

# Adventures in Supercomputing with R

## Lecture 2: Hardware & Software Overview

George Ostrouchov

Oak Ridge National Laboratory and University of Tennessee

2022/01/19 (updated: 2022-02-23)

# Summary of Last Lecure

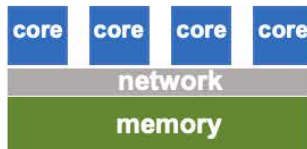
- Two parallel computing communities
  - Parallel numerical math (generating data) - supercomputing
  - Parallel databases (storing and serving data) - data centers
- Statistical computing left batch environment to be interactive
  - Developed S and then R
  - High-level, extensible, and interactive
  - Use numerical libraries
- Supercomputing takes R back to batch
  - High-level, extensible, but batch
  - Use scalable numerical libraries
- Workflow, accounts, tools
  - RStudio, git, GitHub, ssh, unix

# Parallel Hardware

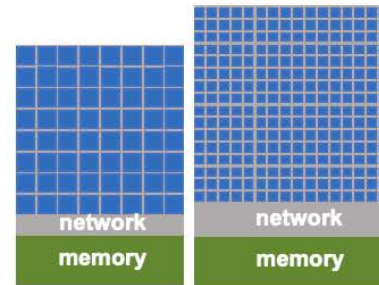
**A high level look at what matters ...**

# Three Basic Concepts in Hardware

Shared Memory  
Multicore Processor

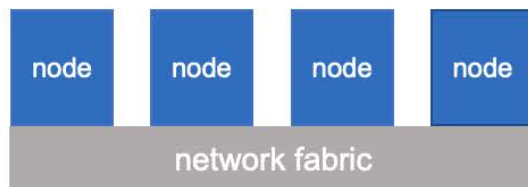


Shared Memory  
Co-Processor



Manycore

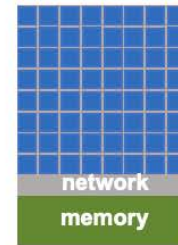
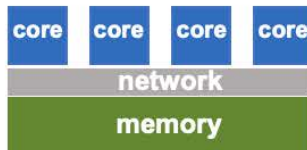
GPU



