JmxInterface

If you start it using the standard startup script, Cassandra will listen for connections on port 8080 (port 7199 starting in 0.8.0-beta1) to view and tweak variables which it exposes via JMX. This may be helpful for debugging and monitoring. See also JmxGotchas.

The MemtableThresholds page describes how to use Jconsole as a client for this.

Domain	Source Location	Type	Keyspace	Cache	Attributes	Operations
org.apache.cassandra.concurrent	./concurrent/JMXEnabledThreadPoolExecutorMBean.java	AE-SERVICE-STAGE			ActiveCount CompletedTasks PendingTasks	
org.apache.cassandra.concurrent	./concurrent/JMXEnabledThreadPoolExecutorMBean.java	COMMITLOG			ActiveCount CompletedTasks PendingTasks	
org.apache.cassandra.concurrent	./concurrent/JMXEnabledThreadPoolExecutorMBean.java	CONSISTENCY-MANAGER			ActiveCount CompletedTasks PendingTasks	
org.apache.cassandra.concurrent	./concurrent/JMXEnabledThreadPoolExecutorMBean.java	FILEUTILS-DELETE-POOL			ActiveCount CompletedTasks PendingTasks	
org.apache.cassandra.concurrent	./concurrent/JMXEnabledThreadPoolExecutorMBean.java	FLUSH-SORTER-POOL			ActiveCount CompletedTasks PendingTasks	
org.apache.cassandra.concurrent	./concurrent/JMXEnabledThreadPoolExecutorMBean.java	FLUSH-WRITER-POOL			ActiveCount CompletedTasks PendingTasks	
org.apache.cassandra.concurrent	./concurrent/JMXEnabledThreadPoolExecutorMBean.java	GMFD			ActiveCount CompletedTasks PendingTasks	
org.apache.cassandra.concurrent	./concurrent/JMXEnabledThreadPoolExecutorMBean.java	HINTED-HANDOFF-POOL			ActiveCount CompletedTasks PendingTasks	
org.apache.cassandra.concurrent	./concurrent/JMXEnabledThreadPoolExecutorMBean.java	LB-OPERATIONS			ActiveCount CompletedTasks PendingTasks	
org.apache.cassandra.concurrent	./concurrent/JMXEnabledThreadPoolExecutorMBean.java	LB-TARGET			ActiveCount CompletedTasks PendingTasks	
org.apache.cassandra.concurrent	./concurrent/JMXEnabledThreadPoolExecutorMBean.java	LOAD-BALANCER-STAGE			ActiveCount CompletedTasks PendingTasks	
org.apache.cassandra.concurrent	./concurrent/JMXEnabledThreadPoolExecutorMBean.java	MEMTABLE-POST-FLUSHER			ActiveCount CompletedTasks PendingTasks	
org.apache.cassandra.concurrent	./concurrent/JMXEnabledThreadPoolExecutorMBean.java	MESSAGE-DESERIALIZER-POOL			ActiveCount CompletedTasks PendingTasks	
org.apache.cassandra.concurrent	./concurrent/JMXEnabledThreadPoolExecutorMBean.java	MESSAGE-STREAMING-POOL			ActiveCount CompletedTasks PendingTasks	
org.apache.cassandra.concurrent	./concurrent/JMXEnabledThreadPoolExecutorMBean.java	RESPONSE-STAGE			ActiveCount CompletedTasks PendingTasks	
org.apache.cassandra.concurrent	./concurrent/JMXEnabledThreadPoolExecutorMBean.java	ROW-MUTATION-STAGE			ActiveCount CompletedTasks PendingTasks	
org.apache.cassandra.concurrent	./concurrent/JMXEnabledThreadPoolExecutorMBean.java	ROW-READ-STAGE			ActiveCount CompletedTasks PendingTasks	
org.apache.cassandra.concurrent	./concurrent/JMXEnabledThreadPoolExecutorMBean.java	STREAM-STAGE			ActiveCount CompletedTasks PendingTasks	

