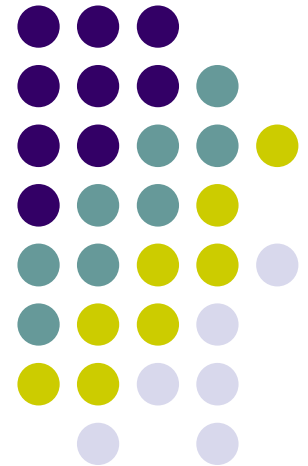


2018
Practical Parallel Computing
(実践的並列コンピューティング)
No. 2

Parallel Programming Models

Toshio Endo
School of Computing & GSIC
endo@is.titech.ac.jp



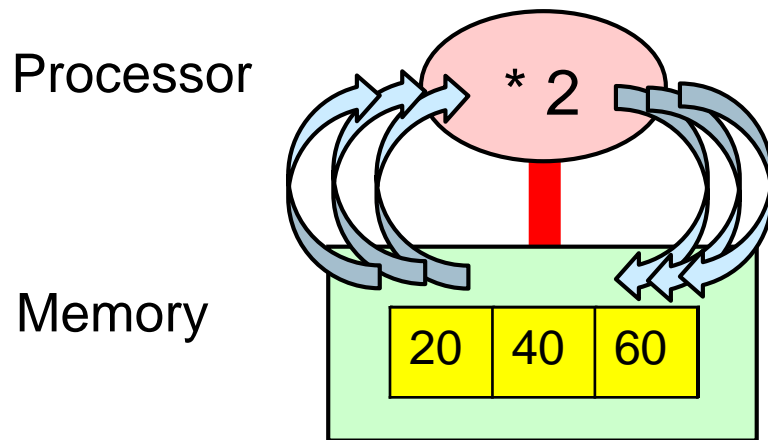
Computation on Computer Architecture



- Computation (Software) = Algorithm + Data
- Architecture (Hardware) = Processor + Memory

Note: This is so simplified discussion

Computer Architecture



Computation Example

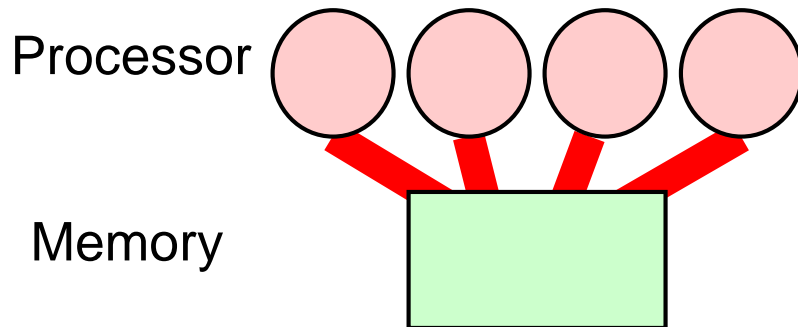
```
int a[3] = {10, 20, 30};  
int i;  
  
for (i = 0; i < 3; i++) {  
    a[i] = a[i] * 2;  
}
```



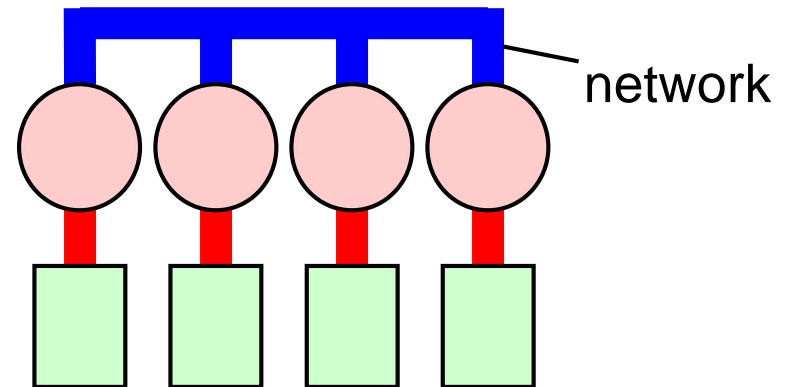
What is Parallel Architecture?

