

# Reunião 02/05/18

CHAVE

Discussão e apresentação dos dados

# Base de consumo de energia

processador.dad

MIN **01** **139.91**

02 147.10

03 153.20

04 157.08

.

30 207.00

31 208.00

MAX **32** **209.00**

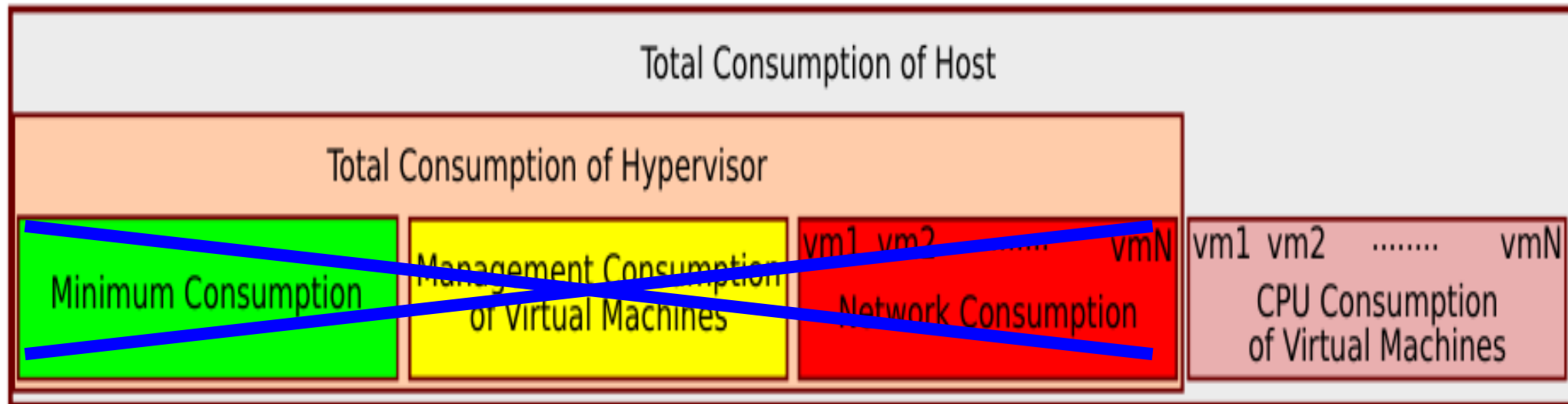
Indica a consumo (Wh) do host em relação ao nº de CPUs usado.

Controle de uso da CPU (Algoritm. dinâmico?)

Ex:  $\frac{1}{2}$  hora \* H<sub>0</sub> 4 CPUs = consumo 78,54Wh!

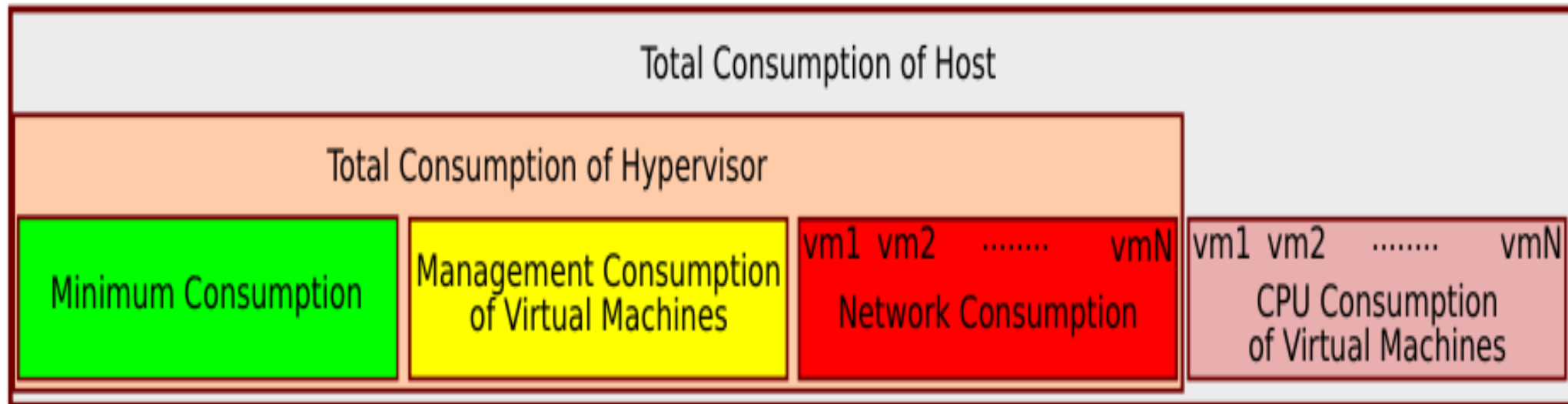
# Modelo do Mauro/Denivy

- Consumo total =
  - Consumo hypervisor
    - { Minimo + gerenciamento + rede virtual }
  - Consumo VMs



# Modelo do Mauro/Denivy

- Consumo total = Temos CT pela tabela anterior
  - Consumo hypervisor
    - { Minimo + gerenciamento + rede virtual }
  - Consumo VMs



# Modelo do Mauro/Denivy

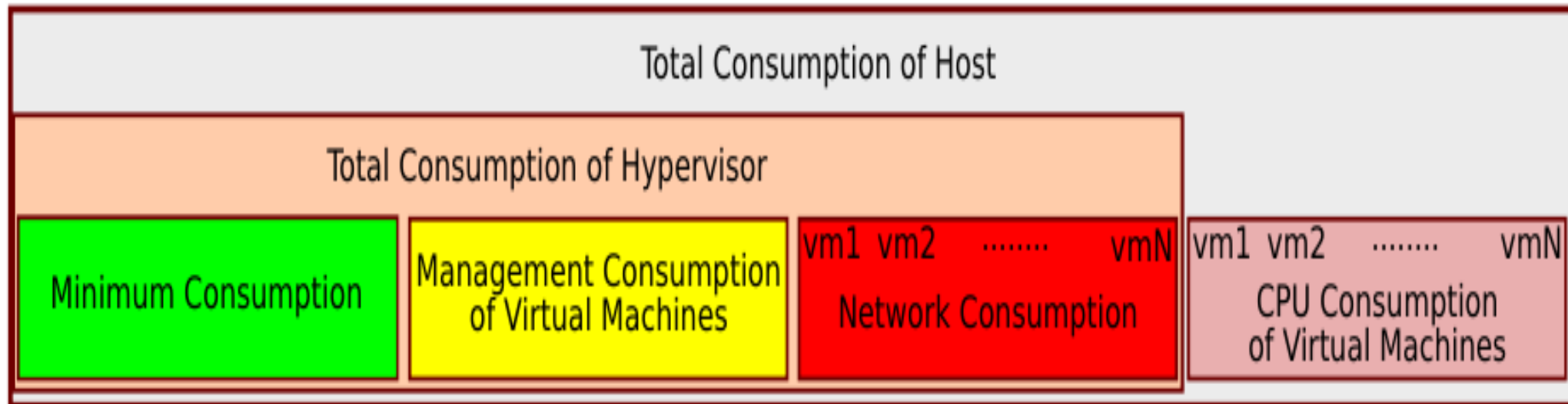
- Consumo total =

- Consumo hypervisor

- { Minimo + gerenciamento + rede virtual }

- Consumo VMs

Como estimar CH e CV?

