



# Energies renouvelables et centres de données

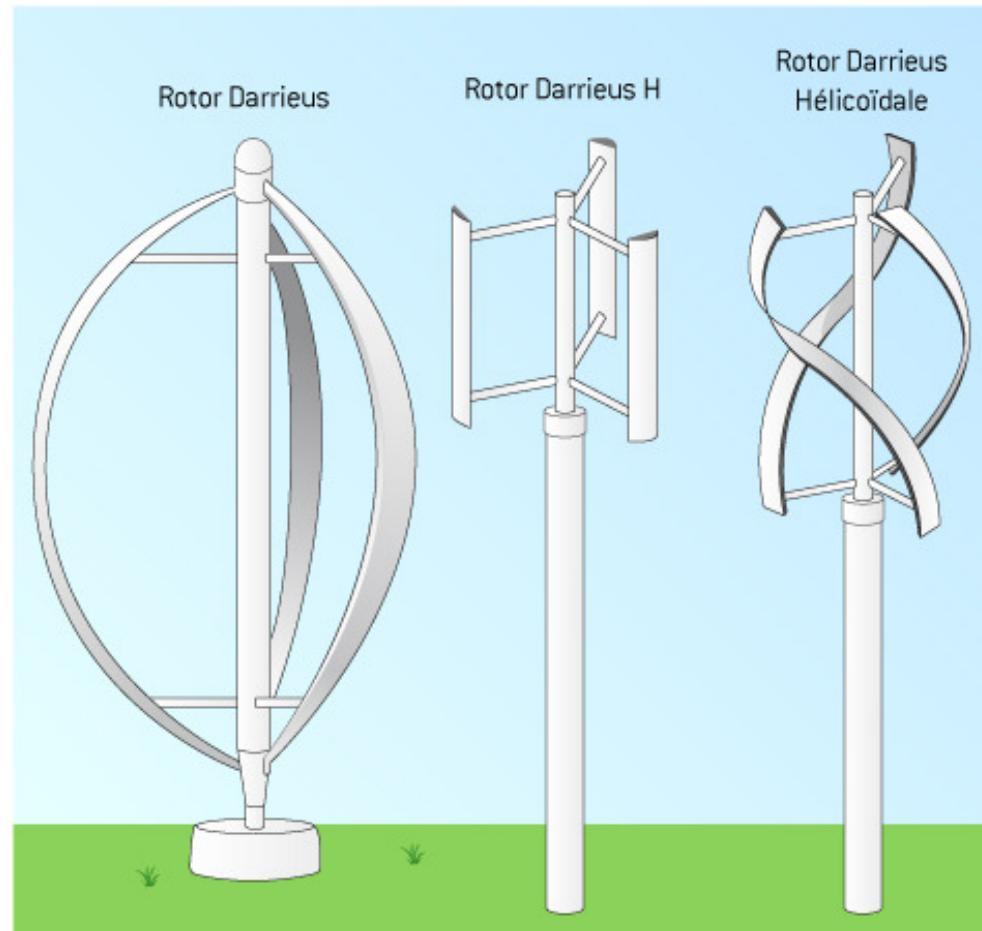
*Jean-Marc Menaud*

*ASCOLA Mines Nantes, INRIA, LINA*

**Ecole Jeunes Chercheurs sur  
l’Efficacité Energétique des  
Réseaux et Systèmes Distribués**

26 Mai 2016  
Mines Nantes

## Eolienne Darrieus





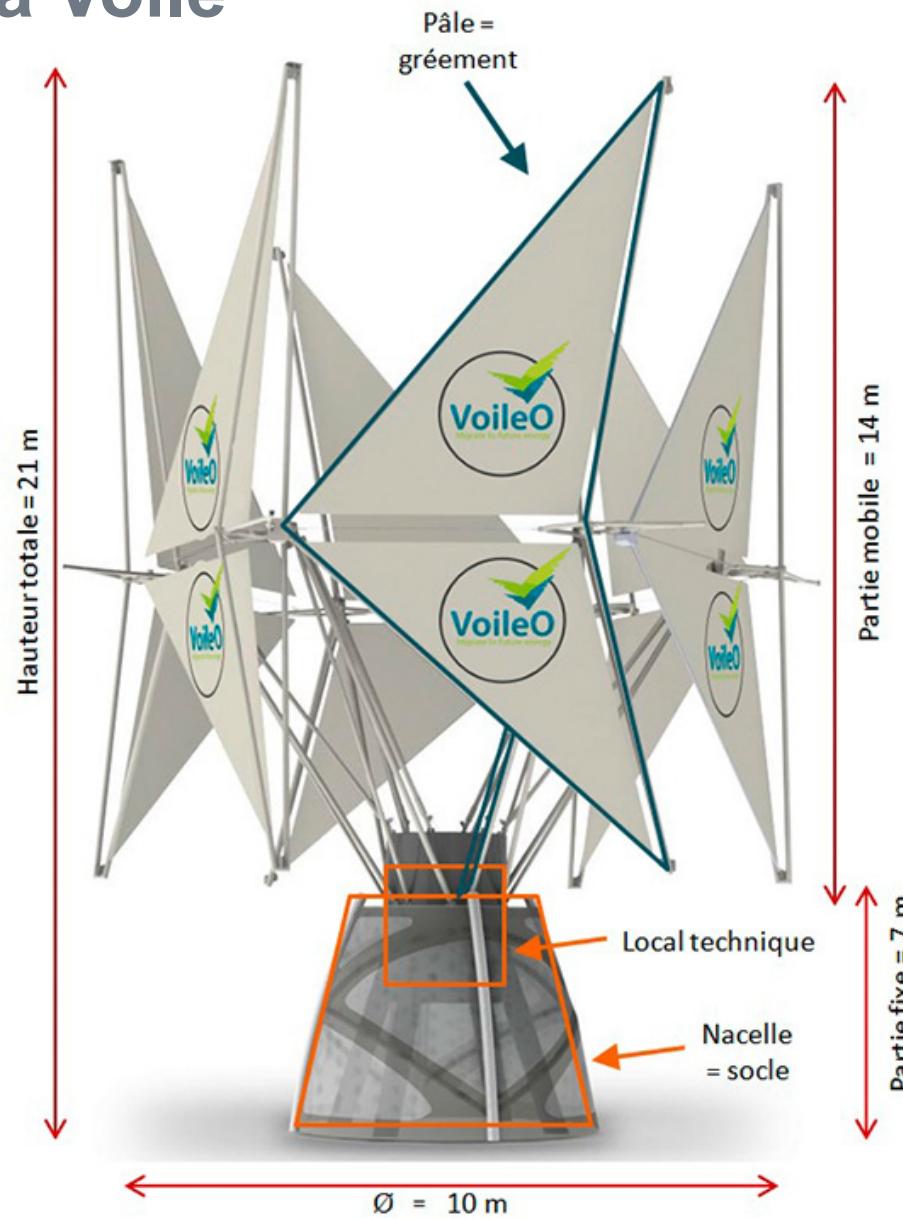
## Pourquoi une éolienne à axe vertical?



- Un rendement inédit, jusqu'à deux fois plus que des éoliennes traditionnelles de taille comparable
- Pas de permis de construire ni étude d'impacts (<12m)
- Très silencieuse
- Omnidirectionnelle, fonctionne en vent turbulent
- Sécurité: homologation bureau Veritas

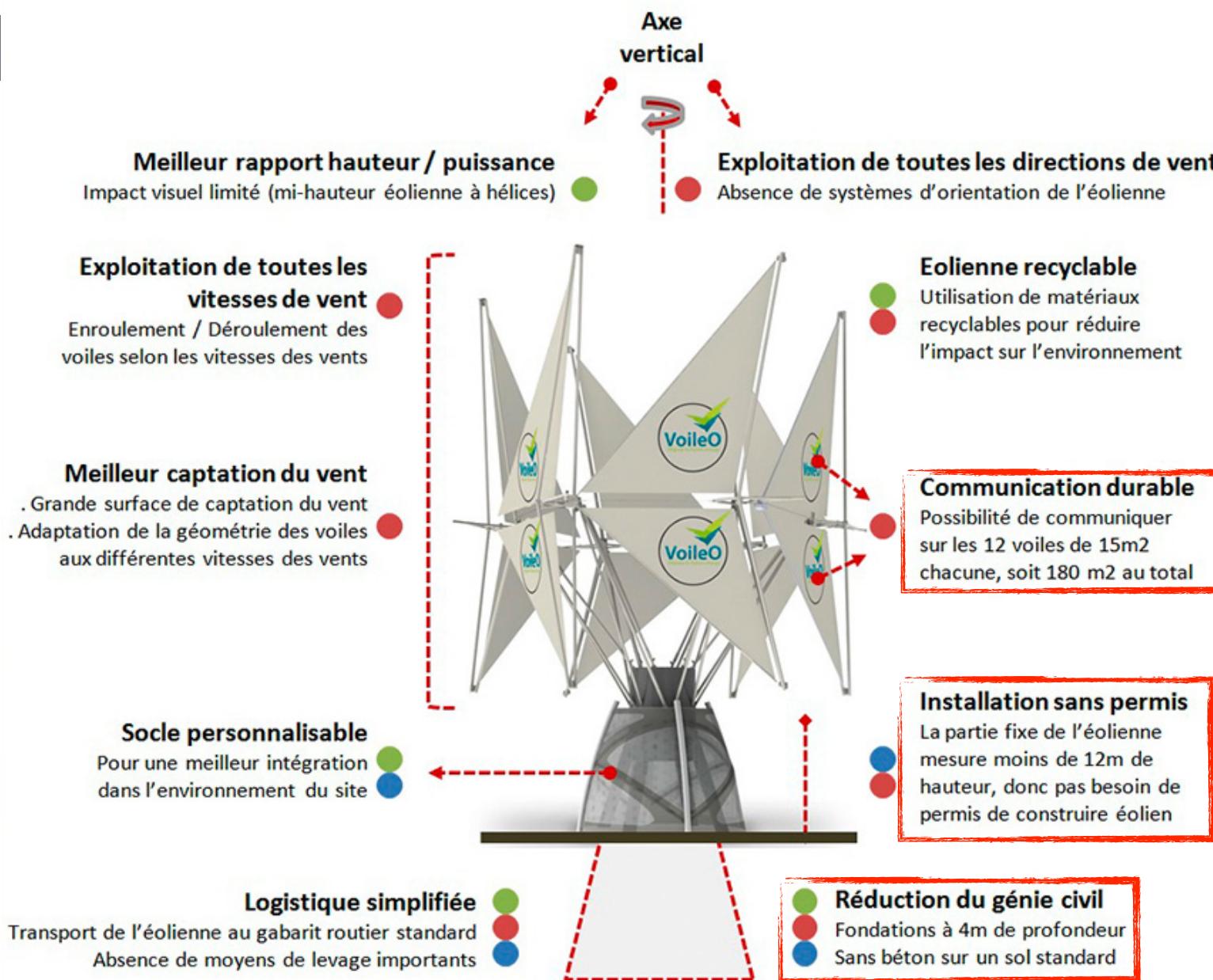


# à Voile



Caractéristiques	Valeur
Puissance crête (kW)	75
Diamètre du rotor (m)	10
Hauteur du socle (m)	7
Hauteur totale (m)	21
Nombre de voiles	12
Surface d'une voile (m <sup>2</sup> )	15
Tension en nominale (V)	400
Vitesse de démarrage (m/s)	1,5

