**Sample configuration files**

|  |  |  |
| --- | --- | --- |
| **Sample file** | **Description** | **Parameters that differ from the other configuration file** |
| Sqliosim.hwcache.cfg.ini | Minimize reads  Files are made small to keep them fully in memory   No sequential reads | For the AuditUser section and for the ReadAheadUser section:  *CacheHitRatio=10000* *UserCount=0* |
| Sqliosim.nothrottle.cfg.ini | Remove I/O throttling  Minimize the time to wait to increase I/O volume | *TargetIODuration=1000000* *AuditDelay=10* *RADelay=10* |
| Sqliosim.seqwrites.cfg.ini | Minimize reads  Files are made small to keep them fully in memory  Files are made non-shrinkable  No sequential reads  No random access  Bulk update in big chunks without delays | *Shrinkable=FALSE*  For the AuditUser section, for the ReadAheadUser section, and for the RandomUser section:  *CacheHitRatio=10000* *ForceReadAhead=FALSE* *BuffersBUMin=600* *BuffersBUMax=1000* *BUDelay=1* *UserCount=0* |
| Sqliosim.sparse.cfg.ini | Use only 32 MB of memory  Make target I/O duration large enough to enable many outstanding I/O requests  Disable scatter/gather APIs to issue separate I/O requests for every 8 KB page  Create a 1 GB non-shrinkable file  Create a 1 GB non-shrinkable secondary sparse stream in the file | *MaxMemoryMB=32* *TestCycles=3* *TestCycleDuration=600* *TargetIODuration=10000* *UseScatterGather=FALSE*  [File1]  *FileName=sqliosim.mdx* *InitialSize=1000 MaxSize=1000* *Increment=10* *Shrinkable=FALSE* *LogFile=FALSE* *Sparse=FALSE*  [File2]  *FileName=sqliosim.ldx* *InitialSize=50* *MaxSize=50* *Increment=0* *Shrinkable=FALSE* *LogFile=TRUE* *Sparse=FALSE*  [File3]  *FileName=sqliosim.mdx:replica* *InitialSize=1000* *MaxSize=1000* *Increment=10* *Shrinkable=FALSE* *LogFile=FALSE* *Sparse=TRUE* |

**SQLIOSim configuration file**

You do not have to use a configuration file. If you do not use a configuration file, all parameters take default values except the data file location and the log file location. You must use one of the following methods to specify the data file location and the log file location:

* Use the command-line parameters in the SQLIOSim.com file.
* Use the **Files and Configuration** dialog box after you run the SQLIOSim.exe file.
* Use the File**x** section of the configuration file.

Notes

* If the name of the parameter indicates that the parameter is a ratio or a percentage, the value of the parameter is expressed as the percentage or the ratio, divided by 0.01. For example, the value of the CacheHitRatio parameter is 10 percent. This value is expressed as 1000 because 10, divided by 0.01, equals 1000. The maximum value of a percentage parameter is 10000.
* If the parameter type is numeric, and you assign a non-numeric value to the parameter, the SQLIOSim utility sets the parameter to 0.
* If the parameter type is Boolean, the valid values that you can assign to the parameter are true and false. Additionally, the values are case sensitive. The SQLIOSim utility ignores any invalid values.
* If a pair of parameters indicates a minimum value and a maximum value, the minimum value must not be larger than the maximum value. For example, the value of the MinIOChainLength parameter must not be larger than the value of the MaxIOChainLength parameter.
* If the parameter indicates a number of pages, the SQLIOSim utility checks the value that you assign to the parameter against the file that the SQLIOSim utility processes. The SQLIOSim utility performs this check to make sure that the number of pages does not exceed the file size.

#### CONFIG section

The SQLIOSim utility takes the values that you specify in the CONFIG section of the SQLIOSim configuration file to establish global testing behavior.

