

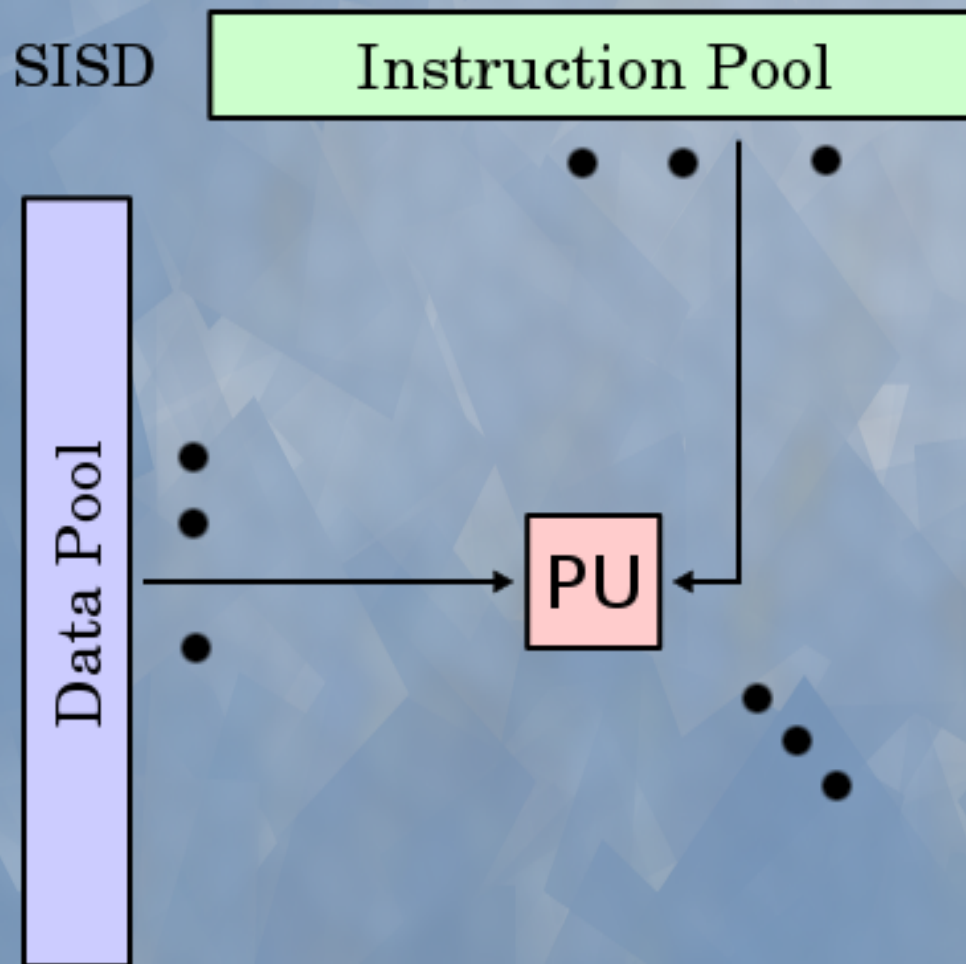
HPC na biurku

Wojciech Dębski

22.01.2015

HPC - co to jest?

Obliczenia szeregowe i równoległe, podział według M. Flyna



Klasyczny komputer

```
for(i=0;i<N;i++)  
    c[i] = a[i] + b[i];
```

```
c[0] = a[0] + b[0];  
c[1] = a[1] + b[1];  
.  
.  
.  
c[N] = a[N] + b[N];
```