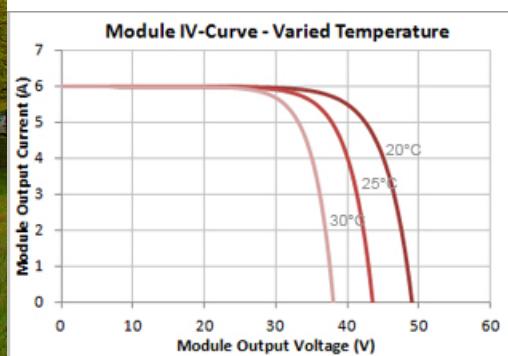
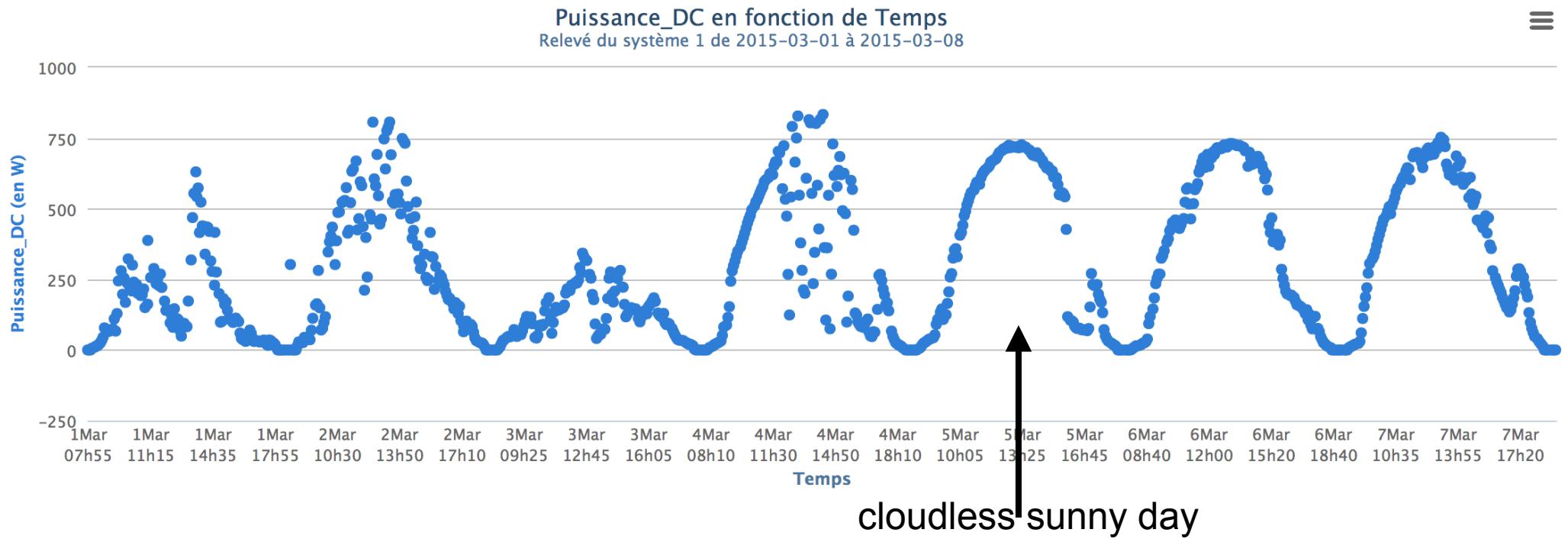
# VoileO, une éolienne résolument orientée Développement Durable



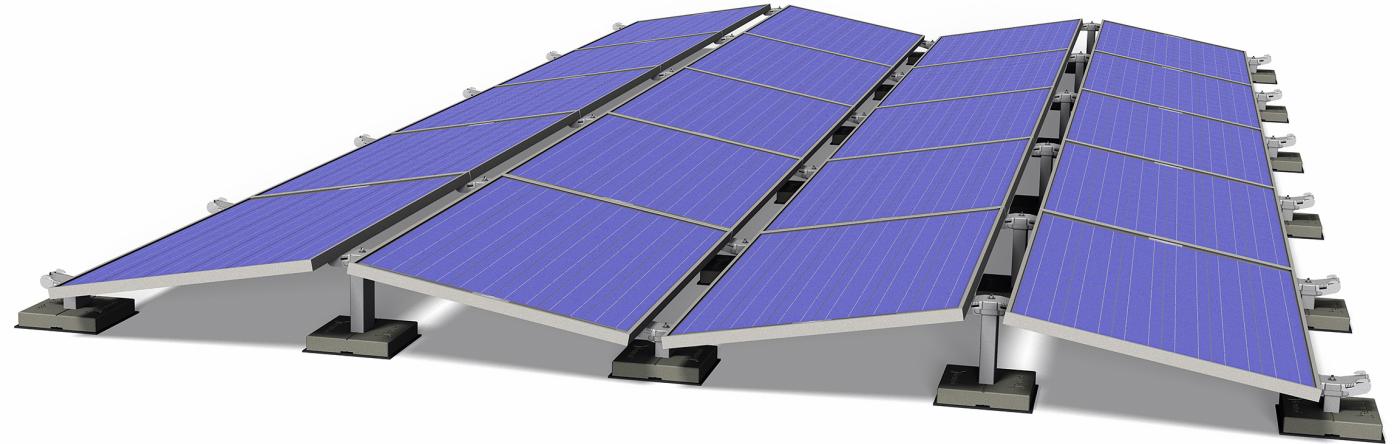


## ■ Panneaux photovoltaïques

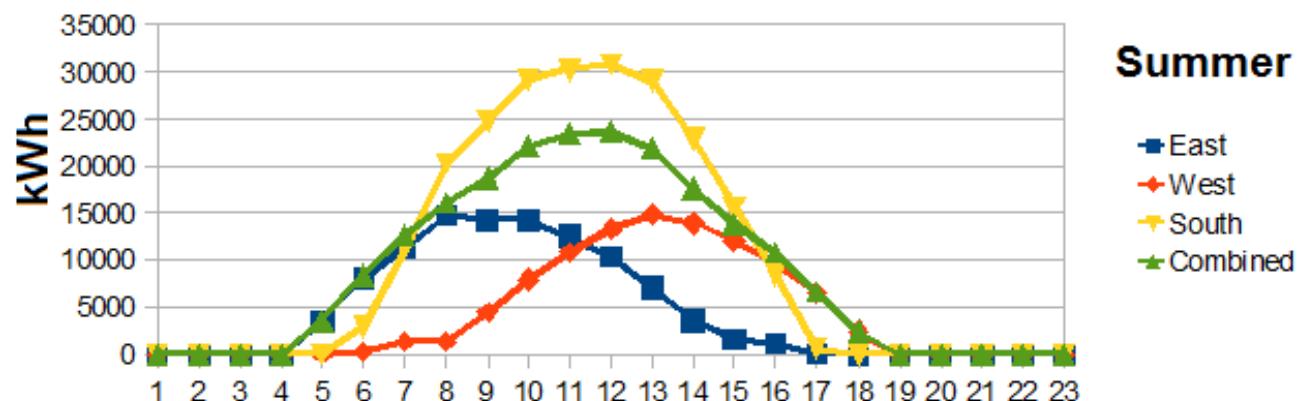
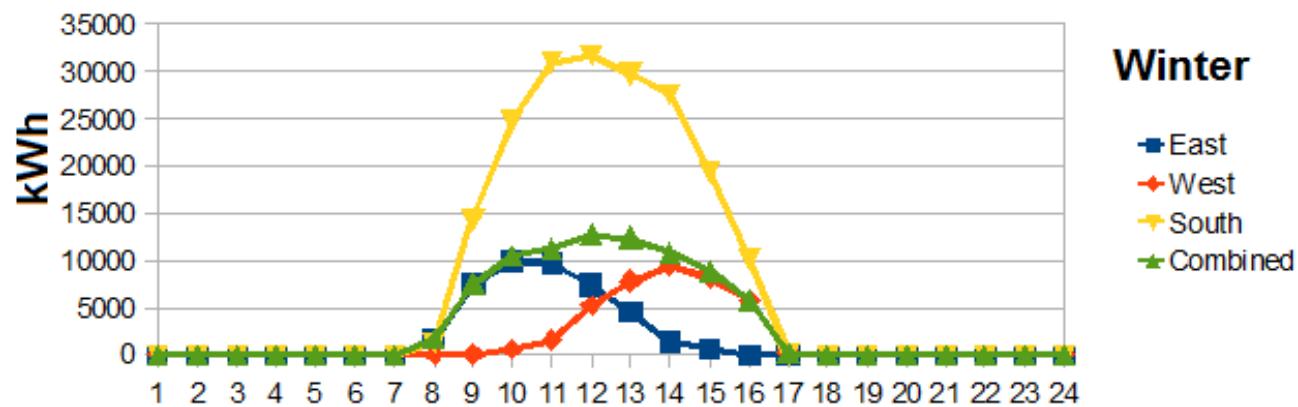
# Energy Production : photovoltaic panel real traces

