

# APPLICATIONS (PARALLEL)

EFFICIENT

↳ "STRUCTURE"

{ PARALLEL PATTERNS }

MURRAY COLE '88

2000

ALGORITHMIC SKELETONS

DESIGN PATTERN

(HPC COMMUNITY)

(PARALLEL Des. Pat)

mid 00s

List<T>

↑  
int  
string

P3L PISA PAR PROCESSING LANG

HP - DI

> p3lcc sample.p3l

↓  
p3lrun

> p3lrun ↩

```
seq f in(int x) out(int y)
${ y = x*x; }$
end seq

form main in(int a) out(int b)
  f in(a) out(b)
end form
```

sample.p3l

APPLICATIONS

↳ { PARALLEL PATTERNS }

MEANING ?

ALGORITHMIC SKELETONS

RECURRING:

SINGLE SKELETONS

↓  
APPL

COMPOSITION

$SK_3(SK_1, SK_2)$

↓  
APPL

----->

KIND OF PAR PATTERNS

ANY APPL

↓  
1 SKEL

HUGE # OF SK

A FEW PATTERNS

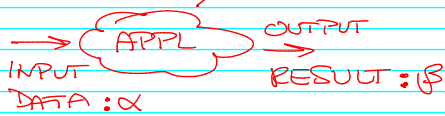
↓  
+ COMPOSABILITY

HOW TO DEFINE

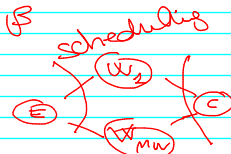
the "SMALLEST"

USEFUL SET OF PATTERNS

USER



$f: \alpha \rightarrow \beta$



APPL  $\equiv$  TASK FARM  
mw ?

Global result

$f, \alpha, \beta$  (APPLICATION DEPENDENT)

CONCERN for APPL PROGRAMMER

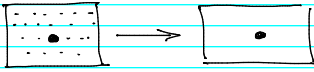
mw task farm SCHEDULING CONCERNS

# RIGHT LEVEL OF ABSTRACTION

## SKELETON / PATTERN

- FUNCTIONAL SEMANTICS
- PARAMETERS
- PARALLEL SEMANTICS

### MAP



$$A : \text{mat } \alpha \quad B : \text{mat } \beta$$

$$b[i][j] = f(a[i][j]) \quad \left. \begin{array}{l} \text{FUN} \\ \text{SEM} \end{array} \right\}$$

$$f : \text{mat } \alpha \rightarrow \text{mat } \beta$$

PARAM:  $f, \alpha, \beta$  (FUNCTIONAL PARAMS)

mw (NON FUNCT. PARAM.)

↳ PARALLELISM DEGREE

PAR SEM: (INFORMAL)

each  $b[i][j]$  may be computed in parallel

## TASK FARM PATTERN

$$\text{FUN SEM: } x_1 \dots x_m \rightarrow f(x_1) \dots f(x_m)$$

$$\hookrightarrow f : \alpha \rightarrow \beta$$

$$\text{PARAM: } f(\alpha, \beta)$$

mw (NON FUNCTIONAL)

PAR SEM: → FORMAL

→ INFORMAL

↳ INFORMAL

each task  $x_i$  may be computed in parallel

general

↓ TARGET ARCHITECTURE

$$\text{img}_1 \dots \text{img}_m \xrightarrow{\text{Bilw}} \text{img}'_1 \dots \text{img}'_m$$

forall(map(f))

↓ pixel → pixel  
col      row

Function computed

form(map(f))

FOR SEM:  $\text{map} : (\alpha \xrightarrow{f} \beta) \rightarrow \text{mat } \alpha \rightarrow \text{mat } \beta$   
 $\text{form} : (\alpha \xrightarrow{g} \beta) \rightarrow \text{stream } \alpha \rightarrow \text{stream } \beta$   
 $\text{form}(\text{map}) : (\alpha \xrightarrow{f} \beta) \rightarrow \text{stream}(\text{mat } \alpha) \rightarrow \text{stream}(\text{mat } \beta)$

f: pixels  $\rightarrow$  pixels  
 col B&W

form(map(f))

stream(img)  $\rightarrow$  stream(img)  
 col B&W

FOR SEM

form  $\forall$  element of the stream is computed in //

map  $\forall$  item of the image " " " //

by composition  $\rightarrow$   $\forall$  item of image on the stream is computed in parallel

concurrent activity graph

P3L

map g in (PIXEL a[N][N])  
 out (PIXEL b[N][N])

each row in parallel

col-to-bdw in (a[\*i][\*j])  
 out (b[\*i][\*j])

in (a[\*i][\*j])  
 out (b[\*i][\*j])

end map

each pixel in parallel

form main in ( ) out ( )

end form