org.apache.cassandra.db	./db/CommitLogExecutorServiceMBean.java	CommitLog			ActiveCount CompletedTasks PendingTasks	
org.apache.cassandra.db	./db/CompactionManagerMBean.java	CompactionManager			BytesCompacted BytesTotalInProgress ColumnFamilyInProgress MaximumCompactionThreshold MinimumCompactionThreshold PendingTasks	
org.apache.cassandra.gms	./gms/FailureDetectorMBean.java	FailureDetector				dumpInterArrivalTimes
org.apache.cassandra.service	./service/StorageServiceMBean.java	StorageService			CurrentGenerationNumber LiveNodes LoadMap LoadString RangeToEndPointMap Token UnreachableNodes	
clearSnapshot decommission forceTableCleanup forceTableCompaction forceTableFlush forceTableRepair getLiveNaturalEndpoints getNaturalEndpoints getRangeToEndPointMap loadBalance move removeToken setLog4jLevel takeAllSnapshot takeSnapshot						
org.apache.cassandra.service	./service/StorageProxyMBean.java	StorageProxy			RangeOperations ReadOperations RecentRangeLatencyMicros RecentReadLatencyMicros RecentWriteLatencyMicros TotalRangeLatencyMicros TotalReadLatencyMicros TotalWriteLatencyMicros WriteOperations	
org.apache.cassandra.streaming	./streaming/StreamingService.java	StreamingService			Status StreamDestinations StreamSources	
getIncomingFiles getOutgoingFiles						

org.apache.cassandra.concurrent

All org.apache.cassandra.concurrent Types are thread pool executors which have Attributes ActiveCount, CompletedTasks and PendingTasks. They are viewable from the command line with **{{'}}nodeprobe tpstats{{'}}**.

They are generally defined as follows :

org.apache.cassandra.concurrent.<THREADPOOL>.Attributes.ActiveCount

Type	Units	Range	Notes
int	tasks	integer >=0	The current number of running tasks.

org.apache.cassandra.concurrent.<THREADPOOL>.Attributes.CompletedTasks

Type	Units	Range	Notes
long	tasks	integer >=0	The number of completed tasks since executor creation.

org.apache.cassandra.concurrent.<THREADPOOL>.Attributes.PendingTasks

Type	Units	Range	Notes
long	tasks	integer >=0	The number of tasks waiting in the queue to be executed.

org.apache.cassandra.concurrent.AE-SERVICE-STAGE

This thread pool is single threaded.

Per src/java/org/apache/cassandra/service/AntiEntropyService.java,

```
AntiEntropyService encapsulates "validating" (hashing) individual column families, exchanging MerkleTrees with
remote nodes via a TreeRequest/Response conversation, and then triggering repairs for disagreeing ranges.

Tree comparison and repair triggering occur in the single threaded AE_SERVICE_STAGE.
```

org.apache.cassandra.concurrent.CONSISTENCY-MANAGER

This pool has 4 threads, which is hardcoded per src/java/org/apache/cassandra/service/StorageService.java,

```
    private static int consistencyThreads_ = 4; // not configurable
```

The CONSISTENCY-MANAGER executor is responsible for passive read repair, for example after a read with ConsistencyLevel.ONE.

Per src/java/org/apache/cassandra/service/StorageService.java,

```
/* This thread pool does consistency checks when the client doesn't care about consistency */
```

org.apache.cassandra.concurrent.FILEUTILS-DELETE-POOL

This thread pool is single threaded. Code is defined in src/java/org/apache/cassandra/io/DeletionService.java. This executor is responsible for deleting files, such as obsolete sstables and obsolete commitlog segments.

org.apache.cassandra.concurrent.FLUSH-SORTER-POOL

This thread pool is multi-threaded, with between 1 and Runtime.getRuntime().availableProcessors() threads in the pool.