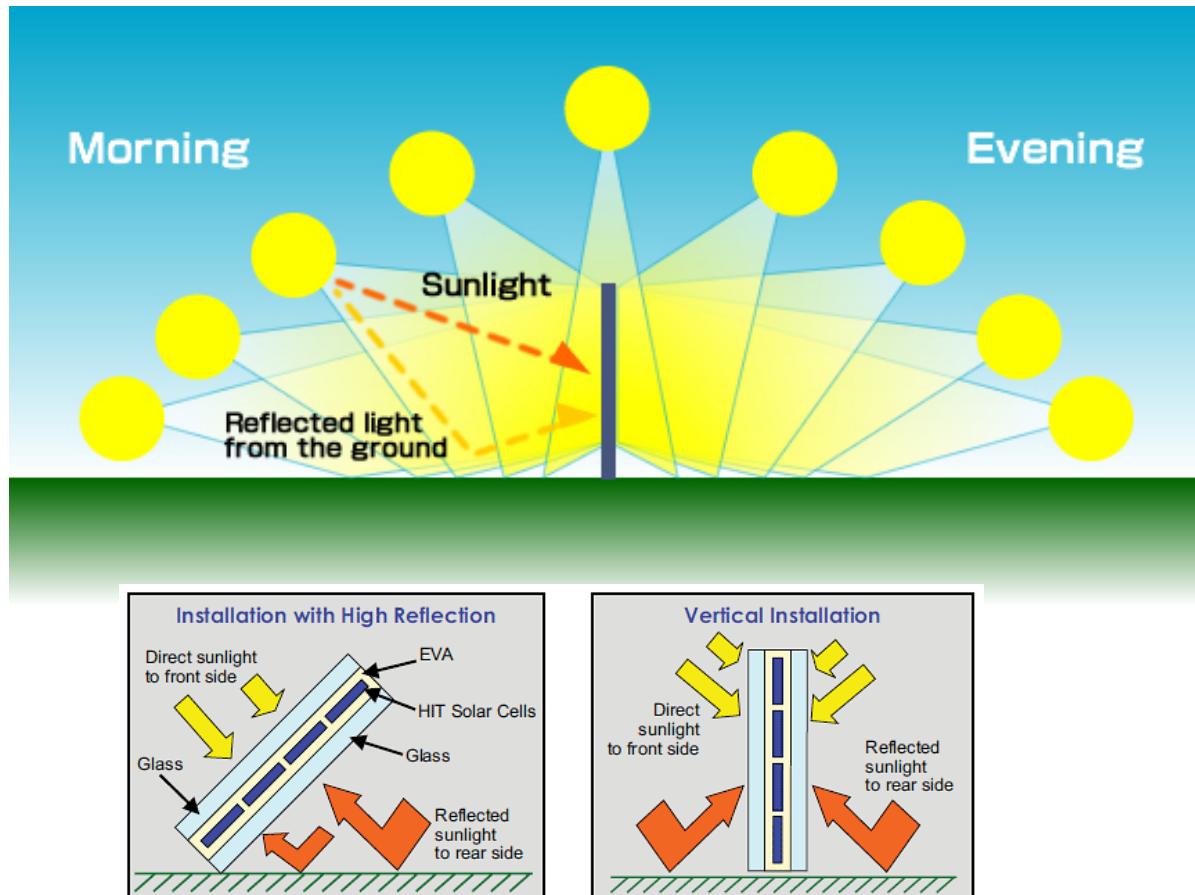


Peaks  
Production are  
more important  
in winter !



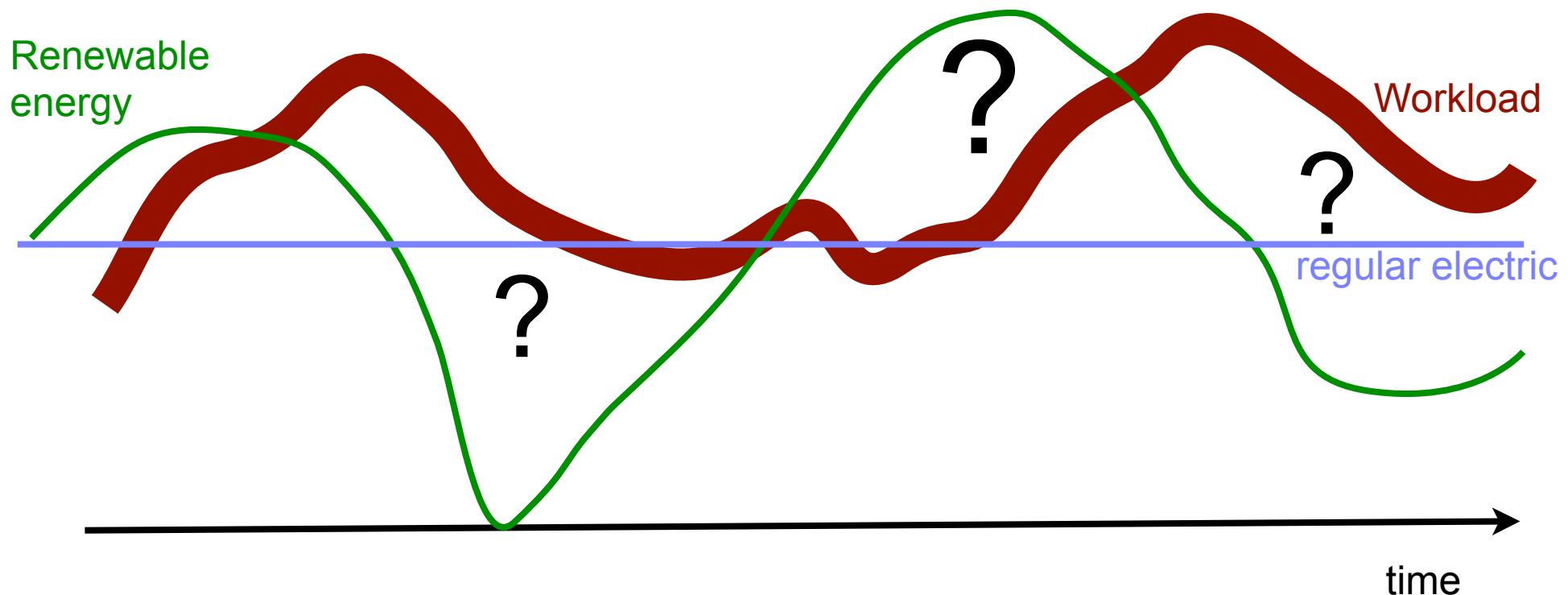
### Comparison between south facing roof and east-west roof

