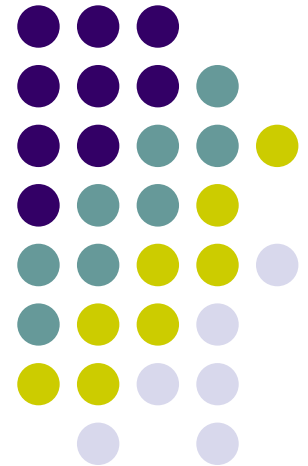


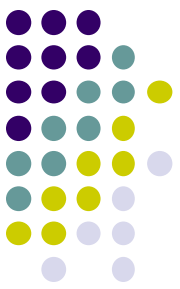
2015年度 実践的並列コンピューティング 第2回

並列アーキテクチャと
並列プログラムモデル

遠藤 敏夫

endo@is.titech.ac.jp

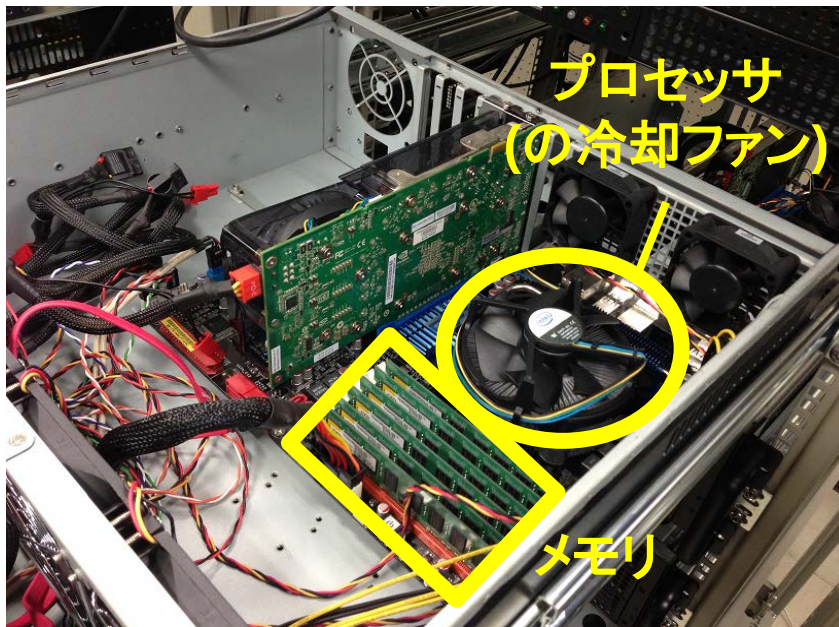
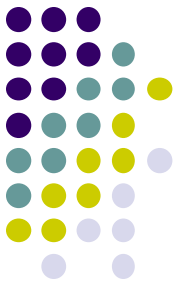




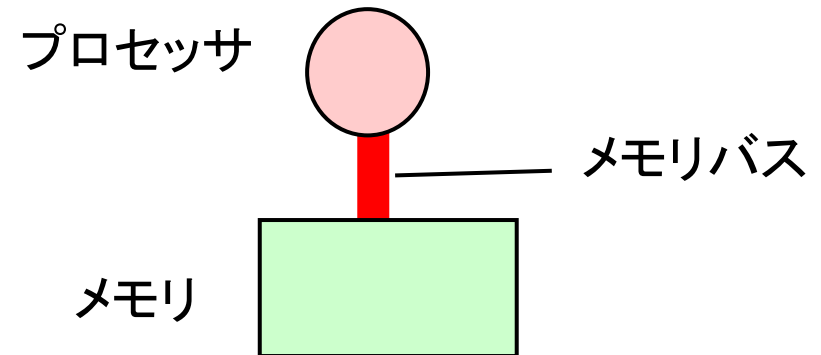
並列計算

- ソフトウェア = アルゴリズム + データ構造
 - ハードウェア = プロセッサ + メモリ・ストレージ
- ※議論を簡単にしています
- 構成要素が複数になるのが並列計算
 - ハードウェア構成を(基本でいいので)知らないとソフトウェアを作れない
 - 更に詳しく知っていれば、「もっと速い」「もっとビッグデータの」「もっと安定な」ソフトウェアが作れる

