



# TIDeFlow: A parallel execution model for High Performance Computing programs.

Computer Architecture and Parallel System Laboratory,  
Electrical and Computer Engineering University of Delaware, Newark Delaware

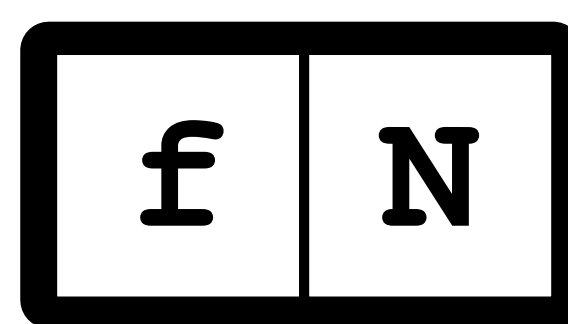
## A parallel representation of programs

TIDeFlow is a collection of theories and tools that offers an alternative to traditional programming and execution models. This work is based on the observation that *HPC programs are mostly composed of loops*. The objective is to provide programmers with tools to naturally express the interplay between loops.

### TIDeFlow: Basic Constructs

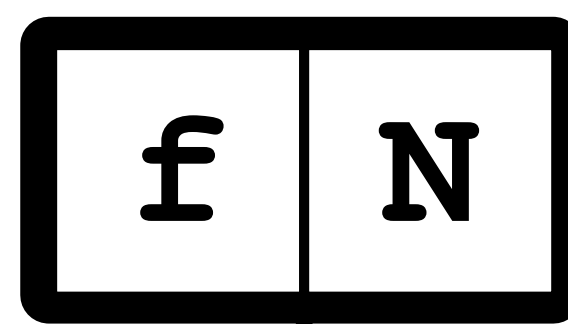
#### Parallel Loops

parallel for  $i$  in 1 to  $N$   
 $f(i)$



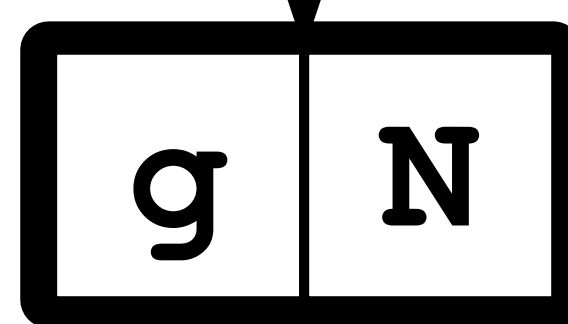
#### Weighted Dependencies

for  $t$  in 1 to  $T$   
parallel for  $i$  in 1 to  $N$   
 $f(i): \{Data(i,t)=...\}$