Distributed Memory Cluster

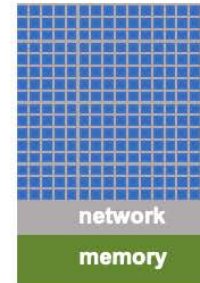
# Three Basic Concepts in Hardware

Shared Memory  
Multicore Processor

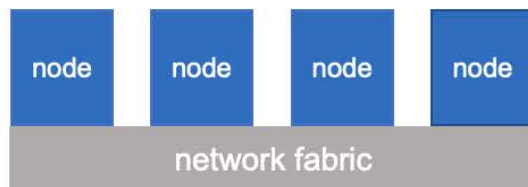


Manycore

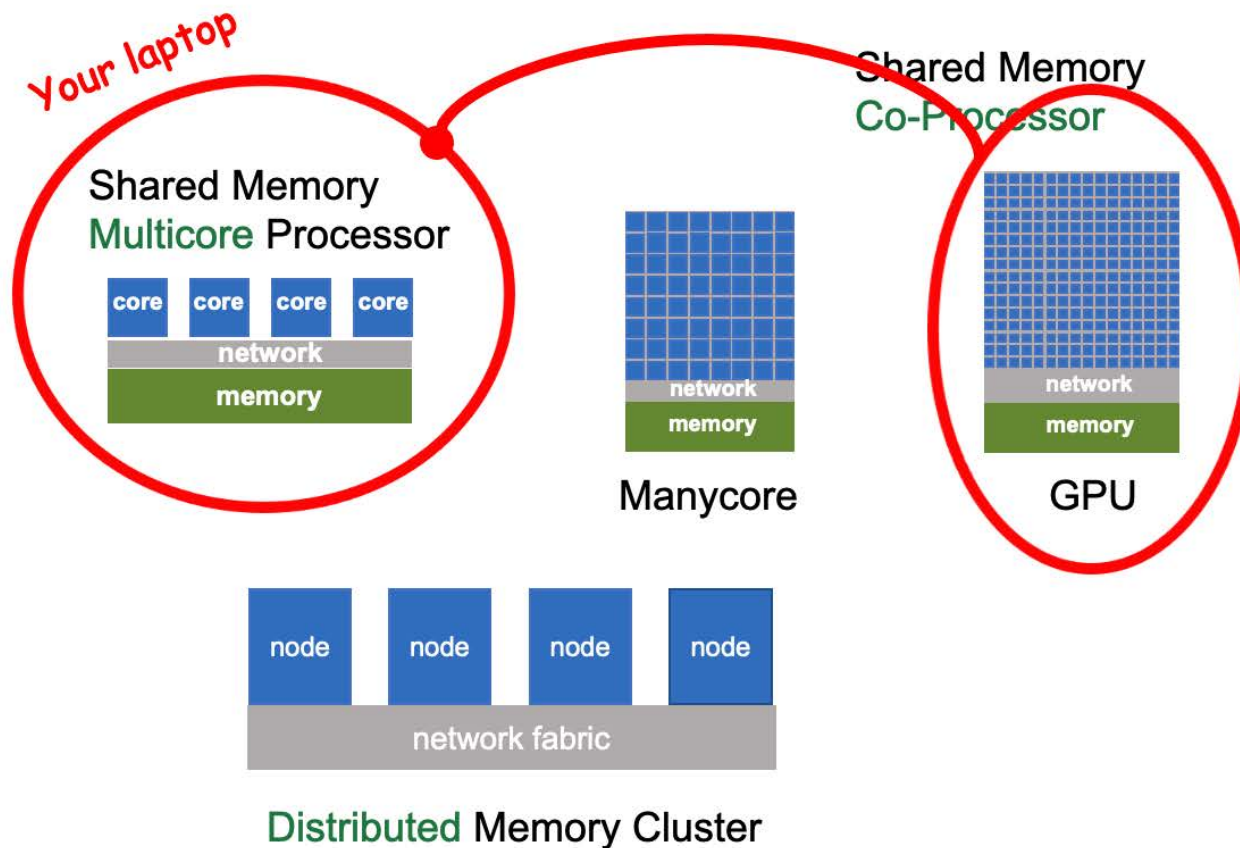
Shared Memory  
Co-Processor

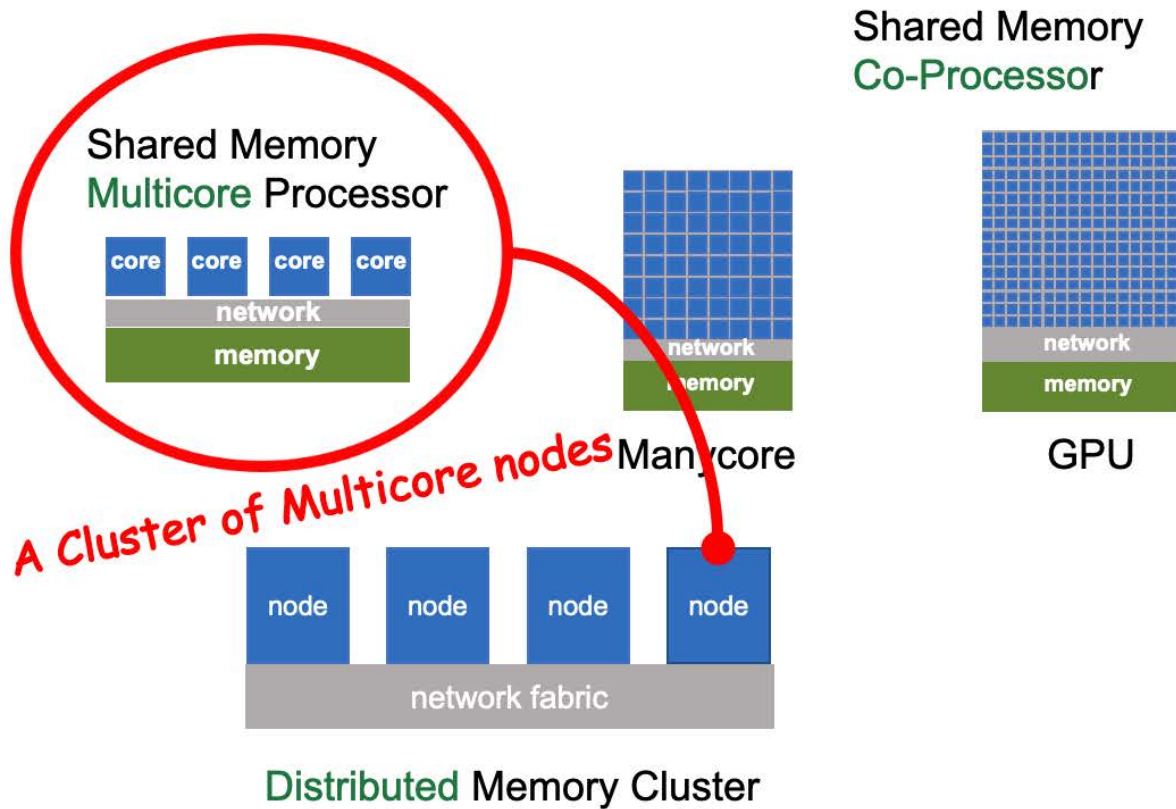


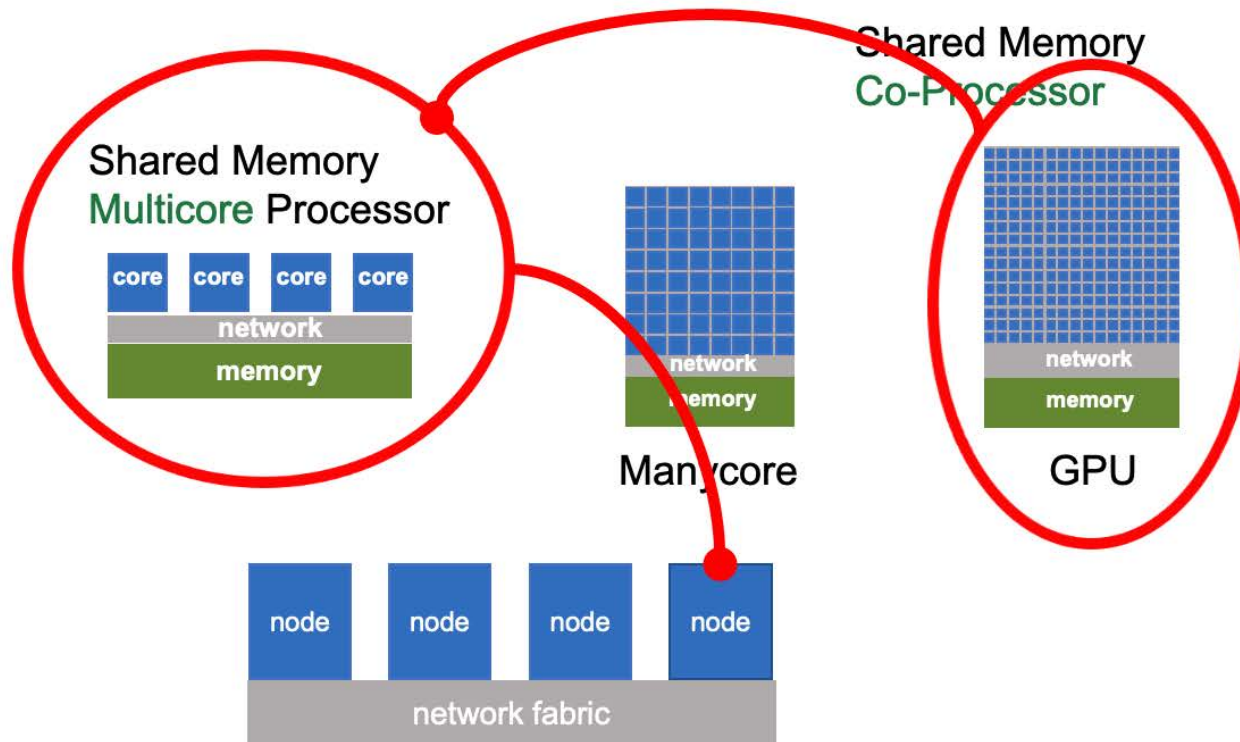
GPU



Distributed Memory Cluster

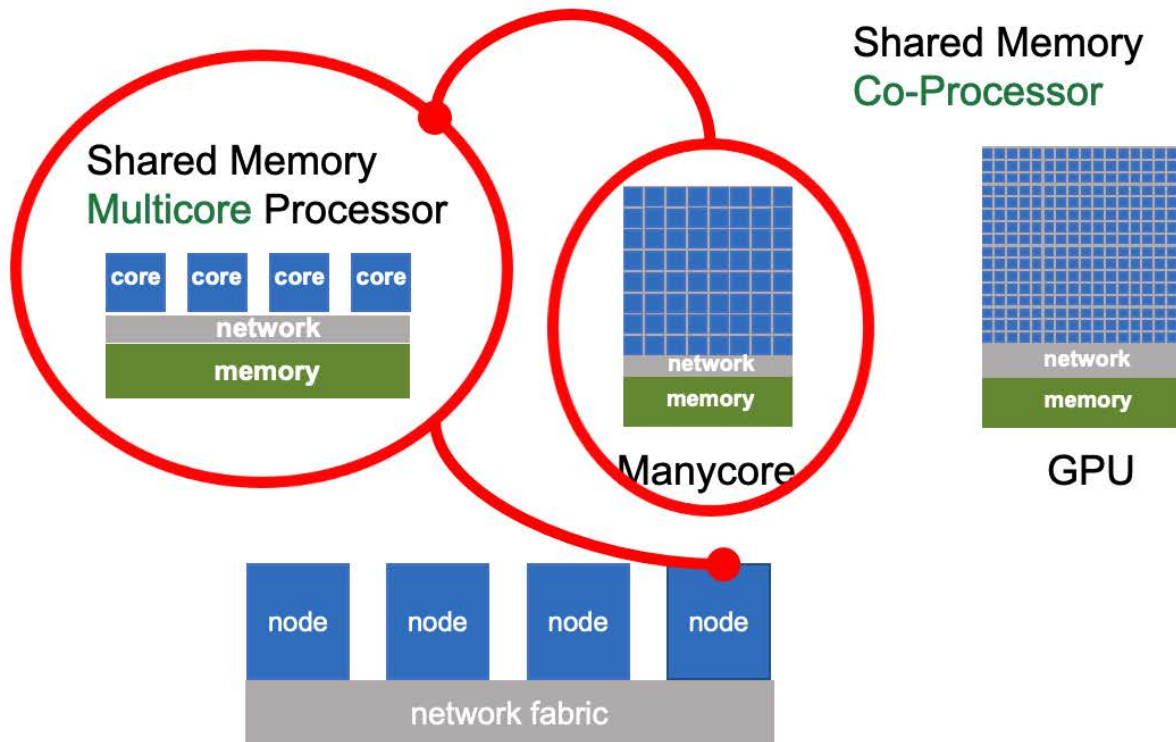






Distributed Memory Cluster  
*A Cluster of Multicore nodes with GPU co-Processors*

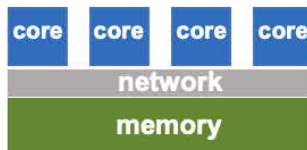




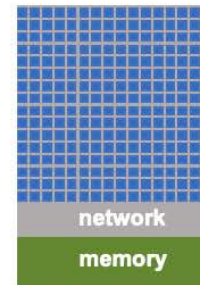
Distributed Memory Cluster