See Also : FLUSH-WRITER-POOL and MEMTABLE-POST-FLUSHER. Per src/java/org/apache/cassandra/db/ColumnFamilyStore.java,

```
     * submitFlush first puts [Binary]Memtable.getSortedContents on the flushSorter executor,
     * which then puts the sorted results on the writer executor.  This is because sorting is CPU-bound,
     * and writing is disk-bound; we want to be able to do both at once.  When the write is complete,
     * we turn the writer into an SSTableReader and add it to ssTables_ where it is available for reads.
     *
     * For BinaryMemtable that's about all that happens.  For live Memtables there are two other things
     * that switchMemtable does (which should be the only caller of submitFlush in this case).
     * First, it puts the Memtable into memtablesPendingFlush, where it stays until the flush is complete
     * and it's been added as an SSTableReader to ssTables_.  Second, it adds an entry to commitLogUpdater
     * that waits for the flush to complete, then calls onMemtableFlush.  This allows multiple flushes
     * to happen simultaneously on multicore systems, while still calling onMF in the correct order,
     * which is necessary for replay in case of a restart since CommitLog assumes that when onMF is
     * called, all data up to the given context has been persisted to SSTables.
```

org.apache.cassandra.concurrent.FLUSH-WRITER-POOL

This thread pool is multi-threaded, with between 1 and Runtime.getRuntime().availableProcessors() threads in the pool.

See Also : FLUSH-SORTER-POOL and MEMTABLE-POST-FLUSHER. Per src/java/org/apache/cassandra/db/ColumnFamilyStore.java,

```
    * submitFlush first puts [Binary]Memtable.getSortedContents on the flushSorter executor,
    * which then puts the sorted results on the writer executor.  This is because sorting is CPU-bound,
    * and writing is disk-bound; we want to be able to do both at once.  When the write is complete,
    * we turn the writer into an SSTableReader and add it to ssTables_ where it is available for reads.
    *
    * For BinaryMemtable that's about all that happens.  For live Memtables there are two other things
    * that switchMemtable does (which should be the only caller of submitFlush in this case).
    * First, it puts the Memtable into memtablesPendingFlush, where it stays until the flush is complete
    * and it's been added as an SSTableReader to ssTables_.  Second, it adds an entry to commitLogUpdater
    * that waits for the flush to complete, then calls onMemtableFlush.  This allows multiple flushes
    * to happen simultaneously on multicore systems, while still calling onMF in the correct order,
    * which is necessary for replay in case of a restart since CommitLog assumes that when onMF is
    * called, all data up to the given context has been persisted to SSTables.
```

org.apache.cassandra.concurrent.GMFD

This thread pool is single threaded. Executor for the Gossiper.

org.apache.cassandra.concurrent.HINTED-HANDOFF-POOL

This thread pool is single threaded.

Per src/java/org/apache/cassandra/db/HintedHandOffManager.java,

```
 * For each table (keyspace), there is a row in the system hints CF.
 * SuperColumns in that row are keys for which we have hinted data.
 * Subcolumns names within that supercolumn are host IPs. Subcolumn values are always empty.
 * Instead, we store the row data "normally" in the application table it belongs in.
 *
 * So when we deliver hints we look up endpoints that need data delivered
 * on a per-key basis, then read that entire row out and send it over.
```

org.apache.cassandra.concurrent.LB-OPERATIONS

This thread pool is single threaded.

See Also : LB-TARGET. Per src/java/org/apache/cassandra/service/StorageLoadBalancer.java,

```
/* This thread pool is used for initiating load balancing operations */
```

org.apache.cassandra.concurrent.LB-TARGET

This thread pool is single threaded.

See Also : LB-OPERATIONS. Per src/java/org/apache/cassandra/service/StorageLoadBalancer.java,

```
/* This thread pool is used by target node to leave the ring. */
```

org.apache.cassandra.concurrent.LOAD-BALANCER-STAGE

This thread pool is single threaded.

org.apache.cassandra.concurrent.MEMTABLE-POST-FLUSHER

This thread pool is single threaded.