Obliczenia szeregowe i równoległe

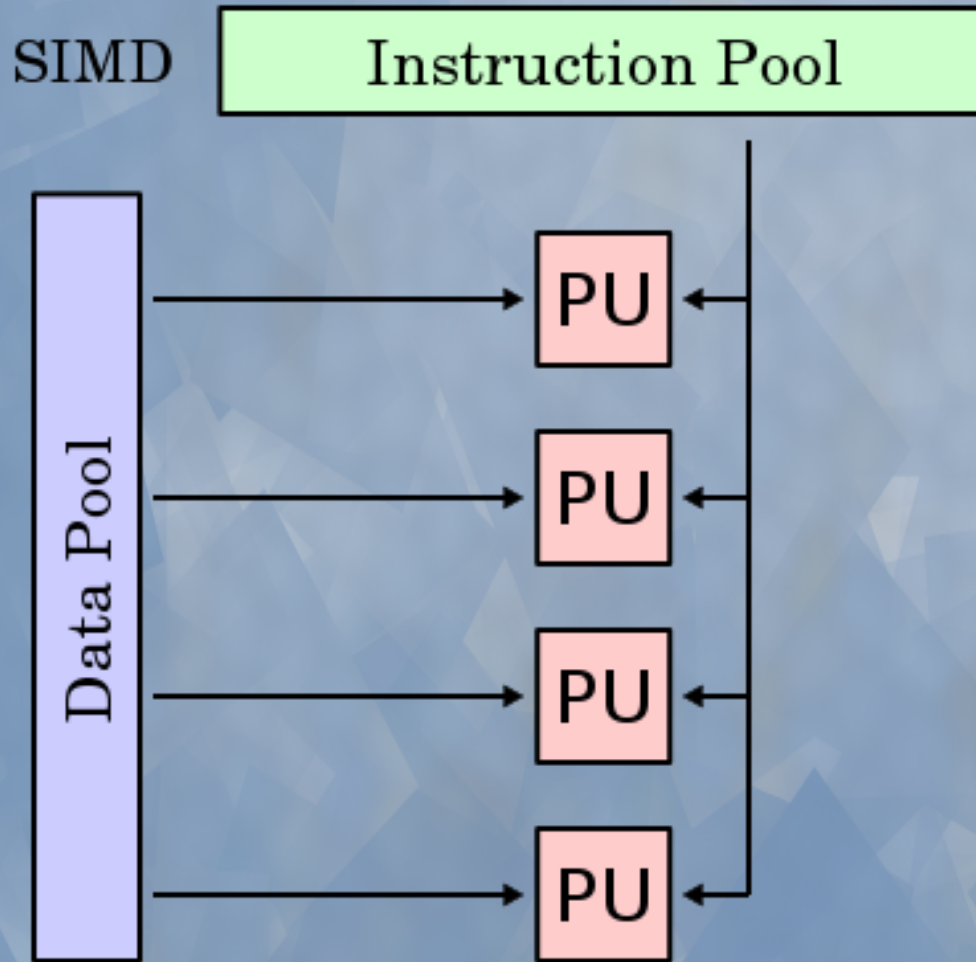


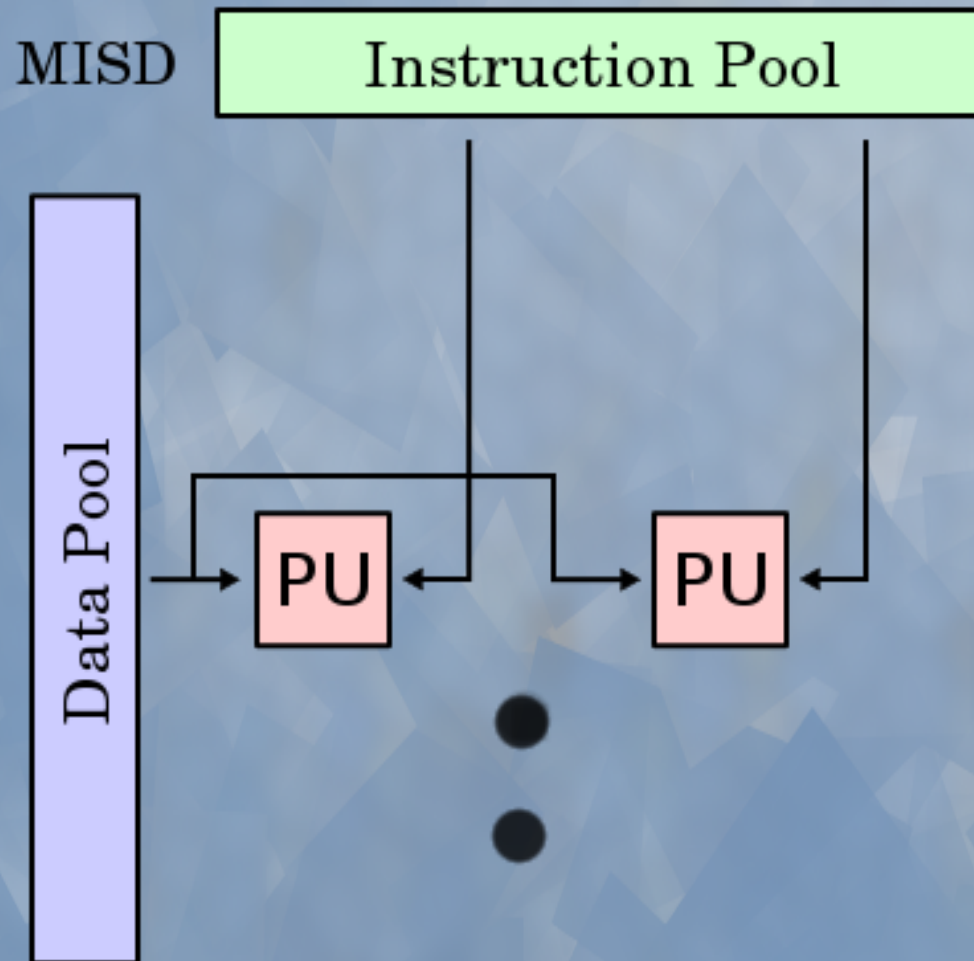
Image processing
(GPU)

$$a = a * 0.5$$

a[0]		a[0] * 0.5
a[1]		a[1] * 0.5
a[2]	==>	a[2] * 0.5
a[3]		a[3] * 0.5
a[4]		a[4] * 0.5



Obliczenia szeregowe i równoległe



Analiza sygnałów
(procesory DSP)