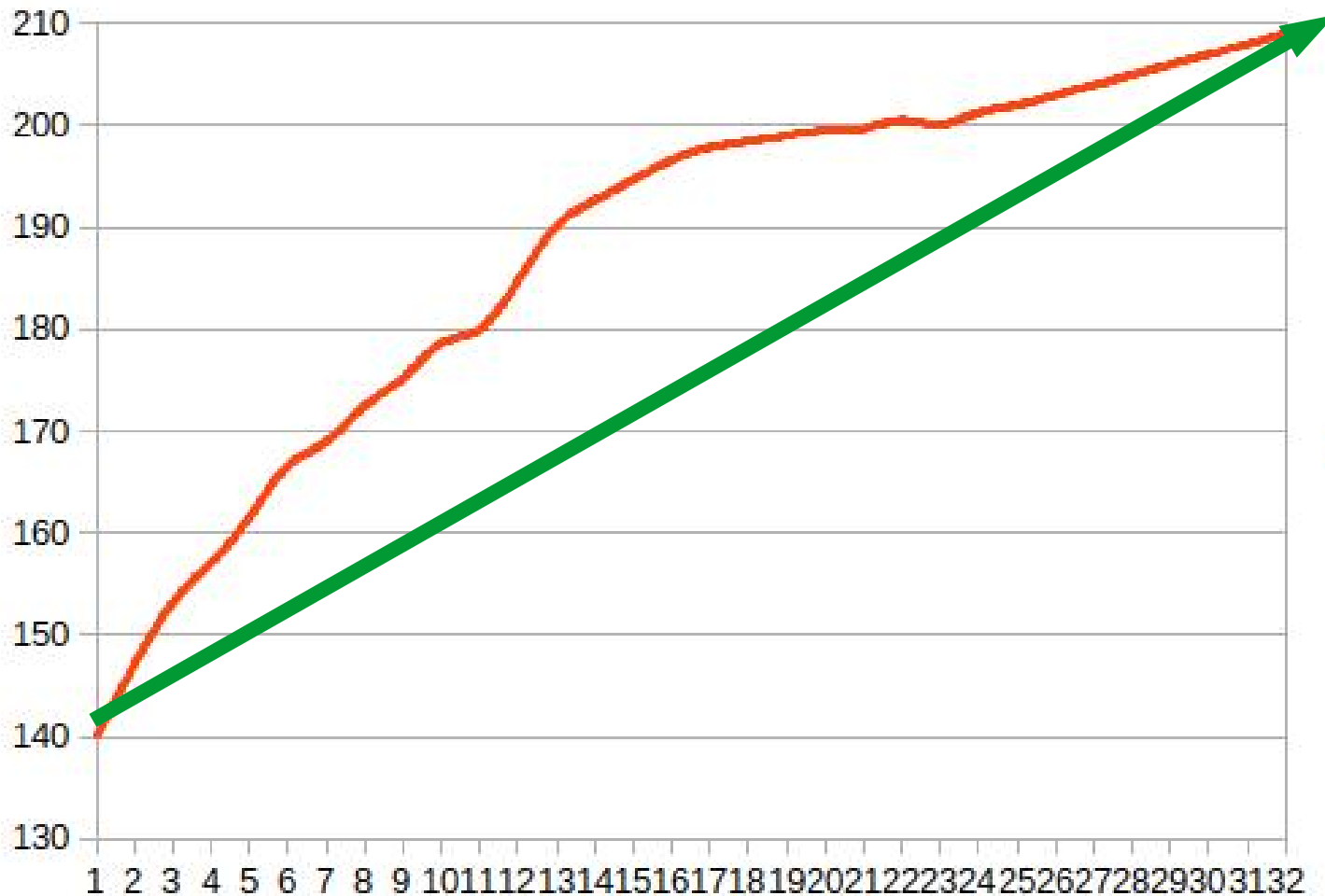


# Estimando consumo do hypervisor

- Hypervisores tipo 1 (Xen, VMware, etc)
  - É reservada uma quantidade de núcleos físicos para o dom0 (**pinados**, sem conflito com outros)
  - Afirma-se que **o consumo do hypervisor refere-se à reserva destes recursos (cpu/ram)**
- Outros tipos de virtualização?
  - Na prática ambientes comerciais não usam, ou podemos afirmar que não é o foco do simulador
  - P: percentual de uso impacta no gerenciamento.

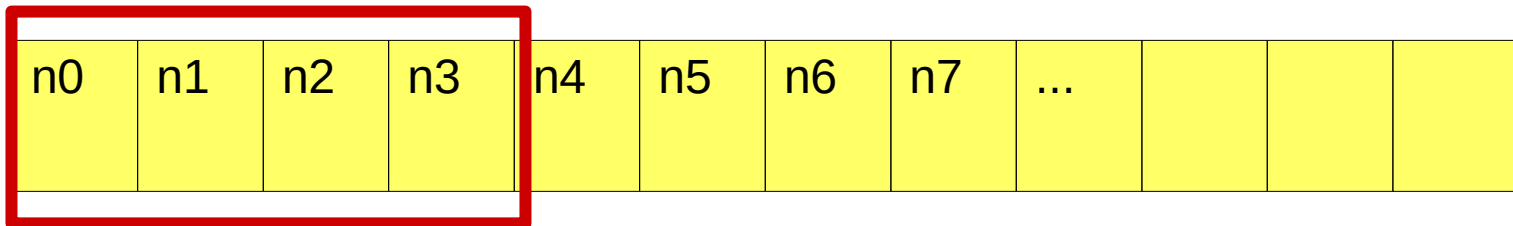
# Estimando consumo do hypervisor

- Quais são os núcleos pinados?



# Estimando consumo do hypervisor

- Quais são os núcleos pinados?
  - Valor médio da diferença entre os núcleos agrupados n-n (janelas)
  - Ou os 'n' primeiros núcleos



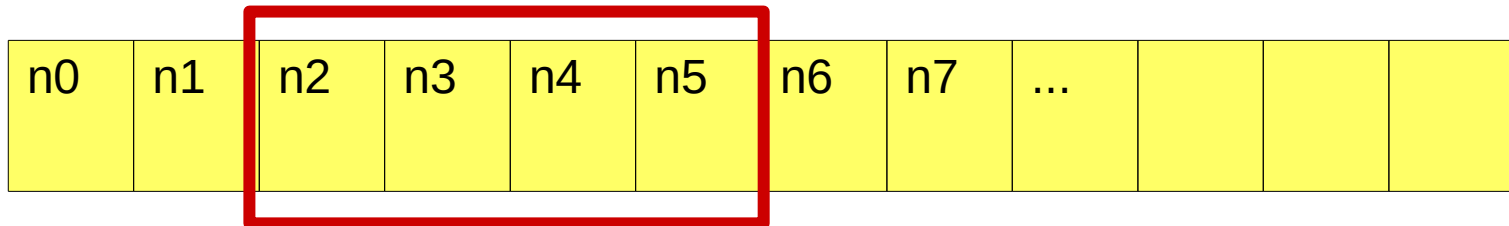
$$J0 = \text{sum}([n1-n0]+[n2-n1]+[n3-n2])$$

$$J0 = n3-n0$$



# Estimando consumo do hypervisor

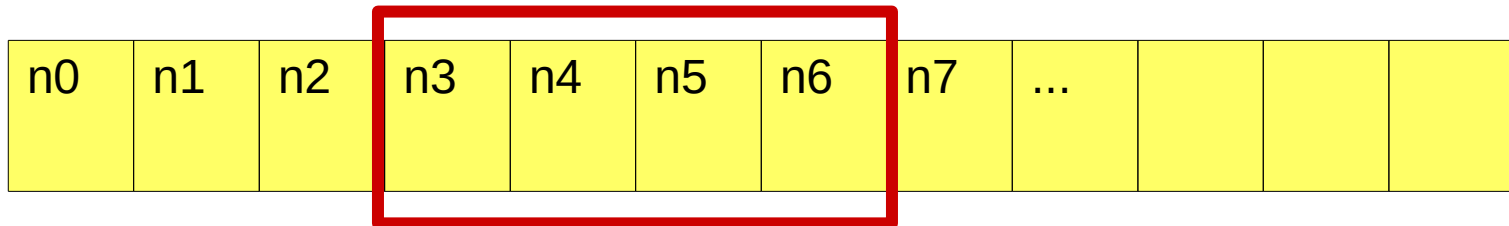
- Quais são os núcleos pinados?
  - Valor médio da diferença entre os núcleos agrupados n-n (janelas)
  - Ou os 'n' primeiros núcleos



$J1 = \text{sum}(\dots)$

# Estimando consumo do hypervisor

- Quais são os núcleos pinados?
  - Valor médio da diferença entre os núcleos agrupados n-n (janelas)
  - Ou os 'n' primeiros núcleos
  - Ao final é feita a **média dessas janelas**