See Also : FLUSH-SORTER-POOL and FLUSH-WRITER-POOL. Per src/java/org/apache/cassandra/db/ColumnFamilyStore.java,

```
     * submitFlush first puts [Binary]Memtable.getSortedContents on the flushSorter executor,
     * which then puts the sorted results on the writer executor.  This is because sorting is CPU-bound,
     * and writing is disk-bound; we want to be able to do both at once.  When the write is complete,
     * we turn the writer into an SSTableReader and add it to ssTables_ where it is available for reads.
     *
     * For BinaryMemtable that's about all that happens.  For live Memtables there are two other things
     * that switchMemtable does (which should be the only caller of submitFlush in this case).
     * First, it puts the Memtable into memtablesPendingFlush, where it stays until the flush is complete
     * and it's been added as an SSTableReader to ssTables_.  Second, it adds an entry to commitLogUpdater
     * that waits for the flush to complete, then calls onMemtableFlush.  This allows multiple flushes
     * to happen simultaneously on multicore systems, while still calling onMF in the correct order,
     * which is necessary for replay in case of a restart since CommitLog assumes that when onMF is
     * called, all data up to the given context has been persisted to SSTables.
```

org.apache.cassandra.concurrent.MESSAGE-DESERIALIZER-POOL

This thread pool is multi-threaded, with between 1 and Runtime.getRuntime().availableProcessors() threads in the pool.

Per src/java/org/apache/cassandra/net/MessagingService.java,

```
// read executor puts messages to deserialize on this.
```

org.apache.cassandra.concurrent.MESSAGE-STREAMING-POOL

This thread pool is single threaded.

org.apache.cassandra.concurrent.RESPONSE-STAGE

This thread pool is mutli-threaded and has between 1 and Runtime.getRuntime().availableProcessors() threads in the pool.

org.apache.cassandra.concurrent.ROW-MUTATION-STAGE

This thread pool is multi-threaded and has between 1 and concurrectWriters threads in the pool.

Per src/java/org/apache/cassandra/concurrent/StageManager.java,

```
    private static int concurrentWriters_ = 32;
```

org.apache.cassandra.concurrent.ROW-READ-STAGE

This thread pool is multi-threaded and has between 1 and concurrectReaders threads in the pool.

Per src/java/org/apache/cassandra/concurrent/StageManager.java,

```
    private static int concurrentReaders_ = 8;
```

org.apache.cassandra.concurrent.STREAM-STAGE

This stage is single threaded.

org.apache.cassandra.db

org.apache.cassandra.db.CommitLog

org.apache.cassandra.db.CommitLog.Attributes.ActiveCount

Type	Units	Range	Notes
int	tasks	>=0	The number of tasks which are currently executing.

org.apache.cassandra.db.CommitLog.Attributes.CompletedTasks

Type	Units	Range	Notes
long	tasks	>=0	The number of completed tasks.

org.apache.cassandra.db.CommitLog.Attributes.PendingTasks

Type	Units	Range	Notes
long	tasks	>=0	The number of tasks waiting in the queue to be executed.

org.apache.cassandra.db.CompactionManager

org.apache.cassandra.db.CompactionManager.Attributes.BytesCompacted

Type	Units	Range	Notes
java.lang.Long	bytes	integer >=0	The number of bytes successfully compacted.

org.apache.cassandra.db.CompactionManager.Attributes.BytesTotalInProgress

Type	Units	Range	Notes
java.lang.Long	bytes	integer >=0	FIXME?? : The total size of the SSTables involved in the current compaction.

org.apache.cassandra.db.CompactionManager.Attributes.ColumnFamilyInProgress

Type	Units	Range	Notes
java.lang.string	name	Any valid Column{{`Family name	The name of the Column}}`Family currently being compacted.

org.apache.cassandra.db.CompactionManager.Attributes.MaximumCompactionThreshold

Type	Units	Range	Notes
int	SSTables	>=0	The maximum number of SSTables in the compaction queue before compaction kicks off.

