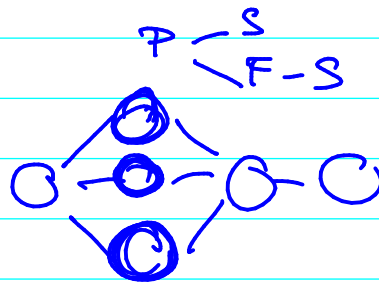


- 1) Generazione dinamica di task
- 2) Espandibilità del set degli skeleton

① mette prog. giro  $\Rightarrow$  generazione di task

$\rightarrow$  D&C

Template  $\Rightarrow$



1 ingresso  $\rightarrow$  n uscite



generare delle terminazioni  
Zion

Macro data flow

$\Rightarrow$  "consequenza" dell'interprete  
Macro data flow

2

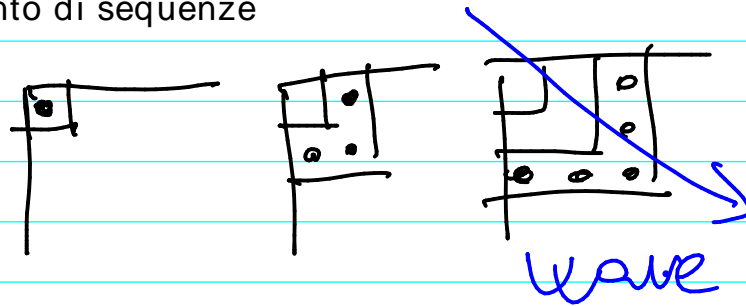
In particolare:

=> generazione di un task

- 1) istanziazione di un nuovo grafo nel TP
- 2) mettere un token nel grafo
- 3) dire dove va finire il token risultato

tipico esempio

allineamento di sequenze



## ② ESPANDIBILITÀ

problema noto: riuscire a coprire tutte le comp par

asindato

- obiettivo: 1) skeleton per le situazioni + comuni  
2) possibilità di "escape" per le situazioni - comuni  
o + particolari

⇒ Template  
aggiungere template

MDF  
aggiungere grafi MDF

3

farm'(w,...)

molto  
specifica

+HW  
+SYS

serve un template

- grafo dei processi
- grafo delle comm
- modello di perf
- regole di riscrittura

dopo

- modificare la proc di compilazione

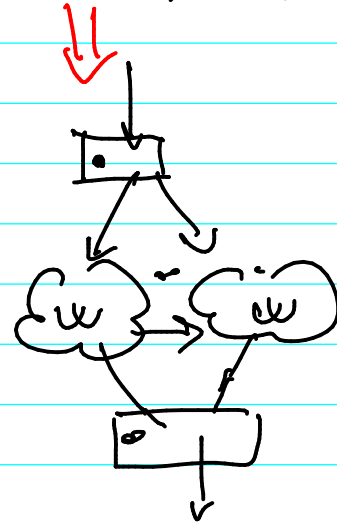
(e una cosa quasi naturale  
: estendo il set di template  
che analizzo per realizzare  
la rete di processi che  
implementa il mio programma)

richiede conoscenze  
che non sono tipiche del  
programmatore applicativo  
(sono + competenze del  
programmatore di sistema)

si parte dal  
modello logico  
"applicativo"<sub>1</sub>

graf MDF

farm'(w, ...)



descritto  
imple mentazione  
vero e propria



descritto stesso  
delle informazioni

# Esempio (MUSKEL)

Forme + pipe

⇒ done MAP 8

flusso logico