単純化した コンピュータアーキテクチャ



単純化したアーキテクチャ



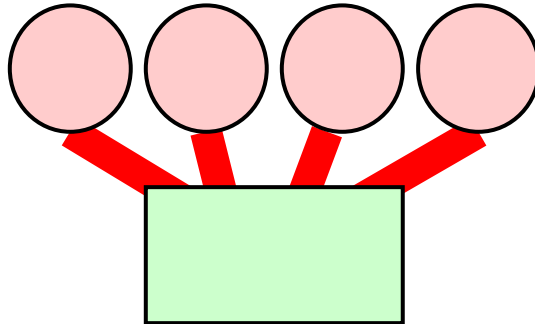
並列アーキテクチャの分類



共有メモリ並列アーキテクチャ

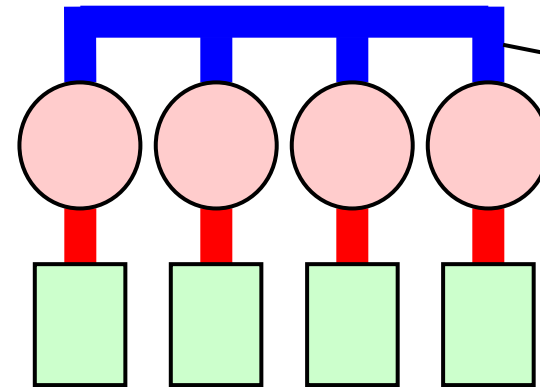
プロセッサ
(コア)