org.apache.cassandra.db.CompactionManager.Attributes.MinimumCompactionThreshold

Type	Units	Range	Notes
int	SSTables	>=0	The minimum number of SSTables in the compaction queue before compaction kicks off.

org.apache.cassandra.db.CompactionManager.Attributes.PendingTasks

Type	Units	Range	Notes
int	tasks	>=0	The number of tasks waiting in the queue to be executed.

org.apache.cassandra.gms

org.apache.cassandra.gms.FailureDetector

org.apache.cassandra.gms.FailureDetector.Operations.dumpInterArrivalTimes

Dump endpoint arrival windows to a file in /var/tmp, per src/java/org/apache/cassandra/gms/FailureDetector.java:

```
FileOutputStream fos = new FileOutputStream("/var/tmp/output-" + System.currentTimeMillis() + ".dat", true);
```

org.apache.cassandra.service

org.apache.cassandra.service.StorageService

org.apache.cassandra.service.StorageService.Attributes.CurrentGenerationNumber

Type	Units	Range	Notes
int	Generation Number	>=0	The number of the current generation in the Gossiper.

org.apache.cassandra.service.StorageService.Attributes.LiveNodes

Type	Units	Range	Notes
java.util.Set	Nodes	n/a	A set of the nodes which are visible and live, from the perspective of this node.

org.apache.cassandra.service.StorageService.Attributes.LoadMap

Type	Units	Range	Notes	Example
java.util.Map	Nodes,Disk Usage	n/a	A map of which nodes have what level of load.	{10.0.0.130=107.86 GB,10.0.0.41=125.82 GB,10.0.0.176=117.47 GB,10.0.0.15=68.65 GB,10.0.0.91=148.25 GB, 10.0.0.165=247.33 GB]

org.apache.cassandra.service.StorageService.Attributes.LoadString

Type	Units	Range	Notes	Example
java.lang.String	Disk usage	n/a	The amount of load on the node being queried.	127.7 GB

org.apache.cassandra.service.StorageService.Attributes.OperationMode

Type	Units	Range	Notes	Examples
java.lang.String	Operation mode string	n/a	A string describing the current operation mode. (FIXME: wikilink on operation mode?)	Decommissioned, Joining: getting bootstrap token, Joining: getting load information, Leaving: streaming data to other nodes, Normal, Joining: sleeping " + RING_DELAY + " for pending range setup, Leaving: sleeping " + RING_DELAY + " for pending range setup

org.apache.cassandra.service.StorageService.Attributes.Token

Type	Units	Range	Notes	Example
java.lang.String	key range start position	n/a	A string describing the start of the range of keys this node is responsible for on the ring.	

org.apache.cassandra.service.StorageService.Attributes.UnreachableNodes

Type	Units	Range	Notes	Example		
<ac:structured-macro ac:name="unmigrated-wiki-markup" ac:schema-version="1" ac:macro-id="093d369a-b21e-4d82-9d4f-d28292ffdc99"><ac:plain-text-body><![CDATA[	java.util.Set	Nodes	n/a	A set of the nodes which this node knows about which are currently unreachable.	[10.0.0.1, 10.0.0.2]	]]></ac:plain-text-body></ac:structured-macro>

org.apache.cassandra.service.StorageService.Operations.clearSnapshot

Arguments	Return Type	Notes
n/a	void	Clear all the snapshots for this node's keyspace.

Per src/java/org/apache/cassandra/db/Table.java :

```
            String snapshotPath = dataDirPath + File.separator + name + File.separator + SNAPSHOT_SUBDIR_NAME;
...
                FileUtils.deleteDir(snapshotDir);
```

org.apache.cassandra.service.StorageService.Operations.decommission

Arguments	Return Type	Notes
n/a	void	Instruct this (live) node to remove itself from the token ring.