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter | Default value | Description | Comments |
| ErrorFile | sqliosim.log.xml | Name of the XML type log file |  |
| CPUCount | Number of CPUs on the computer | Number of logical CPUs to create | The maximum is 64 CPUs. |
| Affinity | 0 | Physical CPU affinity mask to apply for logical CPUs | The affinity mask should be within the active CPU mask. A value of 0 means that all available CPUs will be used. |
| MaxMemoryMB | Available physical memory when the SQLIOSim utility starts | Size of the buffer pool in MB | The value cannot exceed the total amount of physical memory on the computer. |
| StopOnError | true | Stops the simulation when the first error occurs |  |
| TestCycles | 1 | Number of full test cycles to perform | A value of 0 indicates an infinite number of test cycles. |
| TestCycleDuration | 300 | Duration of a test cycle in seconds, excluding the audit pass at the end of the cycle |  |
| CacheHitRatio | 1000 | Simulated cache hit ratio when the SQLIOSim utility reads from the disk |  |
| MaxOutstandingIO | 0 | Maximum number of outstanding I/O operations that are allowed process-wide | The value cannot exceed 140000. A value of 0 means that up to approximately 140,000 I/O operations are allowed. This is the limit of the utility. |
| TargetIODuration | 100 | Duration of I/O operations, in milliseconds, that are targeted by throttling | If the average I/O duration exceeds the target I/O duration, the SQLIOSim utility throttles the number of outstanding I/O operations to decrease the load and to improve I/O completion time. |
| AllowIOBursts | true | Allow for turning off throttling to post many I/O requests | I/O bursts are enabled during the initial update, initial checkpoint, and final checkpoint passes at the end of test cycles. The MaxOutstandingIO parameter is still honored. You can expect long I/O warnings. |
| NoBuffering | true | Use the FILE\_FLAG\_NO\_BUFFERING option | SQL Server opens database files by using FILE\_FLAG\_NO\_BUFFERING == true. Some utilities and services, such as Analysis Services, use FILE\_FLAG\_NO\_BUFFERING == false. To fully test a server, execute one test for each setting. |
| WriteThrough | true | Use the FILE\_FLAG\_WRITE\_THROUGH option | SQL Server opens database files by using FILE\_FLAG\_WRITE\_THROUGH == true. However, some utilities and services open the database files by using FILE\_FLAG\_WRITE\_THROUGH == false. For example, SQL Server Analysis Services opens the database files by using FILE\_FLAG\_WRITE\_THROUGH == false. To fully test a server, execute one test for each setting. |
| ScatterGather | true | Use ReadScatter/WriteGather APIs | If this parameter is set to true, the NoBuffering parameter is also set to true.  SQL Server uses scatter/gather I/Os for most I/O requests. |
| ForceReadAhead | true | Perform a read-ahead operation even if the data is already read | The SQLIOSim utility issues the read command even if the data page is already in the buffer pool.  Microsoft SQL Server Support has successfully used the true setting to expose I/O problems. |
| DeleteFilesAtStartup | true | Delete files at startup if files exist | A file may contain multiple data streams. Only streams that are specified in the File**x** FileName entry are truncated in the file. If the default stream is specified, all streams are deleted. |
| DeleteFilesAtShutdown | false | Delete files after the test is finished | A file may contain multiple data streams. Only data streams that you specify in the File**x** FileName entry are truncated in the file. If the default data stream is specified, the SQLIOSim utility deletes all data streams. |
| StampFiles | false | Expand the file by stamping zeros | This process may take a long time if the file is very large. If you set this parameter to false, the SQLIOSim utility extends the file by setting a valid data marker.  SQL Server 2005 uses the instant file initialization feature for data files. If the data file is a log file, or if instant file initialization is not enabled, SQL Server performs zero stamping. Versions of SQL Server earlier than SQL Server 2000 always perform zero stamping.  You should switch the value of the StampFiles parameter during testing to make sure that both instant file initialization and zero stamping are operating correctly. |

#### File**x** section

The SQLIOSim utility is designed to allow for multiple file testing. The File**x** section is represented as [File1], [File2] for each file in the test.