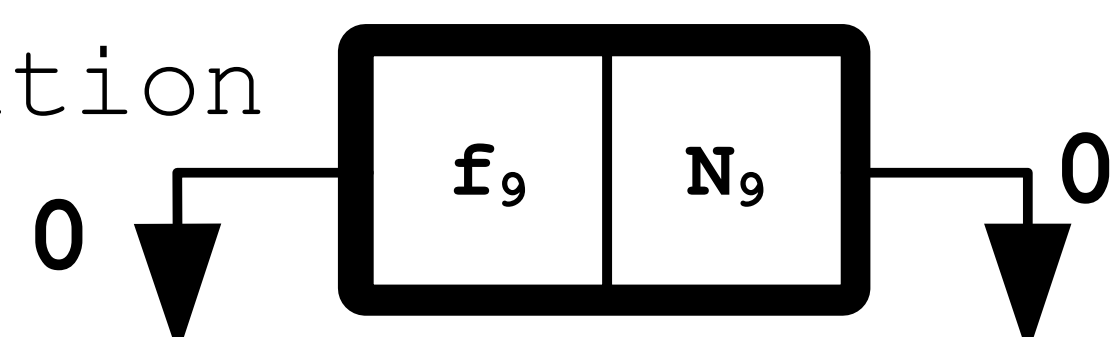
$k$

parallel for  $i$  in 1 to  $N$   
 $g(i): \{...=Data(i,t-k)\}$



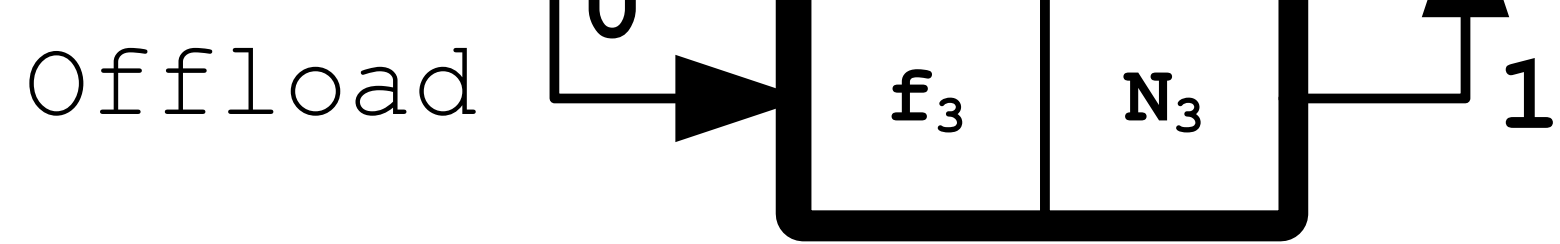
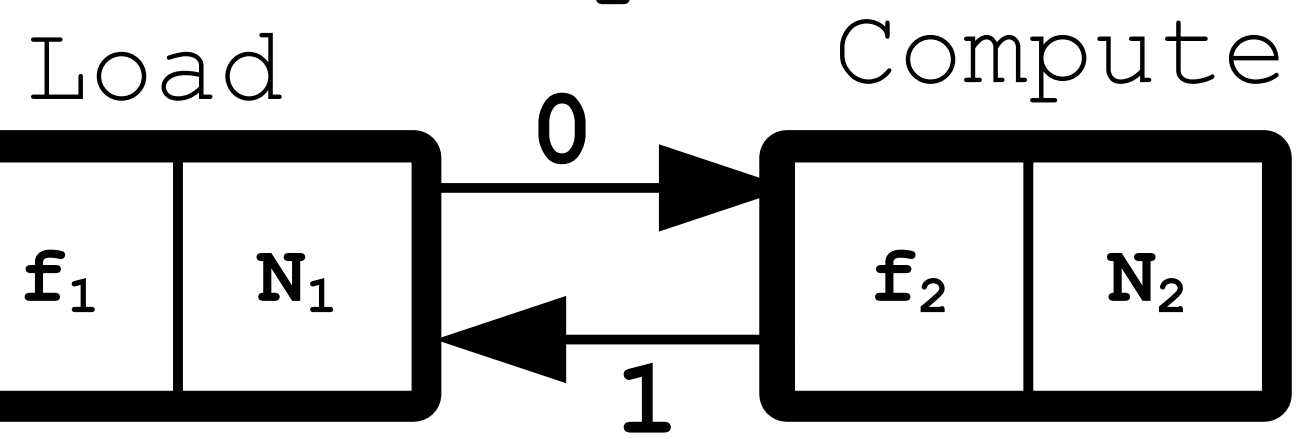
### Example: Matrix Multiply

Initialization

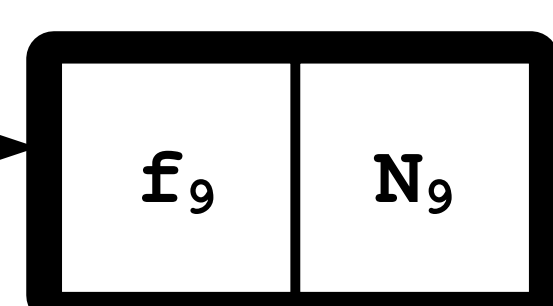


Buffer 1  
(For Double Buffering)