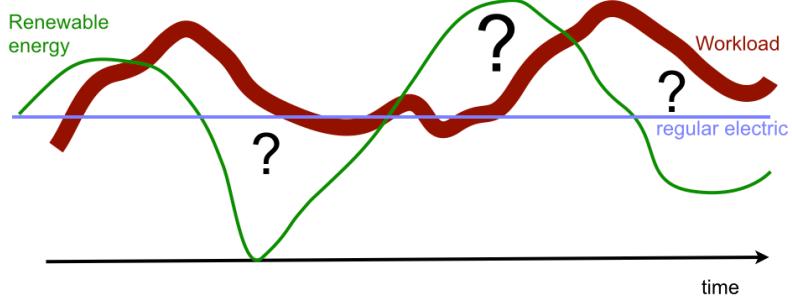






# Workload Traces

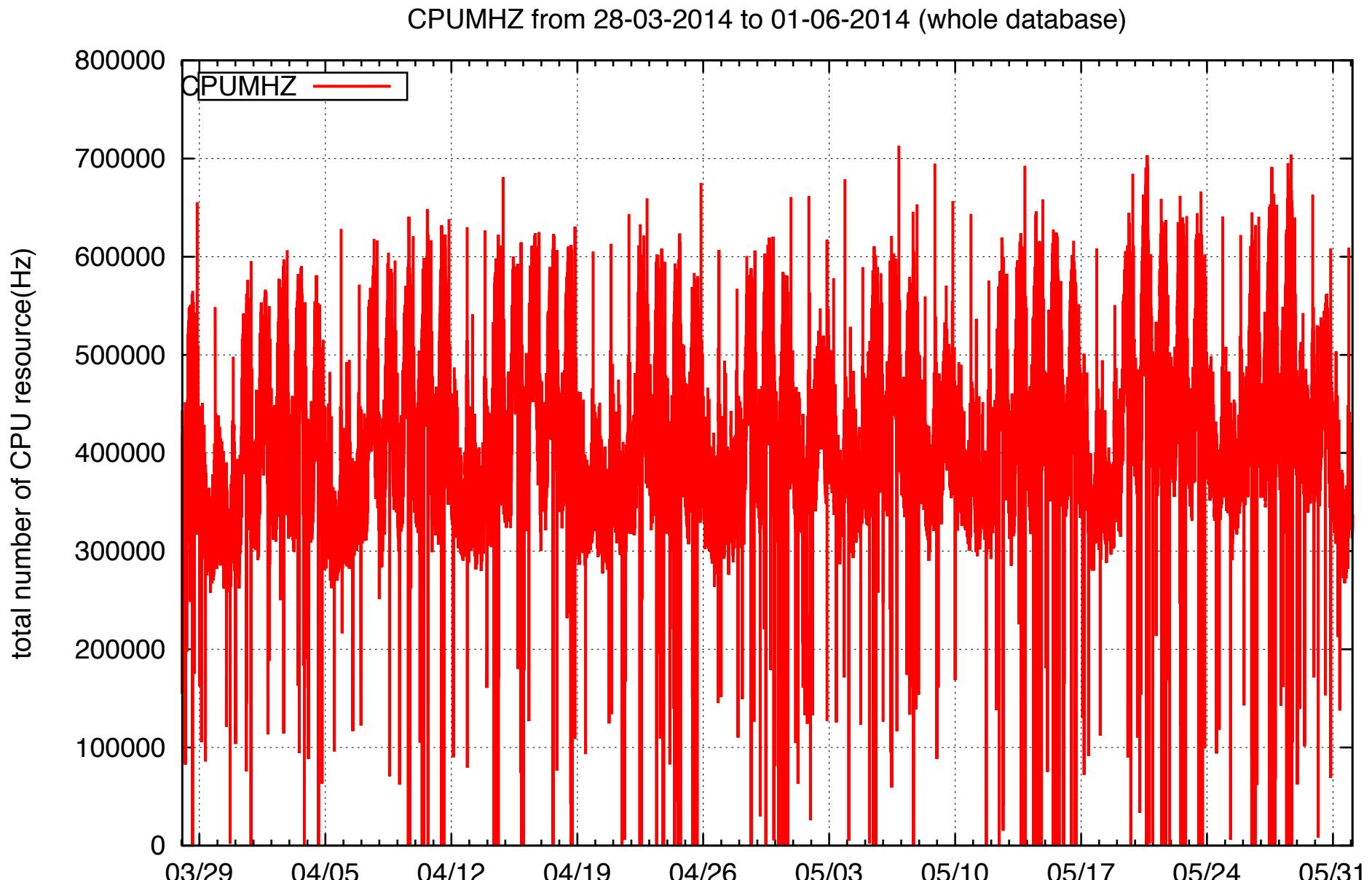




## Traces d'un centre de données

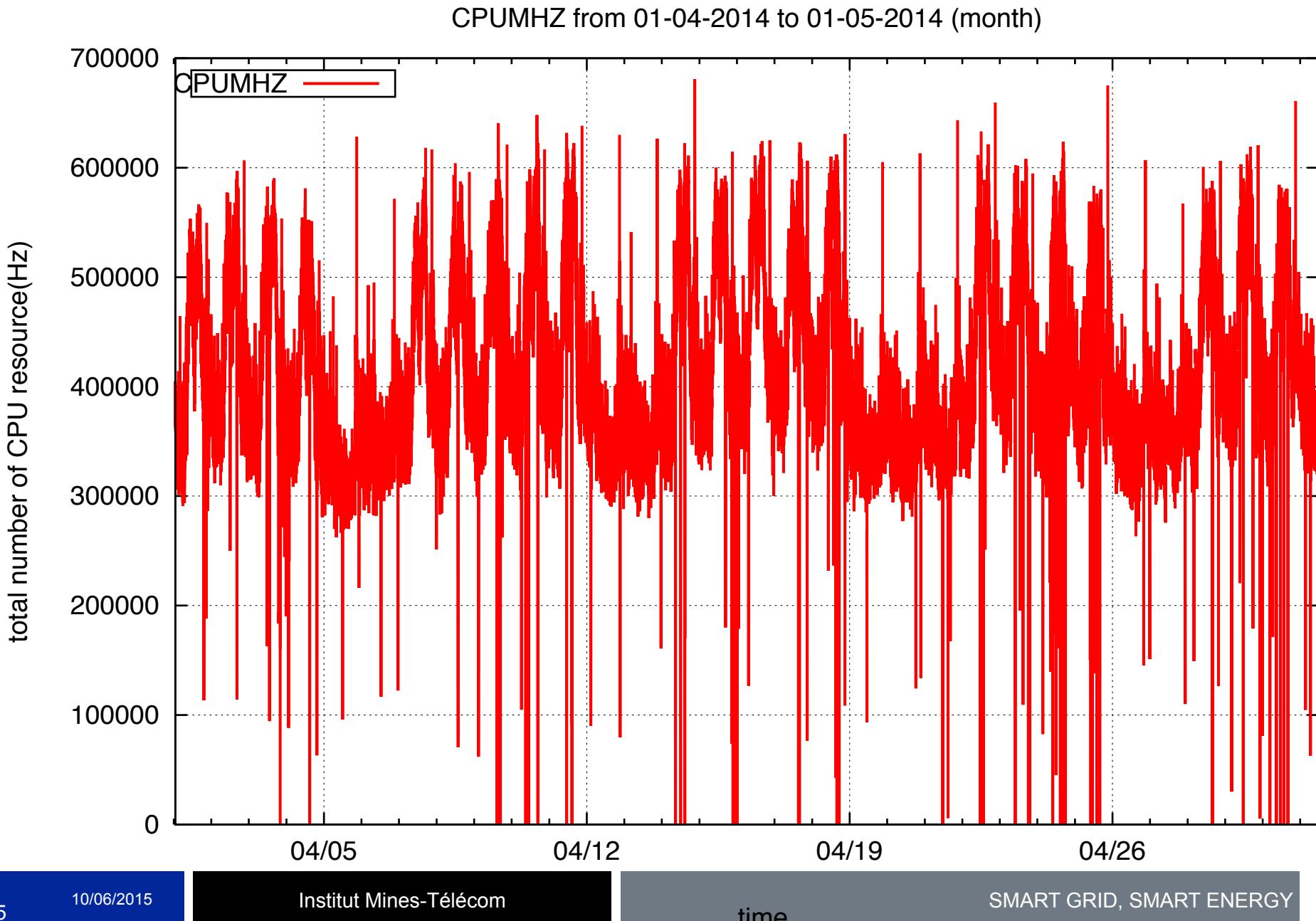
# Virtual Machines workload : Data center real traces

2 Months



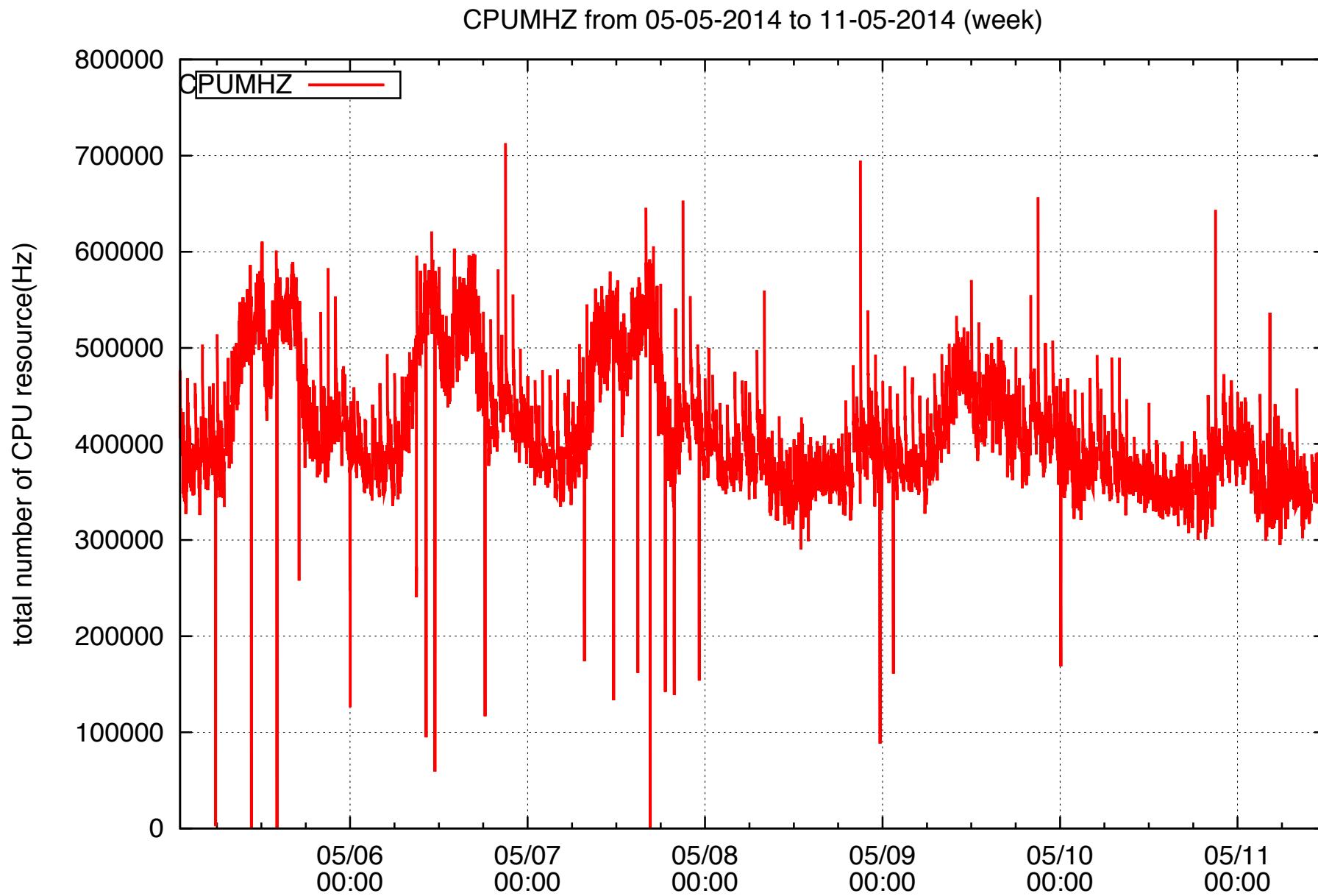
# Virtual Machines workload : Data center real traces

1 Month



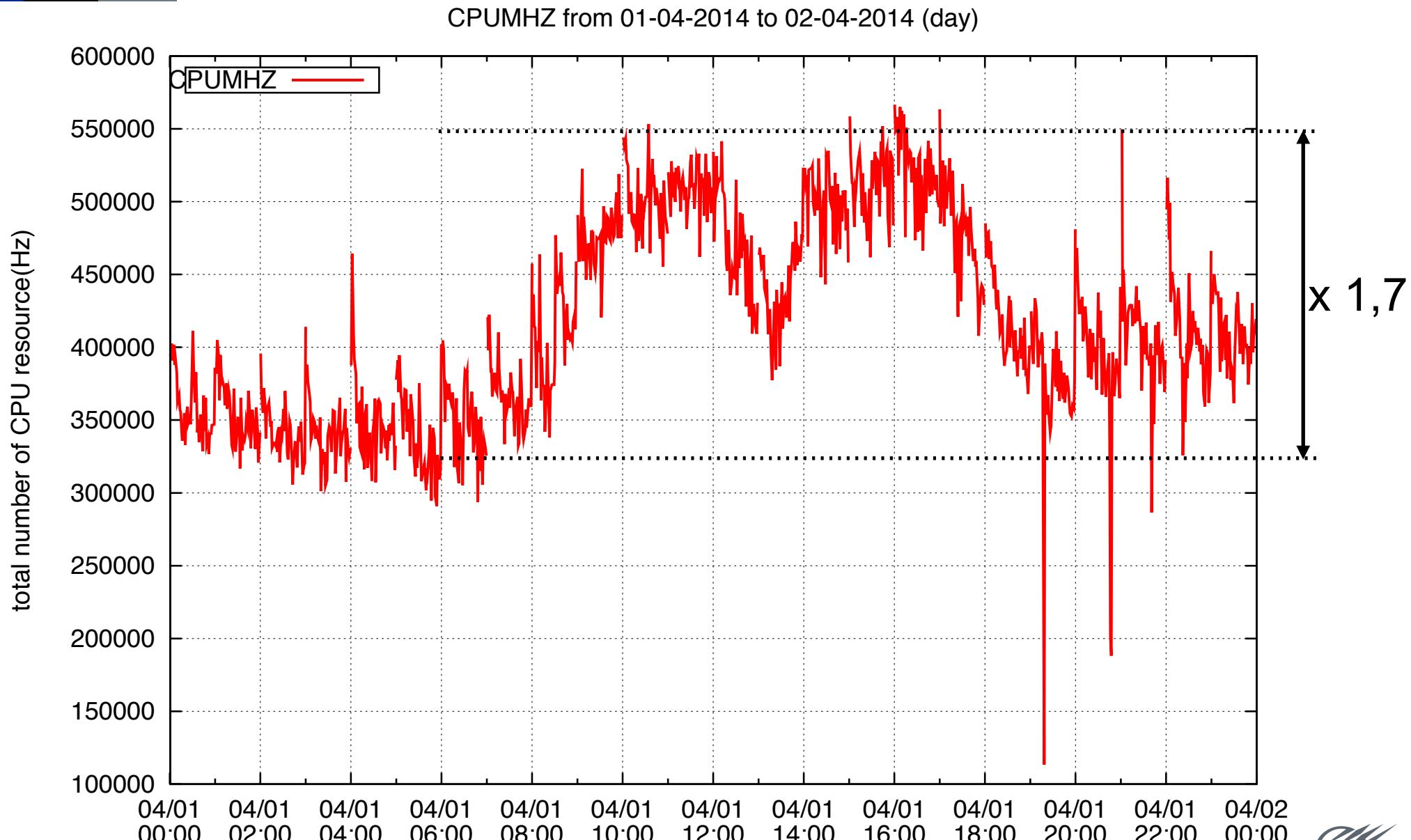
# Virtual Machines workload : Data center real traces