$a = \sin(x)$

$b = \cos(x)$

$a[0] = \sin(x)$ $b[0] = \cos(x)$

$a[1] = \sin(x)$ $b[1] = \cos(x)$

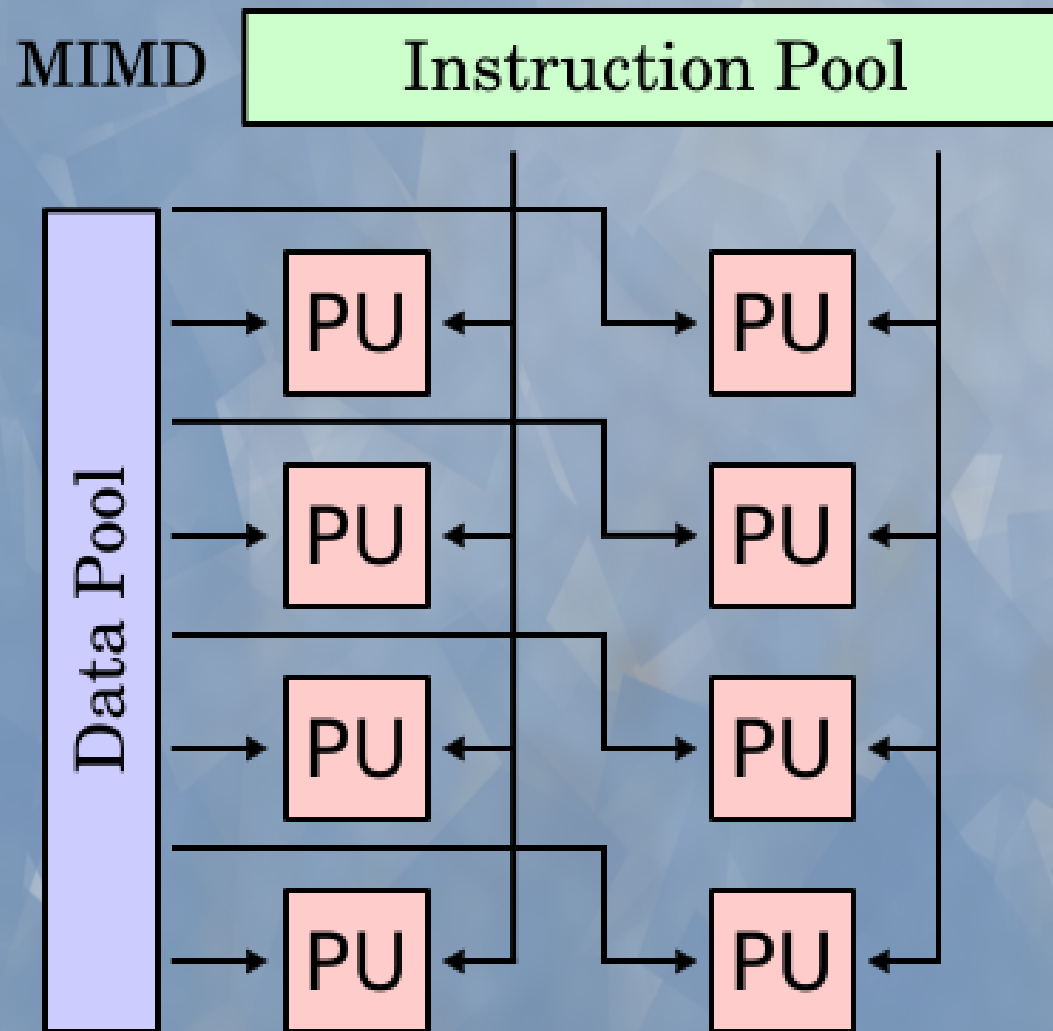
$a[2] = \sin(x)$ $b[2] = \cos(x)$