- Parallel architecture has MULTIPLE components
- Two basic types:

Shared memory
parallel architecture



Distributed memory
parallel architecture



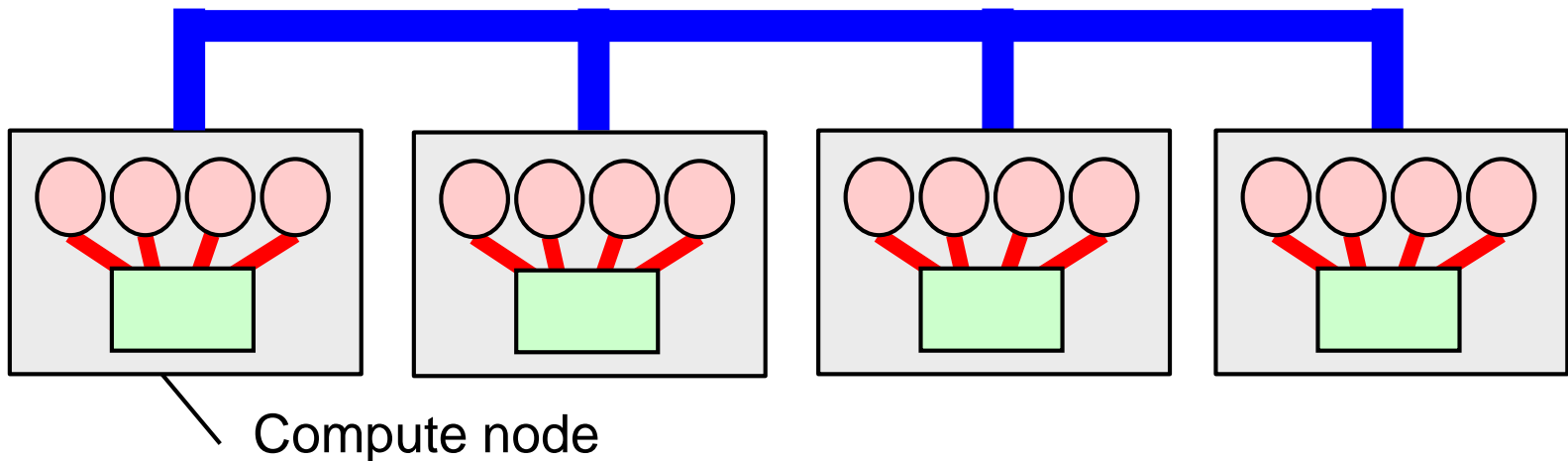
- Different programming methods are used for different architecture

Modern SCs use Both!



Modern SCs are combination of “shared” and “distributed”
“shared memory” in a node

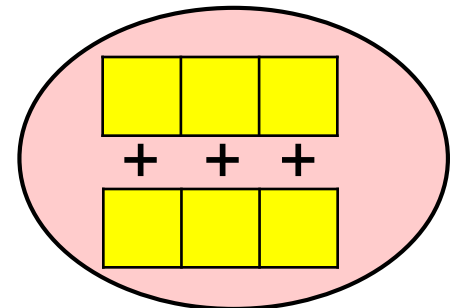
“distributed memory” among nodes, connected by network



✂ Moreover, each processor (core) may have *SIMD parallelism*, such as SSE, AVX...