メモリ

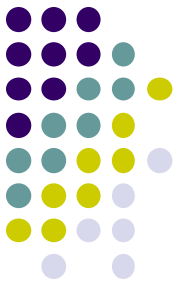


分散メモリ並列アーキテクチャ

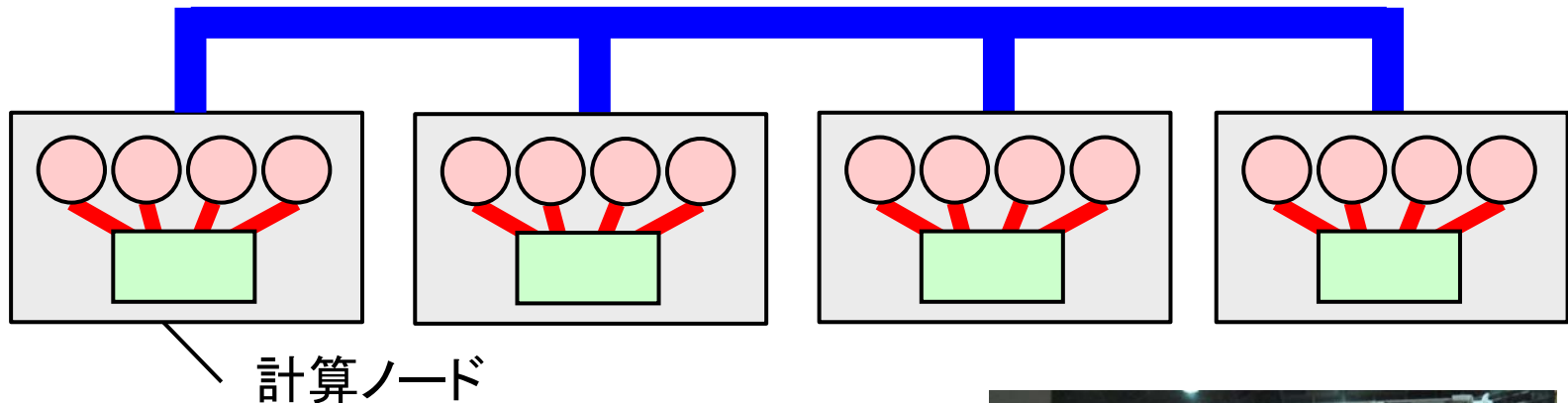
ネット
ワーク



近年の並列アーキテクチャ



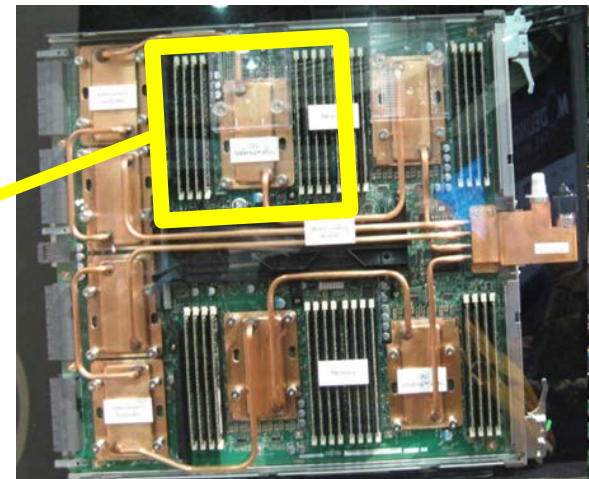
近年のスパコンのほとんど全ては共有と分散の組み合わせ
計算ノード内は共有メモリ，計算ノード間は分散メモリ



TSUBAME2
の計算ノード



京の
計算ノード

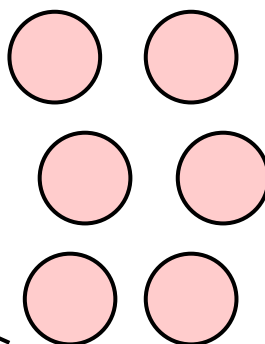


用語:ソケットやコア

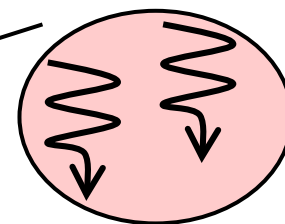
- マルチコア時代になり, プロセッサの定義が複雑に
 - プロセッサの機能を持つ「コア/CPUコア」を複数, パッケージに詰め込むようになった
 - HyperThreadingによりさらに複雑に. 1コアを2つのハードウェアスレッドが共有する. OSからはハードウェアスレッドがプロセッサに見える



ソケット
(パッケージ)