•

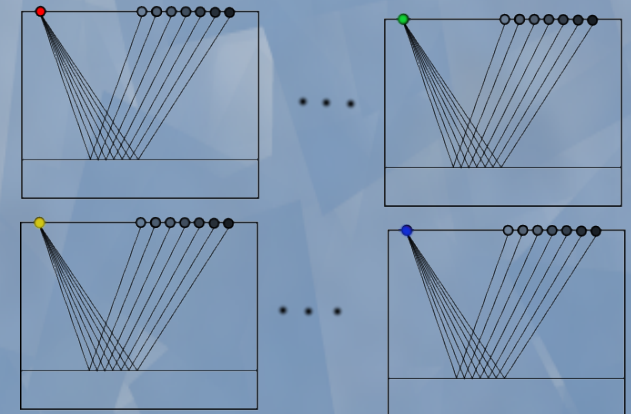
•

•

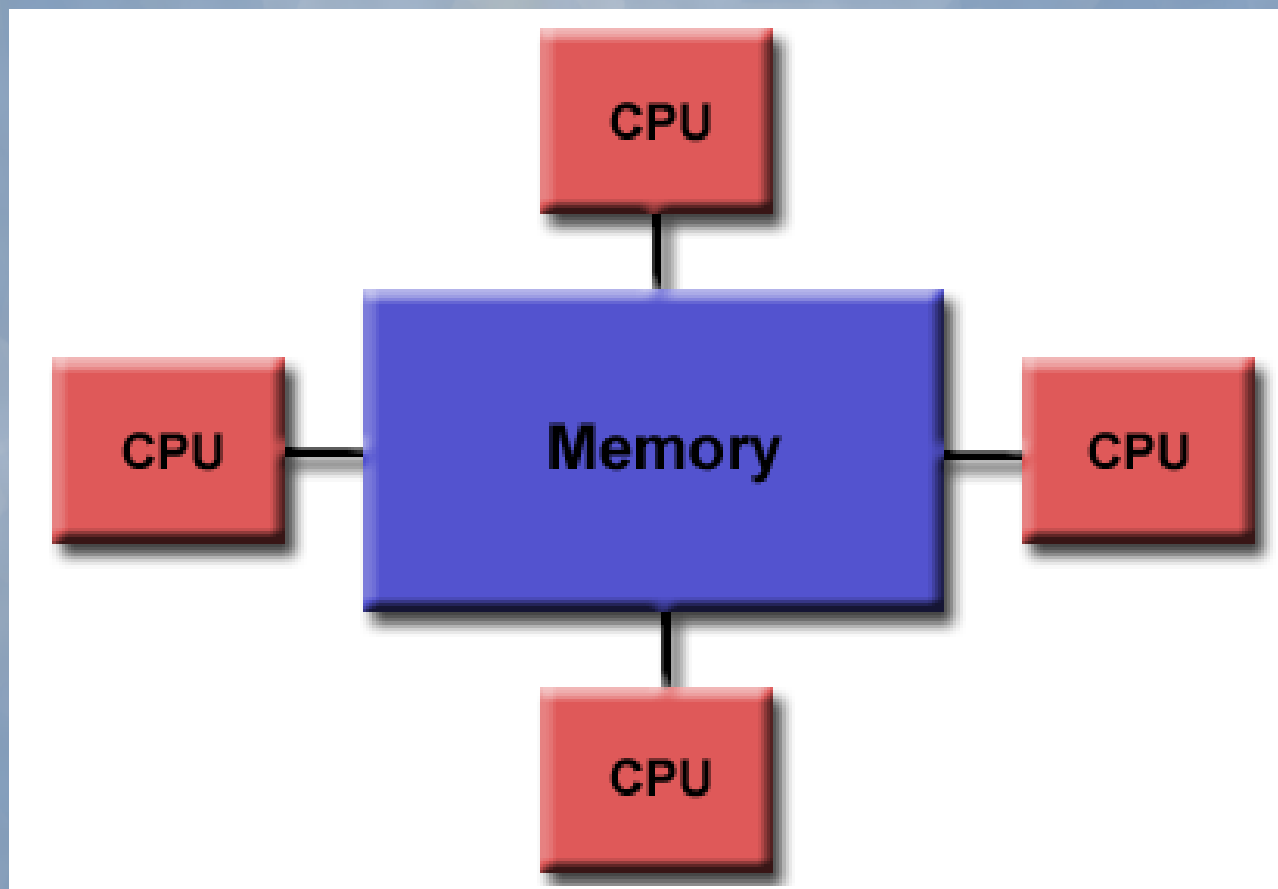
Obliczenia szeregowe i równoległe



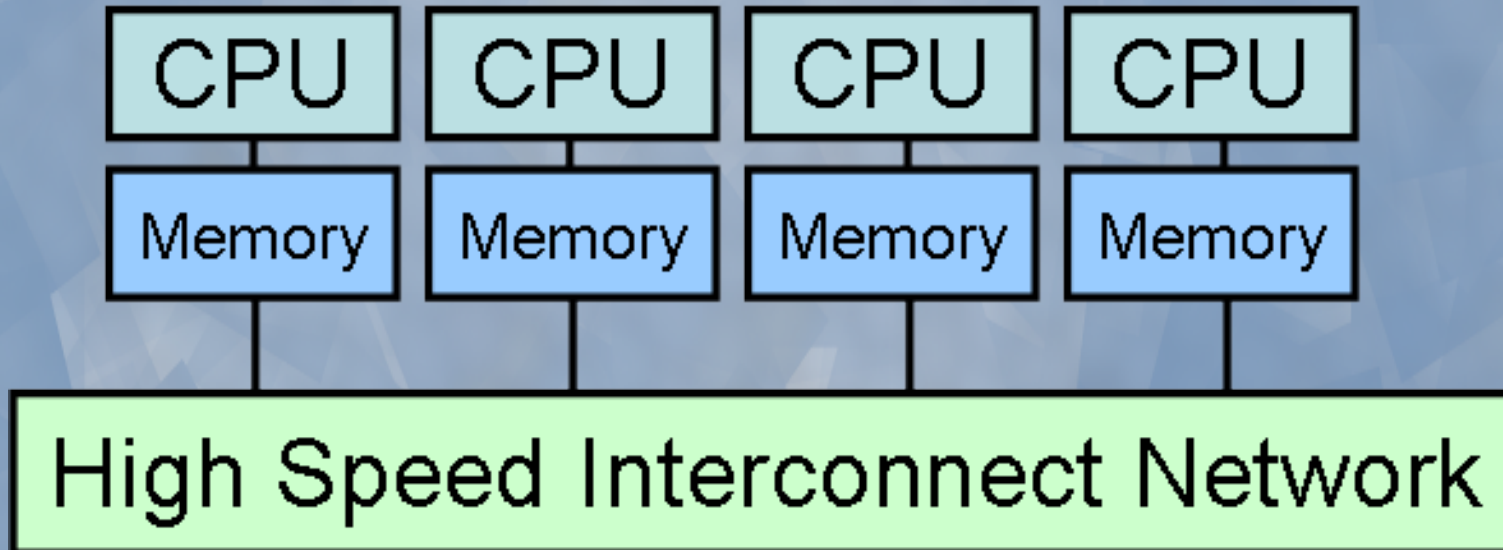
