MPI Partitioned Device side options

Make init different

- But this would make it blocking if we integrate pbuf_prepare at this stage
  - Not ideal for programming ease

Add PStart

- Changes the nature of Pstart – make sure you've got the recv-side ready (not proposing ready type user guarantees)
  - Could output a new type of request that can be used on the device.
- Questions

  Can you mix Start and Pstart?

  What's the ramifications on the rest of the language in the standard?

  Start is non-blocking, this would make Pstart blocking

  Can we Wait on start to guarantee readiness? May not need to always wait…

  MPI_Pstart(req_in, req_ready_out) wait(req_ready_out)

Pbuf_Prepare

- Should we create a new request type here?
- Make it work with a pready_fast type call that takes a new request type?
  - Do we need a new request type?

Pexport_to_device

- Do we want to change Pbuf_prepare to a new call, meant to change the request type and get things ready for the device side?
  - Could have different device types with different requirements
    - One could require recv-ready, another might not
    - Change request behind the scenes to whatever you need on that device
    - Can device be local CPU where you are currently executing?
    - How do we get the right list of devices? Standardize or else implementations bear the burden
- Pready_device
  - Instead of Pready_fast, maybe we just create a specific call meant for devices
- Parrived_device
  - We probably want to support this…do we allow the semantics to change per device? Seems like trouble, but could be useful for power users….

Pexport_device Example

MPI_PSEND_INIT
MPI_START
MPI_Pexport_device (blocking/non-local)
MPI_PREADY_device…(nonblocking)
MPI_WAIT (completing)

MPI_START, MPI_Pexport_device
MPI_PREADY…MPI_PREADY
MPI_WAIT

MPI_PRECV_INIT
MPI_START
MPI_Pexport_device(blocking/non-local)
Optional – parrived_device (nonblocking)
MPI_WAIT (completing)

MPI_START, MPI_Pexport_device
MPI_PARRIVED…MPI_PARRIVED
MPI_WAIT

Pexport_device CPU Example

MPI_PSEND_INIT
MPI_START
MPI_Pexport_device (device=cpu, nonblocking, don't modify request)
MPI_PREADY_device…(nonblocking pready)
MPI_WAIT (completing)

MPI_START, MPI_Pexport_device
MPI_PREADY...MPI_PREADY
MPI_WAIT

MPI_PRECV_INIT
MPI_START
MPI_Pexport_device(device=cpu, nonblocking, Don't modify request)
Optional – parrived_device (nonblocking)
MPI_WAIT (completing)

MPI_START, MPI_Pexport_device
MPI_PARRIVED...MPI_PARRIVED
MPI_WAIT

Pexport_device GPU Example

MPI_PSEND_INIT
MPI_START
MPI_Pexport_device (device=gpu, block and modify request)
MPI_PREADY_device…(nonblocking pready)
MPI_WAIT (completing)

MPI_START, MPI_Pexport_device
MPI_PREADY...MPI_PREADY
MPI_WAIT

MPI_PRECV_INIT
MPI_START
MPI_Pexport_device(device=gpu, block, mod request)
Optional – parrived_device (nonblocking)
MPI_WAIT (completing)

MPI_START, MPI_Pexport_device
MPI_PARRIVED...MPI_PARRIVED
MPI_WAIT

Pexport_device ?PU Example

MPI_PSEND_INIT
MPI_START
MPI_Pexport_device (device=?pu, don't block and modify request)
MPI_PREADY_device…(nonblocking pready)
MPI_WAIT (completing)

MPI_START, MPI_Pexport_device
MPI_PREADY...MPI_PREADY
MPI_WAIT

MPI_PRECV_INIT
MPI_START
MPI_Pexport_device(device=?pu, don't block, mod request)
Optional – parrived_device (nonblocking)
MPI_WAIT (completing)

MPI_START, MPI_Pexport_device
MPI_PARRIVED...MPI_PARRIVED
MPI_WAIT