This operation may fail if :

- The node is not a member of the token ring yet.
- There are no other normal nodes in the ring.
- Data is currently moving to the node.

org.apache.cassandra.service.StorageService.Operations.forceTableCleanup

Arguments	Return Type	Notes
n/a	void	Trigger a cleanup compaction. Goes over each file and removes the keys that the node is not responsible for, and only keeps keys that this node is responsible for.

org.apache.cassandra.service.StorageService.Operations.forceTableCompaction

Arguments	Return Type	Notes
n/a	void	Trigger a major compaction (of all SSTables on disk).

org.apache.cassandra.service.StorageService.Operations.forceTableFlush

Argument p1	Argument p1 Type	Argument p2	Argument p2 Type	Return Type	Notes	Example
tableName	java.lang.String	columnFamilies	java.lang.String...	void	Flush all memtables for a table and column families.	

org.apache.cassandra.service.StorageService.Operations.forceTableRepair

Argument p1	Argument p1 Type	Argument p2	Argument p2 Type	Return Type	Notes	Example
tableName	java.lang.String	columnFamilies	java.lang.String...	void	Trigger proactive repair for a table and column families	

org.apache.cassandra.service.StorageService.Operations.getLiveNaturalEndpoints

Argument p1	Argument p1 Type	Argument p2	Argument p2 Type	Return Type	Notes	Example
tableName	java.lang.String	token	java.lang.String	java.lang.String	Attempts to return N endpoints that are responsible for storing the specified key i.e for replication.	

org.apache.cassandra.service.StorageService.Operations.getNaturalEndpoints

Argument p1	Argument p1 Type	Argument p2	Argument p2 Type	Return Type	Notes	Example
tableName	java.lang.String	token	java.lang.String	java.lang.String	Returns the N endpoints that are responsible for storing the specified key i.e for replication.	

org.apache.cassandra.service.StorageService.Operations.getRangeToEndPointMap

Argument p1	Argument p1 Type	Return Type	Notes	Example
keyspace	java.lang.String	java.lang.String	For a keyspace, return the ranges and corresponding hosts for a given keyspace.	

org.apache.cassandra.service.StorageService.Operations.loadBalance

Arguments	Return Type	Notes
n/a	void	Generate new auto-assigned token for this node, between the two most heavily loaded nodes.

org.apache.cassandra.service.StorageService.Operations.move

Argument p1	Argument p1 Type	Return Type	Notes
keyspace	java.lang.String	void	Move this node to a new token, specified as the argument.

This operation may fail if :

- The target token is already owned by another node.
- Data is currently moving to this node.

org.apache.cassandra.service.StorageService.Operations.removeToken

Argument p1	Argument p1 Type	Return Type	Notes
keyspace	java.lang.String	void	Remove the specified token from the ring. Used on a live node to remove the token of a dead node from all nodes in the ring.

This operation may fail if :

- There is a live node which owns this token.

org.apache.cassandra.service.StorageService.Operations.setLog4jLevel

Argument p1	Argument p1 Type	Argument p2	Argument p2 Type	Return Type	Notes	Example
classQualifier	java.lang.String	rawLevel	java.lang.String	void	Set the level of logging in Log4j.	DEBUG INFO WARN ERROR FATAL ALL OFF - http://jakarta.apache.org/log4j/docs/api/index.html

org.apache.cassandra.service.StorageService.Operations.takeAllSnapshot

Argument p1	Argument p1 Type	Return Type	Notes	Example
tag	java.lang.String or null	void	Take a snapshot of all keyspaces on this node and optionally name it with a (non-null) tag name.	

org.apache.cassandra.service.StorageService.Operations.takeSnapshot

Argument p1	Argument p1 Type	Argument p2	Argument p2 Type	Return Type	Notes	Example
tableName (Keyspace Name)	java.lang.String	tag	java.lang.String or null	void	Take a snapshot of a keyspace and optionally name it with a (non-null) tag name.	

org.apache.cassandra.service.StorageProxy

org.apache.cassandra.service.StorageProxy.Attributes.RangeOperations

Type	Units	Range	Notes
long	Operations	>=0	The number of range operations since executor start.

org.apache.cassandra.service.StorageProxy.Attributes.ReadOperations

Type	Units	Range	Notes
long	Operations	>=0	The number of read operations since executor start.

org.apache.cassandra.service.StorageProxy.Attributes.RecentRangeLatencyMicros

Type	Units	Range	Notes
double	Microseconds	>=0	The latency of range operations since the last time this attribute was read.