Symulacja pola falowego



Architektury komputerów - wspólna pamięć



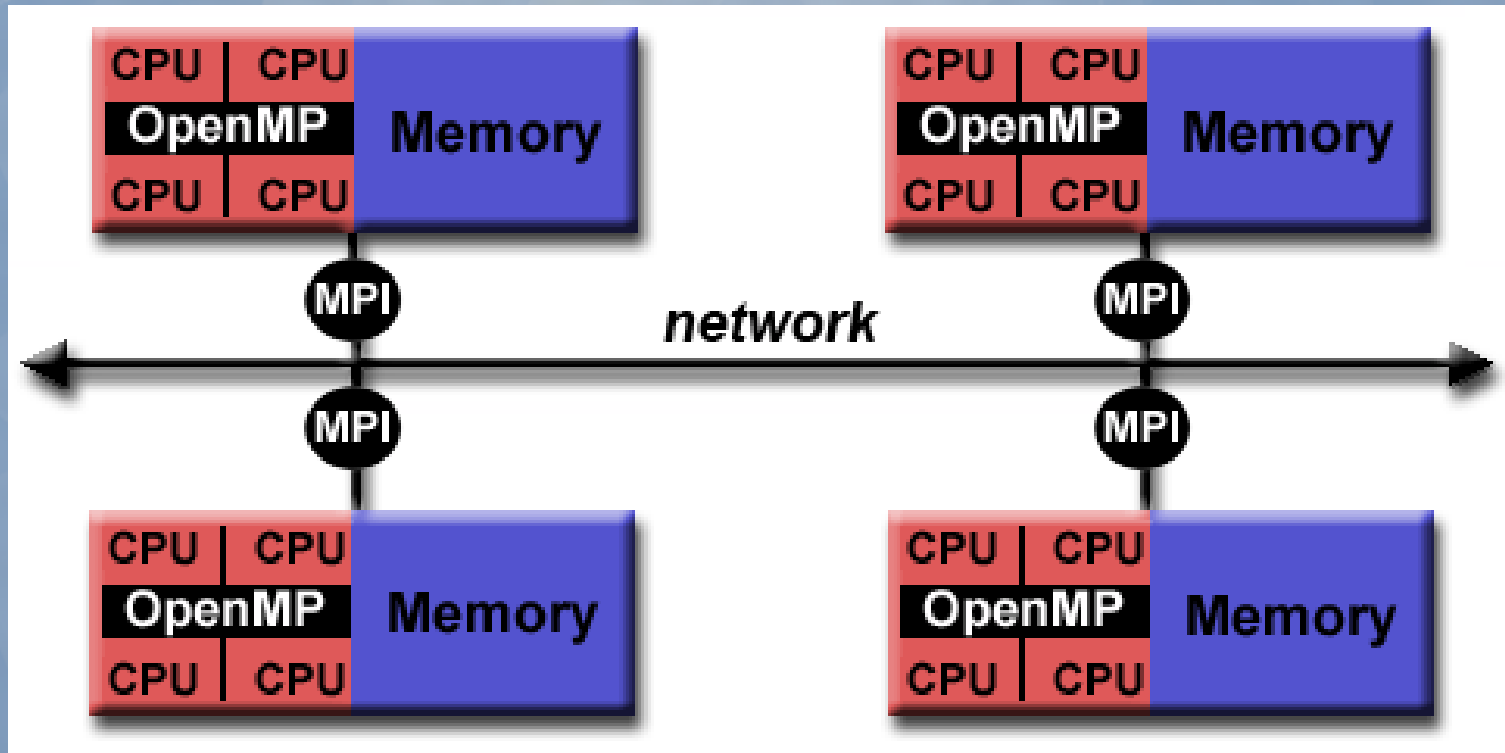
Architektury komputerów - oddzielna pamięć