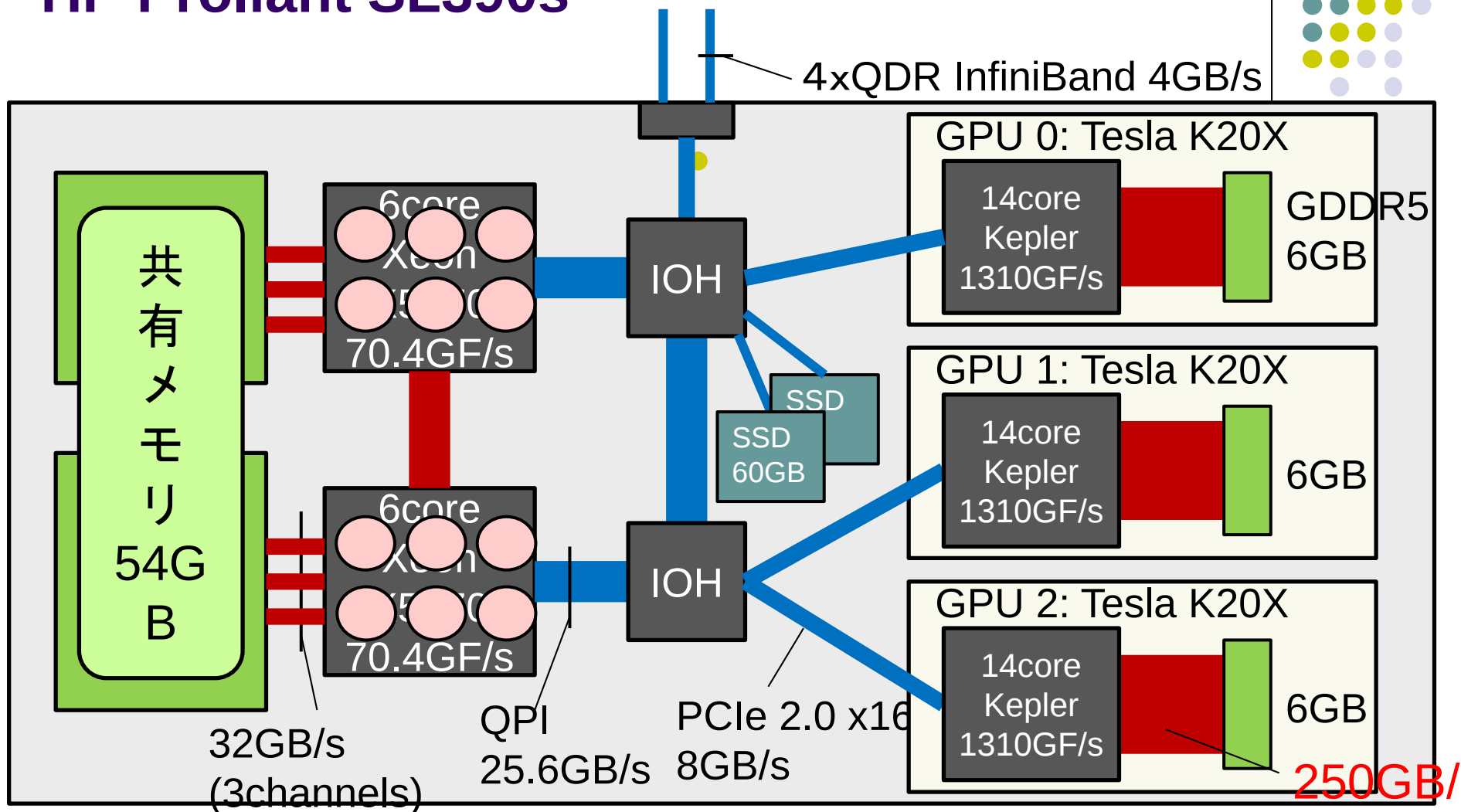
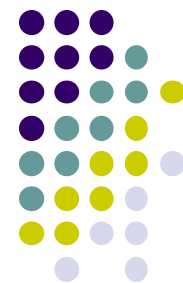
6コア



2ハードウェア
スレッド
(hyper thread)

TSUBAME2のノードアーキテクチャ

HP Proliant SL390s



HyperThreadingのため、2ソケット×6コア×2HT=24プロセッサに見える。/proc/cpuinfoファイルの内容を参照

TSUBAME2のネットワーク構成(簡略化)

