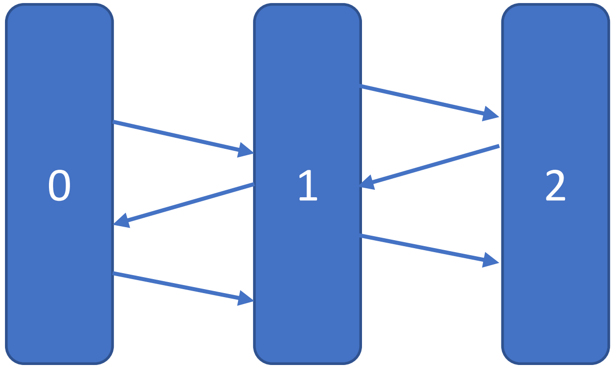
Introduction to MPI and MPI4py

Valentin Dallerit

10/6/2020

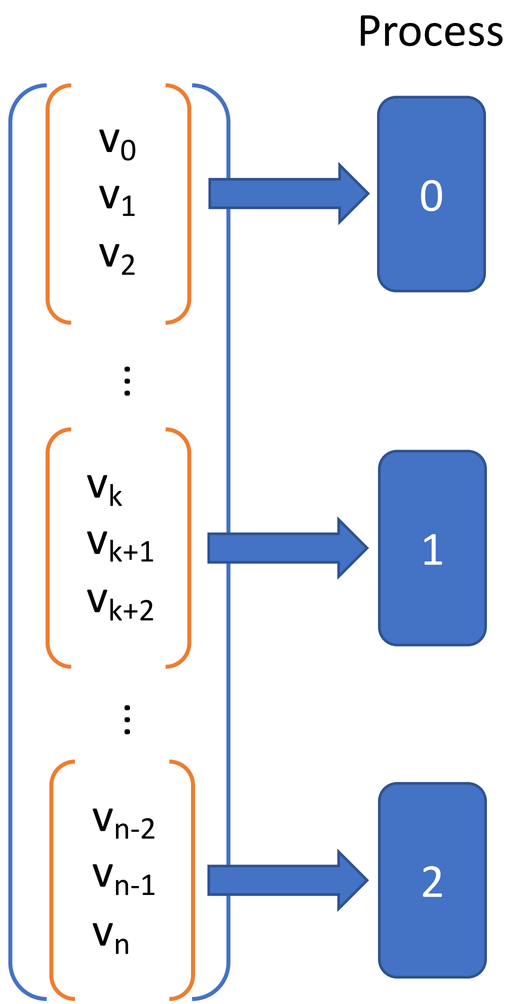
What is MPI?

- MPI = Message Passing Interface
- Single Program Multiple Data
  - Each process has its own memory
- MPI is a specification
  - Different implementations available
- Interface for C/C++, Fortran, Python, Julia ...



Example in Scientific computing

- Spreading vector over processes:
  - Each process stores one part of the vector
  - Communication with other processes when data stored by others is needed



How to run an MPI program?

$ mpiexec –n 16 <MPI program>

- Options:
  - `-n`: Number of processes
  - `--tag-output`: Add tag to each output to identify processes
  - `--output-filename`: Write output with 1 file/process

- In your slurm job, remember to load MPI

Communicators and Ranks

- Communicator = collection of processes that can communicate
  - MPI_COMM_WORLD: Default communicator containing all processes

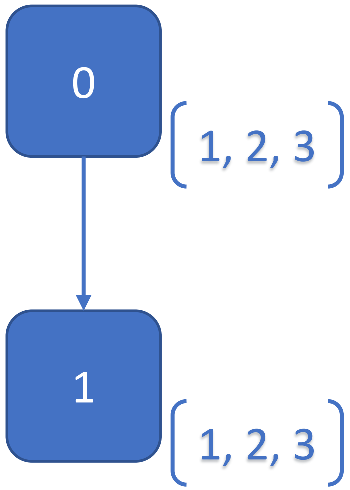
- Rank (also task id) = unique identifier for a process in a communicator

```
comm = MPI.COMM_WORLD
rank = comm.Get_rank()
size = comm.Get_Size()
```

Point-to-Point Communication

- Send data from one process to another

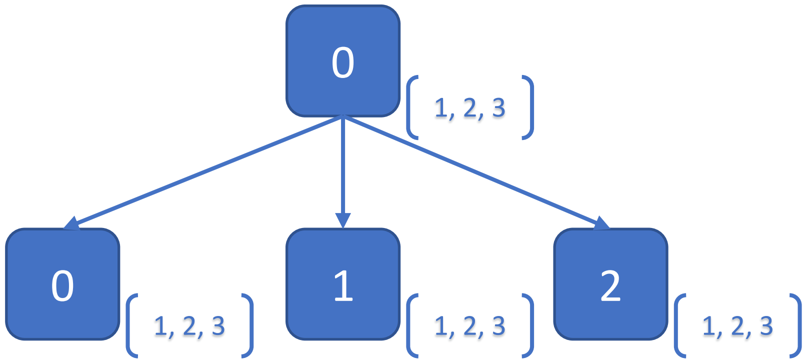
```
if rank == 0:
      data = [1,2,3]
      comm.send(data, dest=1)
elif rank == 1:
      data = comm.recv(source=0)
```



