MPI_Stream

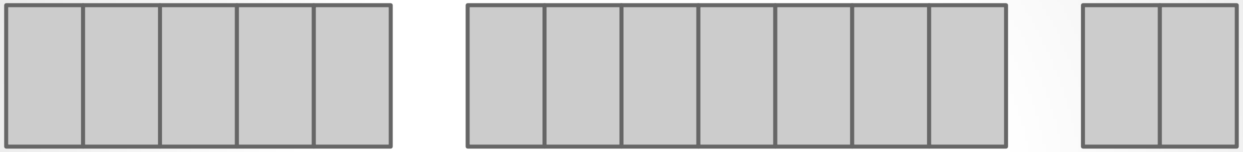
Canvassing feedback on initial ideas

What is an “MPI stream”?

★ From sender to receiver only
  ○ Joined by an existing communicator
★ Sender and receiver use intrinsic data-types as the smallest unit of transmission
  ○ Message boundaries are ignored
★ Sender can send any amount(s) of data
★ Receiver can receive any amount up to what is available
  ○ Buffer underrun - receiver asks for 33 integers but only 16 available so gets 16 integers rather than blocking, receiver must ask again to get the rest
★ Ordered and reliable

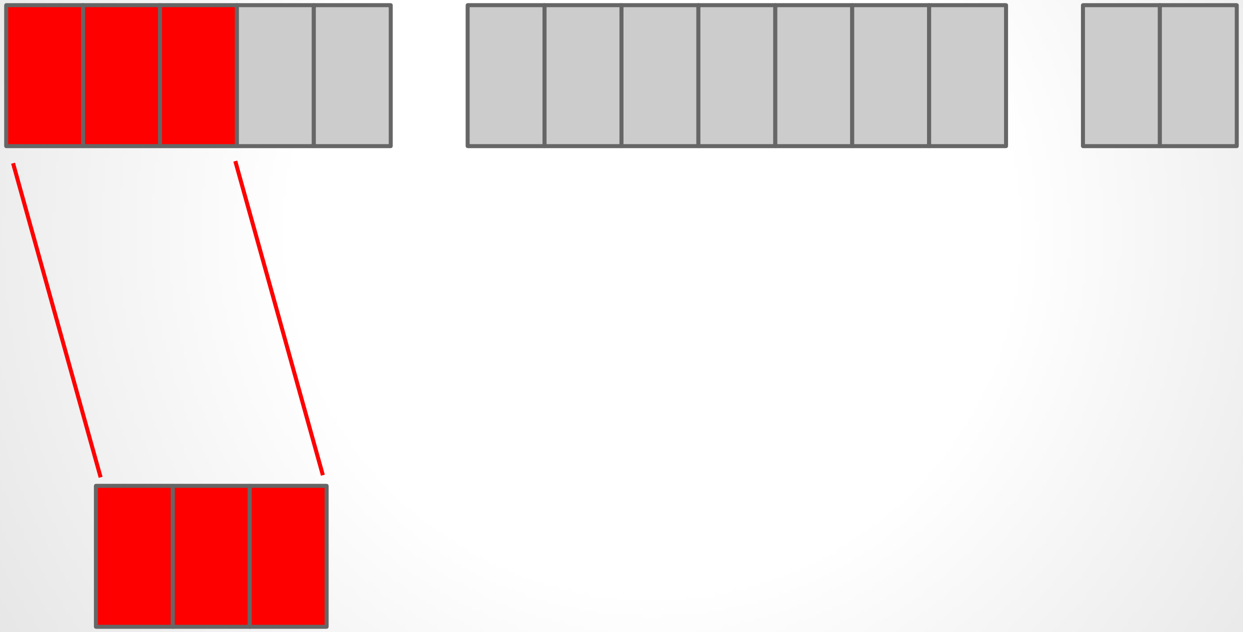
MPI stream example

Sender sends 5 items, then 7 items, then 2 items



MPI stream example

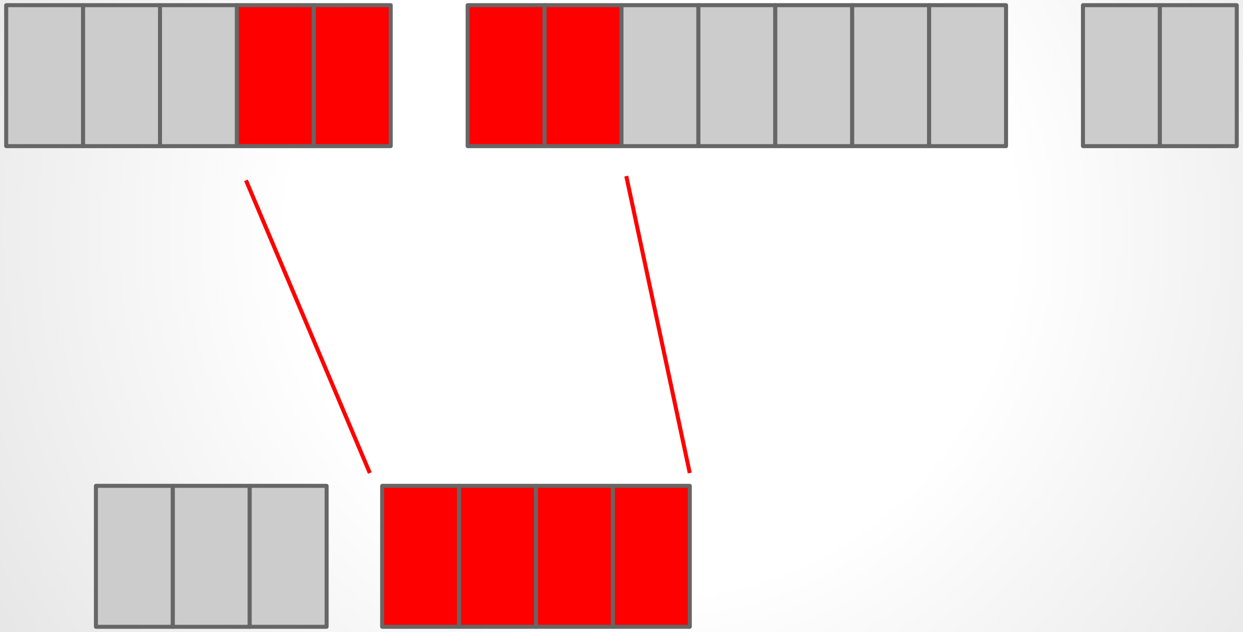
Sender sends: 5 items, then 7 items, then 2 items



Receiver receives first 3 items

MPI stream example

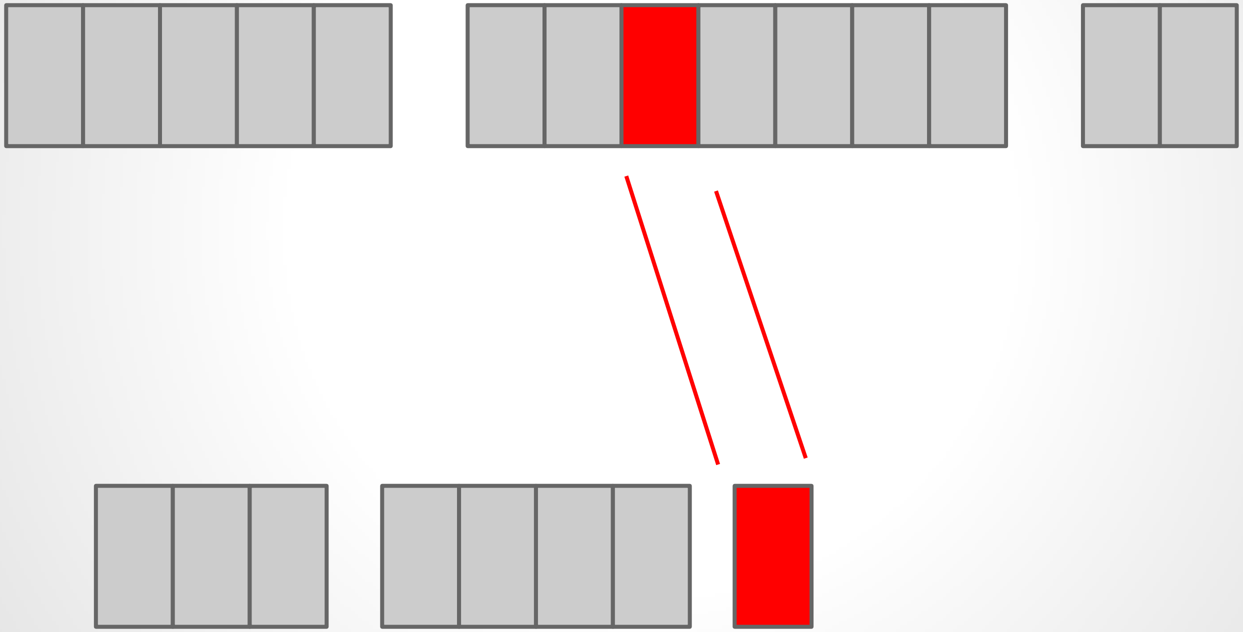
Sender sends: 5 items, then 7 items, then 2 items