A processor (core) can do several computations at once

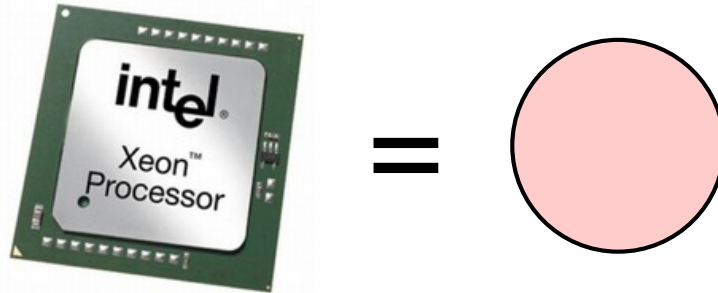
SIMD is out of scope of this class



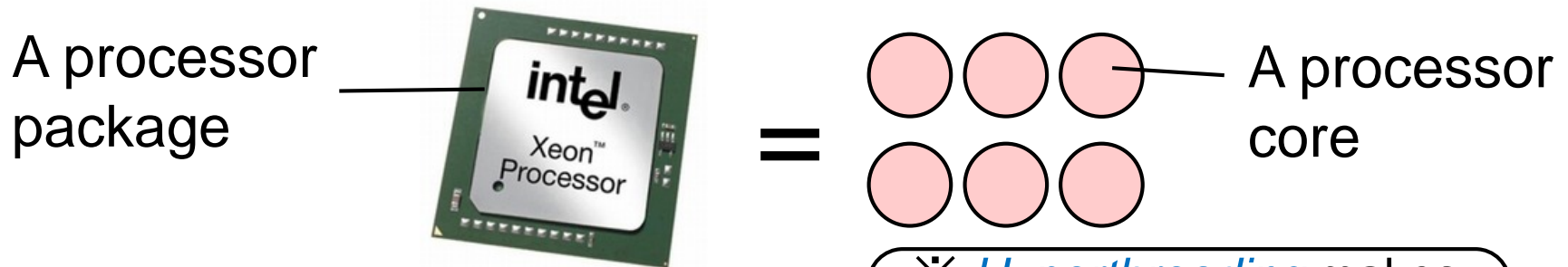


(Confusing) Terminology

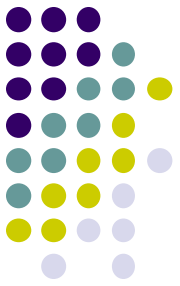
- In old days, definition of “processor” was simple