Tile Computation

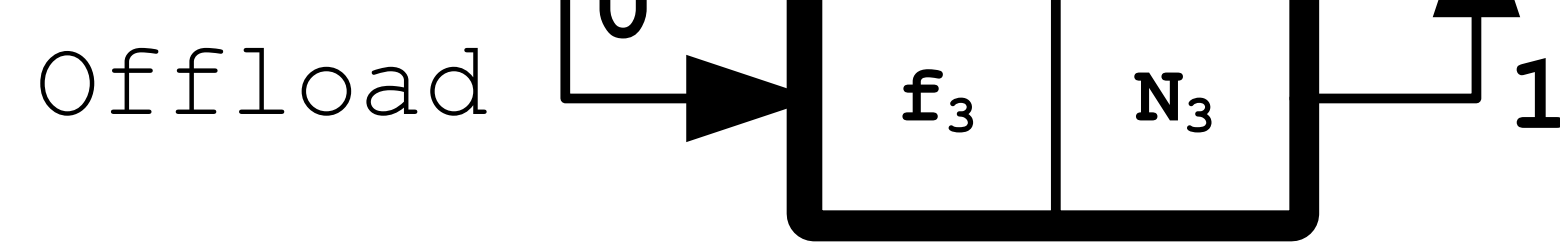
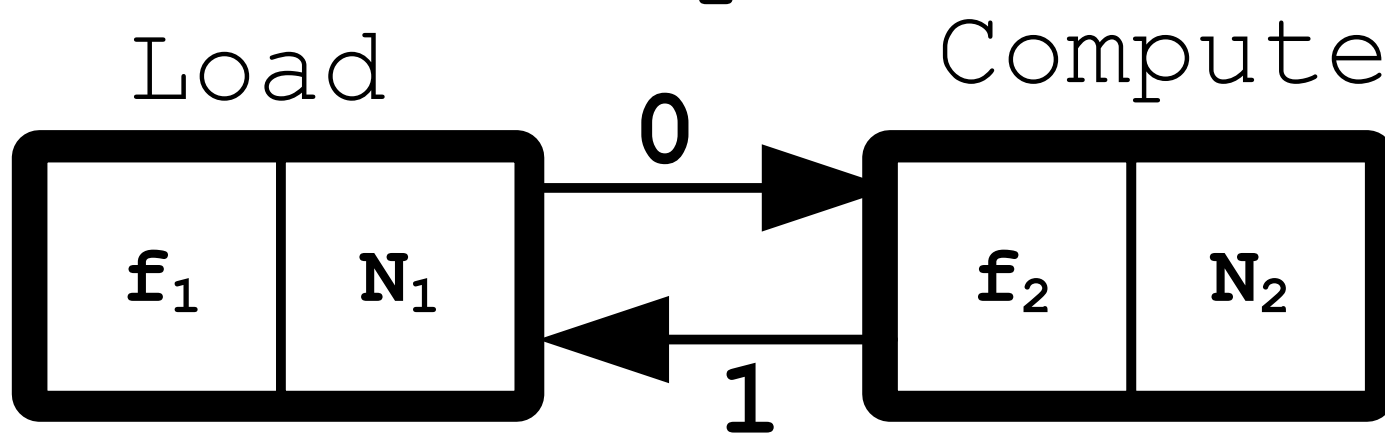


