ERIDIS: Energy-efficient Reservation Infrastructure for large-scale DIstributed Systems

Anne-Cécile Orgerie

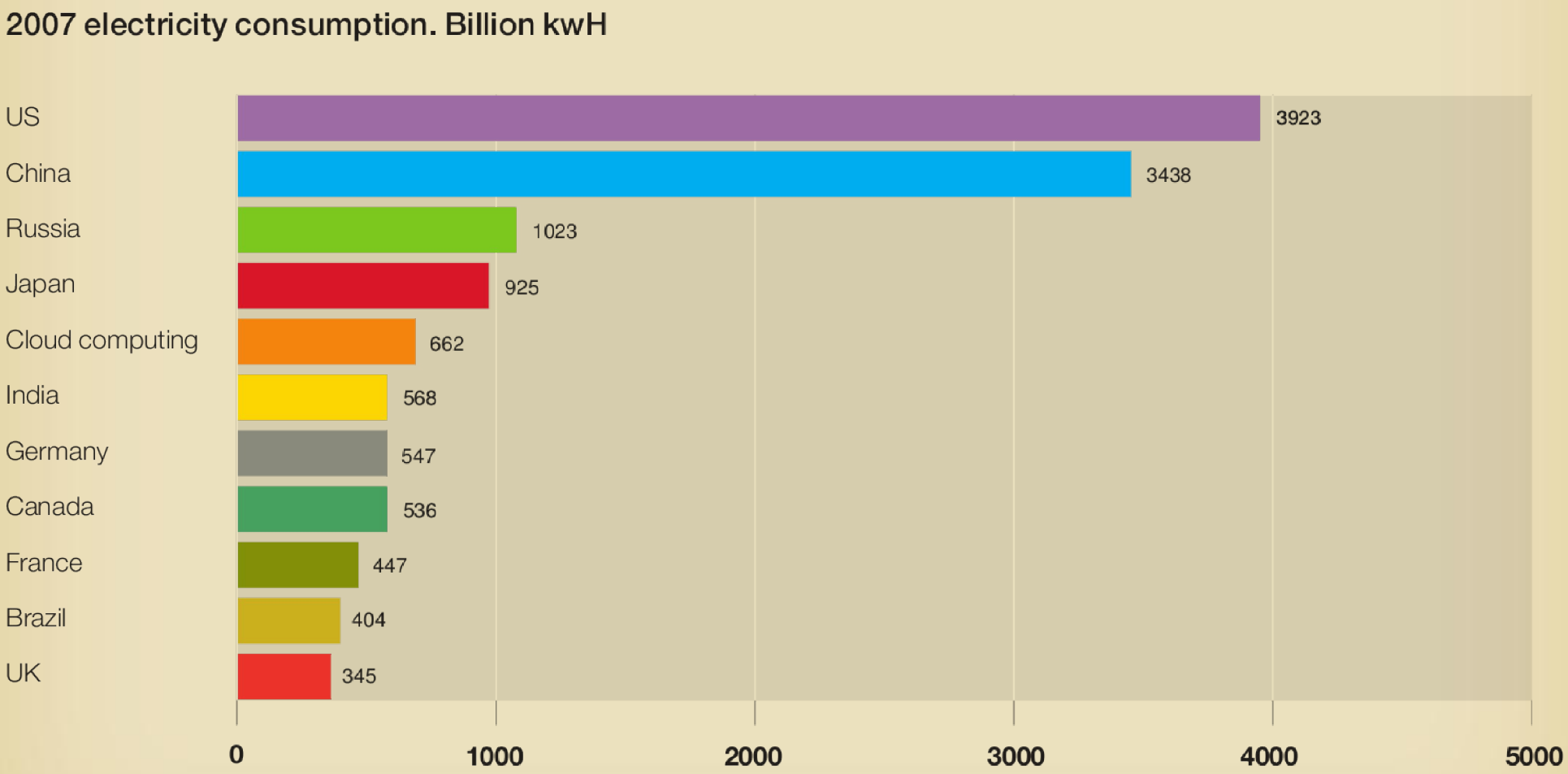
ENS de LYON, FRANCE

annececile.orgerie@ens-lyon.fr

31st May 2011, GreenDays, Paris, France

ENS DE LYON INRIA UCBL Lyon 1 LIP cnrs

Internet + data centers global consumption



Source: ”How dirty is your data?” Greenpeace report, April 2011.

How to decrease the consumption without impacting the performances?

Context:

→ Reservation infrastructures

→ Resource management level

Outline

- ERIDIS
- EARI for data centers and Grids
- GOC for Clouds
- HERMES for dedicated networks
- Conclusions

ERIDIS: Energy-efficient Reservation Infrastructure for large-scale Distributed Systems

Reservation-based systems

Computing reservation:

- Deadline
- Number of resources
- duration

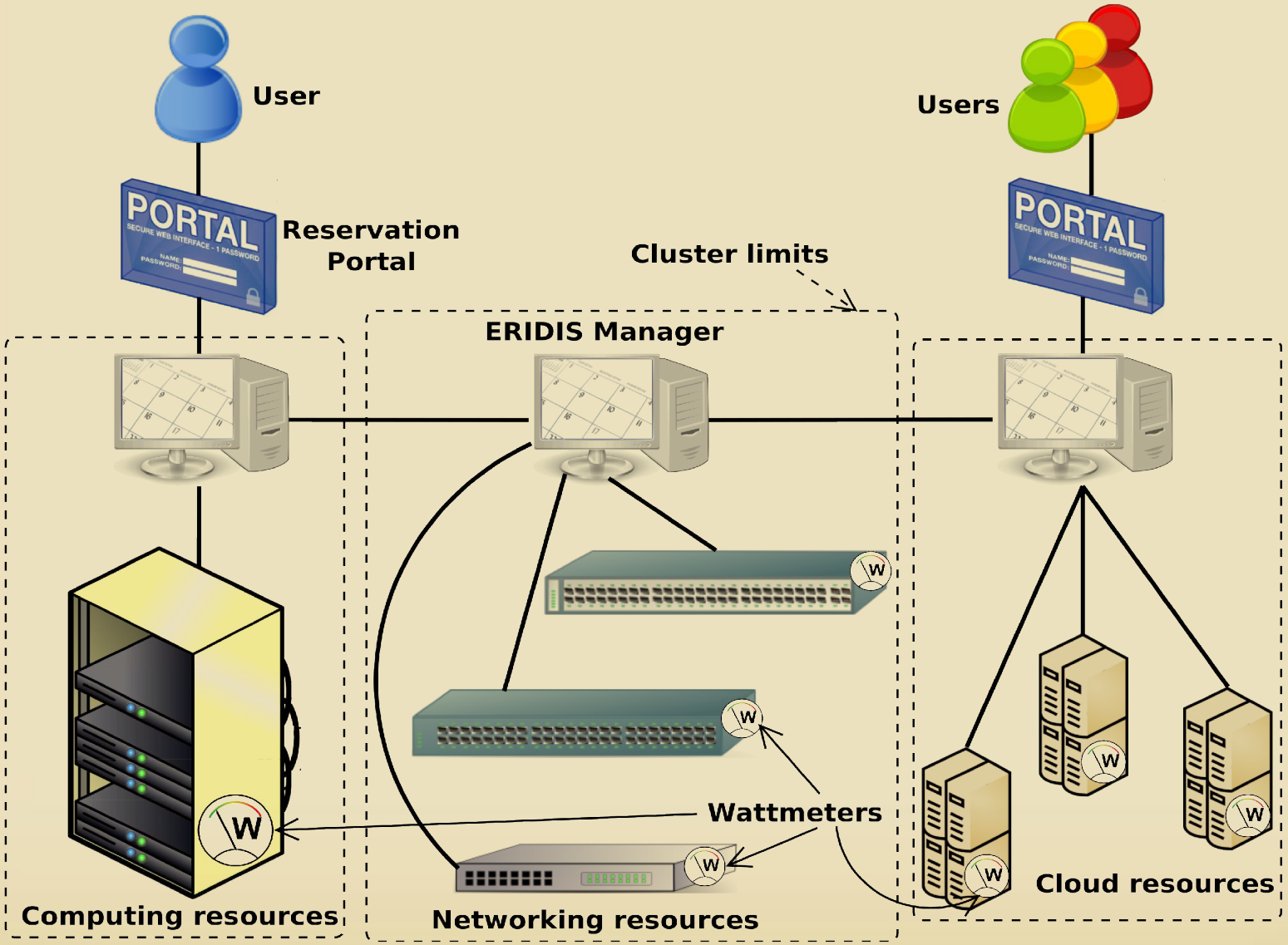
Networking reservation:

- Deadline
- Data volume
- Source and destination

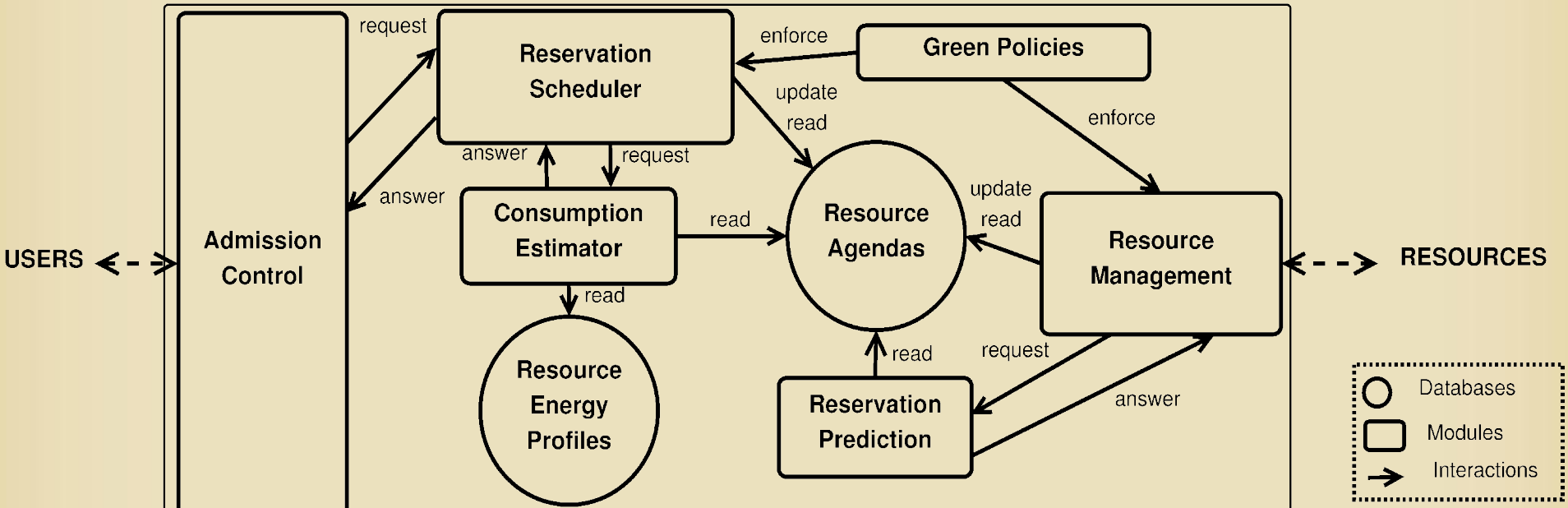
ERIDIS

- Energy sensors
- Allocating and scheduling algorithms
- On/off facilites
- Prediction algorithms
- Workload aggregation policies