- Around 2005 or later, “multicore processor” became popular

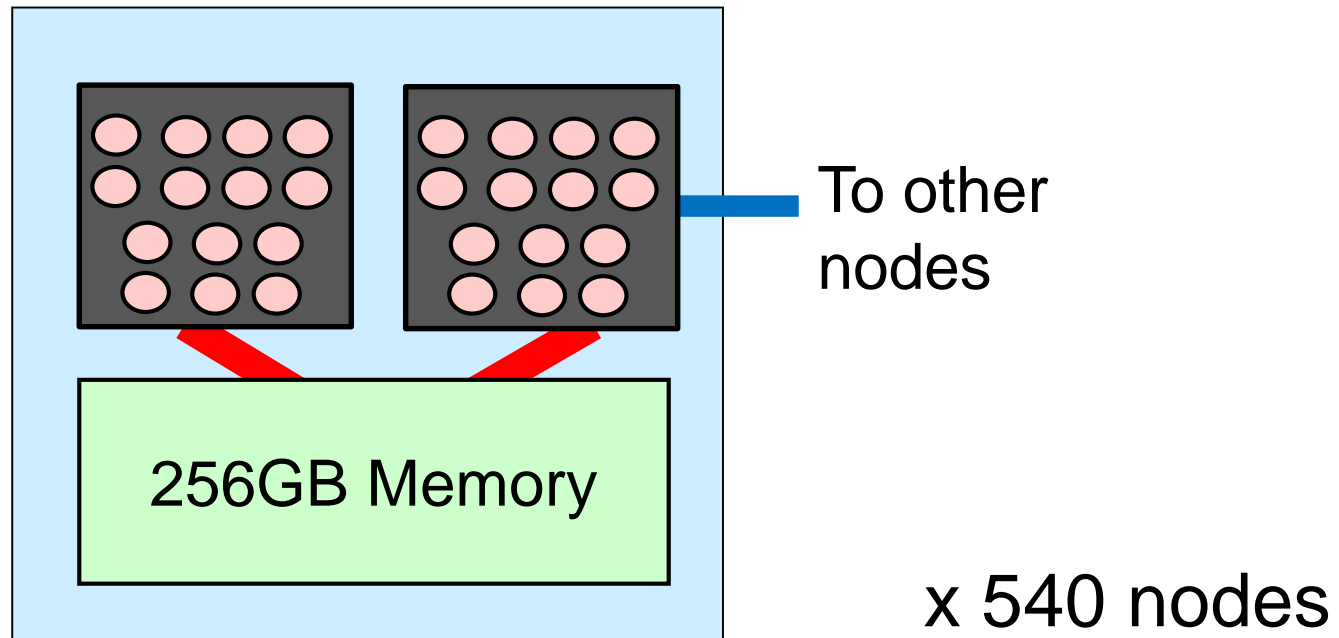


✂ *Hyperthreading* makes discussion more complex, but skipped