Receiver receives next 4 items

MPI stream example

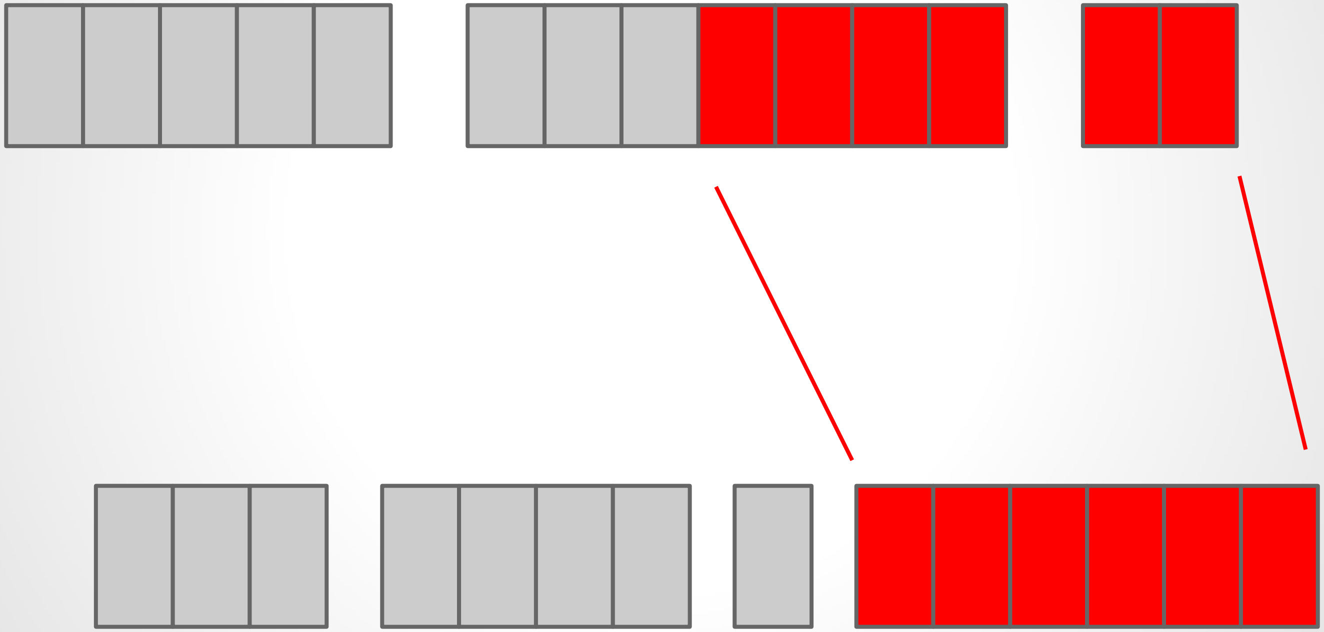
Sender sends: 5 items, then 7 items, then 2 items



Receiver receives next 1 item

MPI stream example

Sender sends: 5 items, then 7 items, then 2 items



Receiver tries to receive next 20 items, gets 6

Doubts

- Why is this better than TCP?
    - It drops a lot of the stuff from TCP that HPC does not want
    - It adds MPI datatypes
    - Can use high performance networks (without TOE)
- Performance argument
    - Hardware needs to be ready and able to support the proposed semantics
- Is this a solution in search of a problem?

Questions for app developers

- ★Would you use this?
- ★What would it buy you?