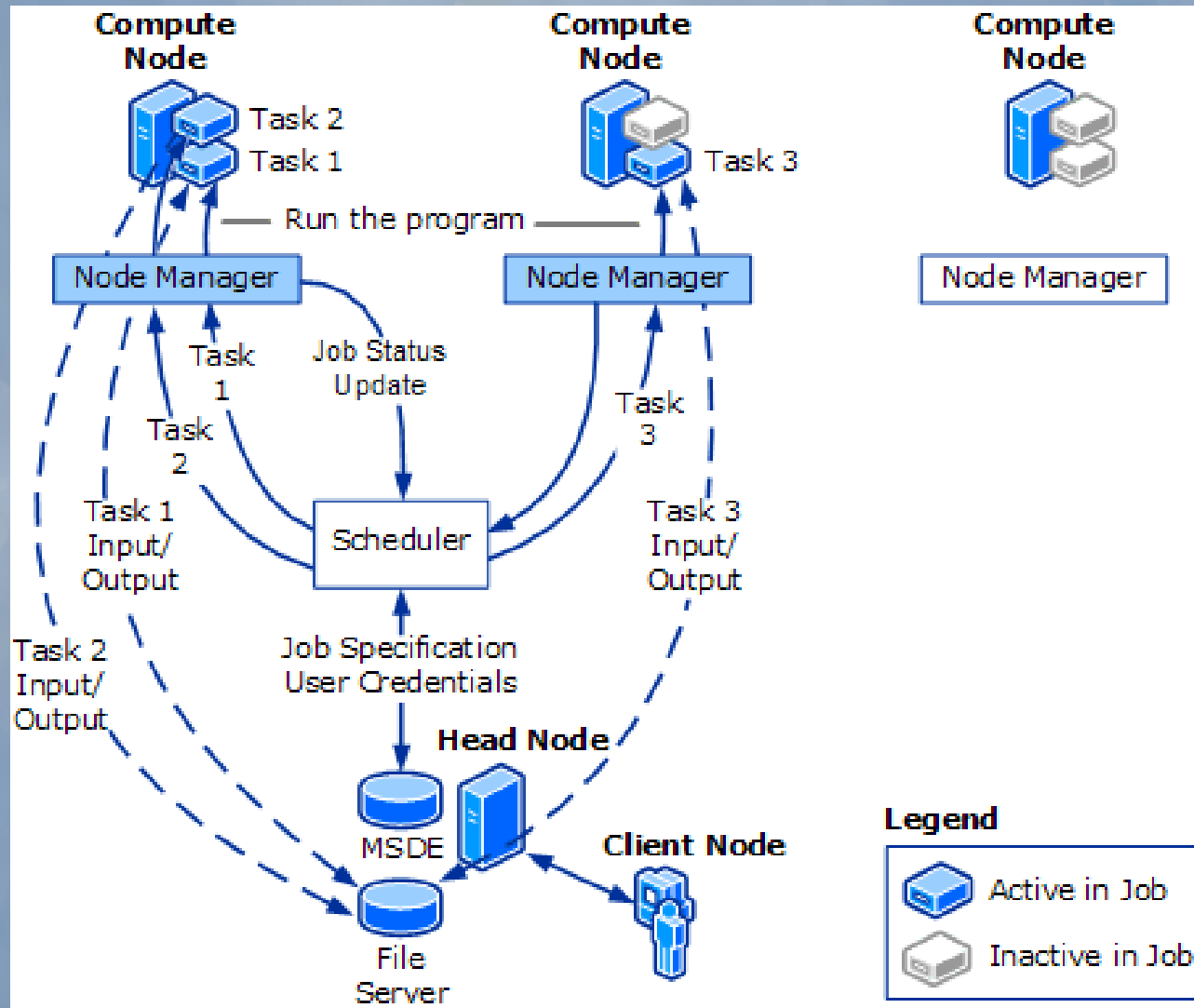
Klastry obliczeniowe - klasyczne rozwiązanie z dzieloną pamięcią