*A Cluster of Multicore nodes with Manycore co-Processors*

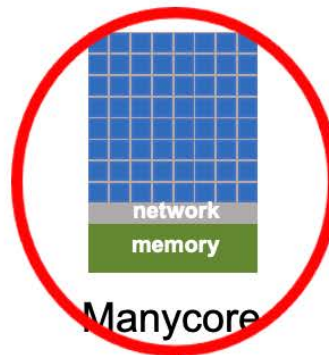
Shared Memory  
Multicore Processor



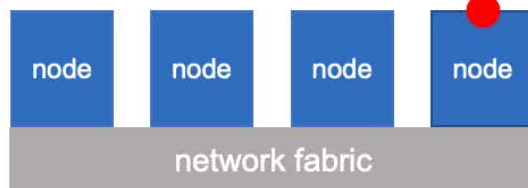
Shared Memory  
Co-Processor



GPU



Manycore



Distributed Memory Cluster

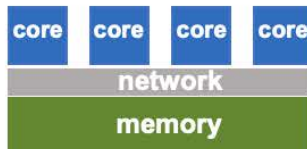
*A Cluster of Manycore Nodes*

*Cray XC40  
2nd generation Manycore  
Intel Xeon Phi "KNL"  
Fugaku ARM*

# Native Programming Mindset

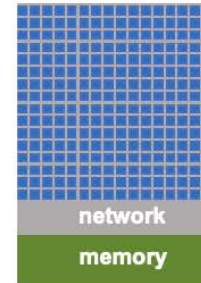
Default is serial: which tasks should be made parallel?

Shared Memory  
Multicore Processor

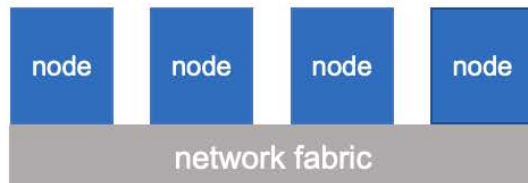


Offload data and tasks: We are slow but many!