$J2 = \text{sum}(\dots)$

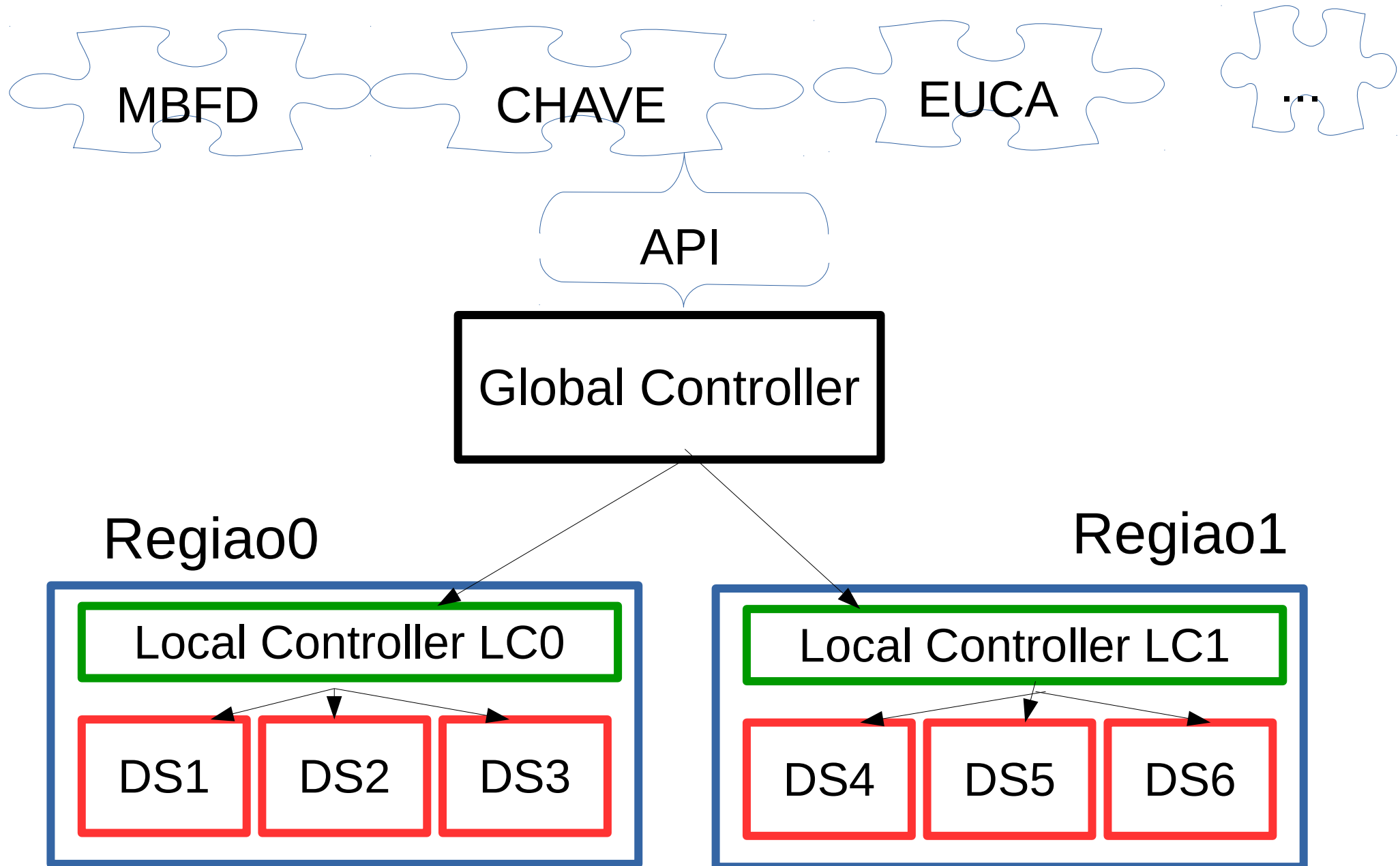
# Estimando o consumo das VMs

- $\text{ConsVM} = \text{ConsTotal} - \text{ConsHyperv}$ 
  - Lógico, não? Não sei!
- Varia da quantidade de núcleos em uso até o máximo de núcleos restante para as VMs
- Útil para quando é habilitado Overbooking..
- **Problemas:** *traces tem VMs com a mesma quantidade de núcleos físicos dos hosts, ie,* inviabiliza considerar CH quando overb. não habilitado

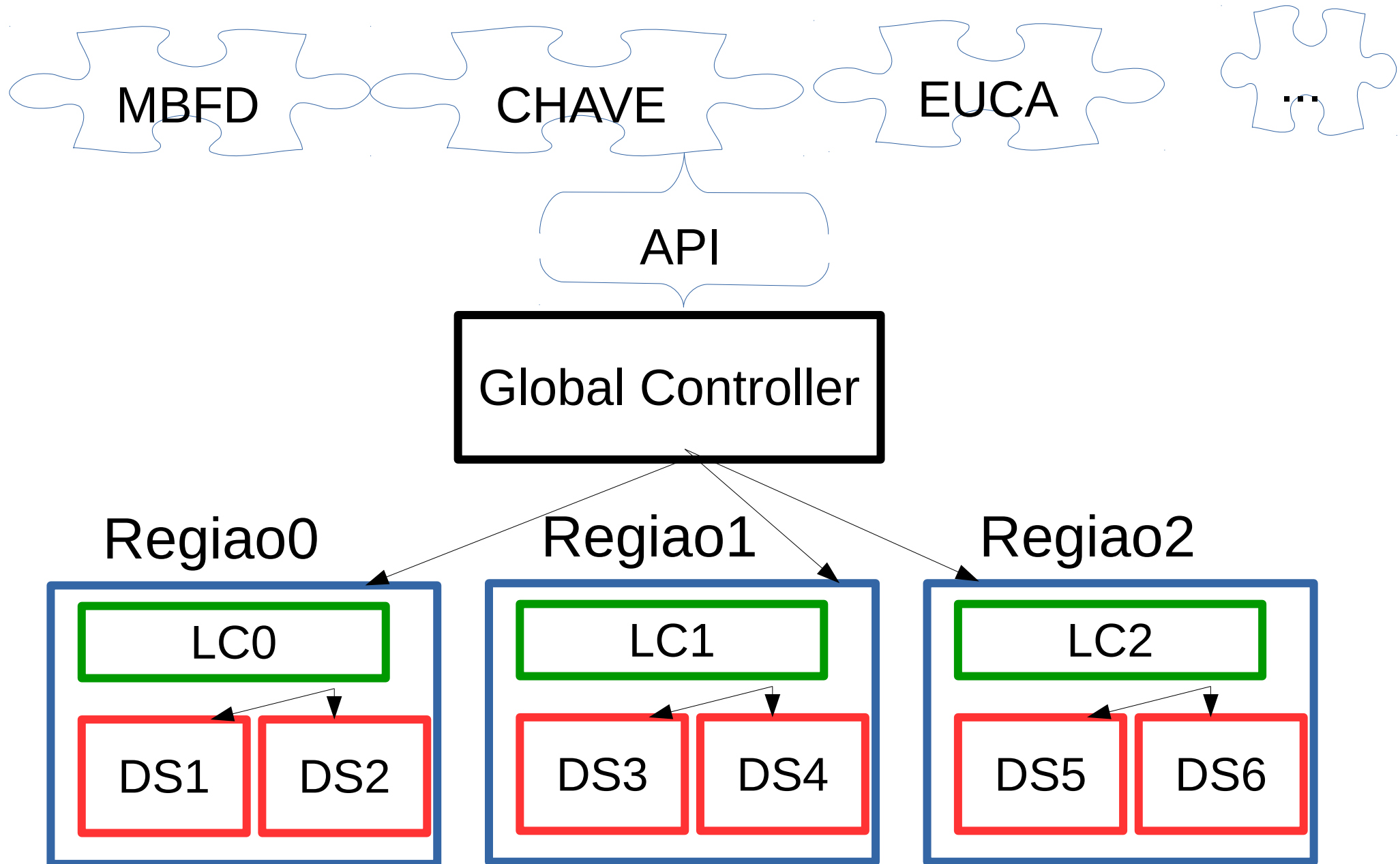
# O Simulador

- Totalmente parametrizado
- Operações nos objetos:
  - **SLA** = parâmetros e métricas
  - **Demanda** = recebe todos os arquivos → VMs
  - **AZs** = old 'datacenter.py' – Hosts, aloca, energia,...
  - **Local Controllers**
  - **Regiões** (opcional)
  - **Global Controller** (API)
  - **Algoritmos de escalonamento** (CHAVE, MBFD, etc)