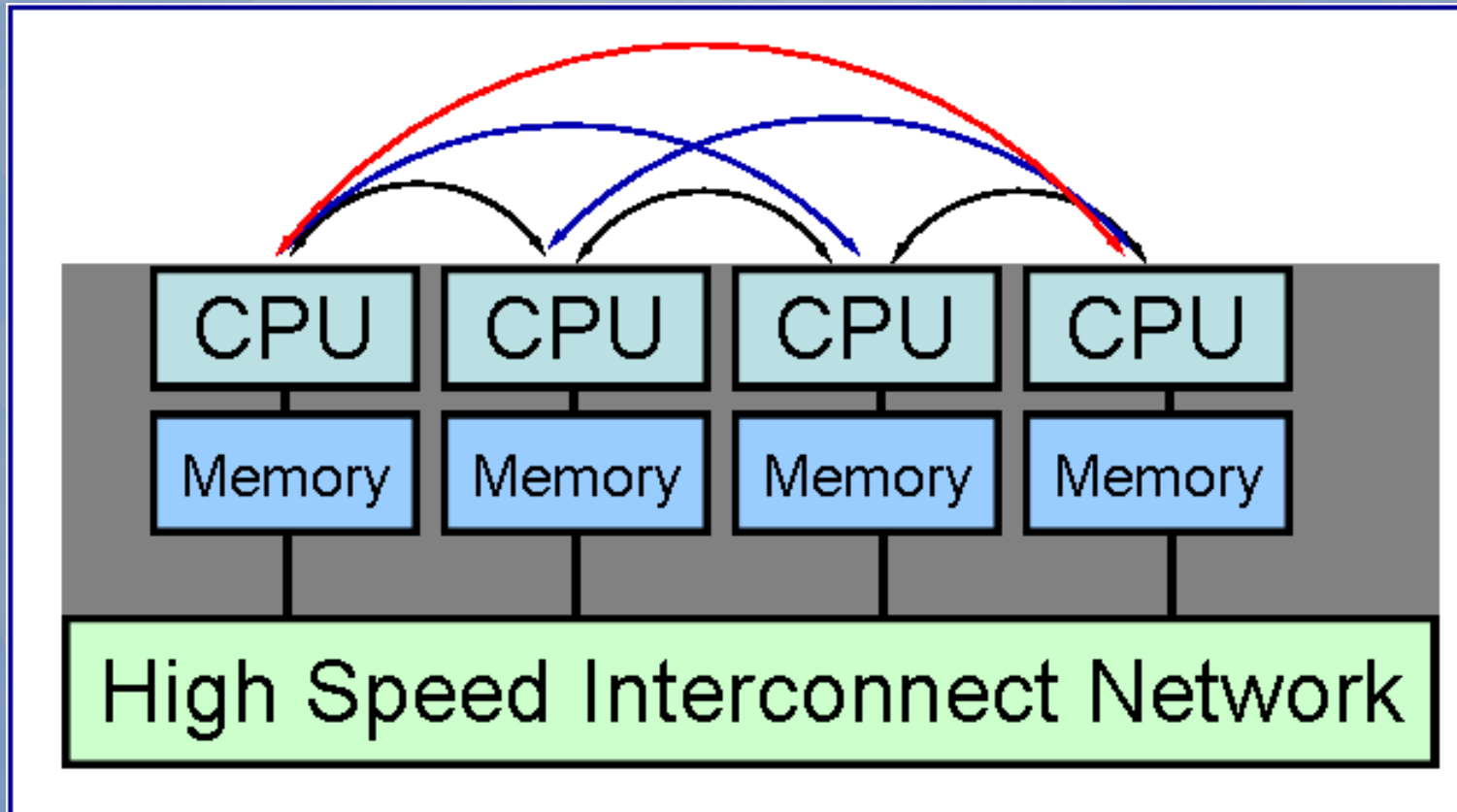
Architektury komputerów - hybrydy



Organizacja klastrów



MPI - Message Passing Interface



MPI - Message Passing Interface

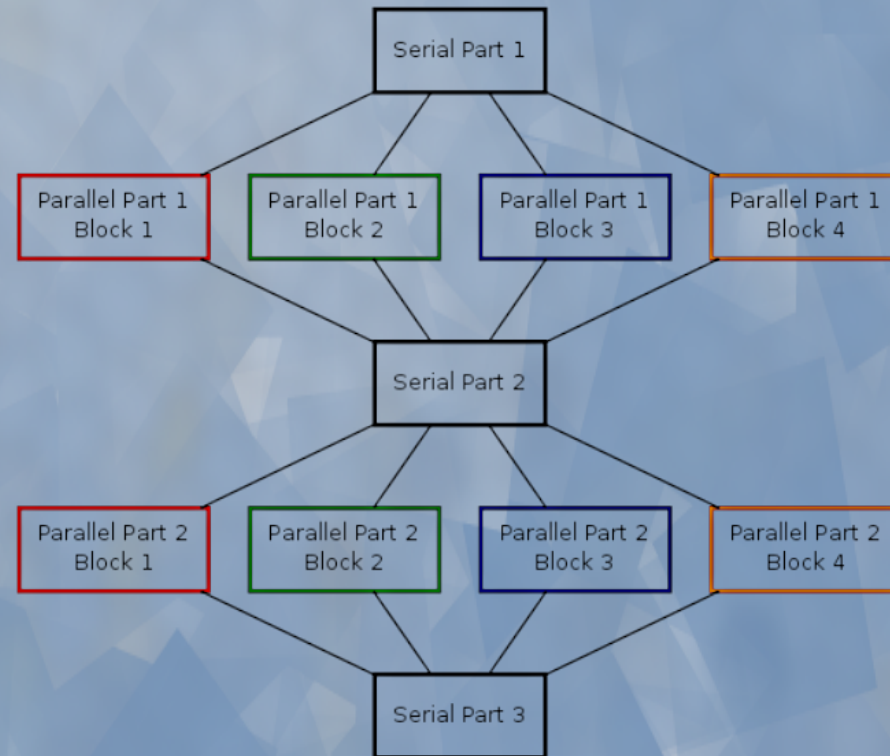
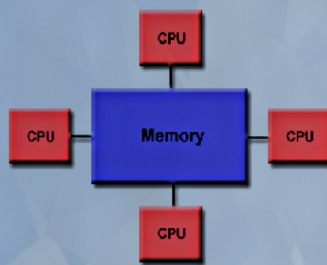
równoległe dodanie wektorów $\vec{c} = \vec{a} + \vec{b}$

```
for (i=0; i<N; i++) c[i] = a[i] + b[i]
```

```
for (i=0; i<=nx; i++)
    MPI_Send(&c33[i][0], nz+1, MPI_FLOAT, dest, 21, MPI_COMM_WORLD);
for (i=0; i<=nx; i++)
    MPI_Send(&c44[i][0], nz+1, MPI_FLOAT, dest, 21, MPI_COMM_WORLD);
    .
    .
    for (i=0; i<=nxs; i++)
        sxm[i][0] = xs + (float) i * dxs;
    .
    .
for (dest=1; dest<=all_proc-1; dest++)
{
    MPI_Recv(&pom, 1, MPI_INT, dest, 101, MPI_COMM_WORLD, &status);
    printf ("Proc:%d Transfer succeed!\n", pom);
}
```

