

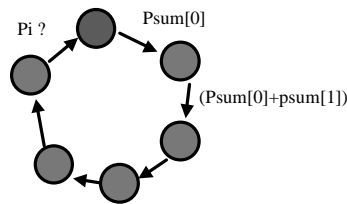
Lecture 7 homework

- Start distributing MPI jobs so we can understand GRID applications in lecture 10.
- MPI Application across multiple systems using different methods to inter-operate, as well as different programming models
 - Single MPI_COMM_WORLD vs multiple MPI_COMM_WORLDs
- Deadline midday 12th March (almost 2 weeks)

The First Application

- MPI application that has the communication topology of a ring.
- The application is from a single piece of code that is structured to allow for two separate MPI executables or a single executable via an IFDEF statement.
- The ring calculates the value π using the integral method.
- I.e. use your first ring code, share the work equally between ranks and use some of the code from the calculate pi routines.

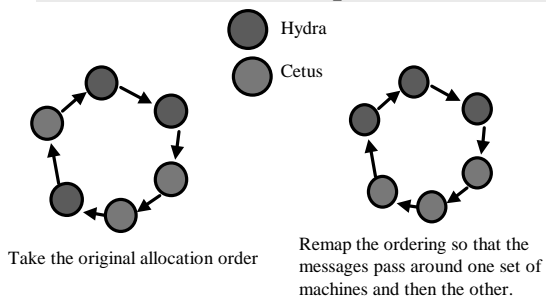
The First application



Part 1 Inter-operating two clusters

- Two runs of all the MPI application are required:
 - (1) that uses the nodes as they are allocated.
 - (2) a version that divides up the nodes so that the first part of the ring is on the cetus machines and the second part of ring is on the hydra machines.
 - This involves making a mapping from the given topology to a new one with the nodes in the right order.

Part 1 example



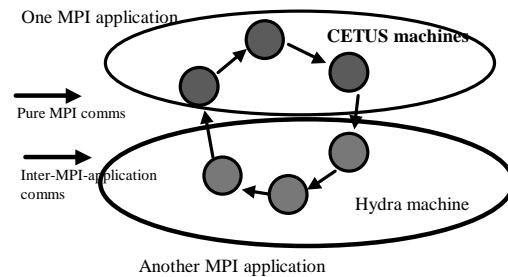
Part 2 Inter-operating two halves

- Using the same source code which is controlled at compile time by precompiler directives (`#ifdef`) alter the application to run as two separate MPI applications that interact to form a single ring and still calculate the same result.

Part 2 Inter-operating two halves

- Run two MPI applications
- They then find each other, exchange contact data.
- They then pass data from one application to the other using a TCP library I will provide.

Application Examples



Bonus points

- Bonus points:
 - how can you check for homogeneous systems and thus reduce any data conversion overheads?
 - Extra points if one is started with an mpirun and the other is a globus run.
 - Can you time around the ring multiple times to prove one is faster than the other?

Write up

- Write a side on any design issues you had and mention any problems encountered.
- Also:
 - Explain how you would handle heterogeneous data representations (and why one ring is easier than the other), different network media etc.
 - If it was not a ring, but a more complex graph, how would you optimize the job placement? Or would you just change the communication ordering...