Dual-rail Full-bisection Fat-tree topology



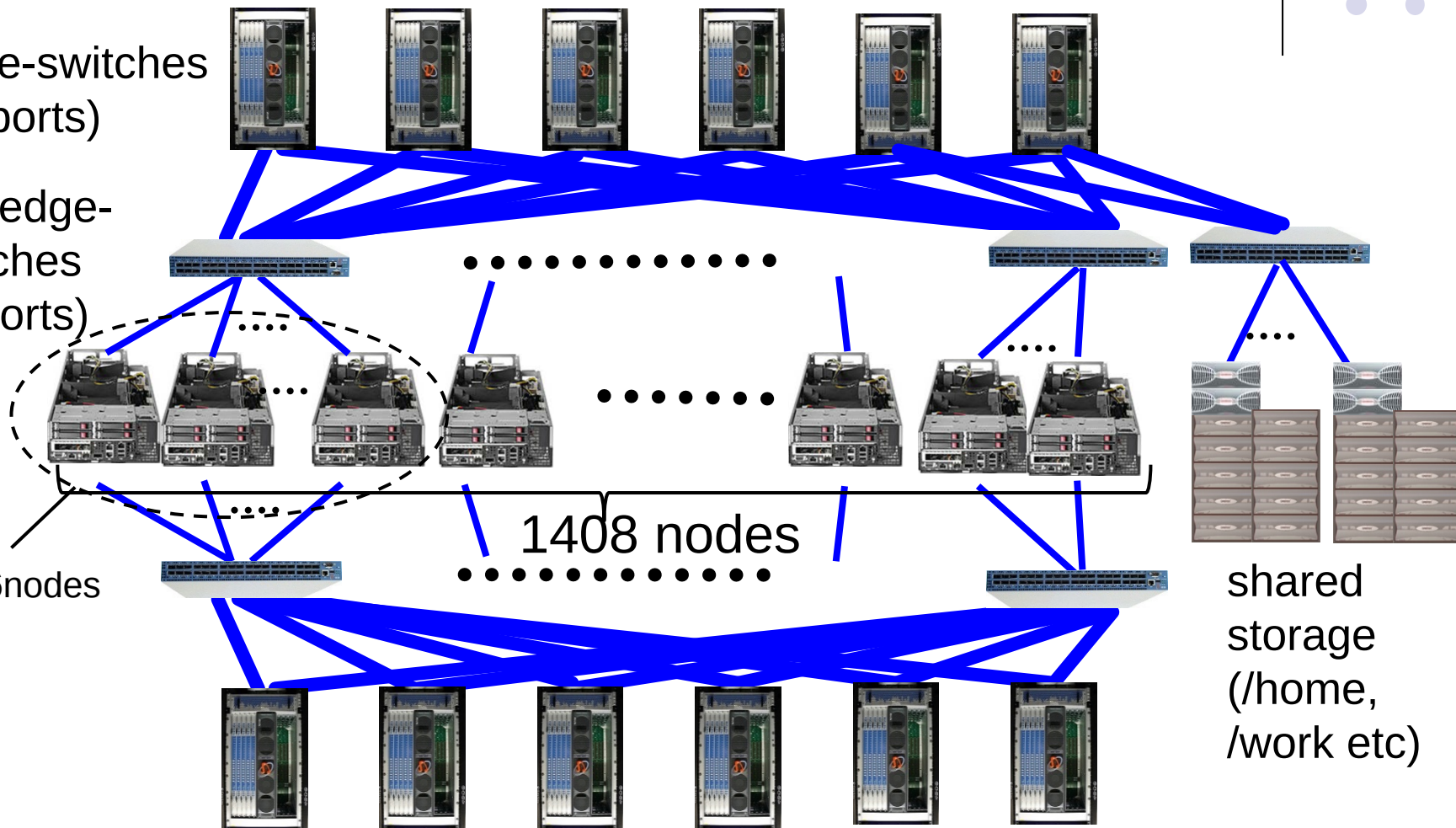
6 core-switches
(324ports)

~90 edge-
switches
(36ports)

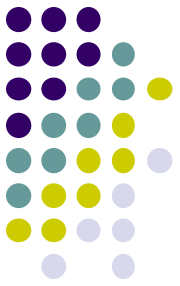
14~16nodes

1408 nodes

shared
storage
(/home,
/work etc)



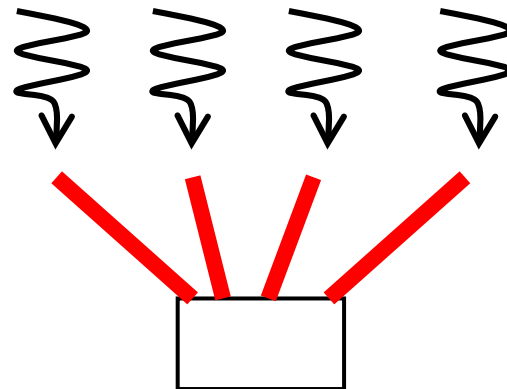
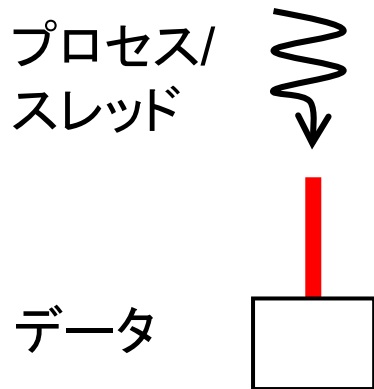
並列プログラミングモデルの、 メモリモデルによる分類