1 week

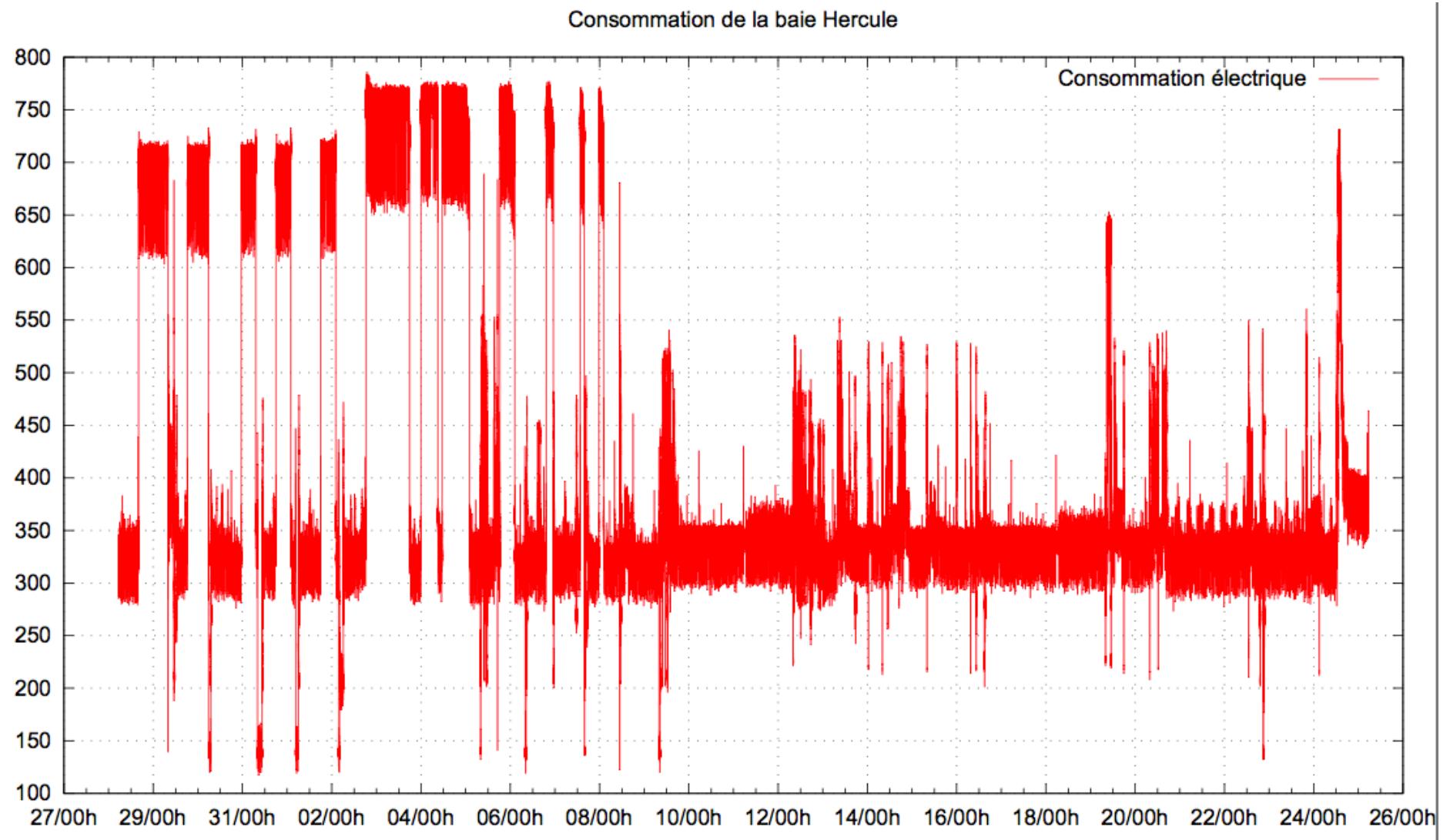


# Virtual Machines workload : Data center real traces

1 day

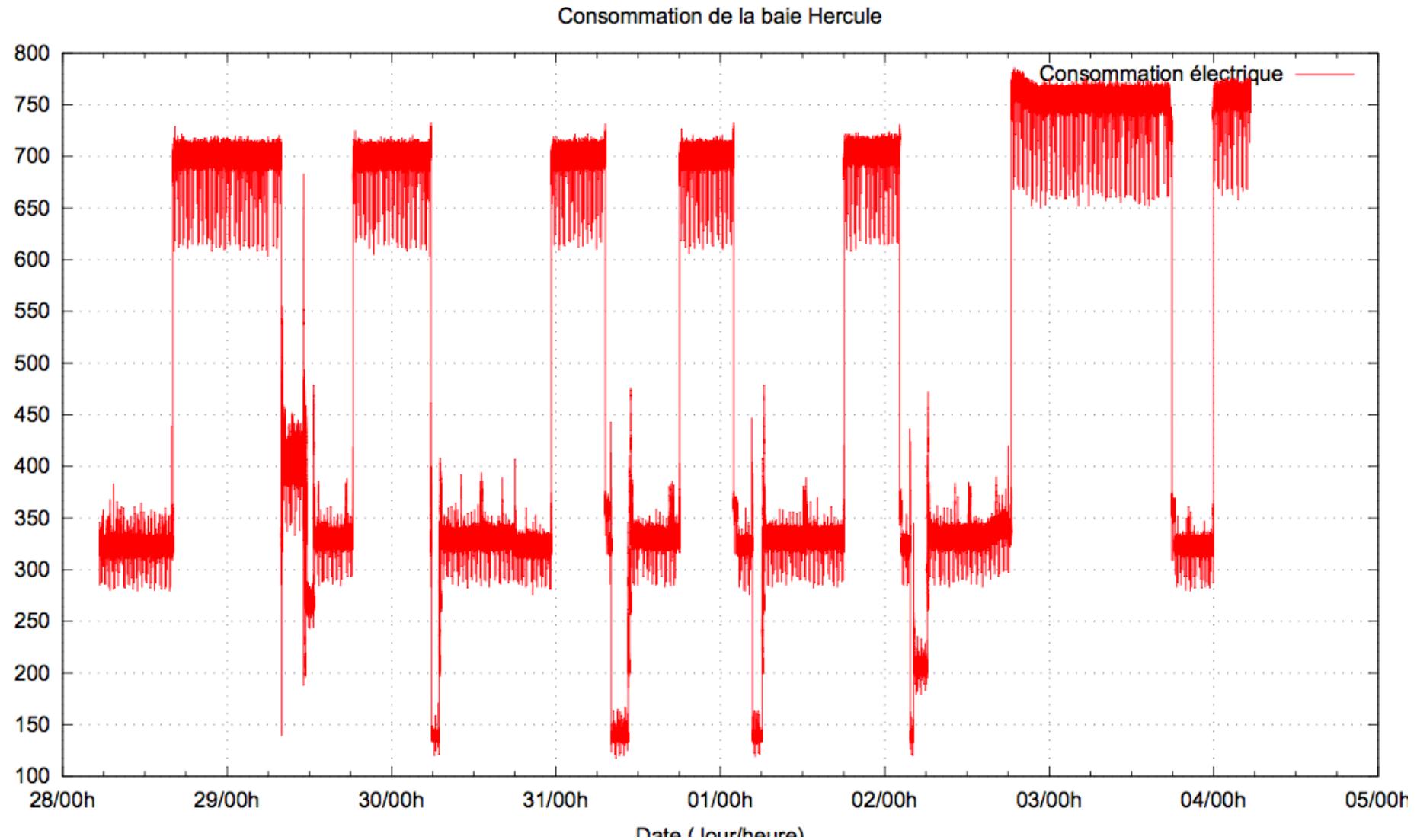


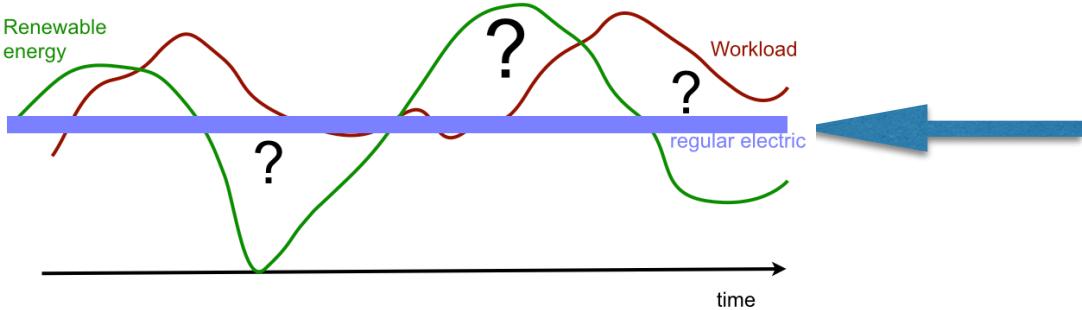
# Batch Jobs



1 week

# Batch Jobs





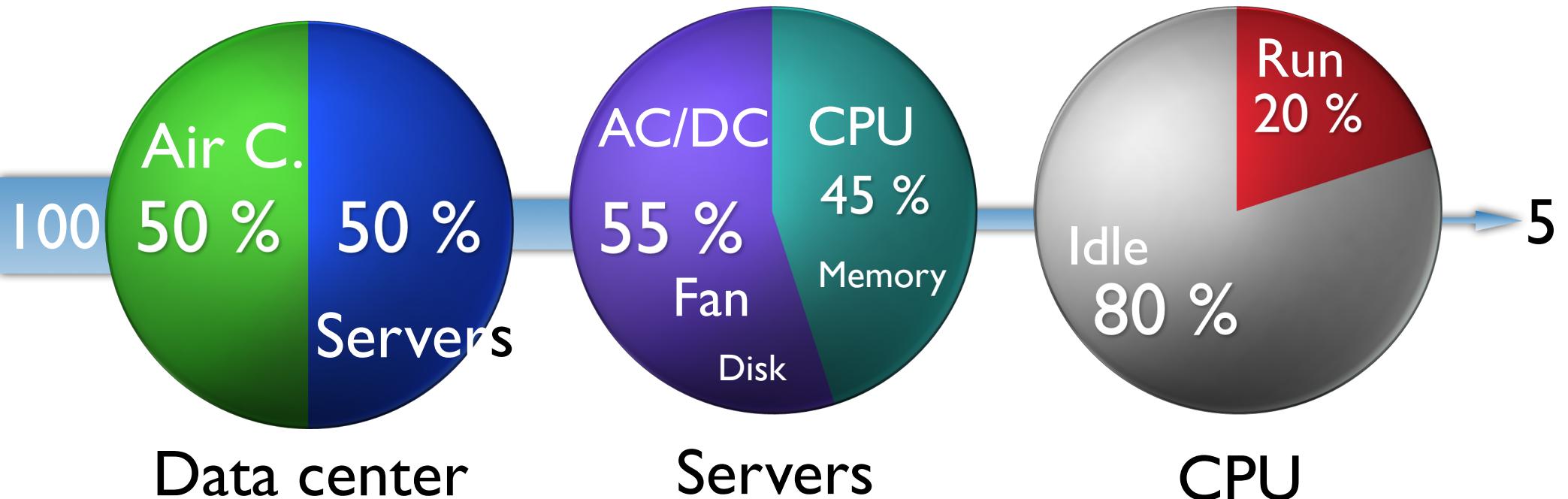
## Solutions logicielles pour la maîtrise énergétique des centres de données

Grands principes  
*Workload driven*

*For a PUE = 2*



## Capacity planning : Energy view



Data center

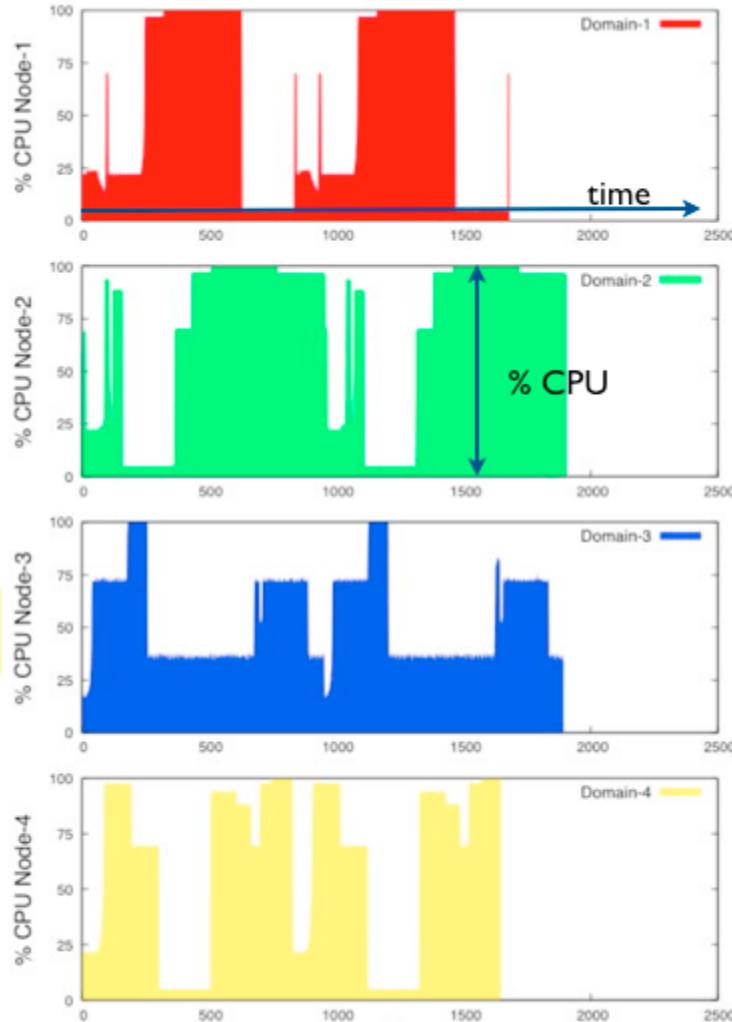
Servers

CPU

- **Analysis of the cost of a 2 MegaWatts DC (5000 nodes, 400w/h)**