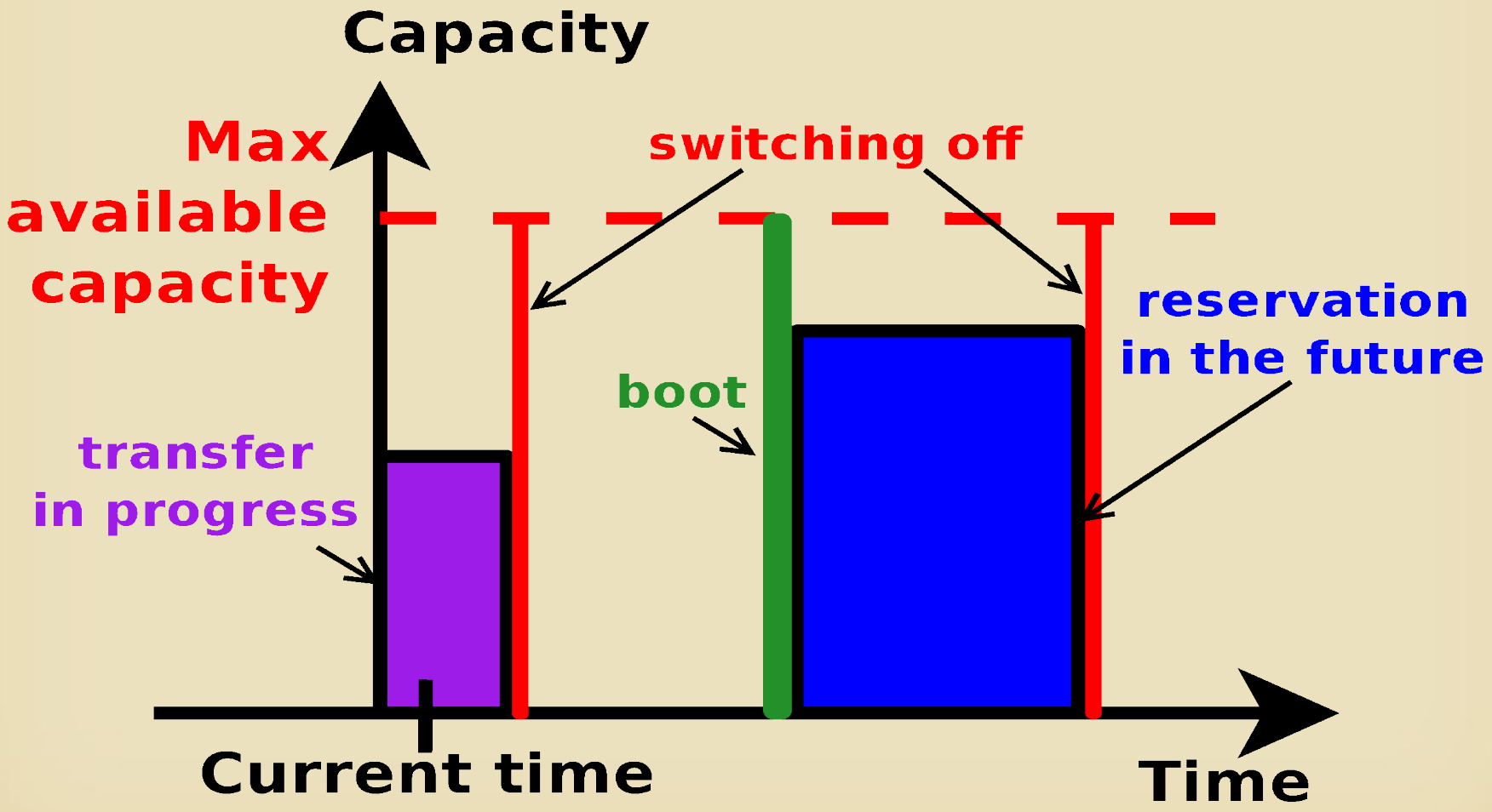
ERIDIS architecture



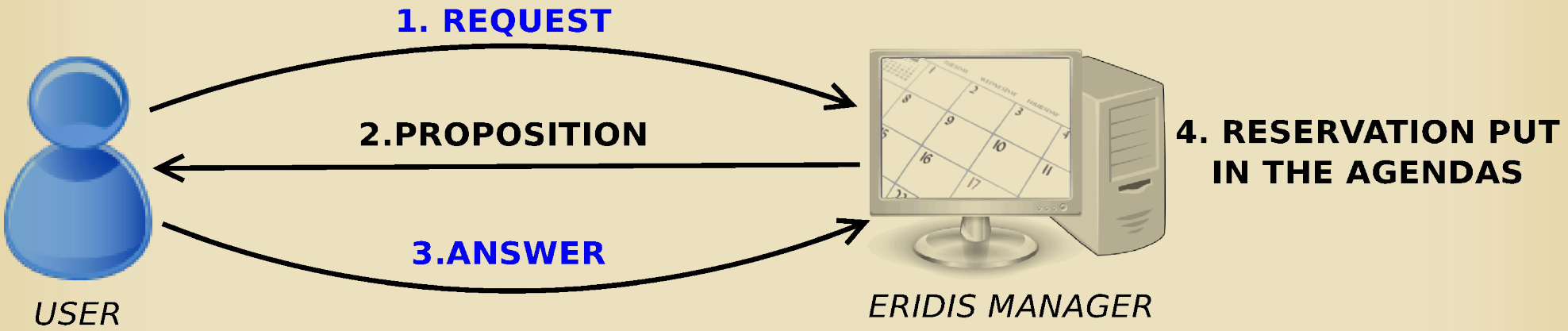
ERIDIS Manager



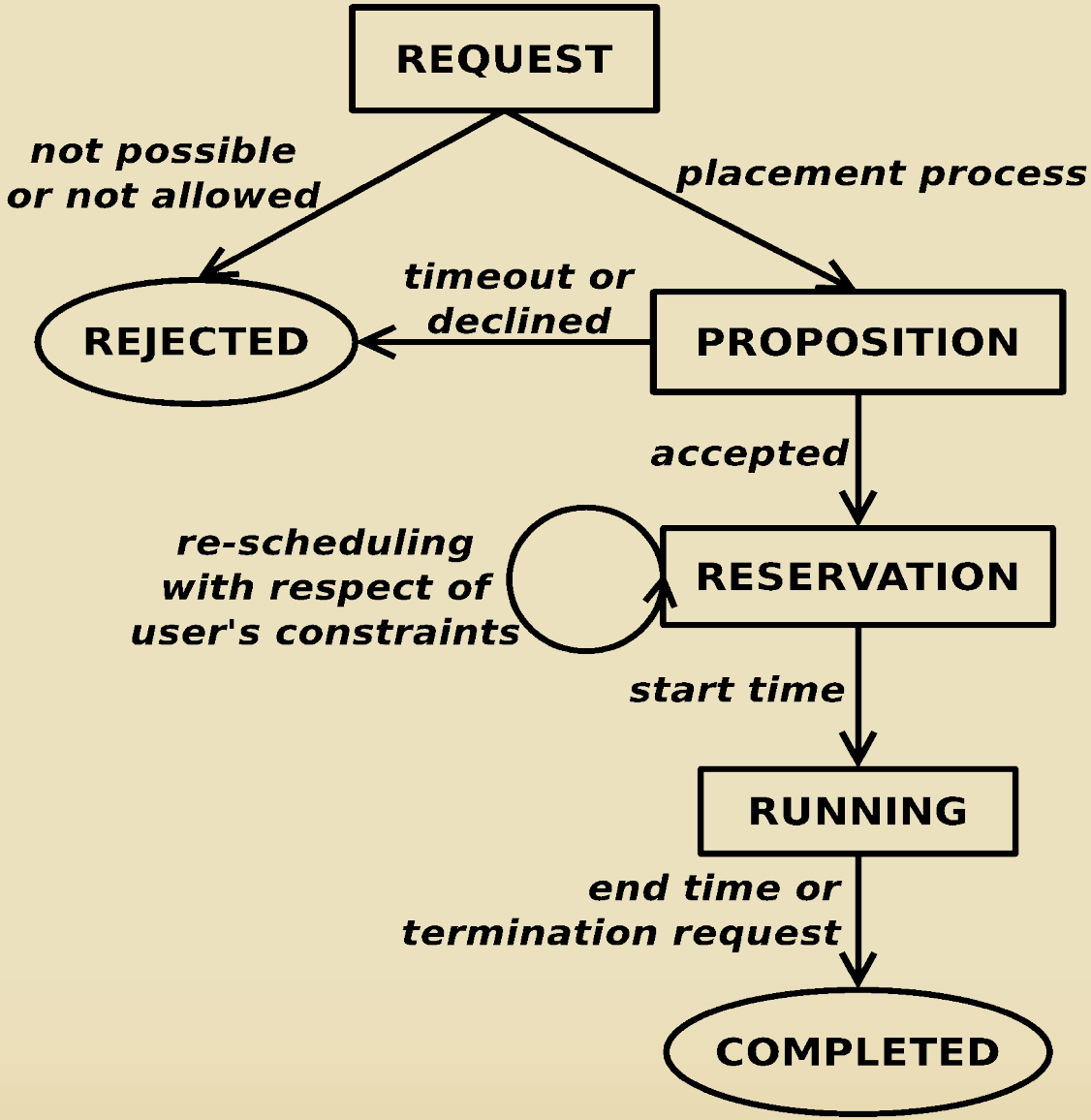
Resource agenda



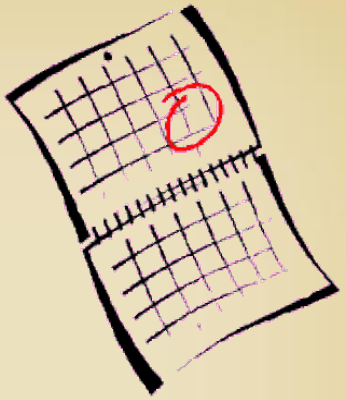
Reservation negociation



Management of a reservation



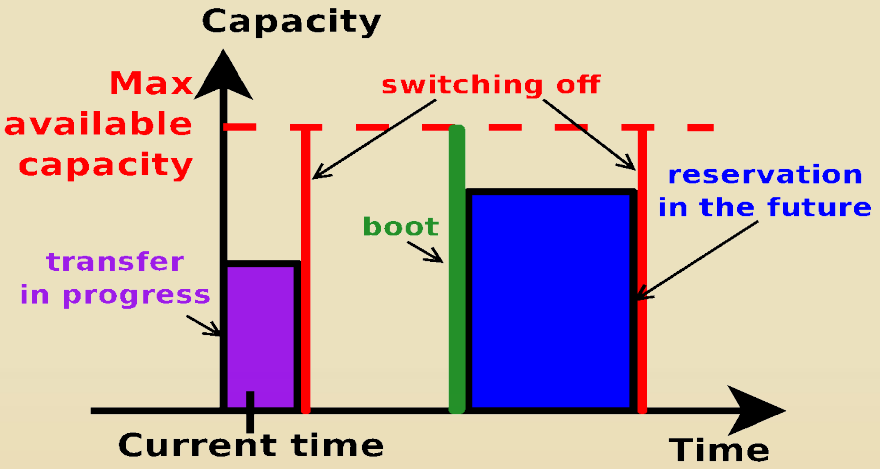
Scheduling



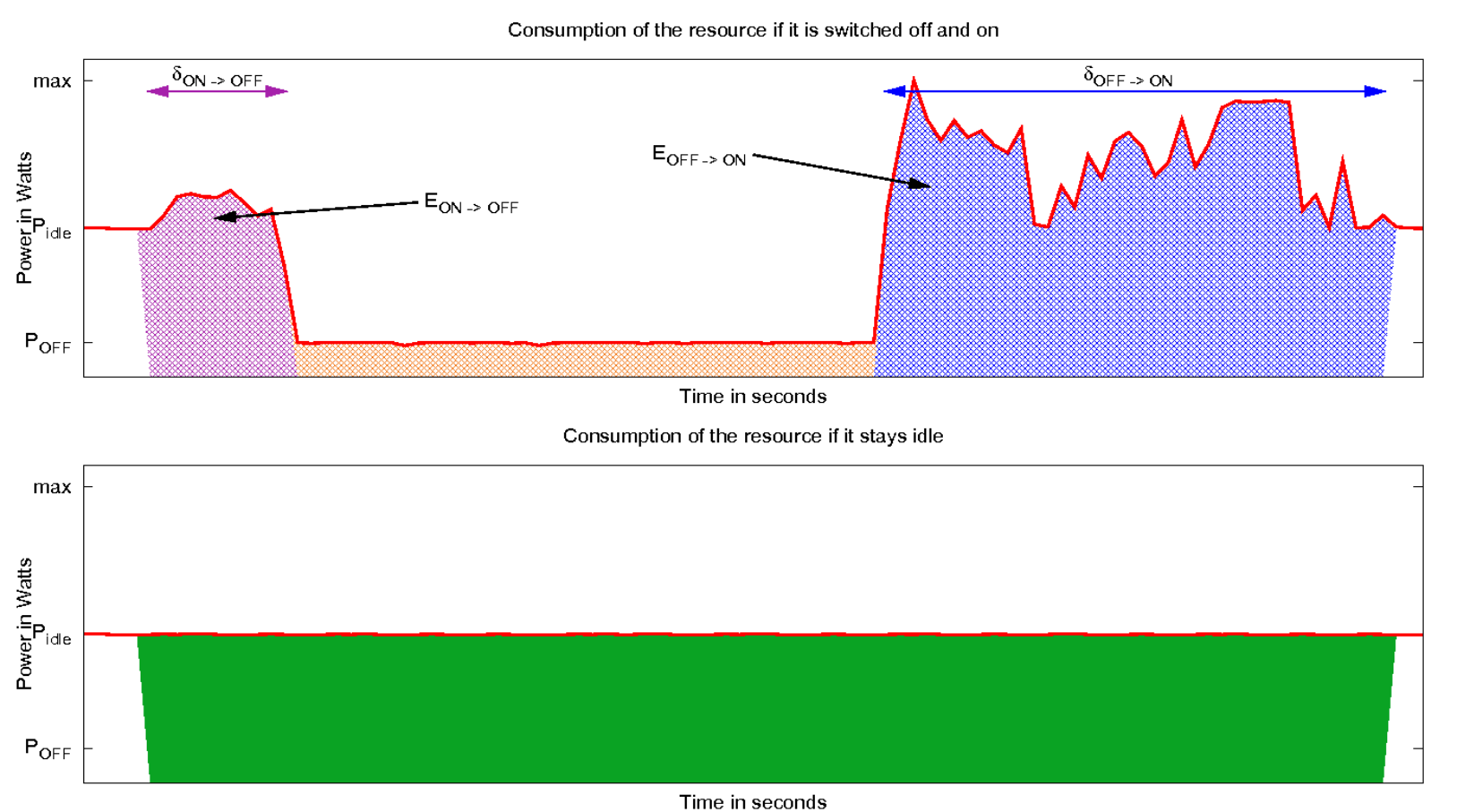
- For each event before the deadline:

  - try to put the reservation here

- Estimate the energy consumption for each possibility
- Pick the least consuming solution



When can we switch off ?



$$T_s = \frac{E_s - P_{OFF}(\delta_{ON \rightarrow OFF} + \delta_{OFF \rightarrow ON}) + E_{ON \rightarrow OFF} + E_{OFF \rightarrow ON}}{P_I - P_{OFF}} + T_r$$

Predictions

What :

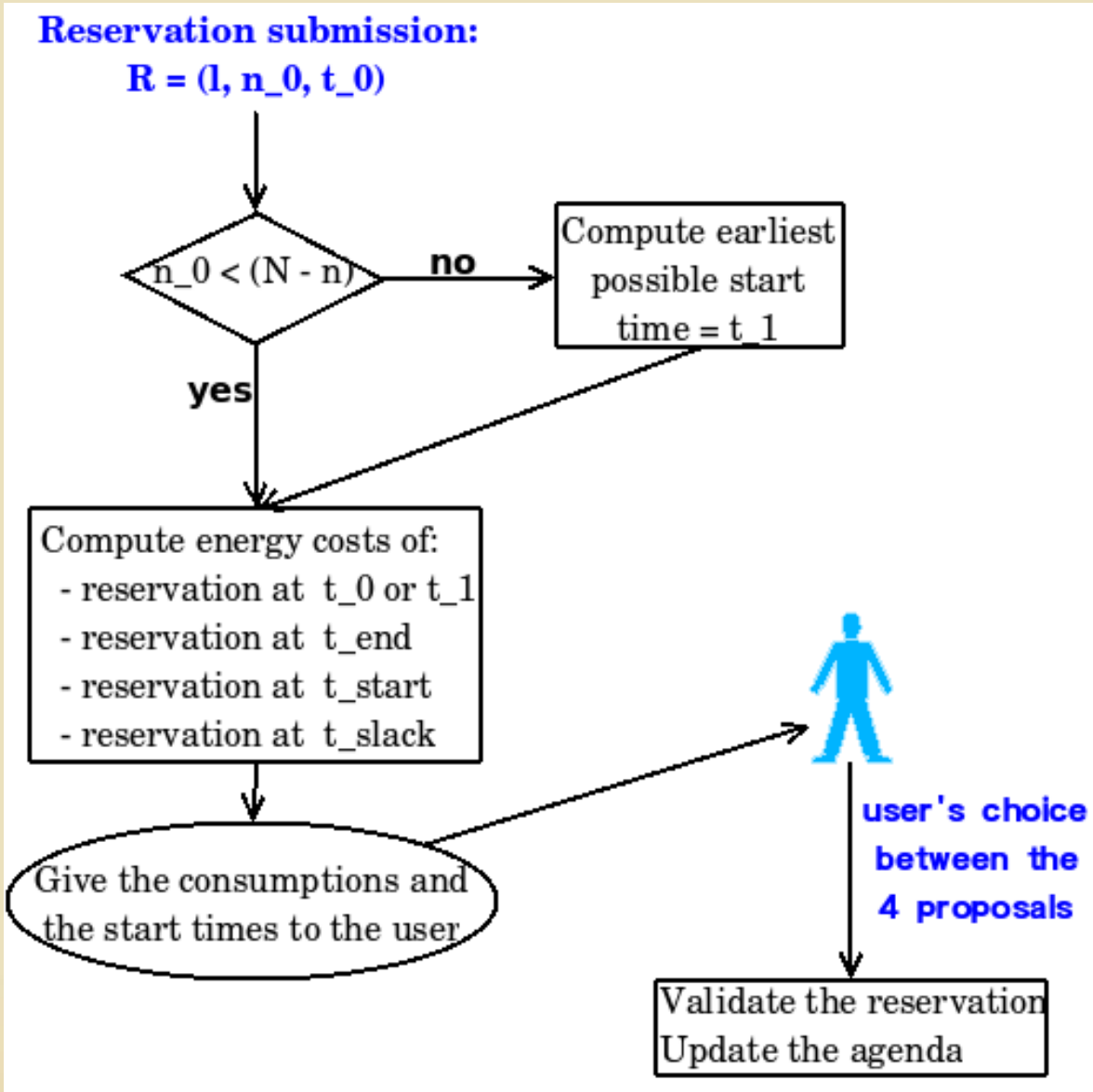
- Next reservation (size, duration, start time)
- Next empty period
- Energy consumption of a reservation

With :

- Recent history (last reservation) + feedback
- Recent reservations days + feedback
- User history + resources