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter | Default value | Description | Comments |
| FileName | **No default value** | File name and path | The FileName parameter can be a long path or a UNC path. It can also include a secondary stream name and type. For example, the FileName parameter may be set to file.mdf:stream2.  Note In SQL Server 2005, DBCC operations use streams. We recommend that you perform stream tests. |
| InitialSize | **No default value** | Initial size in MB | If the existing file is larger than the value that is specified for the InitialSize parameter, the SQLIOSim utility does not shrink the existing file. If the existing file is smaller, the SQLIOSim utility expands the existing file. |
| MaxSize | **No default value** | Maximum size in MB | A file cannot grow larger than the value that you specify for the MaxSize parameter. |
| Increment | 0 | Size in MB of the increment by which the file grows or shrinks. For more information, see the "ShrinkUser section" part of this article. | The SQLIOSim utility adjusts the Increment parameter at startup so that the following situation is established:  Increment \* MaxExtents < MaxMemoryMB / NumberOfDataFiles  If the result is 0, the SQLIOSim utility sets the file as non-shrinkable. |
| Shrinkable | false | Indicates whether the file can be shrunk or expanded | If you set the Increment parameter to 0, you set the file to be non-shrinkable. In this case, you must set the Shrinkable parameter to false. If you set the Increment parameter to a value other than 0, you set the file to be shrinkable. In this case, you must set the Shrinkable parameter to true. |
| Sparse | false | Indicates whether the Sparse attribute should be set on the files | For existing files, the SQLIOSim utility does not clear the Sparse attribute when you set the Sparse parameter to false.  SQL Server 2005 uses sparse files to support snapshot databases and the secondary DBCC streams.  We recommend that you enable both the sparse file and the streams, and then perform a test pass.  Note If you set Sparse = true for the file settings, do not specify NoBuffering = false in the config section. If you use these two conflicting combinations, you may receive an error that resembles the following from the tool:  Error:-=====Error: 0x80070467 Error Text: While accessing the hard disk, a disk operation failed even after retries. Description: Buffer validation failed on C:\SQLIOSim.mdx Page: 28097 |
| LogFile | false | Indicates whether a file contains user or transaction log data | You should define at least one log file. |

#### RandomUser section

The SQLIOSim utility takes the values that you specify in the RandomUser section to simulate a SQL Server worker that is performing random query operations, such as Online Transaction Processing (OLTP) I/O patterns.

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter | Default value | Description | Comments |
| UserCount | -1 | Number of random access threads that are executing at the same time | The value cannot exceed the following value:  CPUCount\*1023-100  The total number of all users also cannot exceed this value. A value of 0 means that you cannot create random access users. A value of -1 means that you must use the automatic configuration of the following value:  min(CPUCount\*2, 8)  NoteA SQL Server system may have thousands of sessions. Most of the sessions do not have active requests. Use the count(\*) function in queries against the sys.dm\_exec\_requests dynamic management view (DMV) as a baseline for establishing this test parameter value.  CPUCount here refers to the value of the CPUCount parameter in the CONFIG section.  The min(CPUCount\*2, 8) value results in the smaller of the values between CPUCount\*2 and 8. |
| JumpToNewRegionPercentage | 500 | The chance of a jump to a new region of the file | The start of the region is randomly selected. The size of the region is a random value between the value of the MinIOChainLength parameter and the value of the MaxIOChainLength parameter. |
| MinIOChainLength | 1 | Minimum region size in pages |  |
| MaxIOChainLength | 100 | Maximum region size in pages | SQL Server 2005 Enterprise Edition and SQL Server 2000 Enterprise Edition can read ahead up to 1,024 pages.  The minimum value is 0. The maximum value is limited by system memory.  Typically, random user activity causes small scanning operations to occur. Use the values that are specified in the ReadAheadUser section to simulate larger scanning operations. |
| RandomUserReadWriteRatio | 9000 | Percentage of pages to be updated | A random-length chain is selected in the region and may be read. This parameter defines the percentage of the pages to be updated and written to disk. |
| MinLogPerBuffer | 64 | Minimum log record size in bytes | The value must be either a multiple of the on-disk sector size or a size that fits evenly into the on-disk sector size. |
| MaxLogPerBuffer | 8192 | Maximum log record size in bytes | This value cannot exceed 64000. The value must be a multiple of the on-disk sector size. |
| RollbackChance | 100 | The chance that an in-memory operation will occur that causes a rollback operation to occur. | When this rollback operation occurs, SQL Server does not write to the log file. |
| SleepAfter | 5 | Sleep time after each cycle, in milliseconds |  |

#### AuditUser section

The SQLIOSim utility takes the values that you specify in the AuditUser section to simulate DBCC activity to read and to audit the information about the page. Validation occurs even if the value of the UserCount parameter is set to 0.