  - PUE of 2, 0.06€/kWh => 2 120 886 €
  - A decrease of 5% enables a gain of 110K€
- ■ **Managing DC resources finely becomes a major challenge**

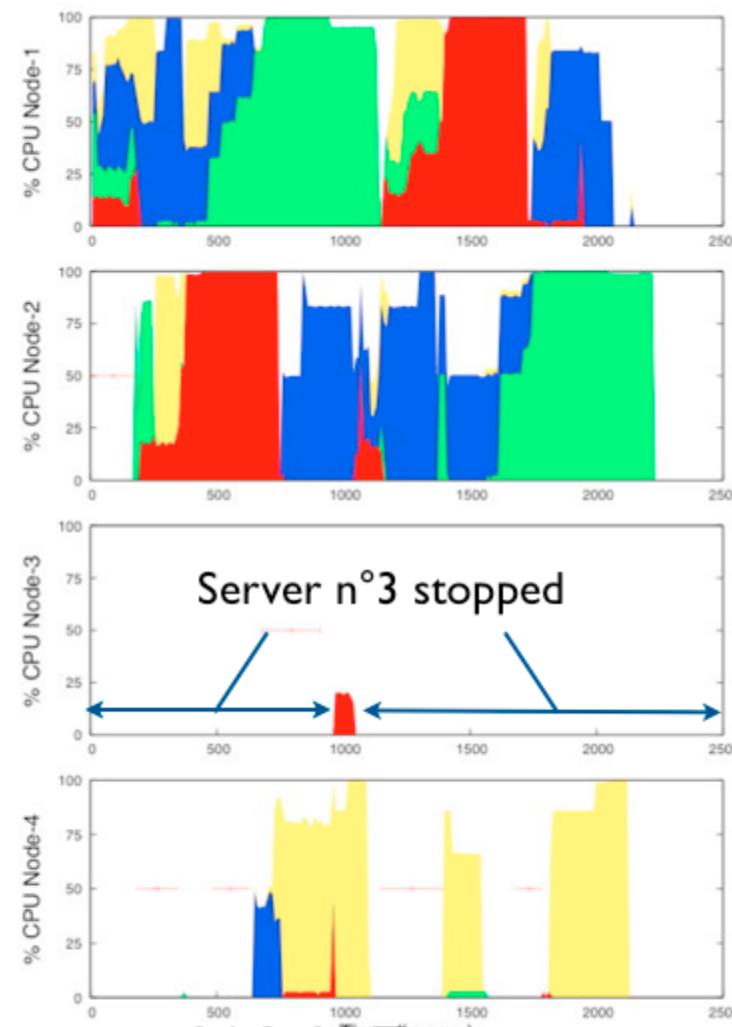
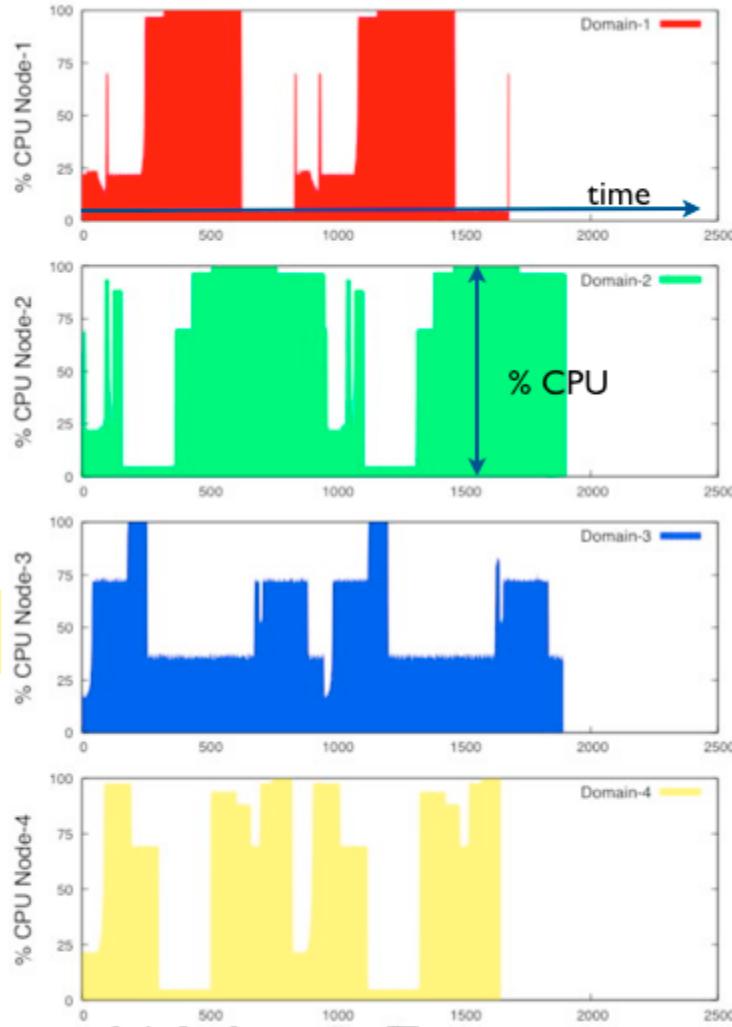
# Entropy: Virtualization & Dynamic Consolidation

4 Tasks ( ) 4 servers



# Entropy: Virtualization & Dynamic Consolidation

4 Tasks ( ) 4 servers



4 Tasks, 3 or 4 Servers  
Consumption is reduced by 25%



... similar to the multi-dimensional bin packing problem know to be NP-Hard ...

