EmitClusterProcessesV1

*italicised* names are shared channels, dashed channels are point-to-point data channel arrays

emitCluster

data.1.n

*One collector per emitNode*

collector

Cluster 1

WBv0

emitter

emitter

*iTOs / sTowb*

emitNode

rts

usessert Systtem :[deterministic]

e

EmitNode comprises a number of Emitter processes and a single WriteBuffer, which contains no queue structure.

Emitter creates and writes data objects to its WriteBuffer until it has no more data objects to produce. It then sends a UT to the WriteBuffer and then terminates.

WriteBuffer reads data objects from its Emitters. It then sends its nodeID to its Manger process using the shared readyToSend channel. It then reads the nodeID of the node to which the data object should be written and writes the data object accordingly. The WriteBuffer process continues doing this until all the Emitter processes have sent it a UT, when it:

Writes a single UT on each of its data channels to the reading nodes.

Writes a single UT to the Manager using the readyToSend channels.

WriteBuffer then terminates.

WorkNode comprises a number of Worker processes plus one ReadBuffer and one WriteBuffer.

CollectorNode comprises a number of Collector processes plus one ReadBuffer.

ReadBuffer contains a queue the size of the number of internal Worker or Collector processes in the node. The internal buffer is organised as a circular queue.

Alternates over:

Reading data objects into its queue buffer until the number of UT’s read matches the number of nodes that write to this node at which point the process enters a termination phase.

Reading the index of a Worker or Collector process that is ready to read another data object. It then writes the next data object from the buffer to the requesting process, modifying the queue variables accordingly. It then writes it nodeID to the Manager process indicating that it can read another data object into its buffer.

In the termination phase the ReadBuffer process first ensures that any data objects still residing in the buffer are processed.

It then sends a UT to each of the Worker of Collector processes.

It then terminates.

Worker process writes its process subscript to the ReadBuffer. It then reads a data object and processes it, using the data object’s processing method, before writing the processed data object to its WriteBuffer. It does this until it reads a UT upon which it writes the UT to the WriteBuffer before it terminates.

A Collector process writes its process subscript to the ReadBuffer. It then reads a data object and processes it using a collate method. It does this until it reads a UT upon which it undertakes any finalise method and then it terminates.