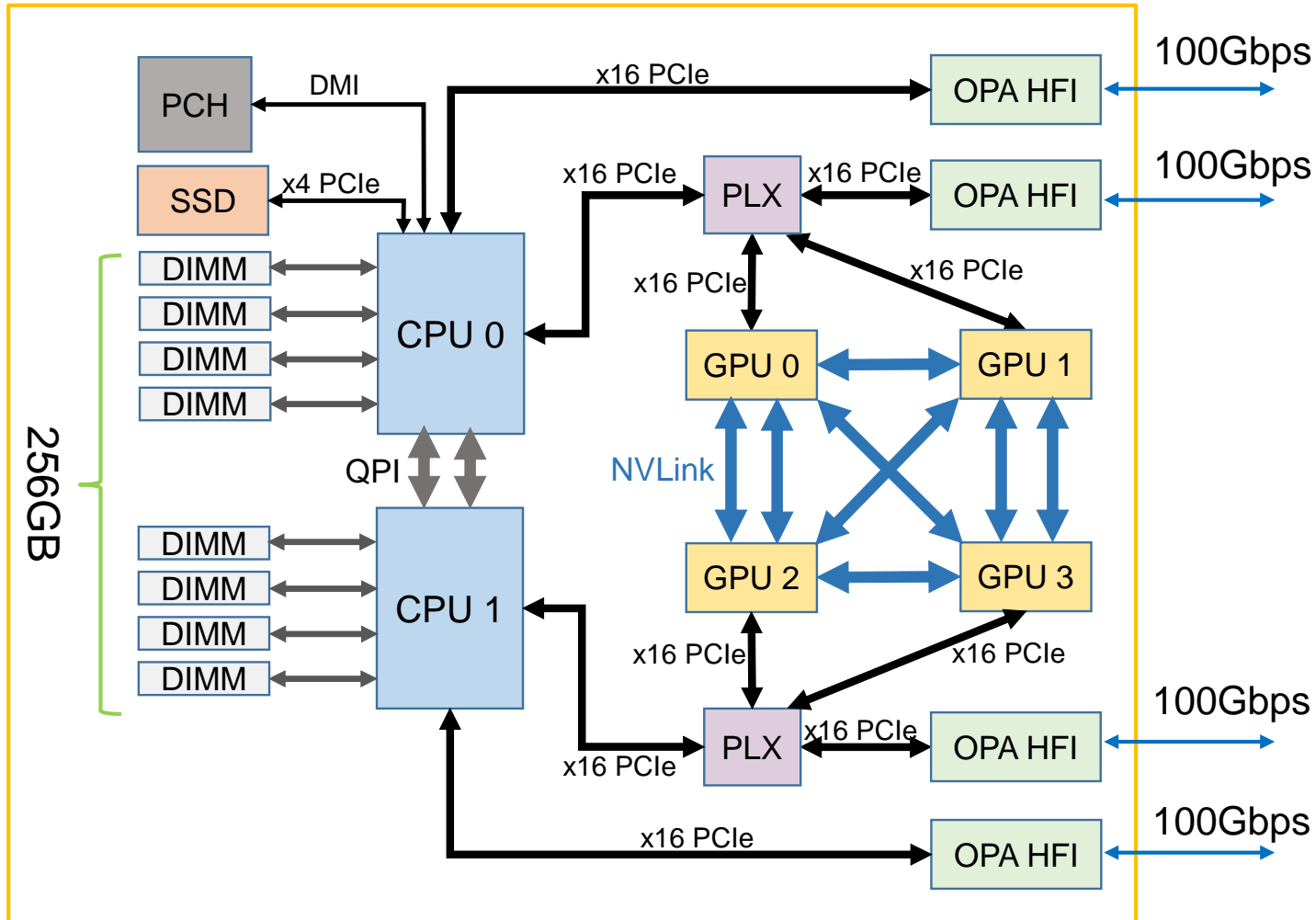
A TSUBAME3 Node

- 2 processor packages × 14 cores
 - 28 cores share memory

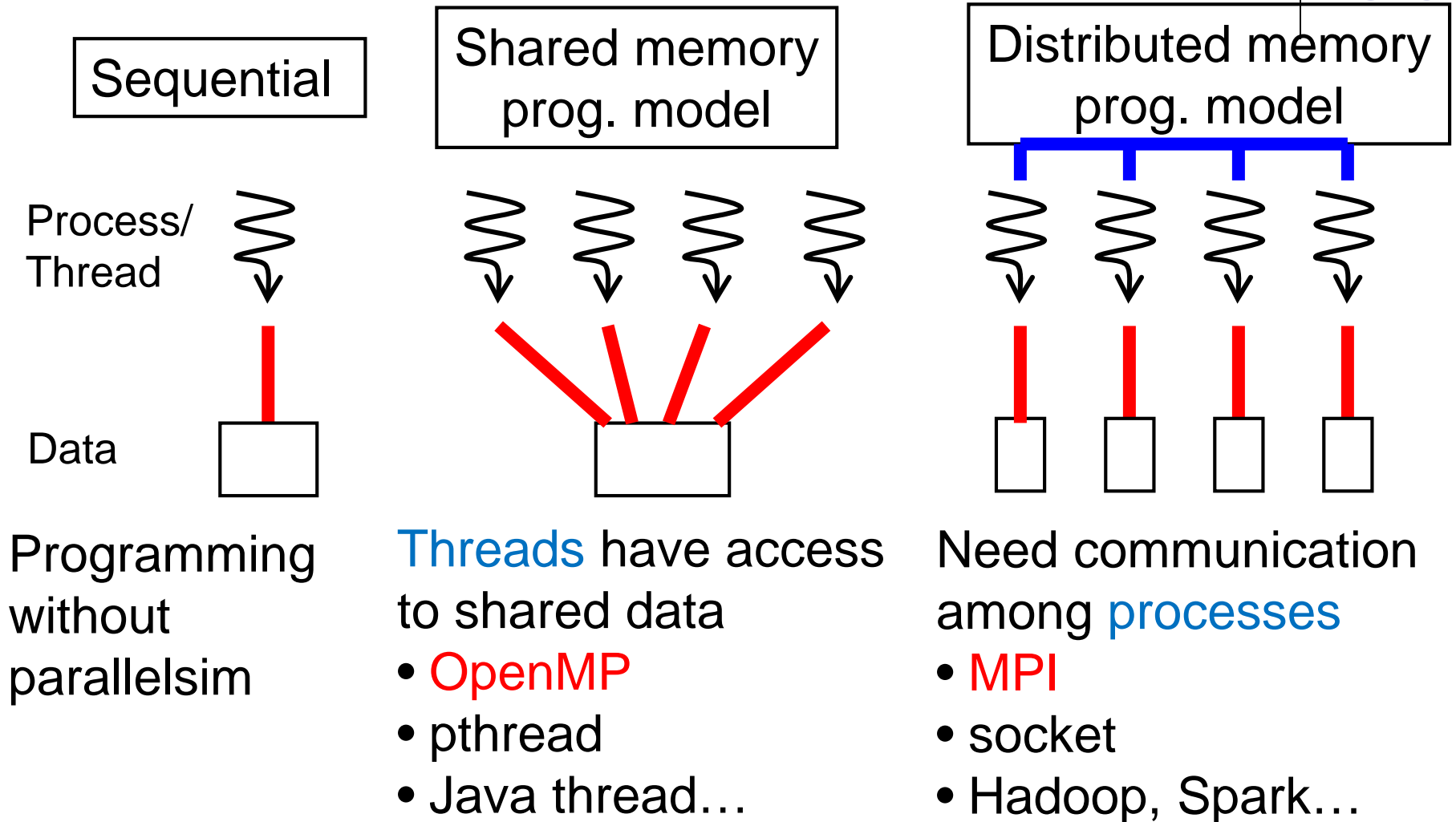
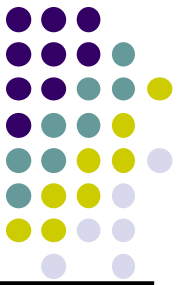




