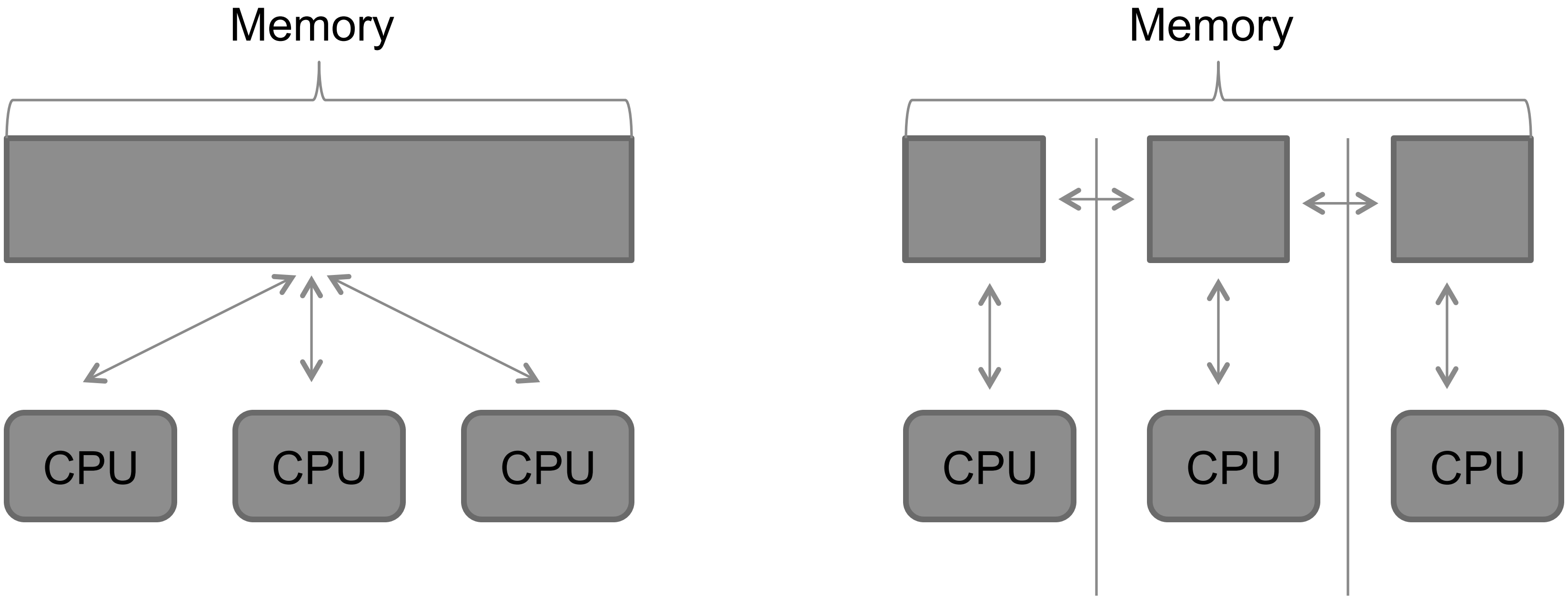
DISTRIBUTED SHARED MEMORY

JASON HOCH
MIKE ROSENSWEIG
SASHKO STUBAILO

GOALS

- **Multiple processes share an address space**
  - Do not share any physical memory



IMPLEMENTATION

- **Memory begins read only**
- **If write is attempted: Page fault**
  - Invalidate other processors' memory
  - Resume execution
- **If read is attempted on invalid memory: Page Fault**
  - Get correct value
  - Mark own memory valid
  - Resume execution

DISTRIBUTION OF MANAGEMENT

- **Each user has to initialize distributed shared memory at the beginning, is assigned ID**
- **Page management split across users**
- **Lazily obtains updates from machines writing to page of memory, handles multiple simultaneous reads**
  - Uses Invalid, Reading, Writing states similar to cache statuses covered in lecture
- **Must be synchronized while receiving updates from other users**
  - Lock per page
  - Queue of conditional waits used to wait for responses