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter | Default value | Description | Comments |
| UserCount | 2 | Number of Audit threads | The value cannot exceed the following value:  CPUCount\*1023-100  The total number of all users also cannot exceed this value. A value of 0 means that you cannot create random access users. A value of -1 means that you must use the automatic configuration of the following value:  min(CPUCount\*2, 8)  NoteA SQL Server system may have thousands of sessions. Most of the sessions do not have active requests. Use the count(\*) function in queries against the sys.dm\_exec\_requests DMV as a baseline for establishing this test parameter value.  CPUCount here refers to the value of the CPUCount parameter in the CONFIG section.  The min(CPUCount\*2, 8) value results in the smaller of the values between CPUCount\*2 and 8. |
| BuffersValidated | 64 |  |  |
| DelayAfterCycles | 2 | Apply the AuditDelay parameter after the number of BuffersValidated cycles is completed |  |
| AuditDelay | 200 | Number of milliseconds to wait after each DelayAfterCycles operation |  |

#### ReadAheadUser section

The SQLIOSim utility takes the values that are specified in the ReadAheadUser section to simulate SQL Server read-ahead activity. SQL Server takes advantage of read-ahead activity to maximize asynchronous I/O capabilities and to limit query delays.

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter | Default value | Description | Comments |
| UserCount | 2 | Number of read-ahead threads | The value cannot exceed the following value:  CPUCount\*1023-100  The total number of all users also cannot exceed this value. A value of 0 means that you cannot create random access users. A value of -1 means that you must use the automatic configuration of the following value:  min(CPUCount\*2, 8)  NoteA SQL Server system may have thousands of sessions. Most of the sessions do not have active requests. Use the count(\*) function in queries against the sys.dm\_exec\_requests DMV as a baseline for establishing this test parameter value.  CPUCount here refers to the value of the CPUCount parameter in the CONFIG section.  The min(CPUCount\*2, 8) value results in the smaller of the values between CPUCount\*2 and 8. |
| BuffersRAMin | 32 | Minimum number of pages to read per cycle | The minimum value is 0. The maximum value is limited by system memory. |
| BuffersRAMax | 64 | Maximum number of pages to read per cycle | SQL Server Enterprise editions can read up to 1,024 pages in a single request. If you install SQL Server on a computer that has lots of CPU, memory, and disk resources, we recommend that you increase the file size and the read-ahead size. |
| DelayAfterCycles | 2 | Apply the RADelay parameter after the specified number of cycles is completed |  |
| RADelay | 200 | Number of milliseconds to wait after each DelayAfterCycles operation |  |

#### BulkUpdateUser section

The SQLIOSim utility takes the values that you specify in the BulkUpdateUser section to simulate bulk operations, such as SELECT...INTO operations and BULK INSERT operations.

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter | Default value | Description | Comments |
| UserCount | -1 | Number of BULK UPDATE threads | The value cannot exceed the following value:  CPUCount\*1023-100  A value of -1 means that you must use the automatic configuration of the following value:  min(CPUCount\*2, 8)  NoteA SQL Server system may have thousands of sessions. Most of the sessions do not have active requests. Use the count(\*) function in queries against the sys.dm\_exec\_requests DMV as a baseline for establishing this test parameter value.  CPUCount here refers to the value of the CPUCount parameter in the CONFIG section.  The min(CPUCount\*2, 8) value results in the smaller of the values between CPUCount\*2 and 8. |
| BuffersBUMin | 64 | Minimum number of pages to update per cycle |  |
| BuffersBUMax | 128 | Maximum number of pages to update per cycle | The minimum value is 0. The maximum value is limited by system memory. |
| DelayAfterCycles | 2 | Apply the BUDelay parameter after the specified number of cycles is completed |  |
| BUDelay | 10 | Number of milliseconds to wait after each DelayAfterCycles operation |  |

#### ShrinkUser section

The SQLIOSim utility takes the values that you specify in the ShrinkUser section to simulate DBCC shrink operations. The SQLIOSim utility can also use the ShrinkUser section to make the file grow.

|  |  |  |
| --- | --- | --- |
| Parameter | Default value | Description |
| MinShrinkInterval | 120 | Minimum interval between shrink operations, in seconds |
| MaxShrinkInterval | 600 | Maximum interval between shrink operations, in seconds |
| MinExtends | 1 | Minimum number of increments by which the SQLIOSim utility will grow or shrink the file |
| MaxExtends | 20 | Maximum number of increments by which the SQLIOSim utility will grow or shrink the file |