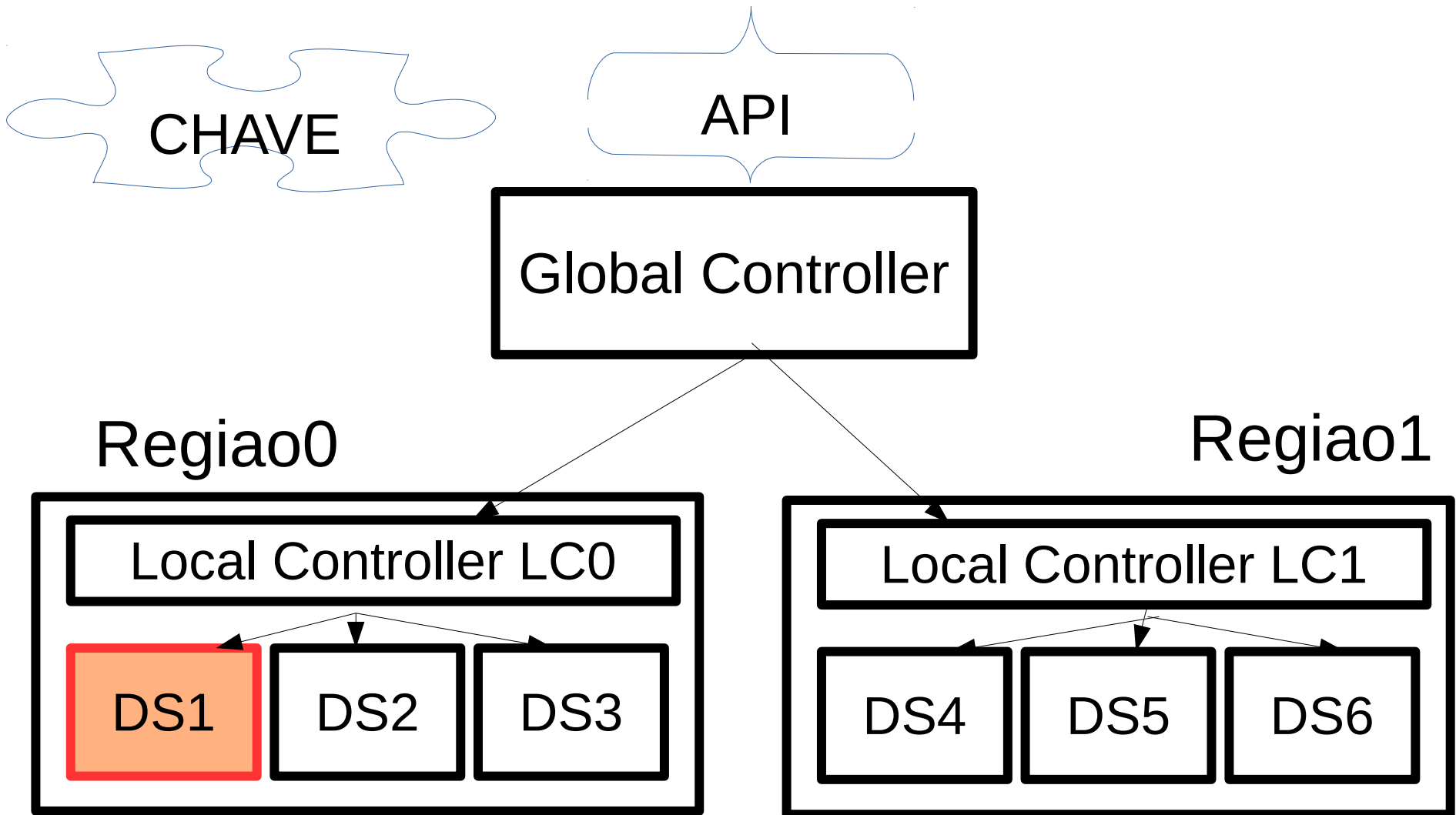
# Topologia usada no teste



# Poderia usar esta topologia no teste

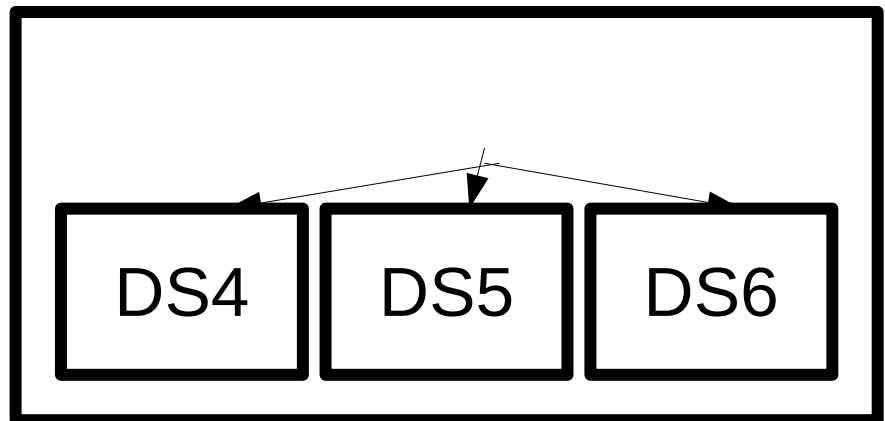
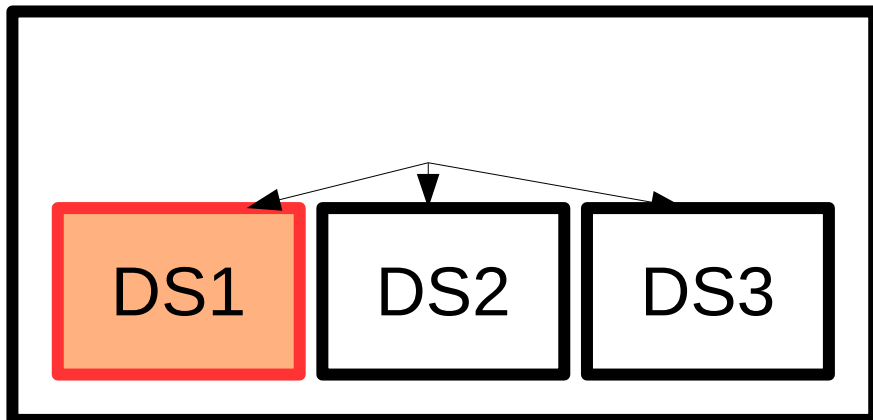


# CHAVE: Consolidação



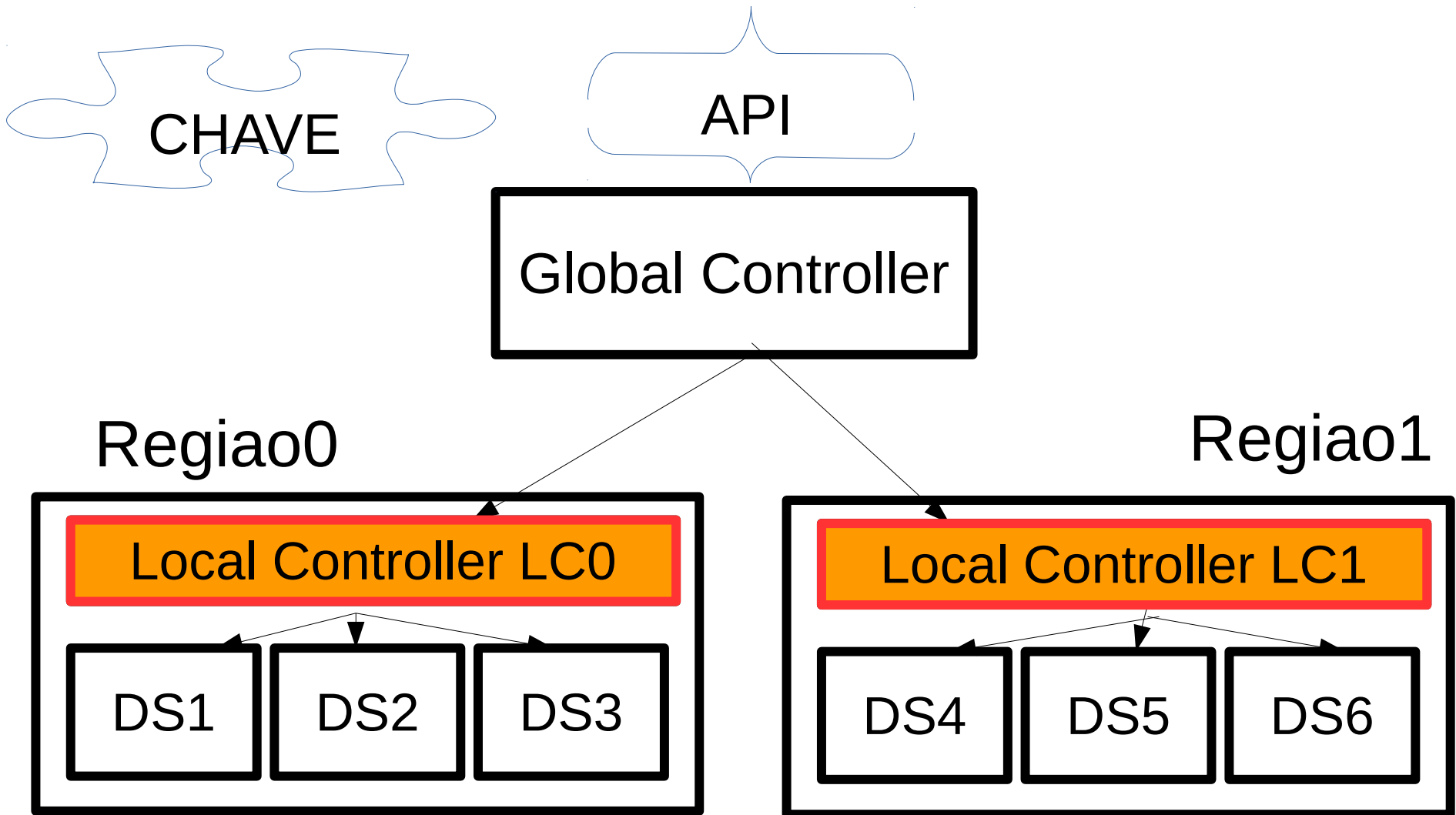
# CHAVE: Consolidação

- Multi-Thread -> execução em **paralelo**?
- Realizado placement otimizado por AZ
- A consolidação é feita por gatilho de **fragmentação (grau de dispersão)**
- Recebe as réplicas das outras AZs



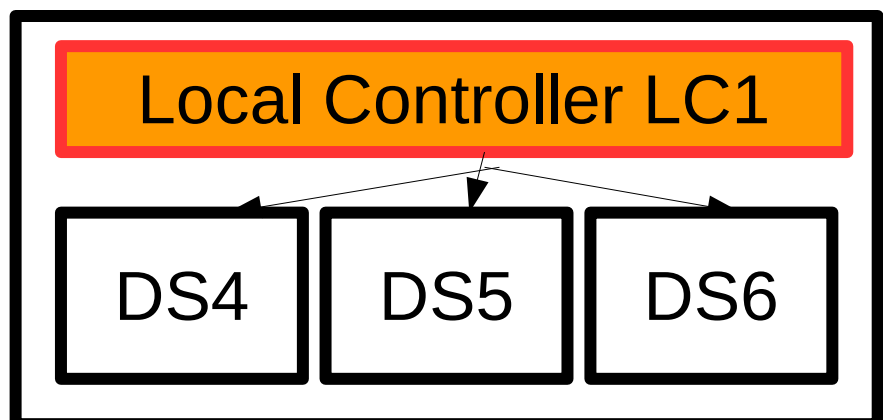
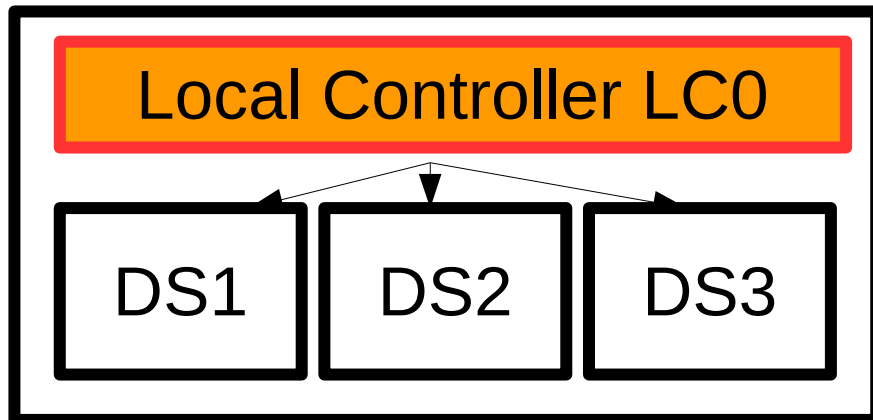