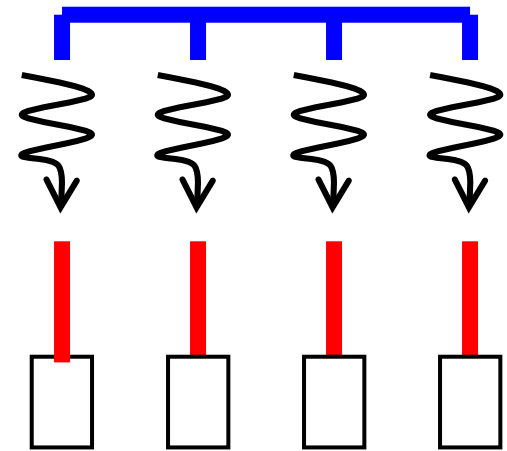
逐次

共有メモリモデル

分散メモリモデル



- スレッド達が共通の
データにアクセス可能
- **OpenMP**(言語拡張)
 - pthread(ライブラリ)
 - Cilk (言語拡張)

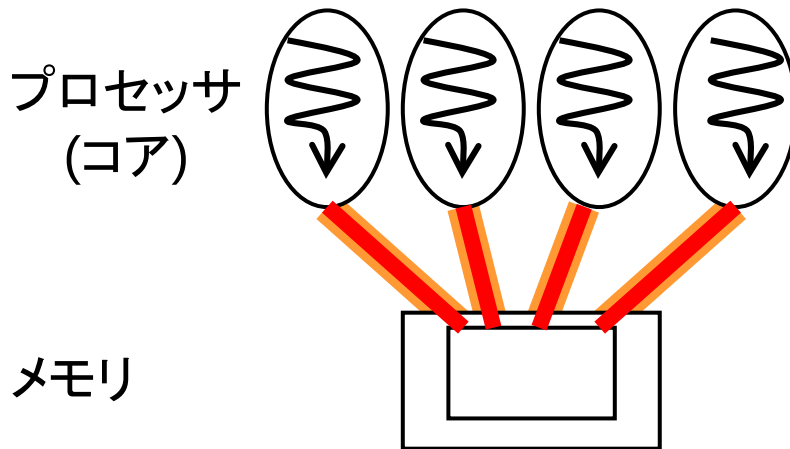


- プロセス間では通信
が必要
- **MPI** (ライブラリ)
 - socket (ライブラリ)
 - UPC, HPF(言語拡張)
 - MapReduce

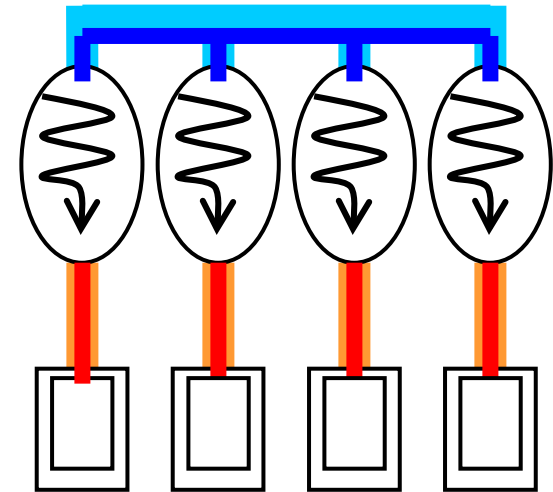
プログラミングモデルとアーキテクチャ (1)