Shared Memory  
Co-Processor



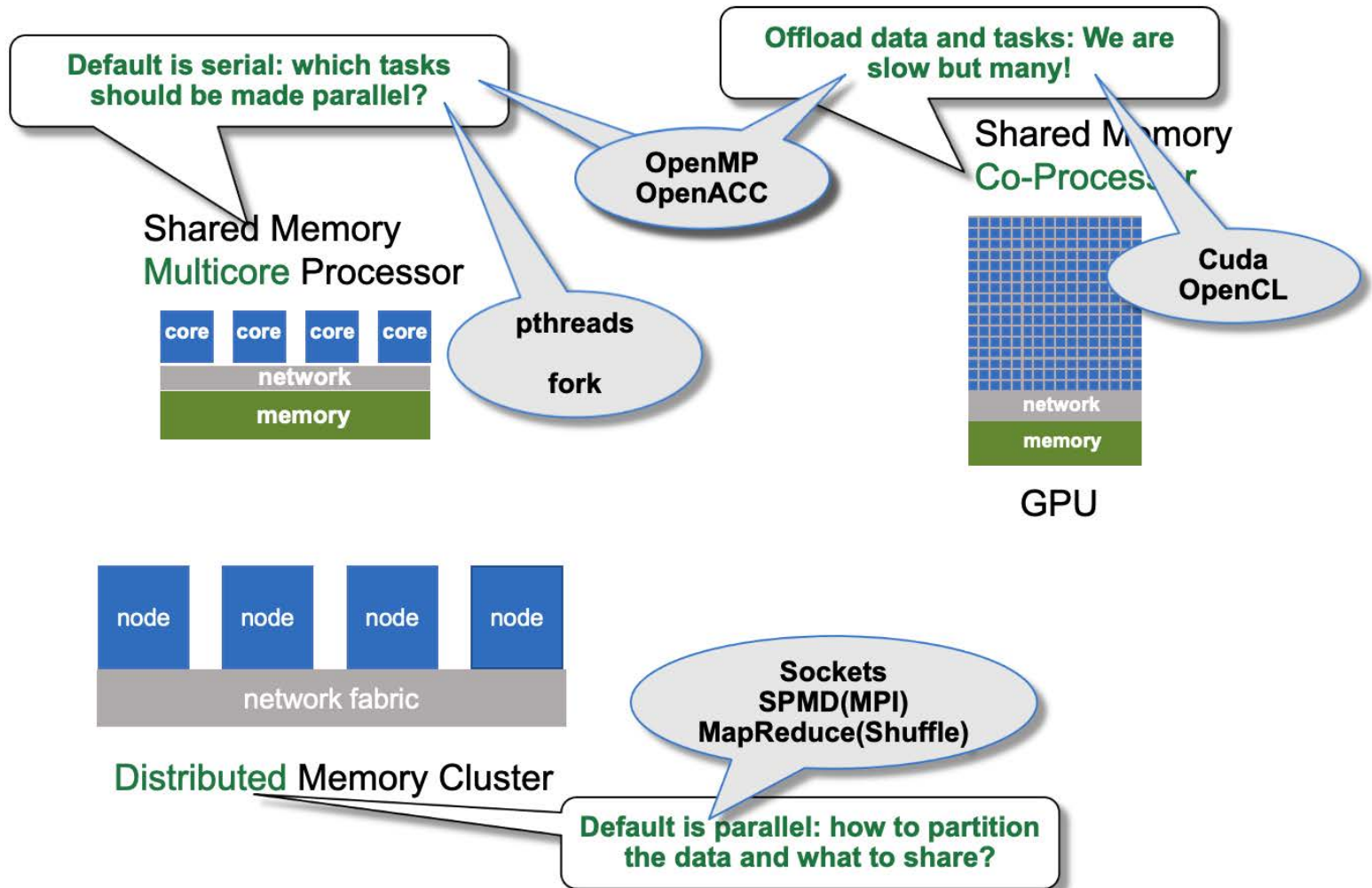
GPU



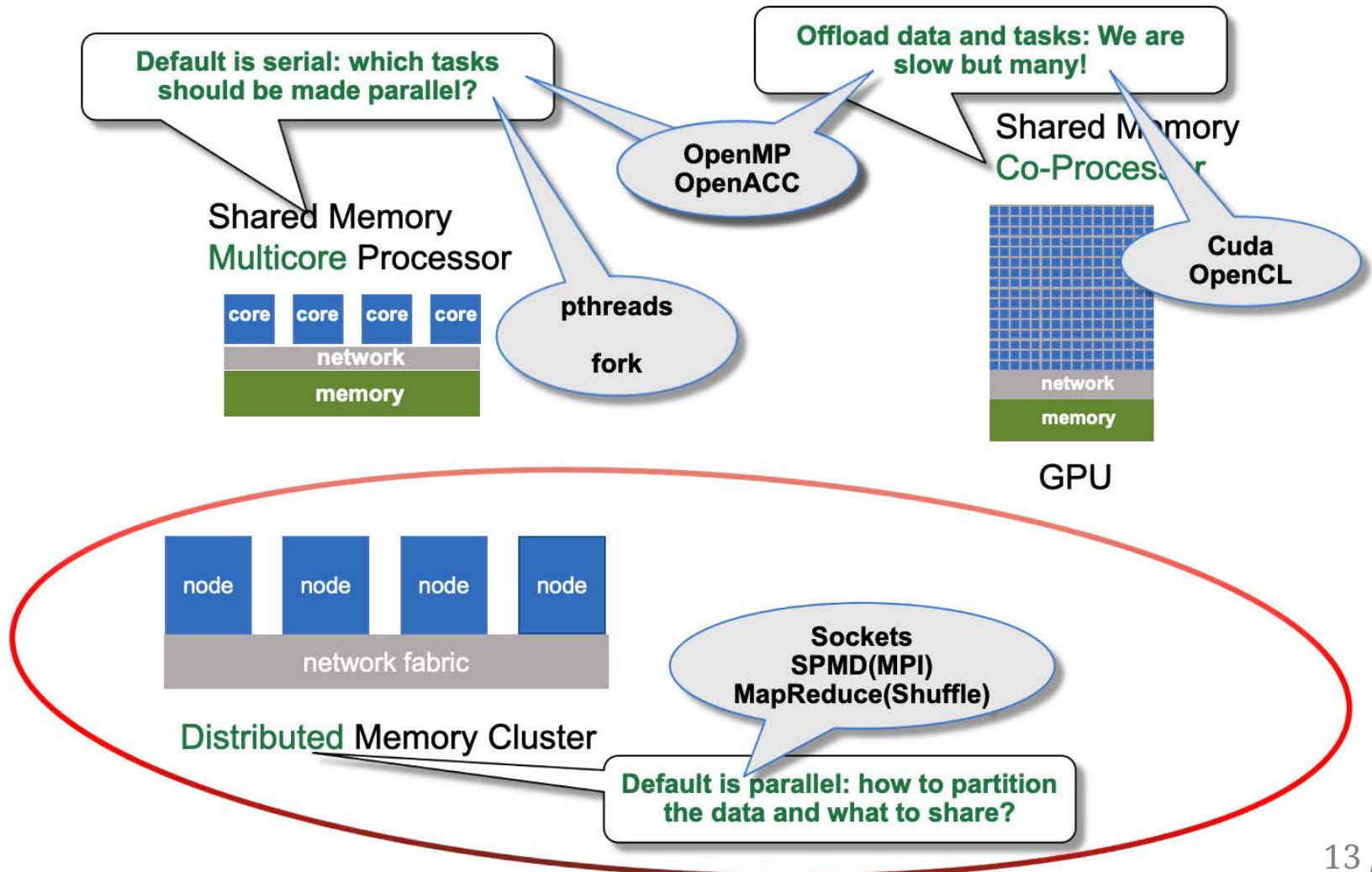
Distributed Memory Cluster

Default is parallel: how to partition the data and what to share?