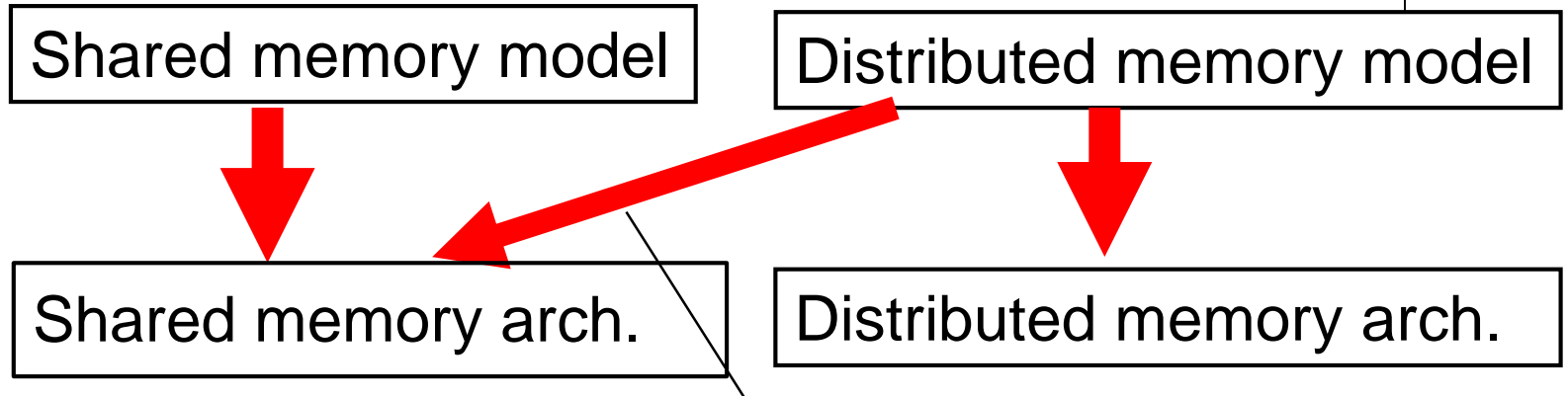
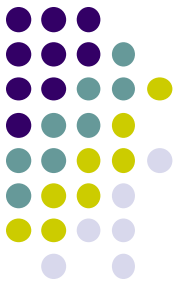
TSUBAME3 Node in More Detail



Classification of Parallel Programming Models



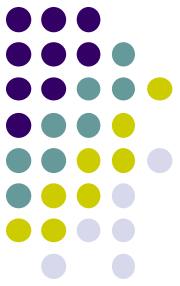
Programming Models on Architecture



It's OK to make multiple
processes on a node

When considering architecture,

- Shared memory model can use only cores in a single node (up to 28 cores on TSUBAME3)
- Distributed memory model supports large scale parallelism (~15,000 cores on TSUBAME3)



In This Course

Part1: Shared Memory Parallel Programming with OpenMP

Part2: Distributed Memory Parallel Programming with MPI

Part3: GPU Programming

- Using OpenACC is planned this year, instead of CUDA
- Uses knowledge both of shared/distributed

TSUBAME Account



- Students of this course will become members of “tga-ppcomp” TSUBAME group
 - Use qsub/qshr command with “-g tga-ppcomp”

Please do the following by Apr 19 (earlier is better)

- Please make your account on TSUBAME3
 - Tokyo Tech Portal (portal.titech.ac.jp) → TSUBAME Portal
- Please send an e-mail to ppcomp@el.gsic.titech.ac.jp

Subject: TSUBAME3 ppcomp account

To: ppcomp@el.gsic.titech.ac.jp

Lab name

Department name

School year

Name

Your TSUBAME account name

Next Class: Introduction to OpenMP



- Shared memory parallel programming API
- Extensions to C/C++, Fortran
- Includes directives & library functions
 - Directives: `#pragma omp ~`

```
int i;  
#pragma omp parallel for  
for (i = 0; i < 100; i++) {  
    a[i] = b[i] + c[i];  
}
```