Broadcasting and Scattering

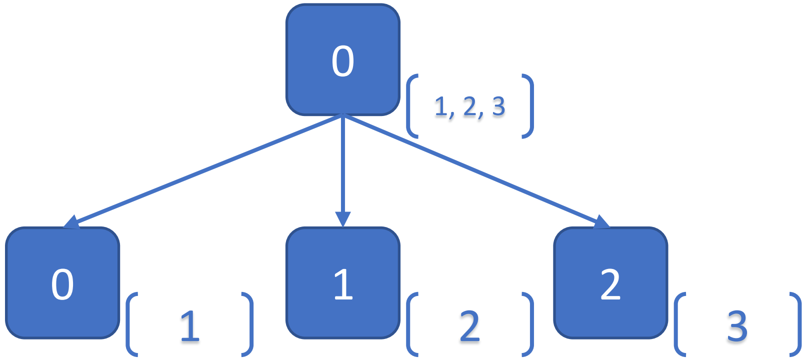
- Broadcast

```
comm.Bcast(data, root=0)
```



- Scatter

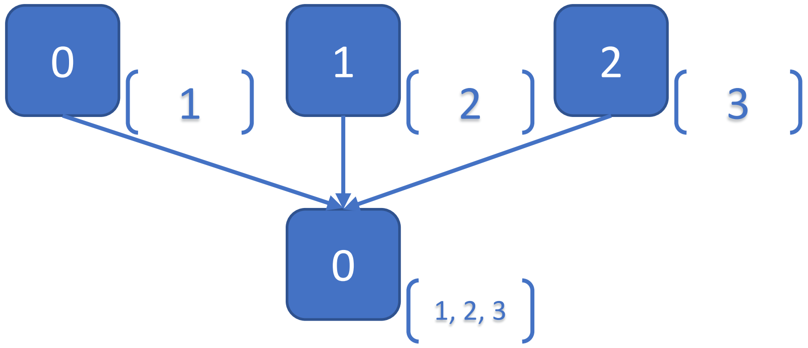
```
comm.Scatter(data, recvbuf, root=0)
```



Gathering and Reduce

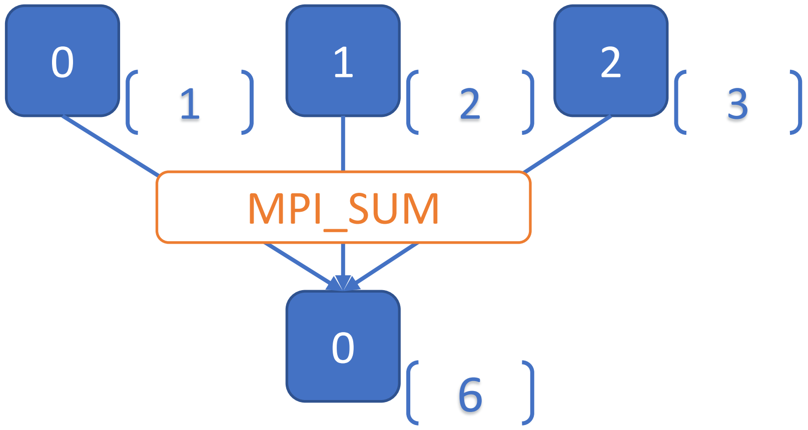
- Gather

  ```
  comm.Gather(sendbuf, recvbuf, root=0)
  ```

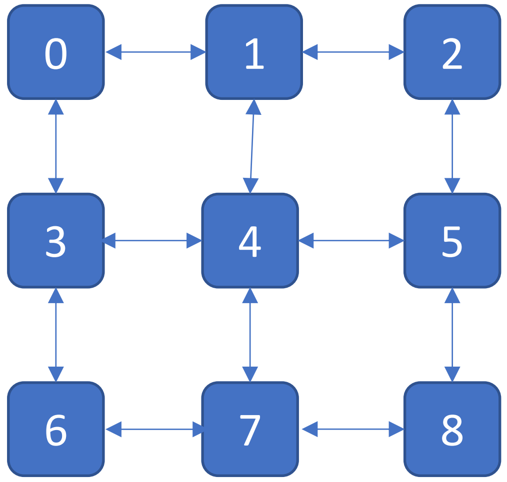


- Reduce:

  ```
  comm.Reduce(v, v_sum, op=MPI.SUM, root=0)
  ```



Virtual Topology



- Cartesian grid:

```
comm.Create_cart(dims, periods)
```

- More general structure:

```
topo = comm.Create_dist_graph_adjacent(srcs, dests)
```

- Communicate to all neighbors:

```
recv = comm.neighbor_alltoall(send)
```

Example: 1D Heat equation

- We want to solve the PDE:
  - $\frac{\partial u}{\partial t} = \frac{\partial^2 u}{\partial x^2}$ $\quad\quad$ $\mathrm{t} \in [0, 0.1]$ $\quad\quad$ $x \in [0, 2\pi]$
  - $u(0, x) = \sin(x) + \sin(3x)$
  - $u(t, 0) = u(t, 2\pi) = 0$

- Discretization:
  - Central finite difference in space: $\frac{\partial^2 u}{\partial x^2} \approx \frac{u_{n-1} - 2u_n + u_{n+1}}{\Delta x^2}$
  - Explicit Euler in time: $u^{n+1} = u^n + \Delta t\, f(u^n)$