## ■ Heuristic methods

- Greedy algorithms Ex: EnaCloud [2009-03]
  - Construct a solution by taking local decision without backtrack.
  - First-Fit Decrease (FFD), Best-Fit (BF), Worst-Fit (WF), Next-Fit (NF) ...
  - Pro: Ease to implement, good worst-case complexity
  - Cons: No optimal solution, not really flexible
- Metaheuristic Ex: Snooze [2012-04]
  - Probabilistic algorithms by searching near optimal solution
  - Genetic, Tabu, Ant colony, Graps ...
  - Pro: Better solution than Greedy algorithms
  - Cons: No optimal solution, not really flexible

## → ■ Exact methods

- Mathematical Ex: Entropy [2009-06]
  - Linear or Constraint programming [1986-05]
  - Compute optimal solution
  - Pro: optimal and flexible
  - Cons: Exponential time solving process

# Entropy Approach

## ■ Constraint programming

- generation of a core model
- placement constraints are translated into "CP constraints"

$$\mathcal{X} = \{x_1, x_2, x_3\}$$

$$\mathcal{D}(x_i) = [0, 4], \forall x_i \in \mathcal{X}$$

$$\mathcal{C} = \begin{cases} c_1 : x_1 < x_2 \\ c_2 : x_1 + x_2 + x_3 = 4 \\ c_3 : \text{allDifferent}(x_1, x_2, x_3) \end{cases}$$

### • Pro

- high-level standardized constraints, portability of a model
- good expressivity
- deterministic composition
- deterministic solving process



### • Cons

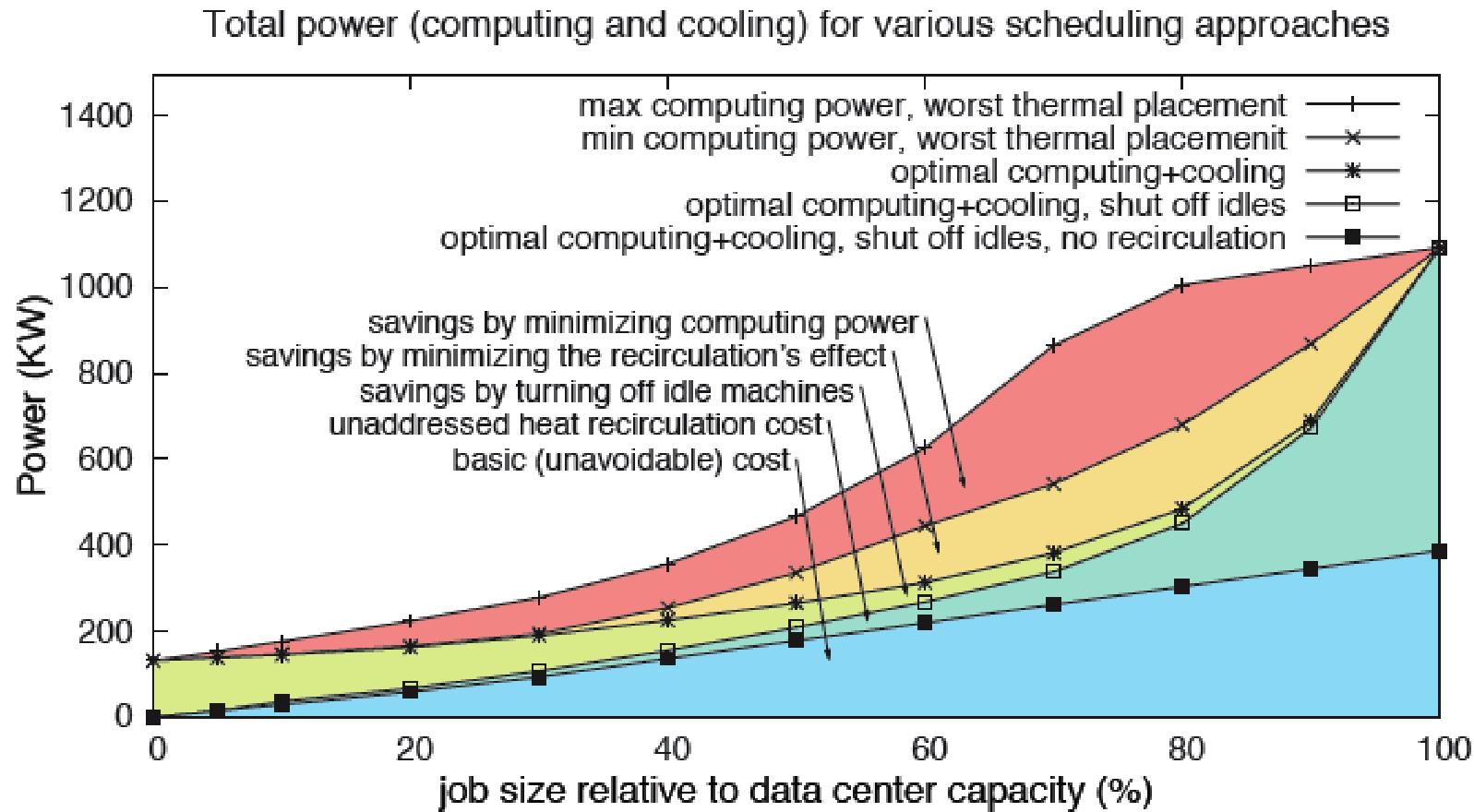
- hard to develop efficient custom constraints
- exact solving duration
- bad model leads to bad performance



## Un peu plus loin dans le placement des tâches

Workload driven

# Holistic System ?



- Thermal-Aware Job Scheduling to Minimize Energy Consumption in Virtualized Heterogeneous Data Centers [2009-18]

# Conceptual overview of thermal-aware task placement



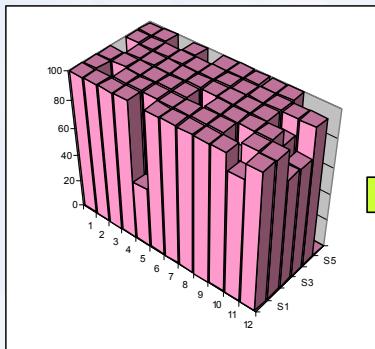
Different task assignments lead to different power consumption distributions



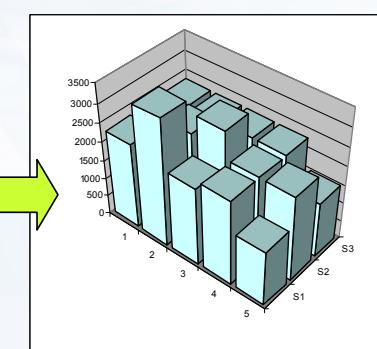
Different power consumption distributions lead to different temperature distributions



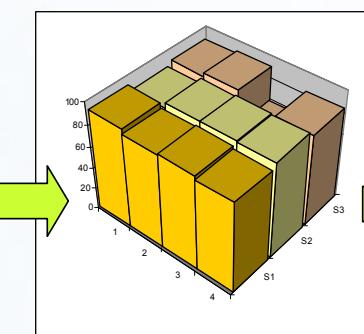
Different temperature distributions lead to different total energy costs