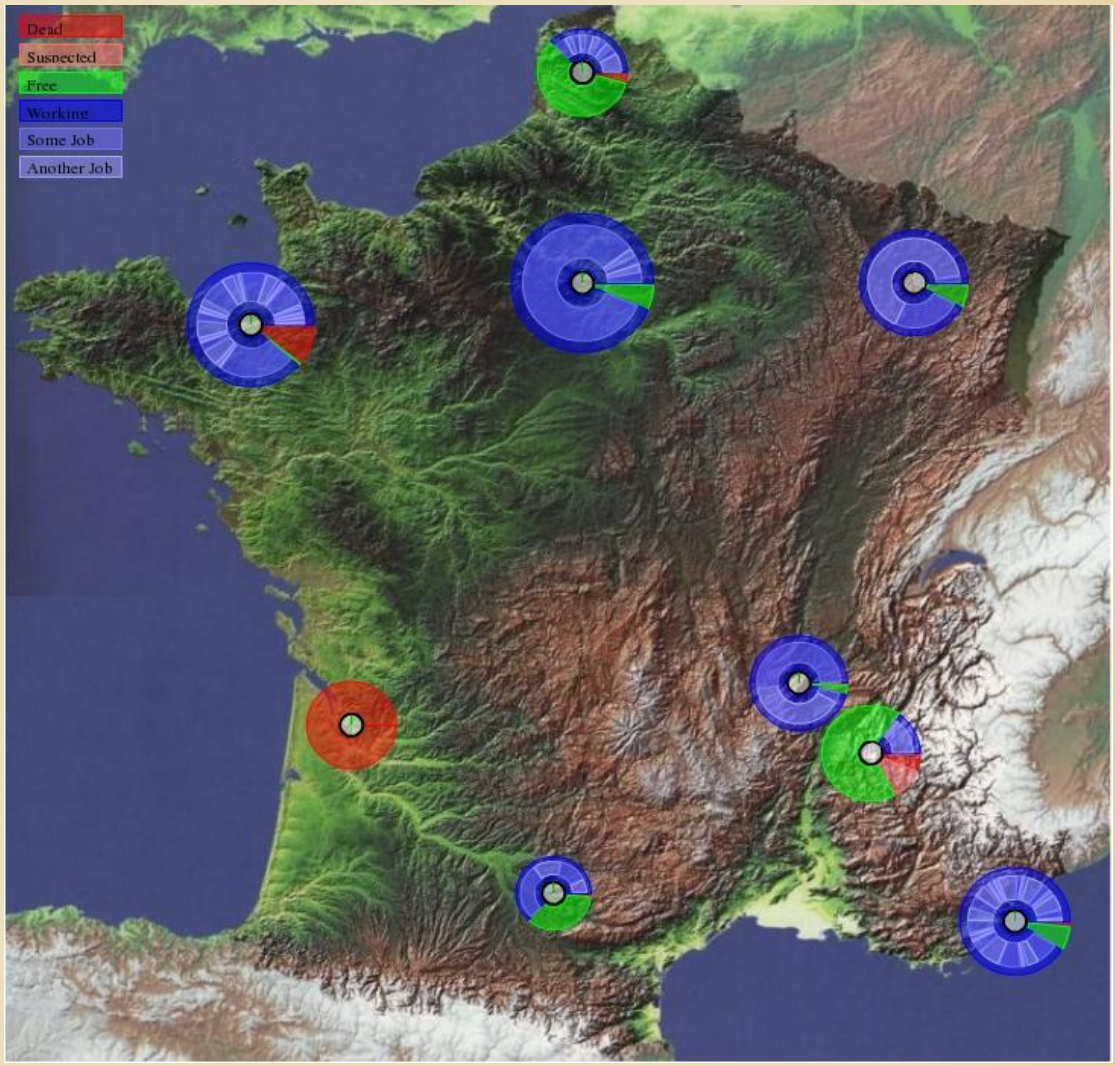
Energy-Aware Reservation Infrastructure

After a reservation request

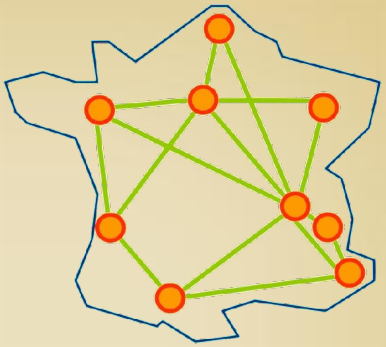


Grid'5000

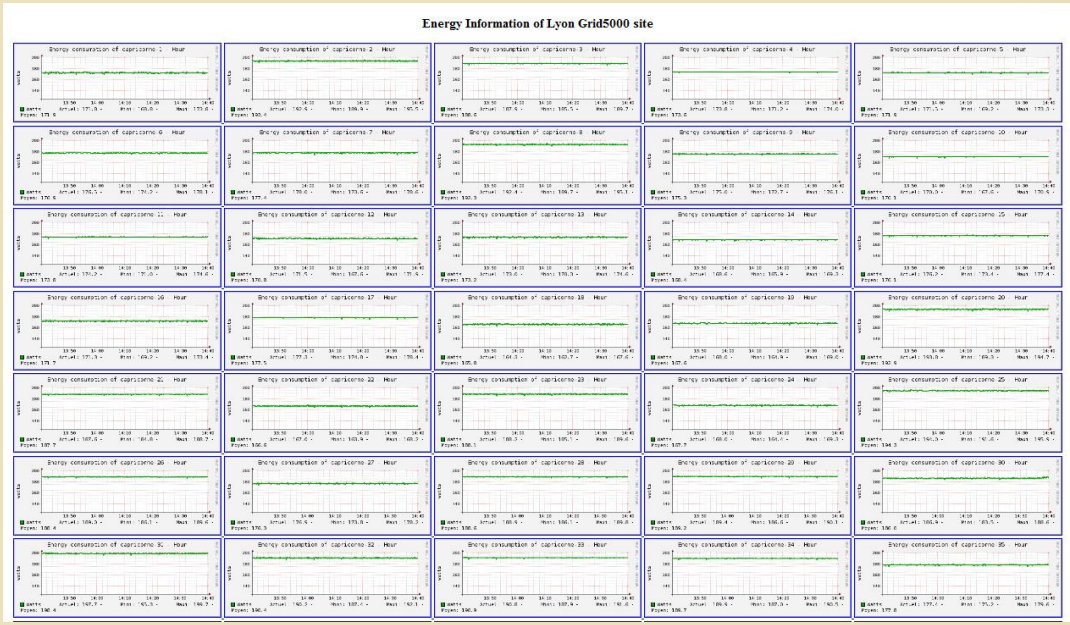
- French experimental testbed
- 5000 cores
- 9 sites
- Dedicated Gb network
- Designed for research on large-scale parallel and distributed systems



Lyon: a Monitored Site



- 135 nodes
- One power measurement per node and per second

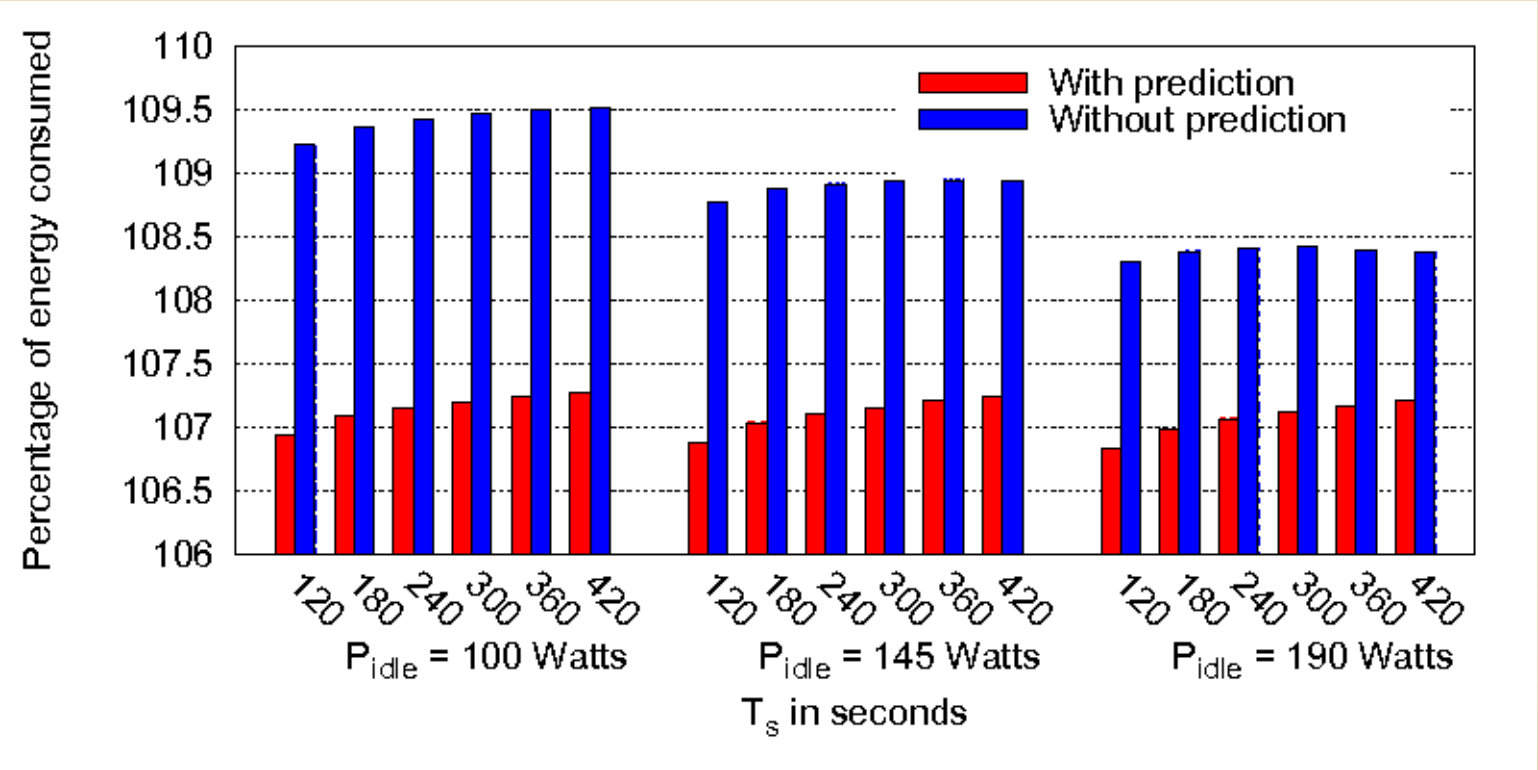


Prediction evaluation based on replay

Example: Bordeaux site (650 cores, 45K reservations, 45% usage)

100 % : theoritical case (future perfectly known)

Currently (always on) : 185 % energy



Green Policies

- **user**: requested date
- **25% green**: 25% of jobs follow Green advices – the rest follows user request
- **50% green**: 50% of jobs follow Green advices – the rest follows user request
- **75% green**: 75% of jobs follow Green advices – the rest follows user request
- **fully green**: solution with uses the minimal amount of energy and follows Green advices
- **deadlined**: fully green for 24h – after: user