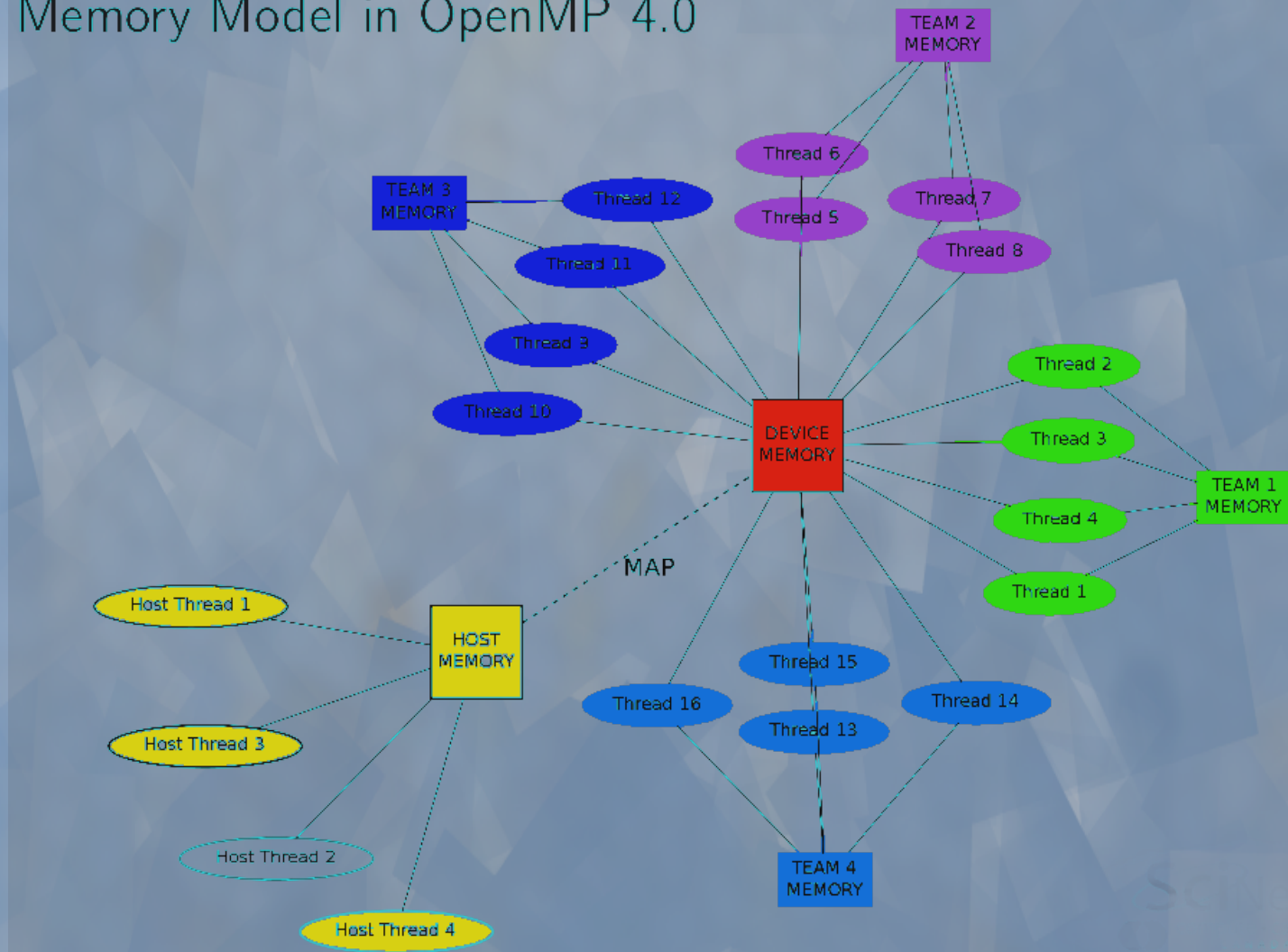
OpenMP 3 - shared memory model

Execution Model in OpenMP



OpenMP 4 - shared memory model

Memory Model in OpenMP 4.0



GPGPU - general purpose graphic processing units

Przykłady

KONIEC