Use Result



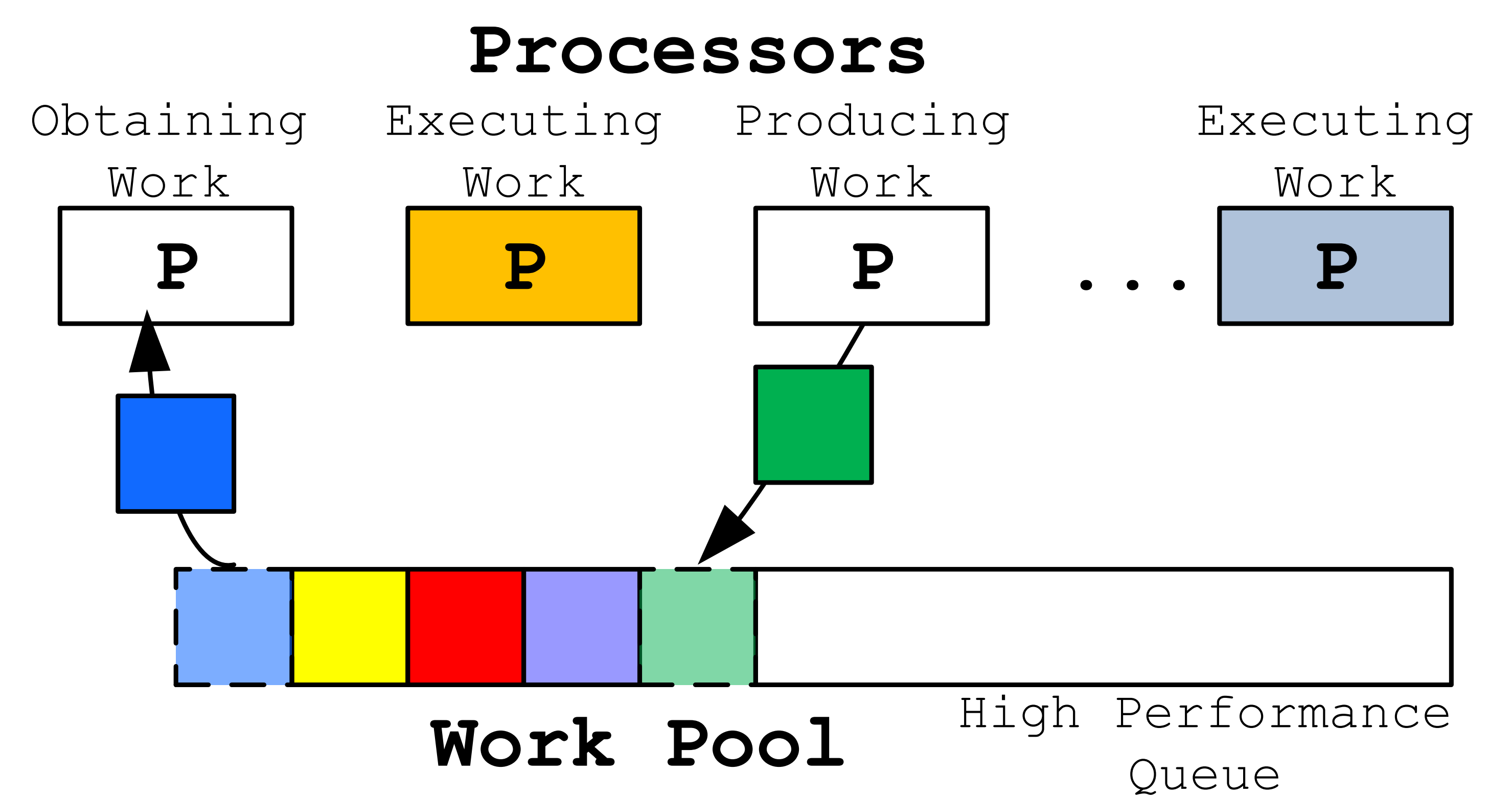
Buffer 2  
(For Double Buffering)

Tile Computation



## A distributed runtime system

Each processor participating in the computation is responsible for obtaining its own work making the intervention of an Operating System Scheduler unnecessary. As a result, finer-grain parallelism is possible.



## Results

Four typical HPC kernels were developed and run using TIDeFlow in Cyclops-64. The results show good scalability with little programmer effort. In each case, the programmer had to write (1) the code for each parallel loop in the application and (2) a graph representing their dependencies.