# CHAVE: Replicação



# CHAVE: Replicação

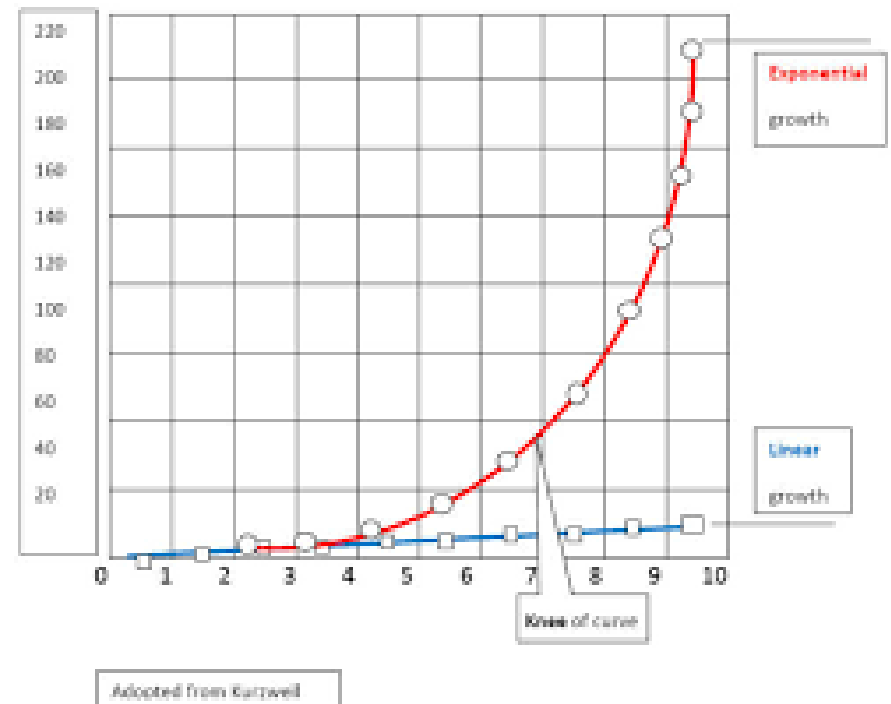
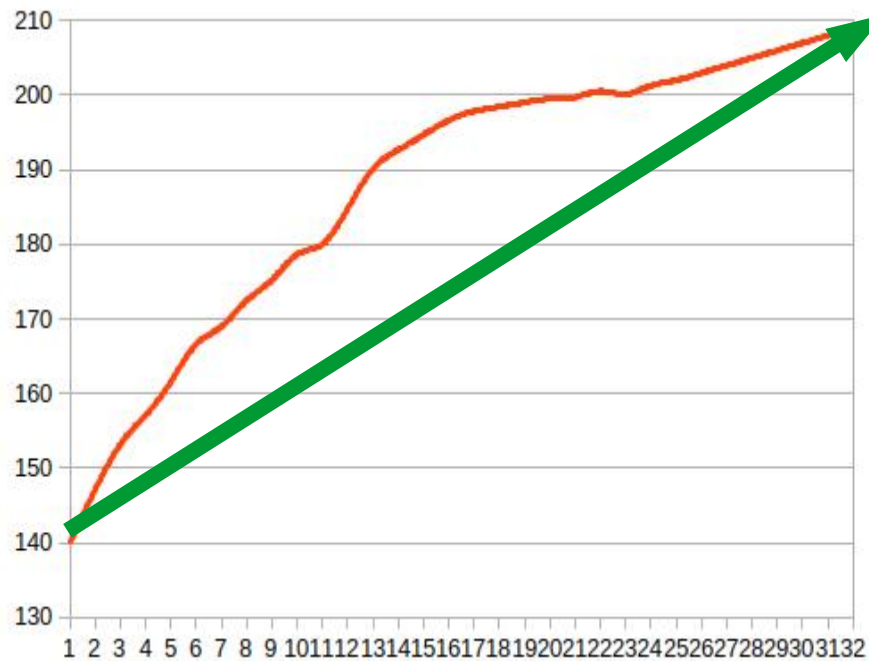
- Multi-thread -> execução em **paralelo**?
- AZ adiciona réplicas em um buffer compartilhado
- LCn Instancia as réplicas em outra AZ:
  - Seleção: Menor uso (LB), melhor disponib. (HA), aleatório, *add++*



# Medição do consumo de energia

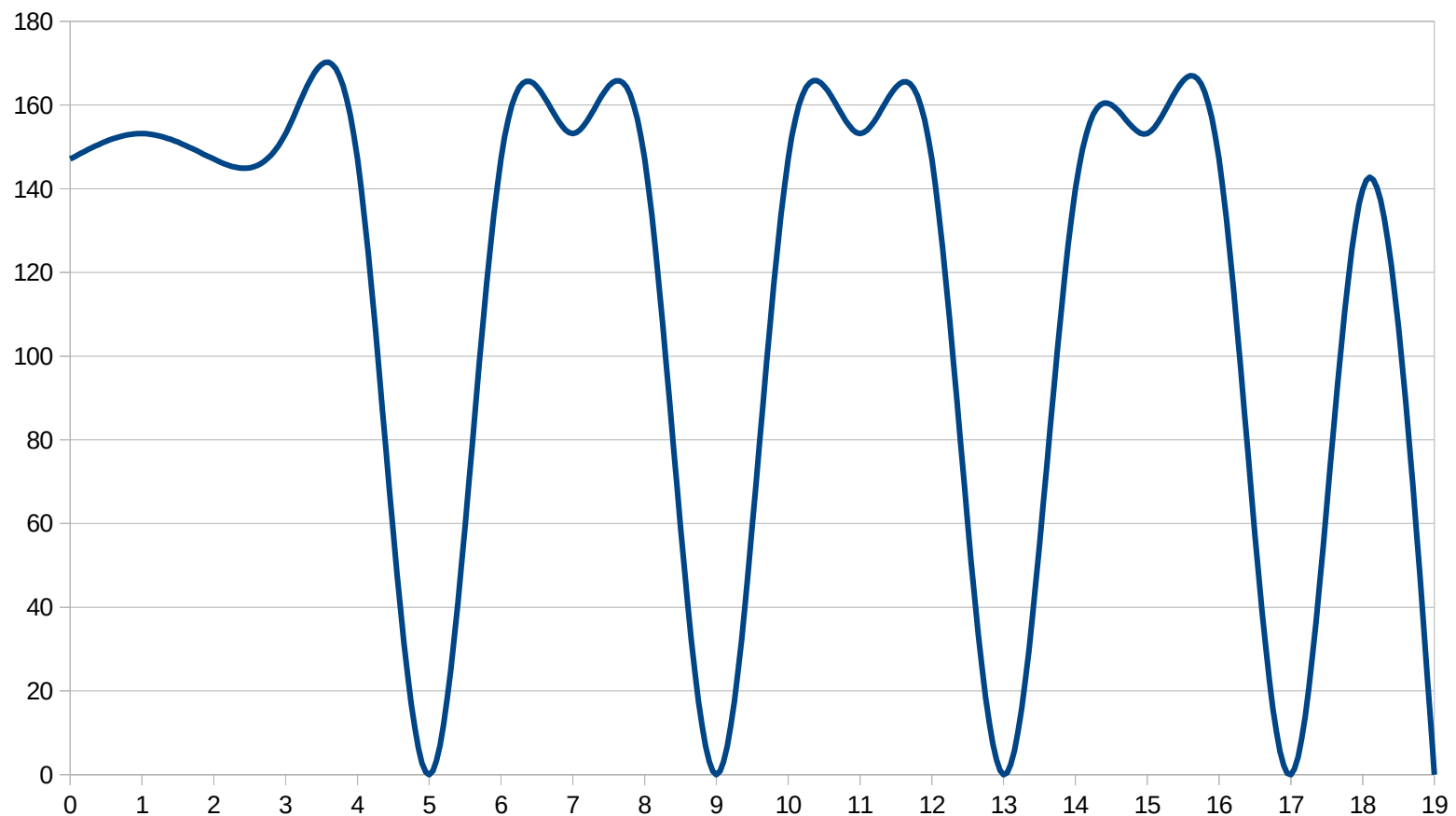
- Média do consumo em cada hora (para Wh)
  - Aproximação, mas não é o ideal
- EAVIRA?:
  - Global: soma/média durante o *'nit'*
  - Host:  $\text{ConsVMs} + \text{minEnergia} * p + \text{ConsGerenc} * p$ 
    - $P = (\text{CPUtotal} - \text{CPUvms}) / \text{CPUtotal}$
    - P seria um fator de aceleração? (*próximo slide*)
  - ConsVMs:  $\text{minEnergia} * p + \text{ConsGerenc} * p + \text{rede} + (\text{tabela}[\text{cpu}] - \text{minEnergia})$

- $P = (\text{CPU}_{\text{total}} - \text{CPU}_{\text{vms}}) / \text{CPU}_{\text{total}}$
- P seria um fator de aceleração linear/exponenc.?