See Also : RecentReadLatencyMicros, RecentWriteLatencyMicros.

Per src/java/org/apache/cassandra/utils/Latency{{`Tracker.}}`java/getRecentLatencyMicros :

```
        long ops = opCount.get();
        long n = totalLatency.get();
        return ((double)n - lastLatency) / (ops - lastOpCount);
        ...
        lastLatency = n;
        lastOpCount = ops;
```

org.apache.cassandra.service.StorageProxy.Attributes.RecentReadLatencyMicros

Type	Units	Range	Notes
double	Microseconds	>=0	The latency of range operations since the last time this attribute was read.

See Also RecentRangeLatencyMicros, RecentWriteLatencyMicros.

org.apache.cassandra.service.StorageProxy.Attributes.RecentWriteLatencyMicros

Type	Units	Range	Notes
double	Microseconds	>=0	The latency of write operations since the last time this attribute was read.

See Also : RecentRangeLatencyMicros, RecentReadLatencyMicros.

org.apache.cassandra.service.StorageProxy.Attributes.TotalRangeLatencyMicros

Type	Units	Range	Notes
long	Microseconds	>=0	The latency of all range operations since executor start.

org.apache.cassandra.service.StorageProxy.Attributes.TotalReadLatencyMicros

Type	Units	Range	Notes
long	Microseconds	>=0	The latency of all read operations since executor start.

org.apache.cassandra.service.StorageProxy.Attributes.TotalWriteLatencyMicros

Type	Units	Range	Notes
long	Microseconds	>=0	The latency of all write operations since executor start.

org.apache.cassandra.service.StorageProxy.Attributes.WriteOperations

Type	Units	Range	Notes
long	Operations	>=0	The number of write operations since executor start.

org.apache.cassandra.streaming

org.apache.cassandra.streaming.StreamingService

org.apache.cassandra.streaming.StreamingService.Attributes.Status

Type	Units	Range	Notes	Example
java.lang.String	Status string	n/a	A string describing the current state of the StreamingService.	Flushing memtables for <tableName>, Performing anticompaction, Sending a stream initiate message to <target>, Done with transfer to <target>

org.apache.cassandra.streaming.StreamingService.Attributes.StreamDestinations

Type	Units	Range	Notes	Example
java.lang.Set	Nodes	n/a	A set of the nodes this node is currently streaming to.	

org.apache.cassandra.streaming.StreamingService.Attributes.StreamSources

Type	Units	Range	Notes	Example
java.lang.Set	Nodes	n/a	A set of the nodes this node is currently streaming from.	

org.apache.cassandra.streaming.StreamingService.Operations.getIncomingFiles

Argument p1	Argument p1 Type	Return Type	Notes	Example
node	java.lang.String	java.Util.List	Given a node as an argument, return a List of filenames, current position in the file and how many total bytes are expected in the stream. Shows incoming streams only.	

See Also : getOutgoingFiles.

Per src/java/org/apache/cassandra/streaming/StreamingService.java :

```
            files.add(String.format("%s %d/%d", f.getFilename(), f.getPtr(), f.getExpectedBytes()));
```

org.apache.cassandra.streaming.StreamingService.Operations.getOutgoingFiles

Argument p1	Argument p1 Type	Return Type	Notes	Example
node	java.lang.String	java.Util.List	Given a node as an argument, return a List of filenames, current position in the file and how many total bytes are expected in the stream. Shows outgoing streams only.	

See Also : getIncomingFiles.