共有メモリアーキテクチャ



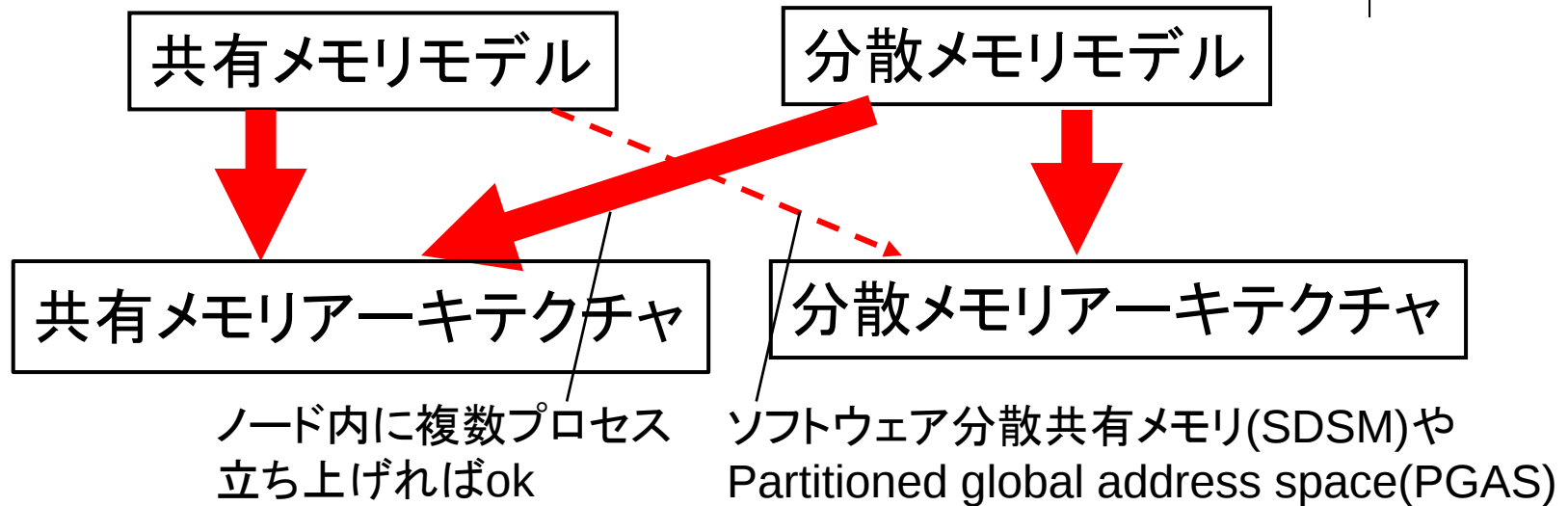
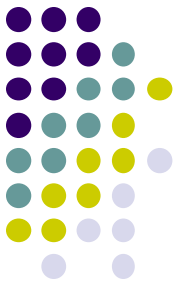
分散メモリアーキテクチャ



この図のように、きれいに対応するケースばかりではない

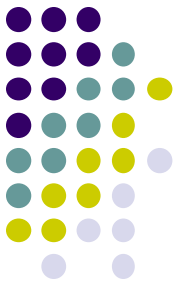
- 1コアに複数スレッド/プロセスが同居しうる
- 共有メモリアーキテクチャ上で分散メモリモデルが走りうる

プログラミングモデルとアーキテクチャ(2)



TSUBAME2の場合:

- 共有メモリモデル(OpenMPなど)で、1ノード内の12コアまで利用できる
- ノードをまたぐ、大規模並列を行いたい場合、分散メモリモデル(MPIなど)でソフトウェアを作る必要
 - 効率向上のために、MPIとOpenMP両方使う場合も



講義では

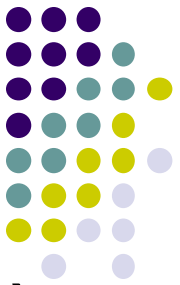
Part1: 共有メモリ並列プログラミング(OpenMP)

Part2: 分散メモリ並列プログラミング(MPI)

Part3: GPUプログラミング(CUDA?)

共有メモリ・並列メモリ両方の知識が必要

Hadoop, Swift並列scriptなど「モダン」な要素もできれば...



About Account

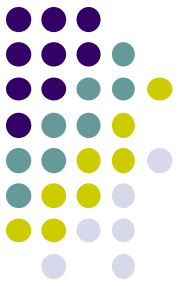
- TSUBAME2のアカウントができたなら, 連絡してください.
授業用のTSUBAMEグループへ登録します.

Subject: TSUBAME2 ppcomp account

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

- 専攻・研究室
- 学年
- 氏名
- アカウント名

次回: OpenMPとは



- 共有メモリモデルによる並列プログラミングAPI
 - Pthread, Java threadより組むのがちょっと楽
- C言語, C++, Fortranに対応
- 並列化のための指示文や, ライブラリ関数
 - 指示文: #pragma omp ~~

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