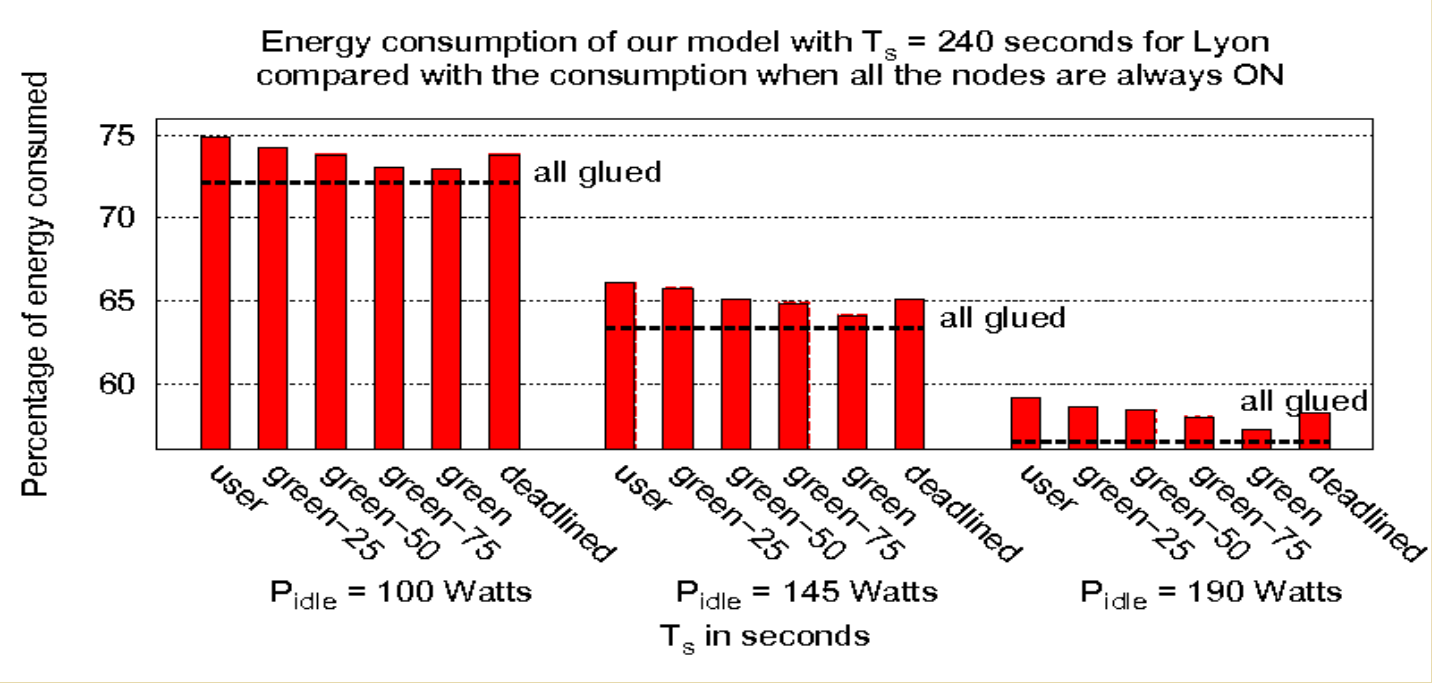
Evaluation on Lyon example

Example of Lyon site (322 cores, 33K reservations, 46% usage)

Current situation: always ON nodes (100 %)

All glued: unreachable theoretical limit

For Lyon site: saving of 73,800 kwh for 2007 period



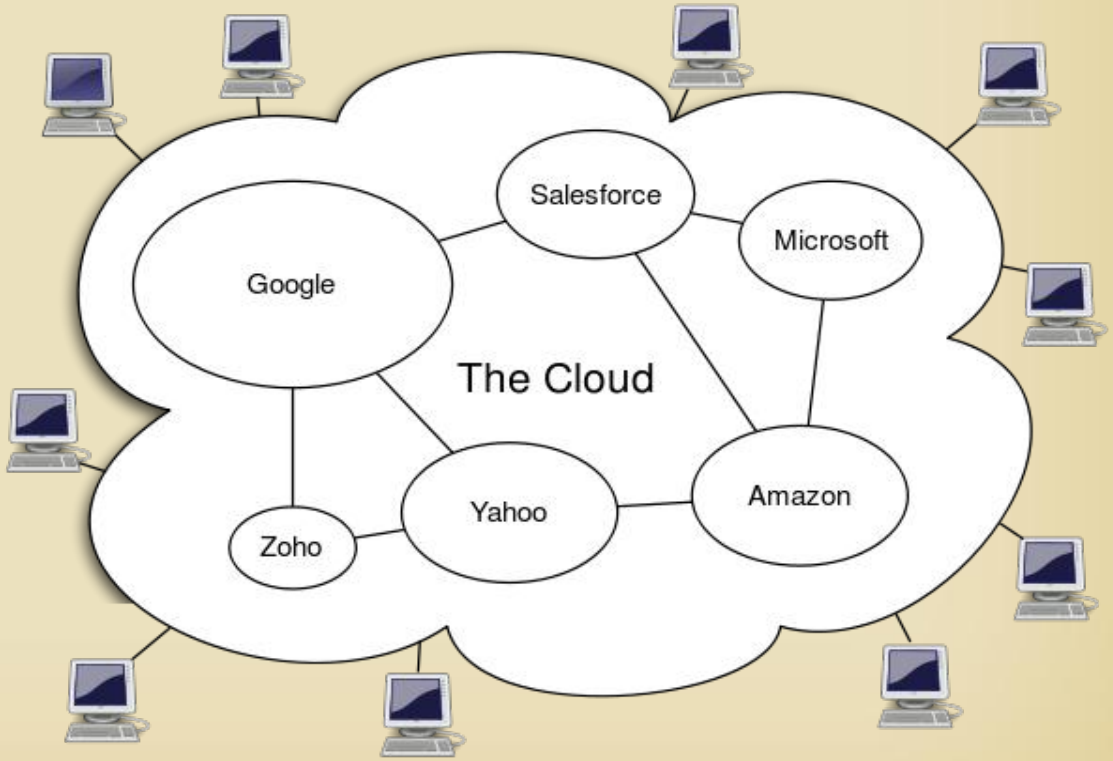
Summary

- Proposition of an energy-aware infrastructure for resource reservation
  - simple and quick in terms of computing time
  - including heuristics
  - proposing energy saving solutions to the users without forcing them and impacting performances
  - leading to important energy savings.

Green Open Cloud

GOC Features

- Virtual machines
- Reservations
- Live migration
- Reduce the number of awake nodes



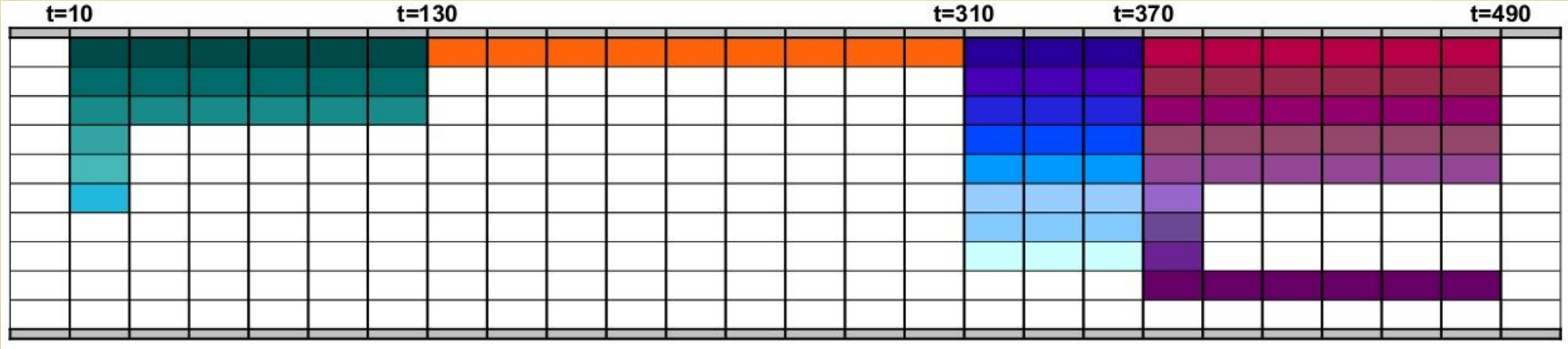
Experimental Methodology

Cloud job arrival example:

- t = 10: 3 jobs of 120 s. + 3 jobs of 20 s.
- t = 130: 1 job of 180 s.
- t = 310: 8 jobs of 60 s.
- t = 370: 5 jobs of 120 s. + 3 jobs of 20 s. + 1 job of 120 s.