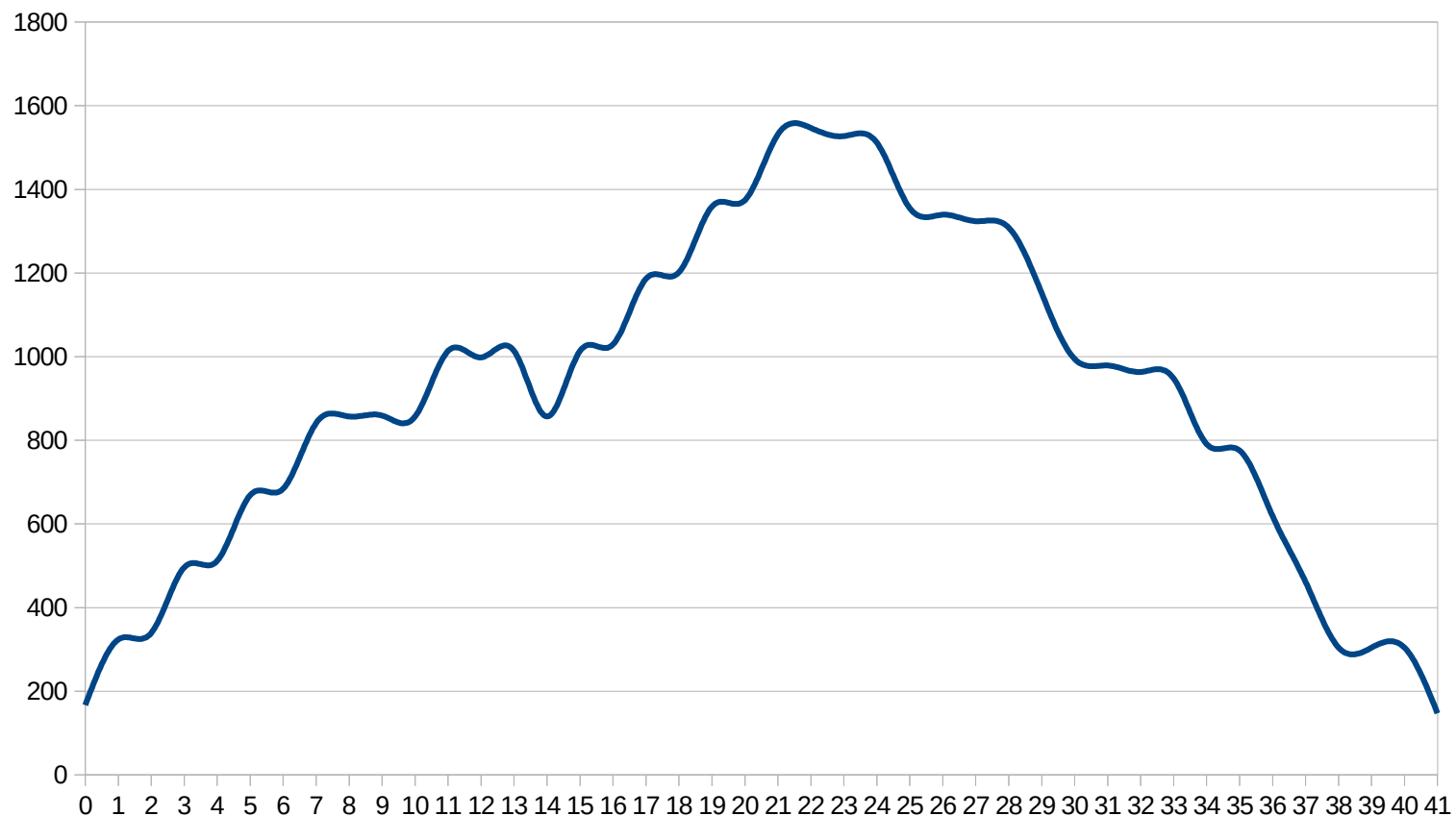
# Medição do consumo de energia

- Média do consumo em cada hora (para Wh)
  - Aproximação, mas não é o ideal

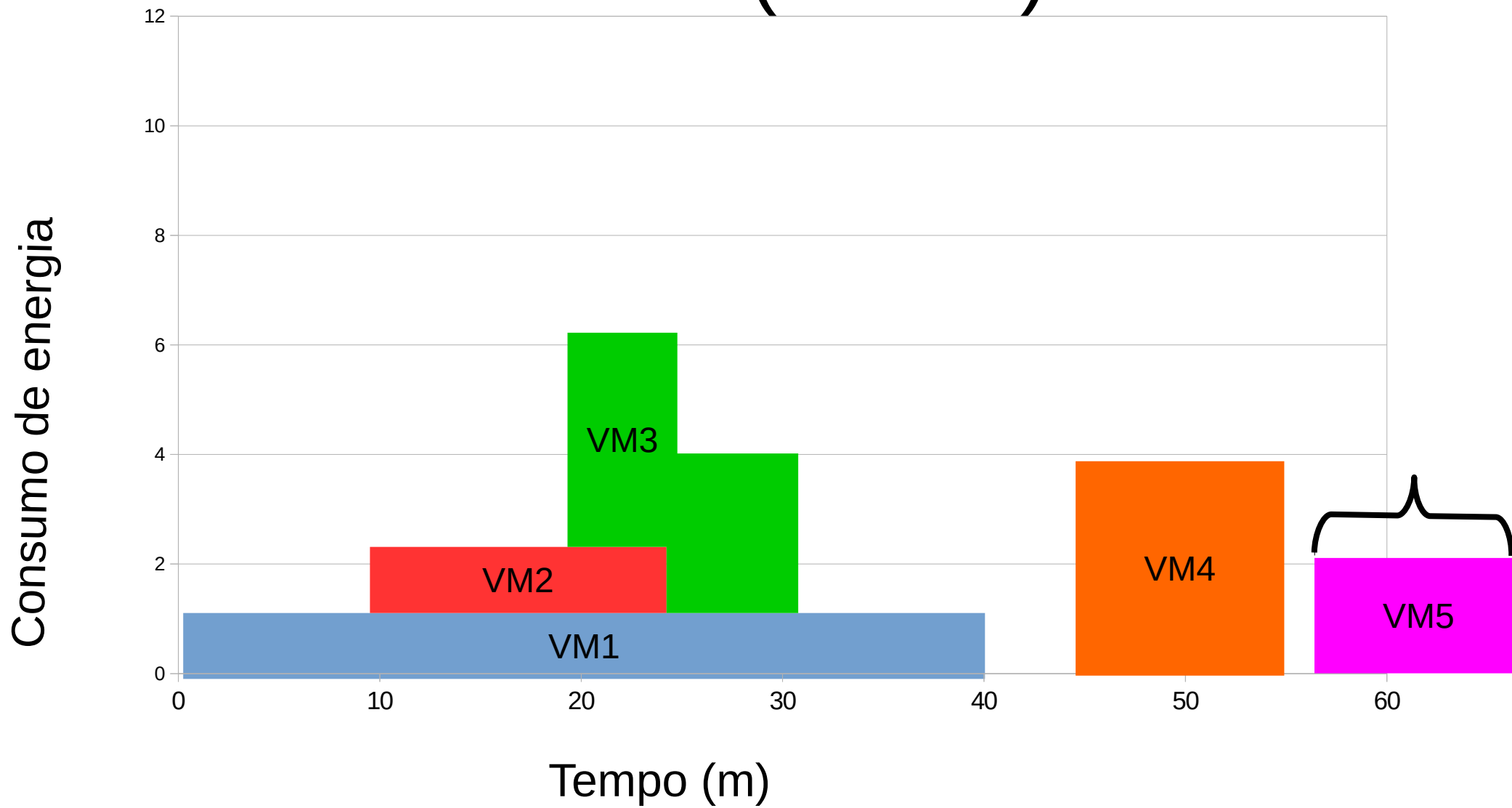


# Medição do consumo de energia

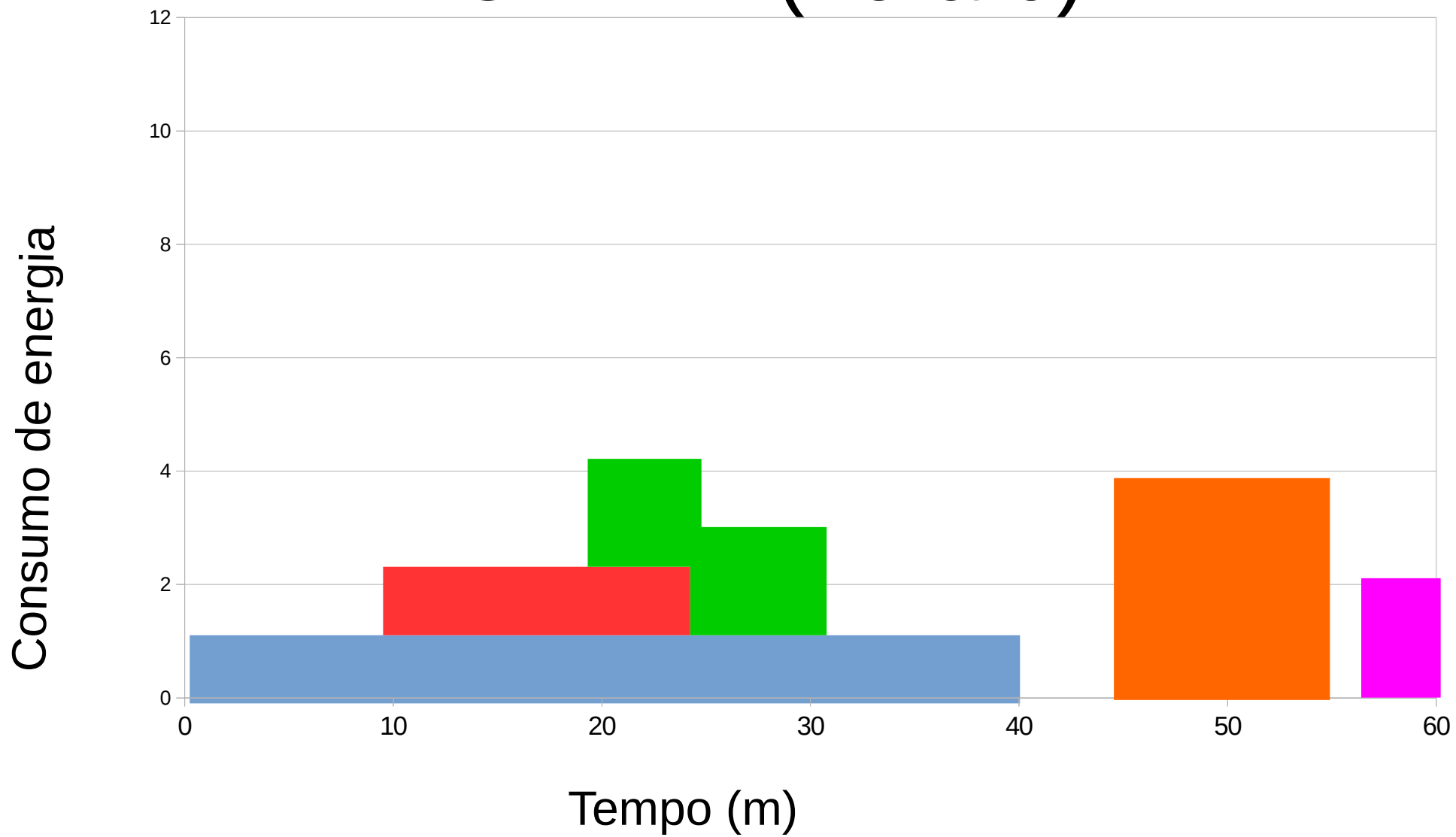
- Média do consumo em cada hora (para Wh)
  - Aproximação, mas não é o ideal