Server task distribution



Power consumption distribution



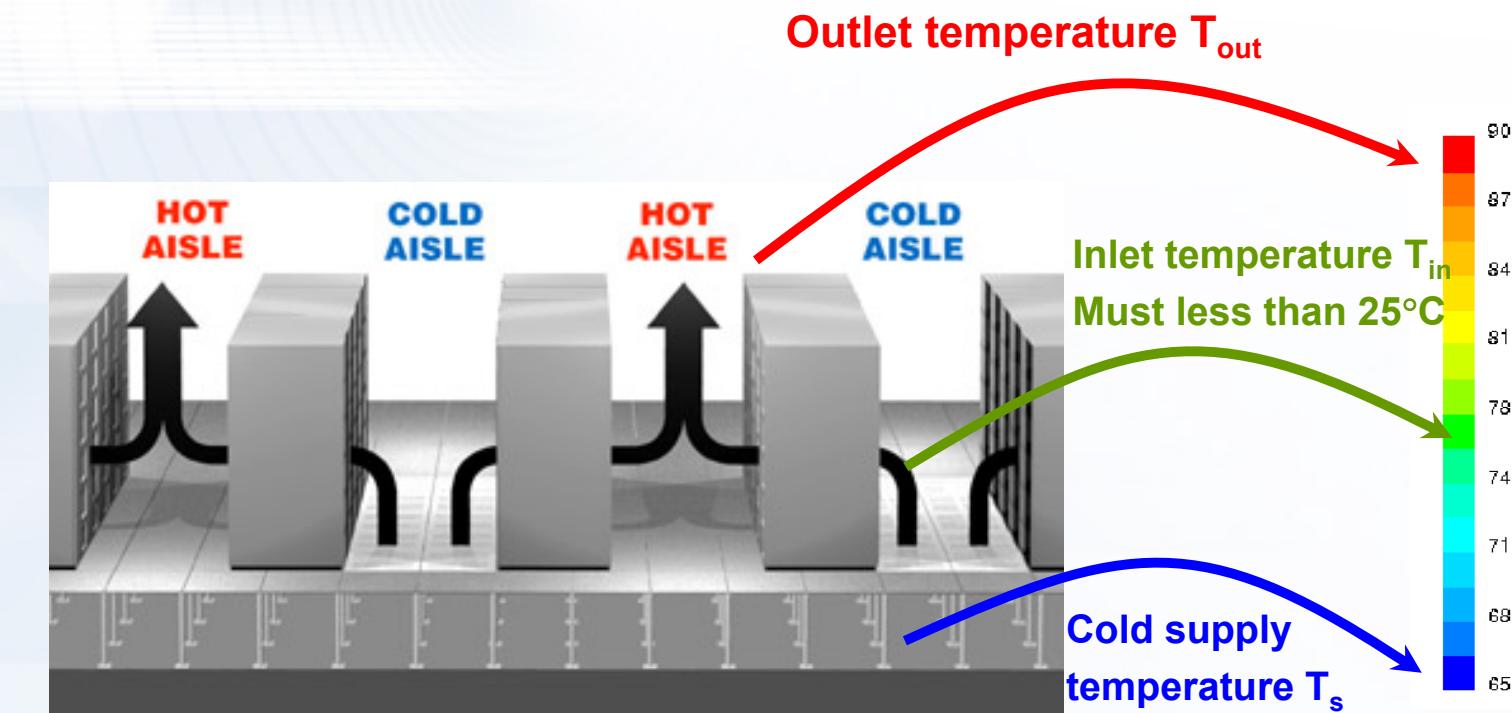
Temperature distribution



Energy cost

19

# Data Center Preliminary: Layout



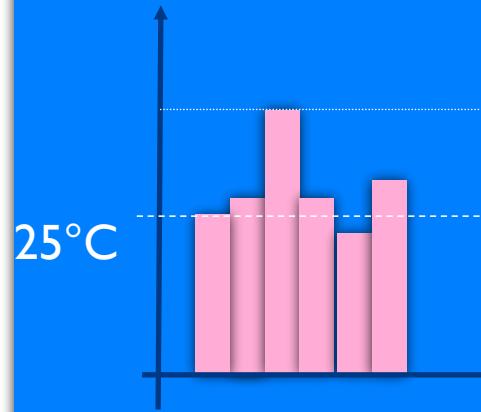
20

**ASU** ARIZONA STATE  
UNIVERSITY

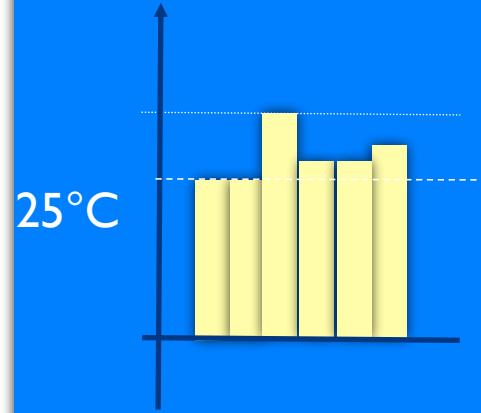
# Scheduling Impacts Cooling Setting

Scheduling 1

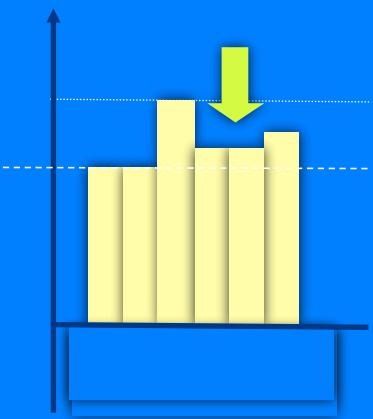
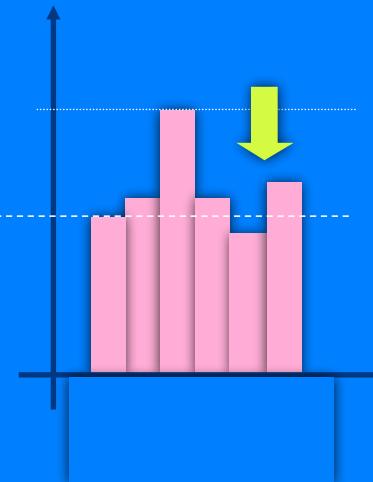
Inlet temperature distribution without Cooling



Scheduling 2



Inlet temperature distribution with Cooling



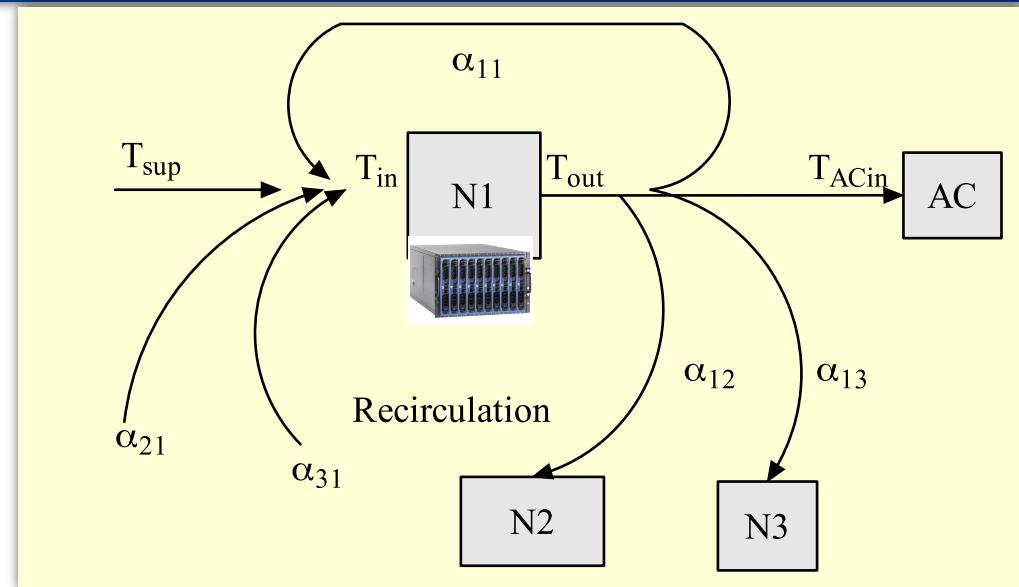
Different demands for cooling capacity

## ► Heat Recirculation Coefficients

- Analytical
- Matrix-based

## ► Properties of model

- Granularity at air inlets (discrete/simplified)
- Assumes steadiness of air flow



$$T_{in} = T_{sup} + D \times P$$

heat distribution

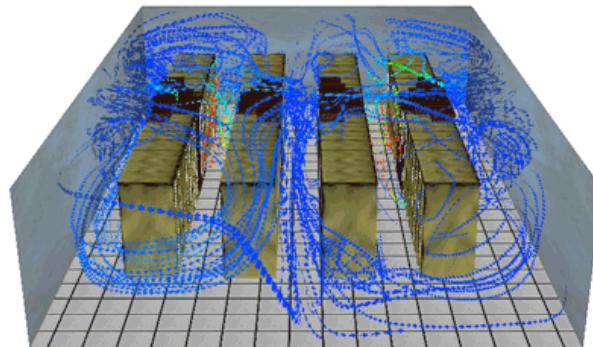
inlet temperatures    supplied air temperatures    power vector

# Benefit: fast thermal evaluation

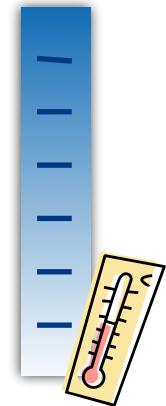
Give workload



Run CFD simulation (days)

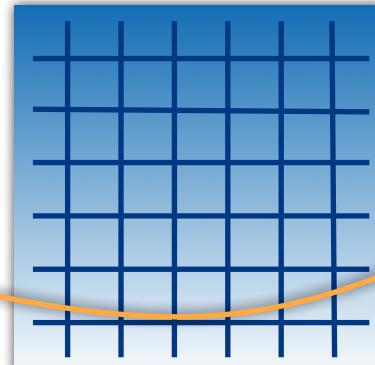


Extract temperatures



**ASU** Give workload

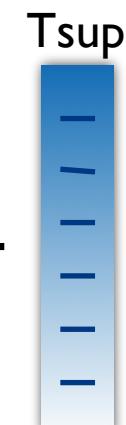
D



x



+



Compute vector (seconds)

Yields temperatures

# Thermal-aware Task Placement Problem

**Given an incoming task, find a task partitioning and placement of subtasks to minimize the (increase of) peak inlet temperature**

## XInt Algorithm

Approximation solution  
(genetic algorithm)

- Take a feasible solution and perform mutations until certain number of iterations

$$T_{in} = T_{sup} + D \times U \quad (a) + b$$

The diagram illustrates the mathematical model for calculating inlet temperatures. It shows the equation  $T_{in} = T_{sup} + D \times U \quad (a) + b$ . The terms are represented as follows:

- $T_{in}$ : A vertical vector of inlet temperatures.
- $T_{sup}$ : A vertical vector of supplied air temperatures.
- $D$ : A square matrix representing the heat distribution.
- $U$ : A vertical utilization vector.
- $a$ : A scalar coefficient.
- $b$ : A constant offset.

Below the vectors, the labels "inlet temperatures" and "supplied air temperatures" are placed under their respective vertical vectors. Below the matrix  $D$ , the label "heat distribution" is centered. Below the utilization vector  $U$ , the label "utilization vector" is centered.

# Contrasted scheduling approaches

## ► Uniform Outlet Profile (UOP)

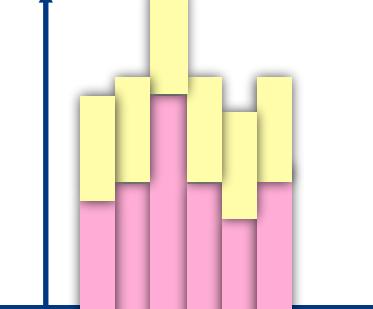
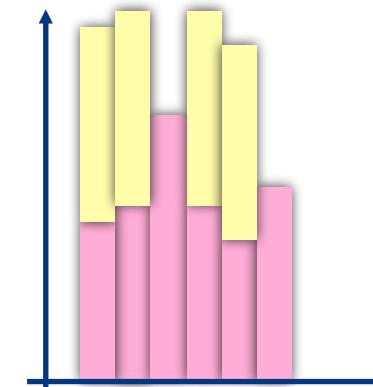
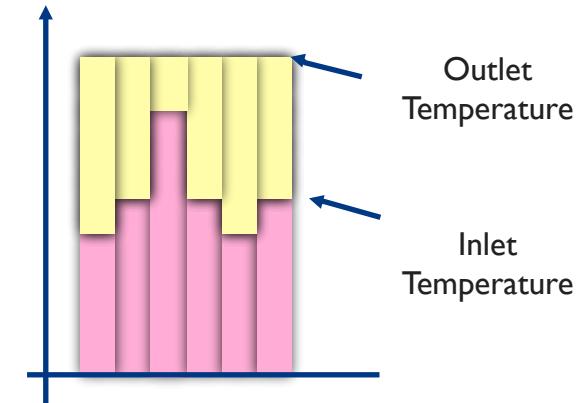
- Assigning tasks in a way that tries to achieve uniform outlet temperature distribution
- Assigning more task to nodes with low inlet temperature (water filling process)

## ► Minimum computing energy

- Assigning tasks in a way that keeps the number of active (power-on) chassis as few as possible
- Server with coolest inlet temperature first

## ► Uniform Task (UT)

- Assigning all chassis the same amount of tasks (power consumptions)
- All nodes experience the same power consumption and temperature rise



# Performance Results

- **Xint outperforms other algorithms**
- **Data Centers almost never run at 100%**
  - Plenty of room for benefits!