→ limited time experiment

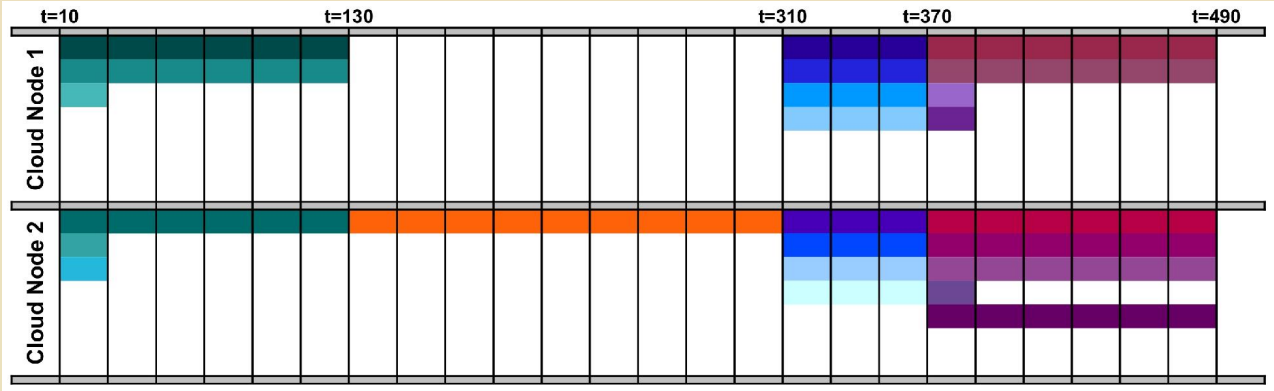
→ identical nodes

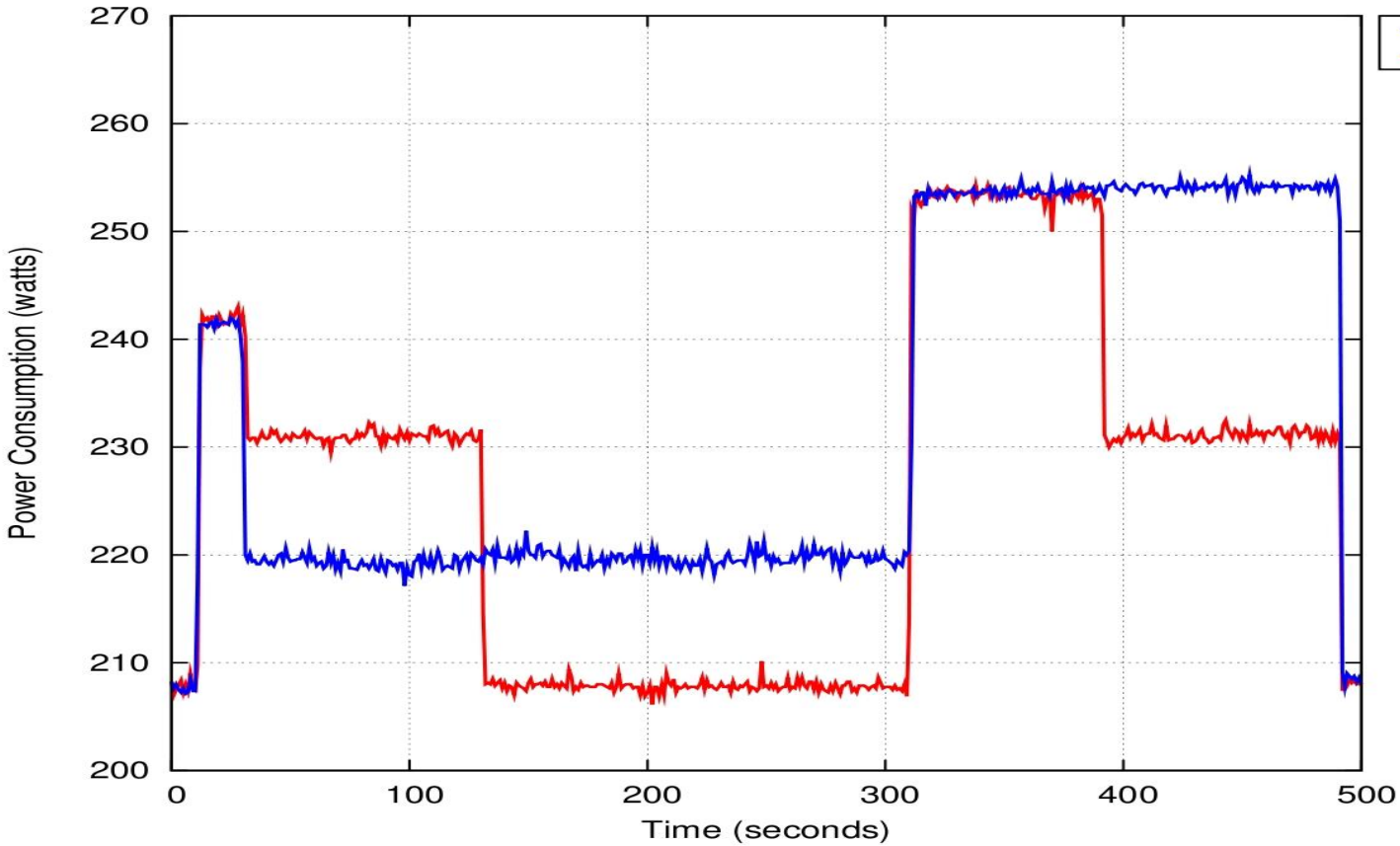


Experimental Methodology

- Two different simple **schedulings**: *round-robin* and *unbalanced.*
- Four **scenarios**:
  - *basic*: nothing to do;
  - *balancing*: use migration to balance the load;
  - *on/off:* switch off unused nodes;
  - *green*: switch off unused nodes and use migration to unbalance the load.