# CHAVE (hora 0)

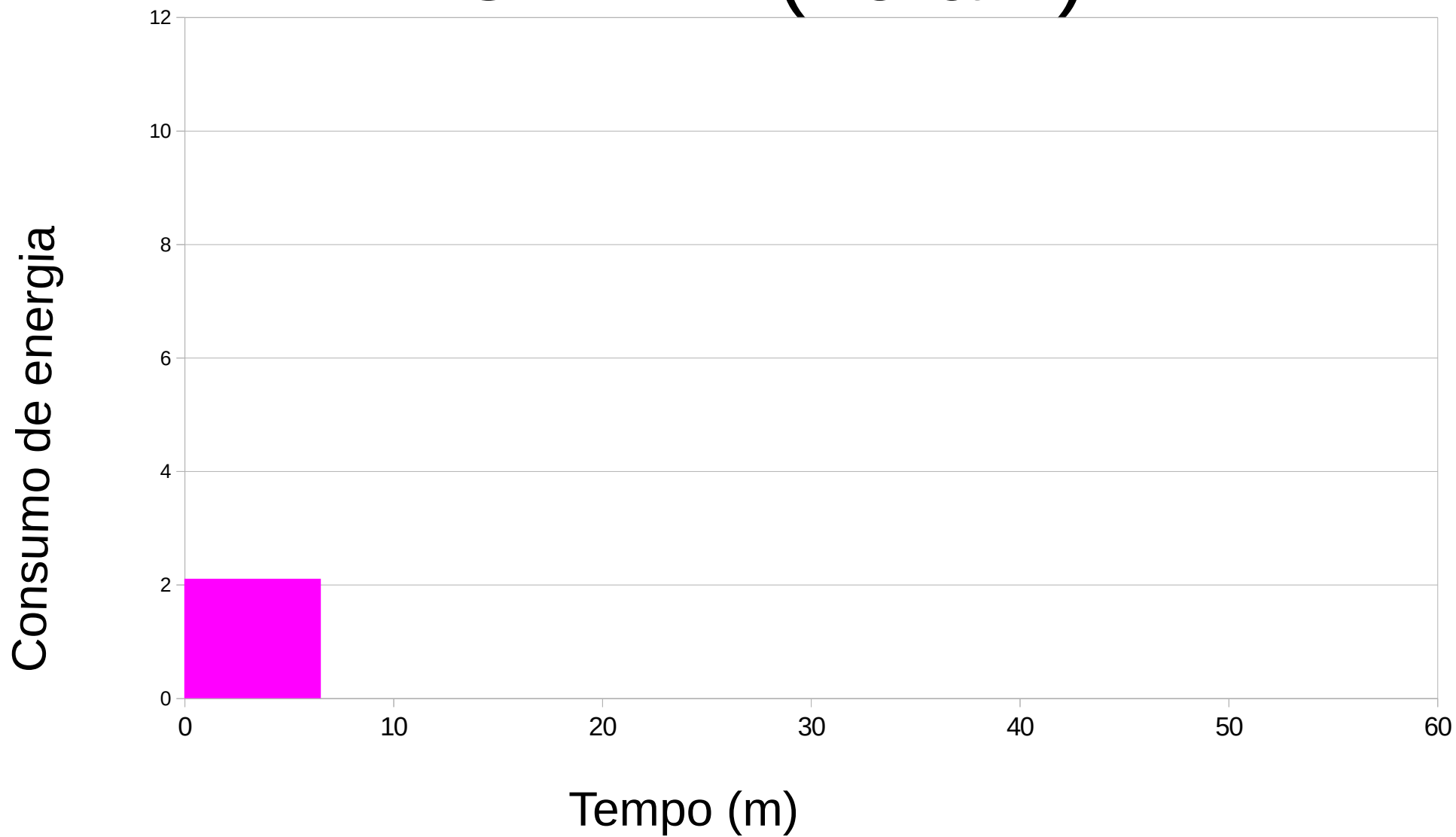


# CHAVE (hora 0)





# CHAVE (hora 1)



r		
Tempo (s)	Id ~> estado	Consumo
0	base	130? 0
0	v1~>E	140
10	v2~>E	155
20	v3~>E	215
25	V2~>S	200
30	v3~>S	140
40	v1~>S	130? 0
45	v4~>E	185
55	v4~>S	130? 0
57	v5~>E	155

Tabela CPU/Wh	
0	130
1	140
2	155
3	170
4	185
5	200
6	215

Histórico de uma hora		
T5_0	Vm5 ^^	c5
t5_f	vm5 vv	c5
t5	vm5++	c5
<b>t4'</b>	<b>vm4--</b>	<b>c4'</b>
t4	vm4++	c4
t-b	base	c0
<b>t1'</b>	<b>vm1--</b>	<b>c1'</b>
	<b>vm3--</b>	
t2'	vm3<<	c2'
<b>25</b>	<b>vm2--</b>	<b>c2'</b>
20	vm3++	c3
10	vm2++	c2
0	vm1++	c1
0	base	c0

# Chave: EnergyMonitor

- Cada host tem seu próprio monitor
- Algoritmo baseado em pilha:
  - Pilha com memória e pilha temporária
  - A cada **inserção** e **remoção** das MVs de um host, é modificada essa pilha
  - A cada hora (3600s) é calculado o consumo total:
    - Em WattsHora, considerando que o tempo é em segundos e que a tabela do Mauro está em WattsHora

# Características dos traces pelo artigo de Zhihao Yao

Trace	Duration (days)	VM created	Host	Cores/Host
DS1	85	9173	13	24
DS2	283	900	7	12
DS3	279	2600	7	8
DS4	94	1436	12	8
DS5	33	8456	31	32
DS6	34	4157	31	32

