Lecture 7 summary

Optimise code so that each thread accesses the memory in an optimal manner such as special locality where each thread accesses data items close by at a time

Temporal locality- data item can be used multiple times when read in from the memory

GPU’s – many core systems, simplified in comparison to multi-processor

Shared memory system is cheaper but less scalable than distributed memory systems

Graphical user interface, text, application

Description automatically generated

MPI Source- returns rank of sender process

MPI Tag- Tag of received message

MPI Error- error code

Wildcards:

MPI\_ANY\_SOURCE

MPI\_ANY\_TAG- don’t want to specify a particular tag message

When they are non-blocking, we cannot safely use send/ receive yet

Graphical user interface, text, application

Description automatically generated

MPI Reduce-> first is local value, buffer of aggregated val, count of how many values to aggregate, datatype of vars, type of operation, target destination process

Text

Description automatically generated with medium confidence

Mostly blocking communications