access Methods: Worktables From Cache Ratio
  - General Statistics: Active Temp Tables
  - General Statistics: Temp Tables Creation Rate
  - General Statistics: Temp Tables For Destruction
- DMV
  - sys.dm_os_waiting_tasks

Contention

- Sysmon counters
  - Temp Tables Creation Rate
  - Temp Tables For Destruction
- Working with TempDB
  - http://technet.microsoft.com/en-us/library/cc966545.aspx
- Concurrency Enhancements
  - http://support.microsoft.com/kb/328551

Demonstration

TempDB DMV's

Questions