# Características dos traces

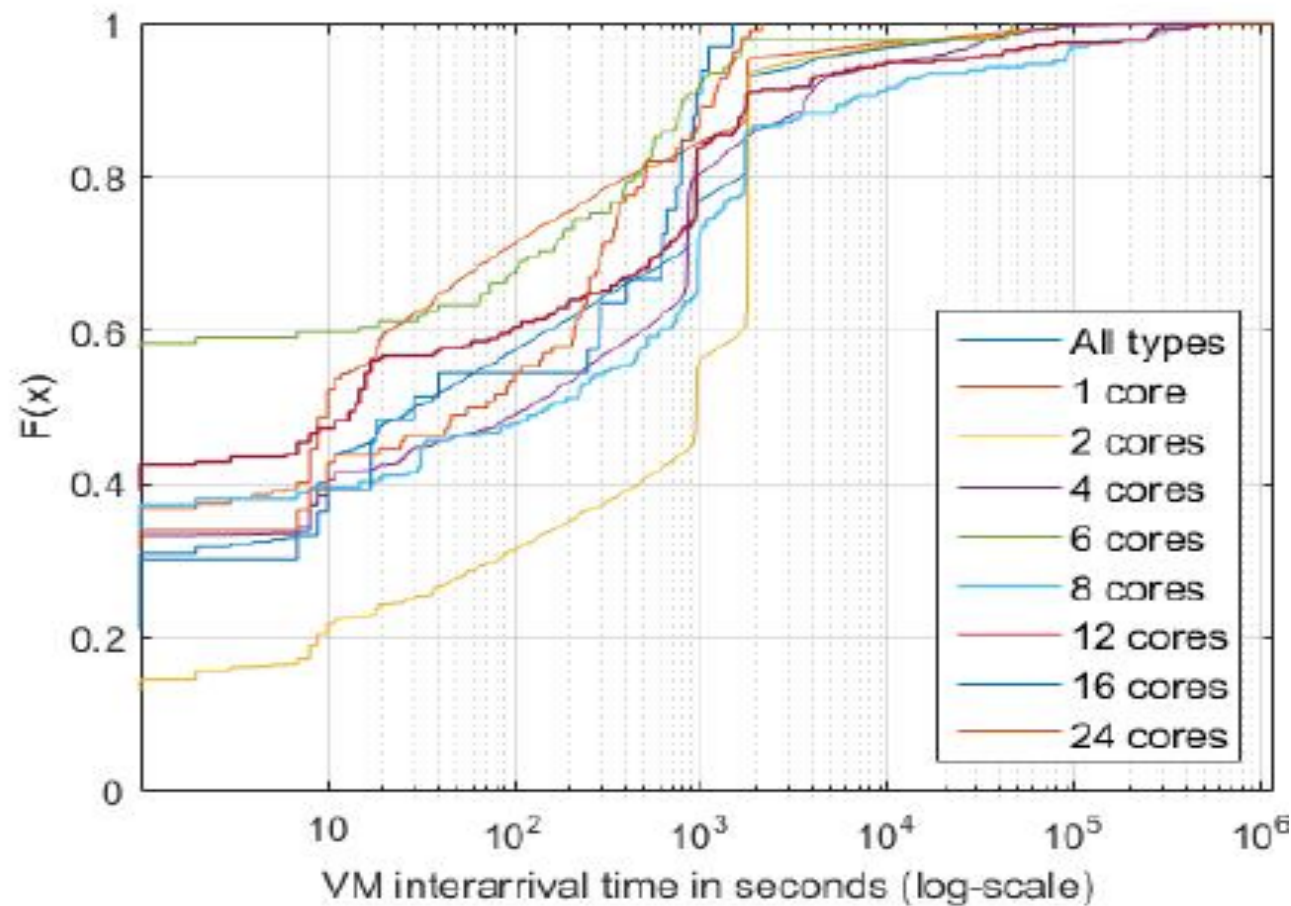


Fig. 2. CDF of VM inter-arrival time of Eucalyptus traces

# Características dos traces

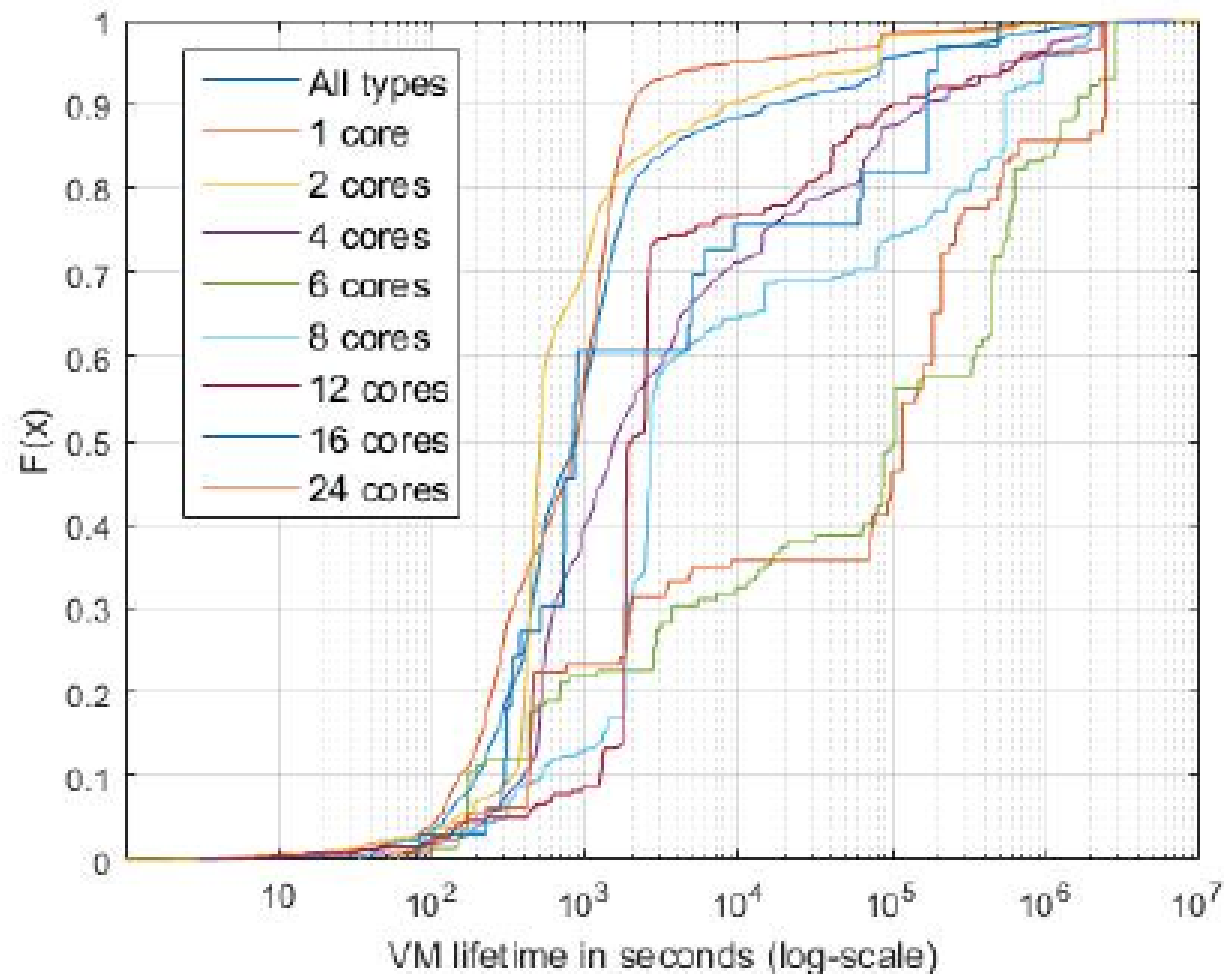
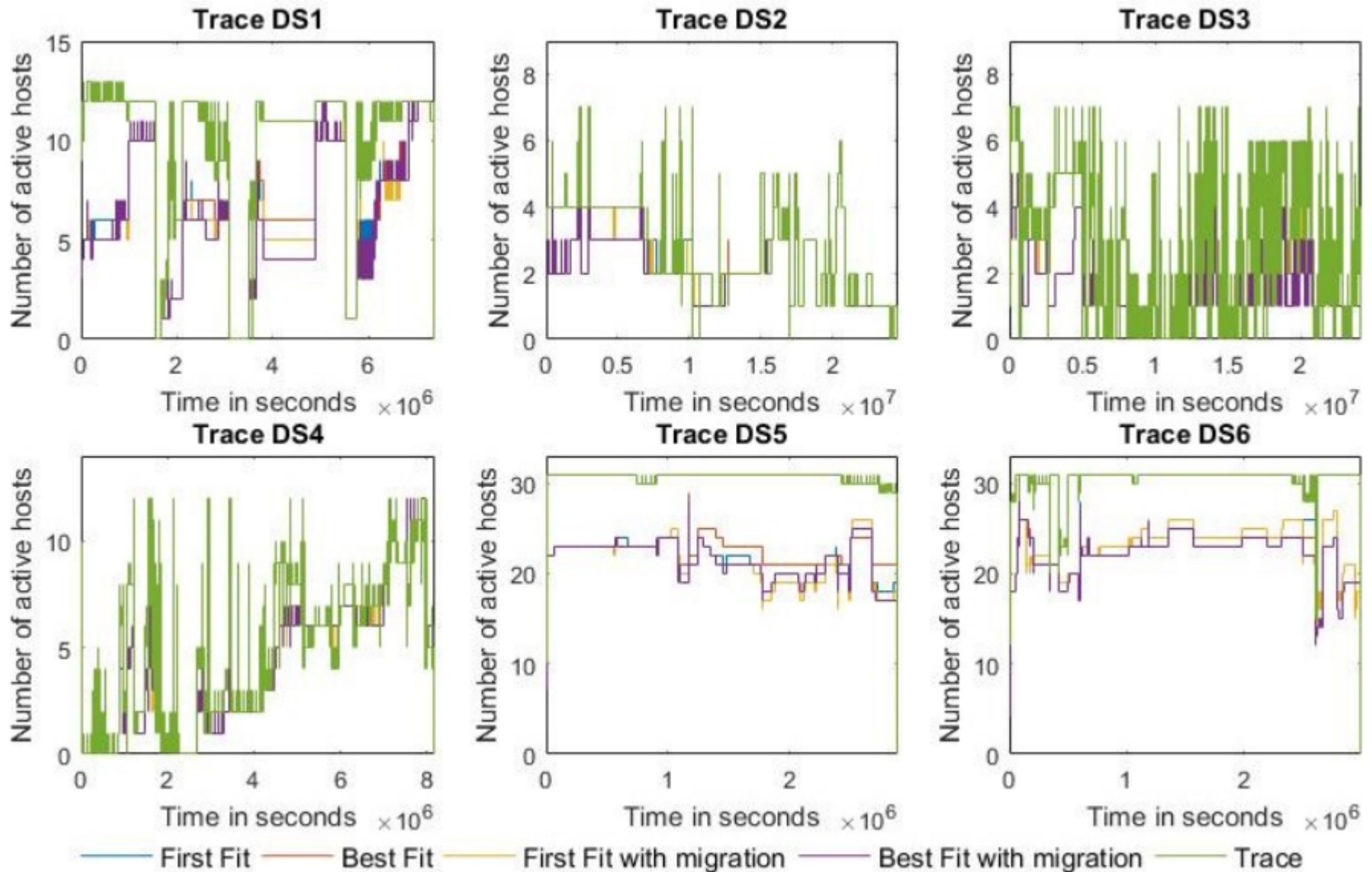


Fig. 3. CDF plot of VM life time in six traces

# Características dos traces



# Entrada de dados

- Se janela de tempo (WT) é 1
  - Tempo real, mas o 'decreasing' do FFD não aplica
  - Apenas a ordem dos hosts