- `StringMemory`: Default size of string memory (0 to 100 = % of maximum, 101+ = actual bytes).
- `TcpBind_INADDR_ANY`: Bind IP_ADDR_ANY so that connections can be made from anywhere (for autogenerated connections).
- `TimeBetweenEpochs`: Time between epochs (synchronization used for replication).
- `TimeBetweenEpochsTimeout`: Timeout for time between epochs. Exceeding causes node shutdown.
- `TimeBetweenGlobalCheckpoints`: Time between group commits of transactions to disk.
- `TimeBetweenGlobalCheckpointsTimeout`: Minimum timeout for group commit of transactions to disk.
- `TimeBetweenInactiveTransactionAbortCheck`: Time between checks for inactive transactions.
- `TimeBetweenLocalCheckpoints`: Time between taking snapshots of database (expressed in base-2 logarithm of bytes).
- `TimeBetweenWatchDogCheck`: Time between execution checks inside data node.
- `TimeBetweenWatchDogCheckInitial`: Time between execution checks inside data node (early start phases when memory is allocated).
- `TotalSendBufferMemory`: Total memory to use for all transporter send buffers..
- `TransactionBufferMemory`: Dynamic buffer space (in bytes) for key and attribute data allocated for each data node.
- `TransactionDeadlockDetectionTimeout`: Time transaction can spend executing within data node. This is time that transaction coordinator waits for each data node participating in transaction to execute request. If data node takes more than this amount of time, transaction is aborted.
- `TransactionInactiveTimeout`: Milliseconds that application waits before executing another part of transaction. This is time transaction coordinator waits for application to execute or send another part (query, statement) of transaction. If application takes too much time, then transaction is aborted. Timeout = 0 means that application never times out.
- `TransactionMemory`: Memory allocated for transactions on each data node.
- `TwoPassInitialNodeRestartCopy`: Copy data in 2 passes during initial node restart, which enables multithreaded building of ordered indexes for such restarts.
- `UndoDataBuffer`: Number of bytes on each data node allocated for writing data undo logs.
- `UndoIndexBuffer`: Number of bytes on each data node allocated for writing index undo logs.
- `UseShm`: Use shared memory connections between this data node and API node also running on this host.

The following parameters are specific to `ndbmtd`:

- `AutomaticThreadConfig`: Use automatic thread configuration; overrides any settings for ThreadConfig and MaxNoOfExecutionThreads.
- `ClassicFragmentation`: When true, use traditional table fragmentation; set false to enable flexible distribution of table fragments among LDMs.
- `MaxNoOfExecutionThreads`: For ndbmtd only, specify maximum number of execution threads.