# Native Programming Models and Tools

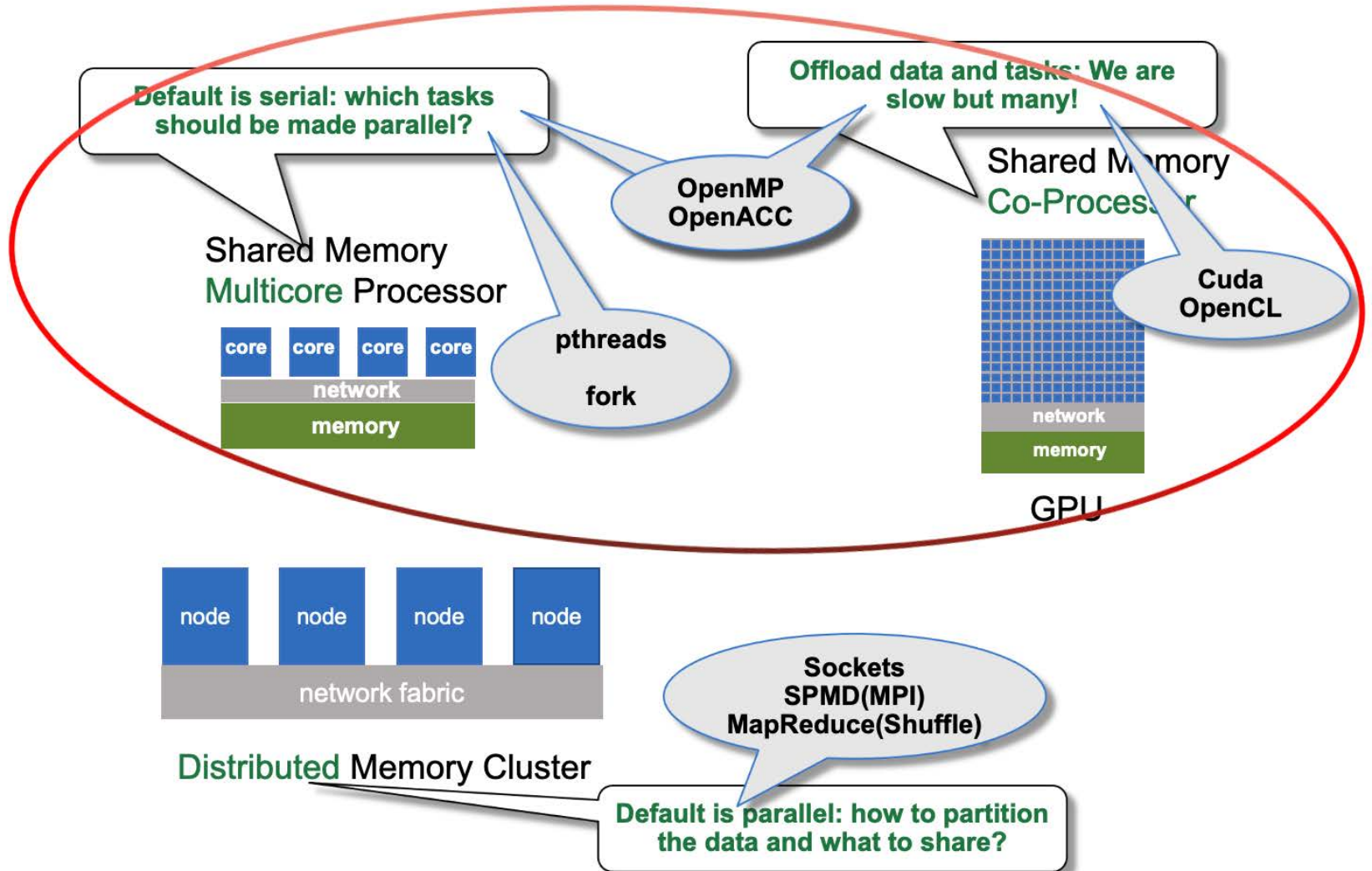


# 35+ Years of Parallel Computing Research

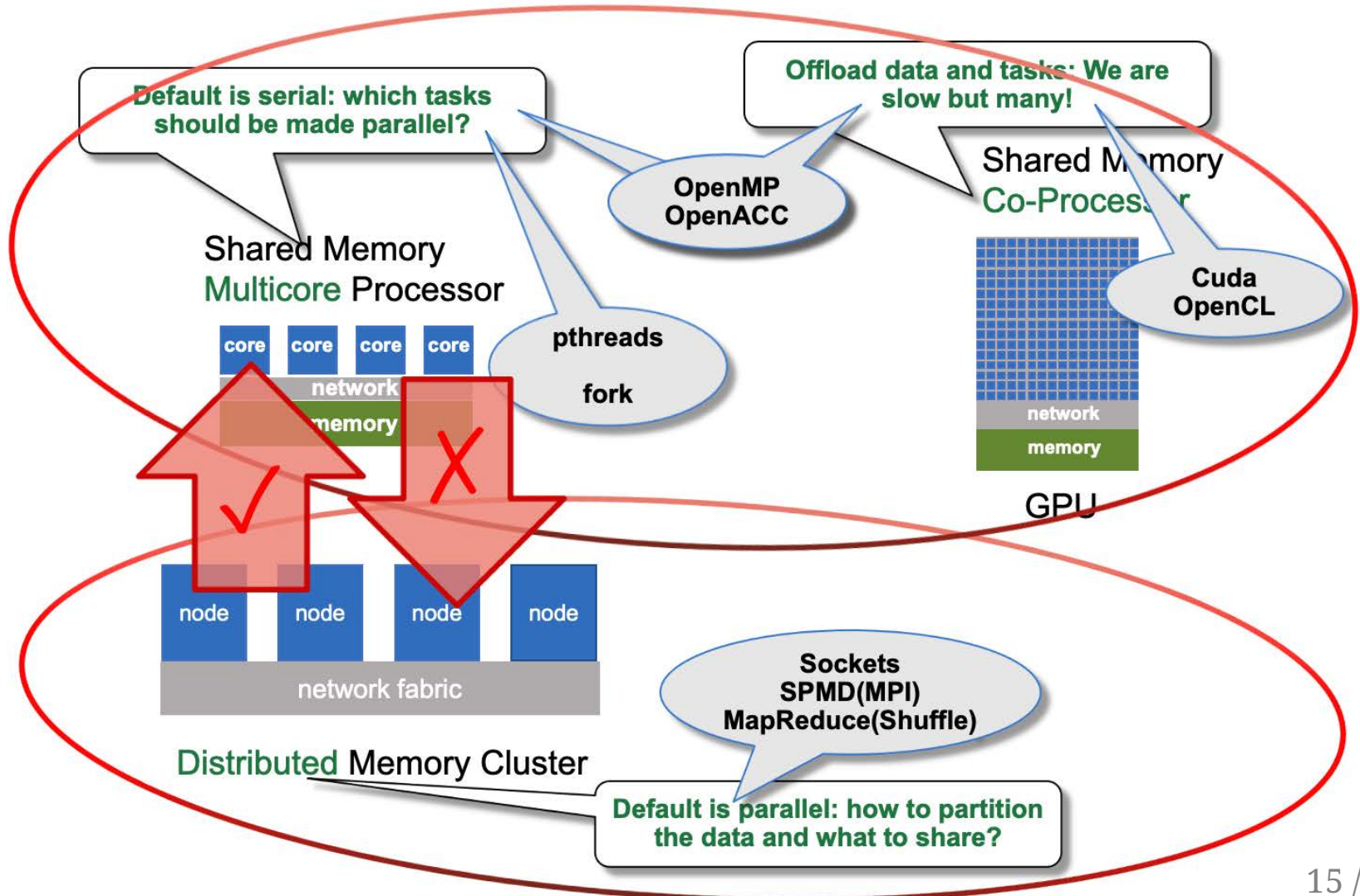




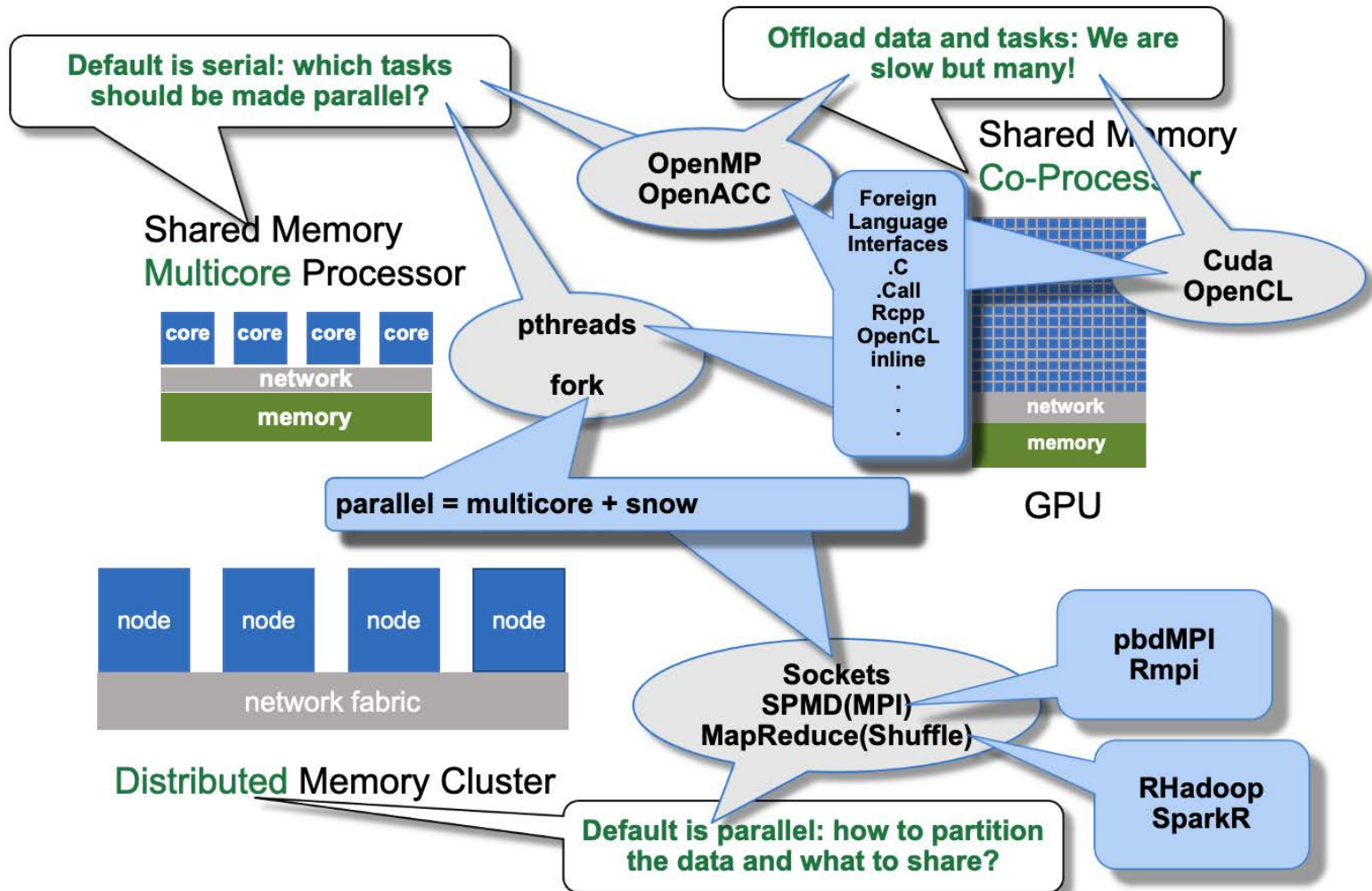
# Last 15+ years of Advances



# Distributed Programming Works in Shared Memory

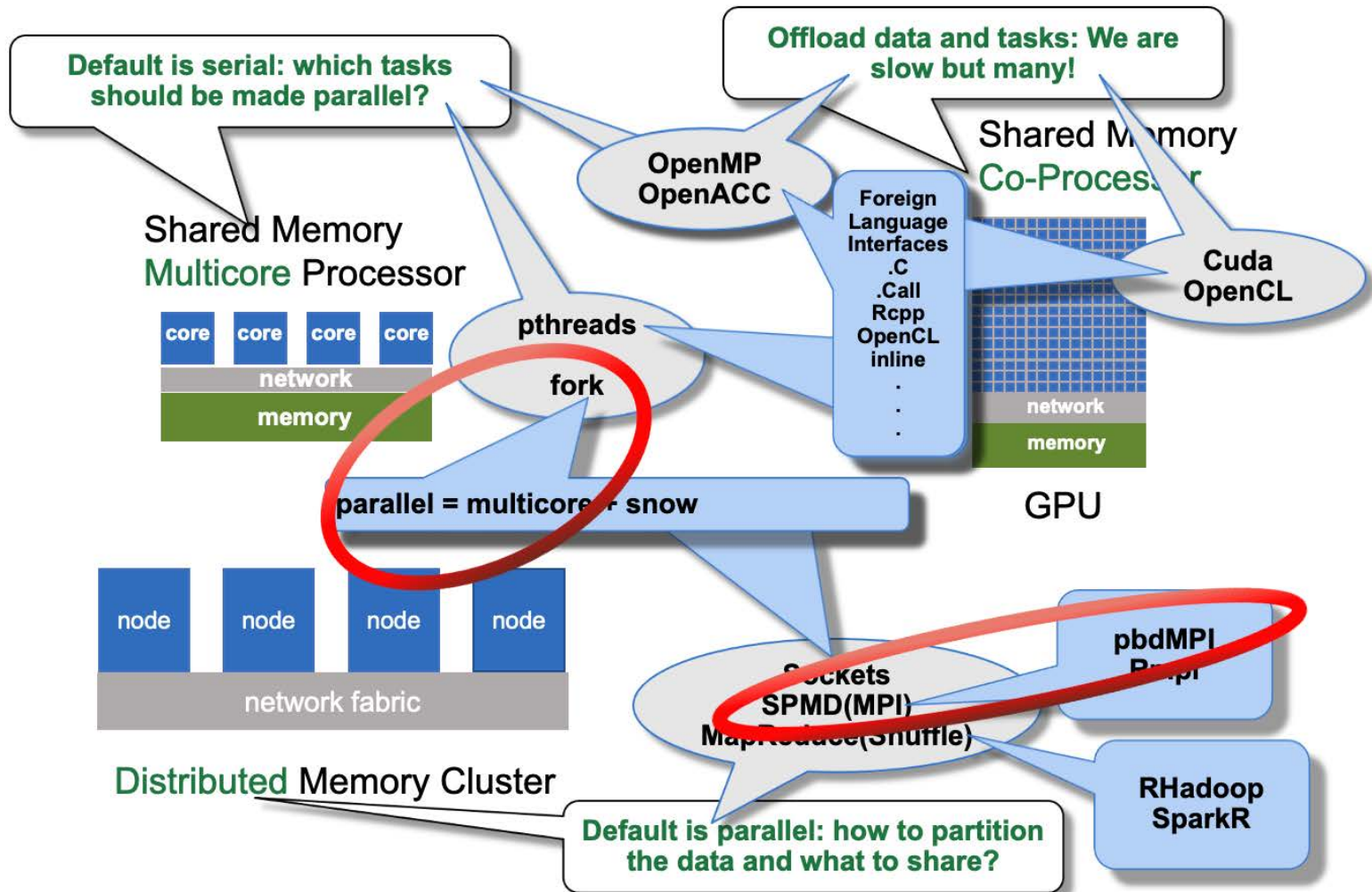


# R Interfaces to Low-Level Native Tools

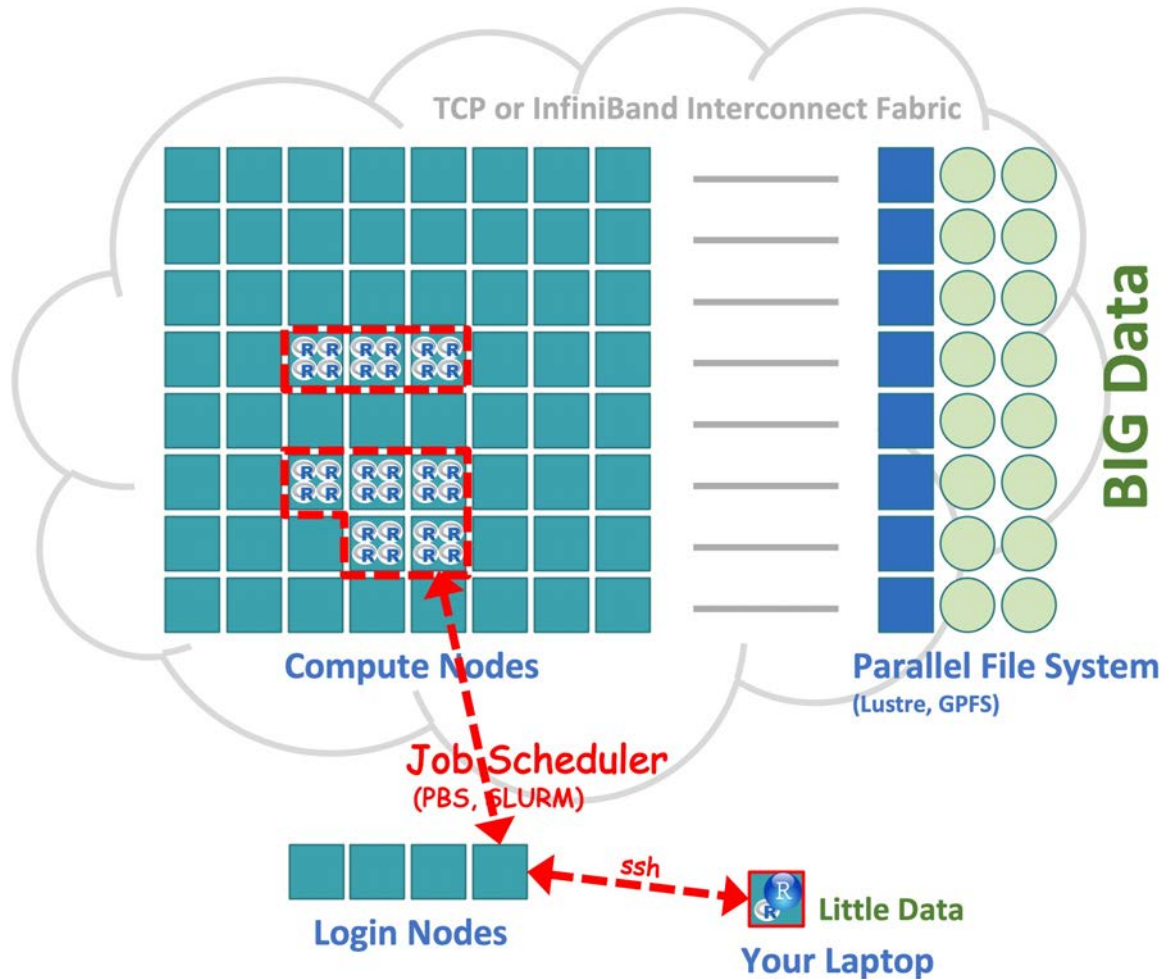




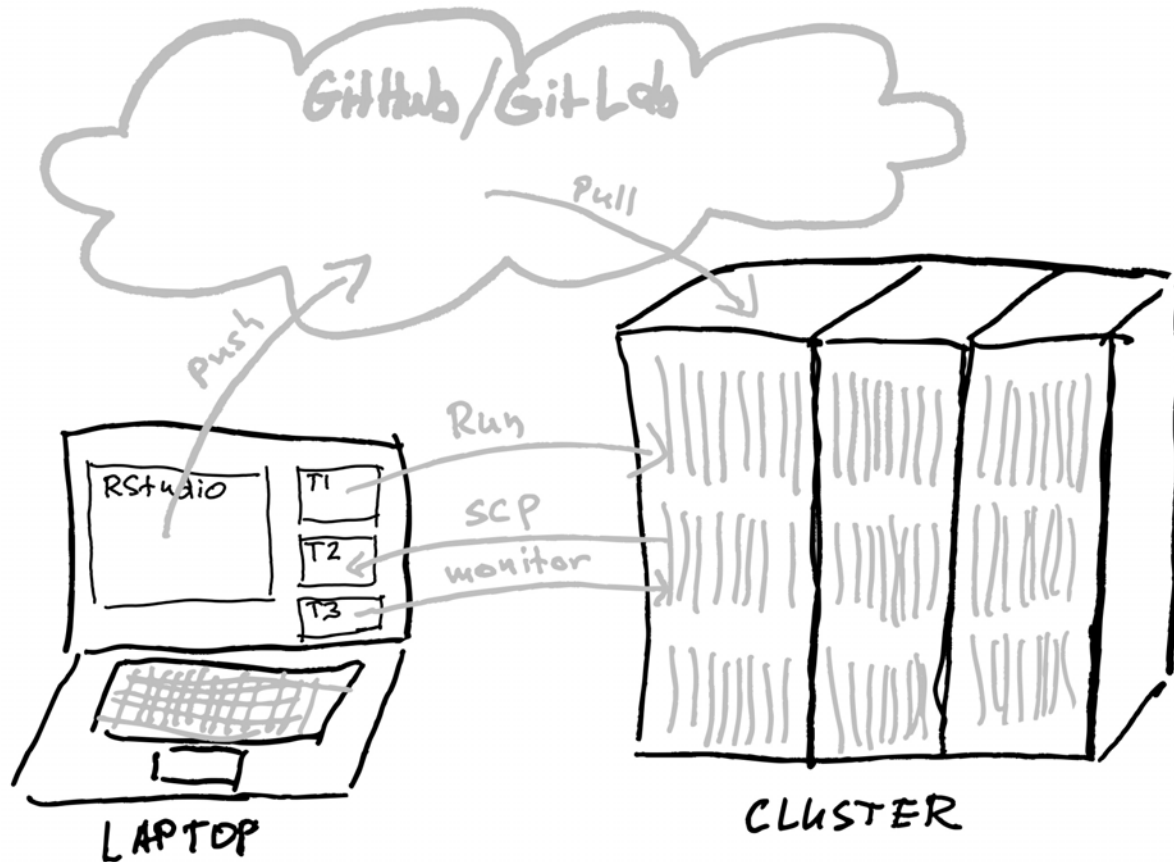
# R Interfaces to Low-Level Native Tools



# Running Distributed on a Cluster



# Typical Workflow from Laptop to Cluster



# Unix Concepts and Starting your Workflow

...

# Some useful Unix concepts

- Linux is one of many descendants of original Unix. Mac OS X is another.
- Like all file systems, Unix files are organized as a tree.
- When in a terminal, you are talking to a *shell* program (*bash* is most common)
- Commands are looked up in directories listed in your PATH variable
- Spaces delimit commands and options
- `>` and `>>` redirect standard output to a file
- `command1 | command2` pipes standard output1 to standard input2
- `$` means substitute variable value
- There are many resources on the web to learn Linux basics.

# Some useful Linux commands

- **pwd** Show current directory
- **ls** List files in current directory
  - **ls -a** Include files that start with .
  - **ls -l** Long listing with *permissions*, *owners*, and *last change time*
- **cd** *dir\_name* Change directory to *dir\_name*
  - **cd** without *dir\_name* or `~` as *dir\_name* changes to home directory
- **mkdir** *dir\_name* Creates directory *dir\_name*
- **rmdir** *dir\_name* Deletes directory (must be empty)
- **rm** *file\_name* Deletes *file\_name*
- **cat** *file\_name* Displays content of entire *file\_name*
- **less** *file\_name* Displays content of *file\_name* with paging
- **man** *command* Displays the manual page for *command* with paging
- **which** *command* Returns location of *command*
- **exit** Quit shell and logout

# Demo follows

Slides created via the R package **xaringan** and converted to pdf with **xaringanBuilder**