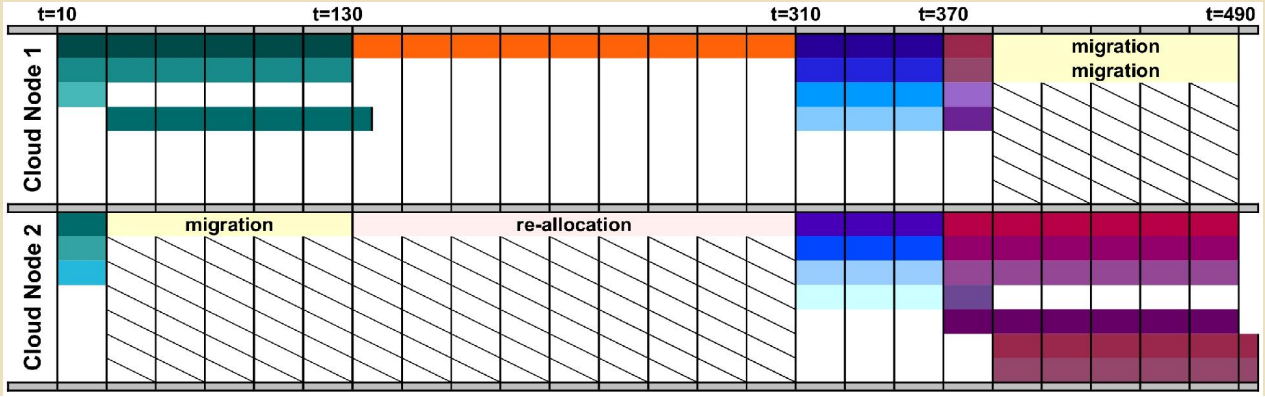
Round-Robin with Basic Scenario

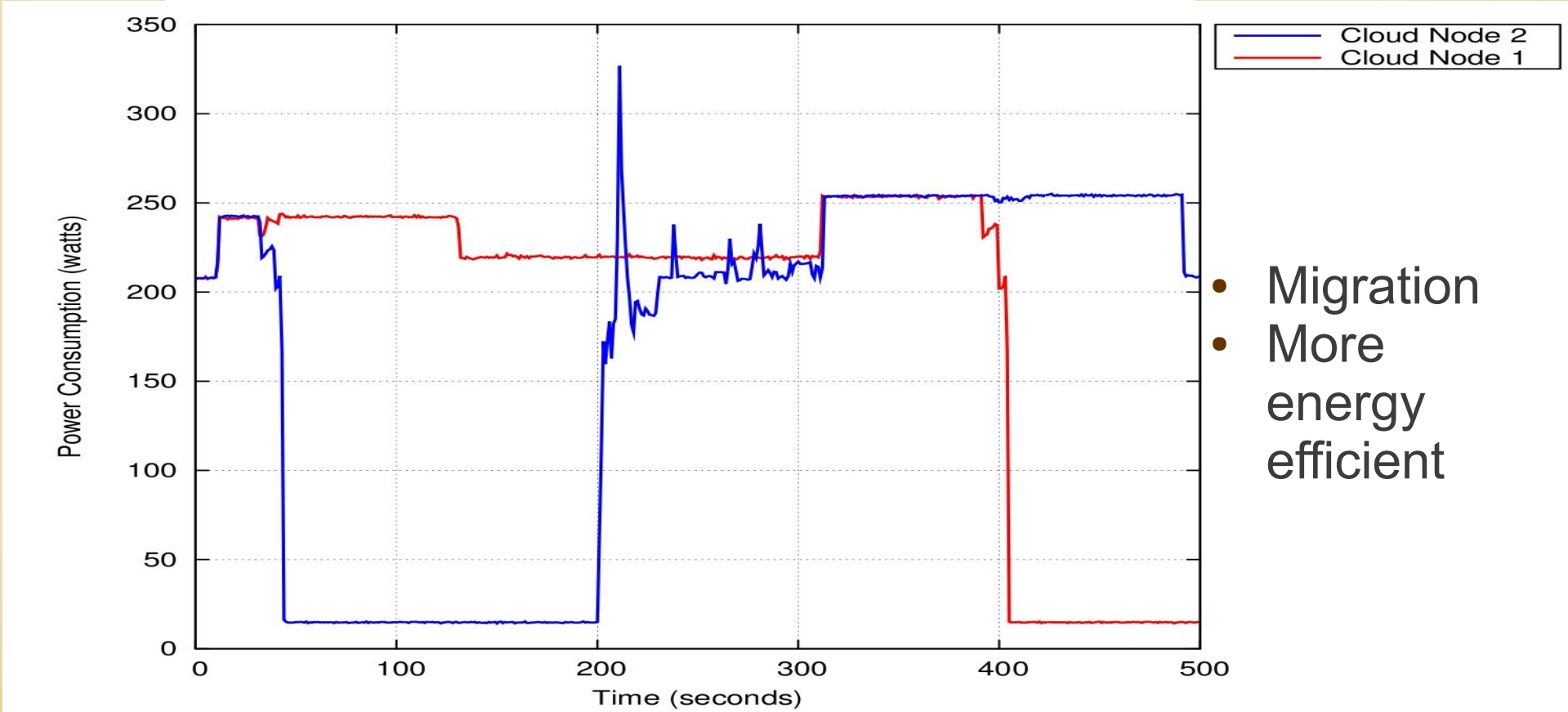




- Identical nodes
- Energy levels

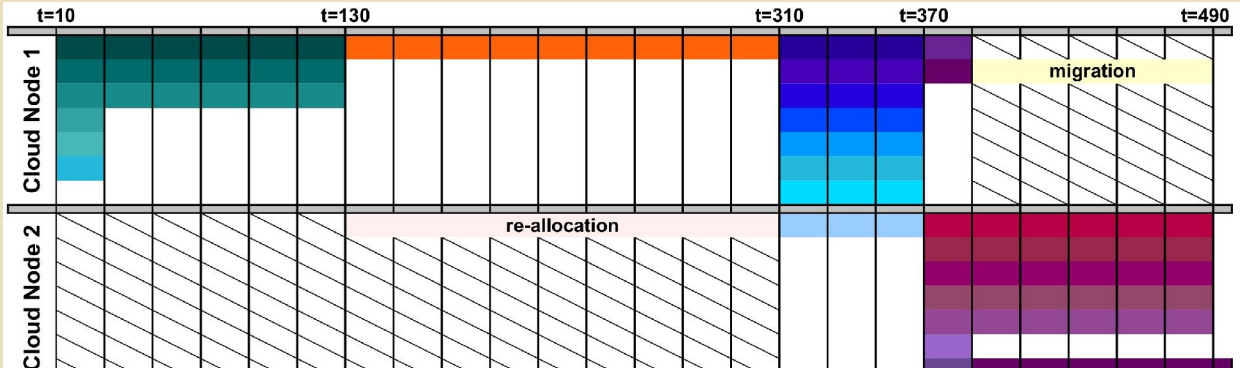
Round-Robin with Green Scenario

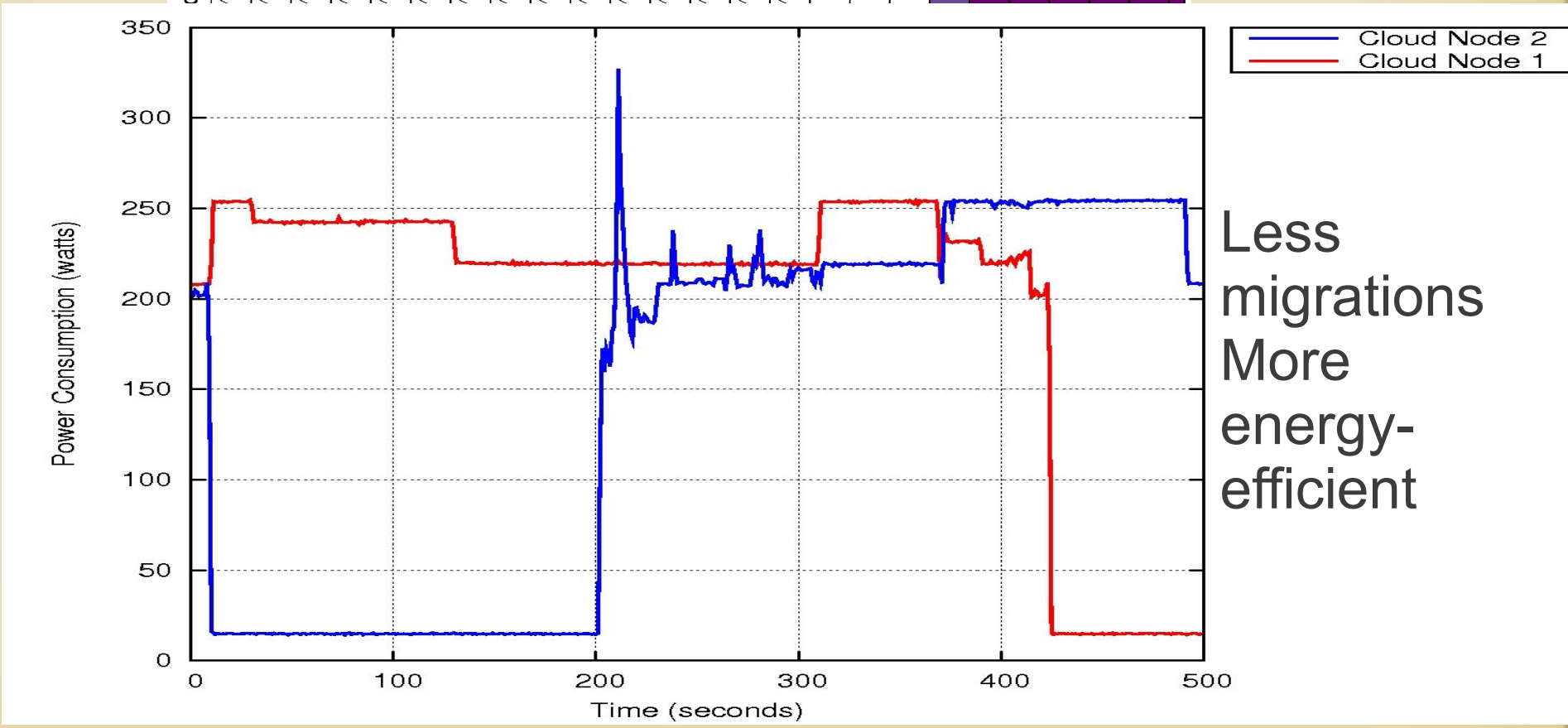




- Migration
- More energy efficient

Unbalanced with Green Scenario

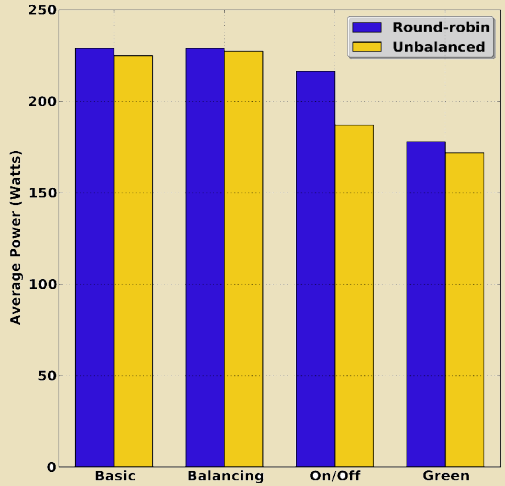




Less migrations
More energy-efficient

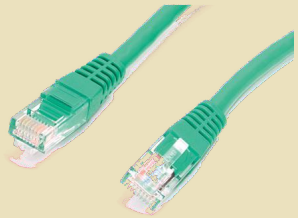
Results

- Test on real nodes leads to 25% of energy saved with GOC



- Significant energy savings are achievable.
- GOC can be integer in current and future Cloud infrastructures (with reservation, accounting, ...)

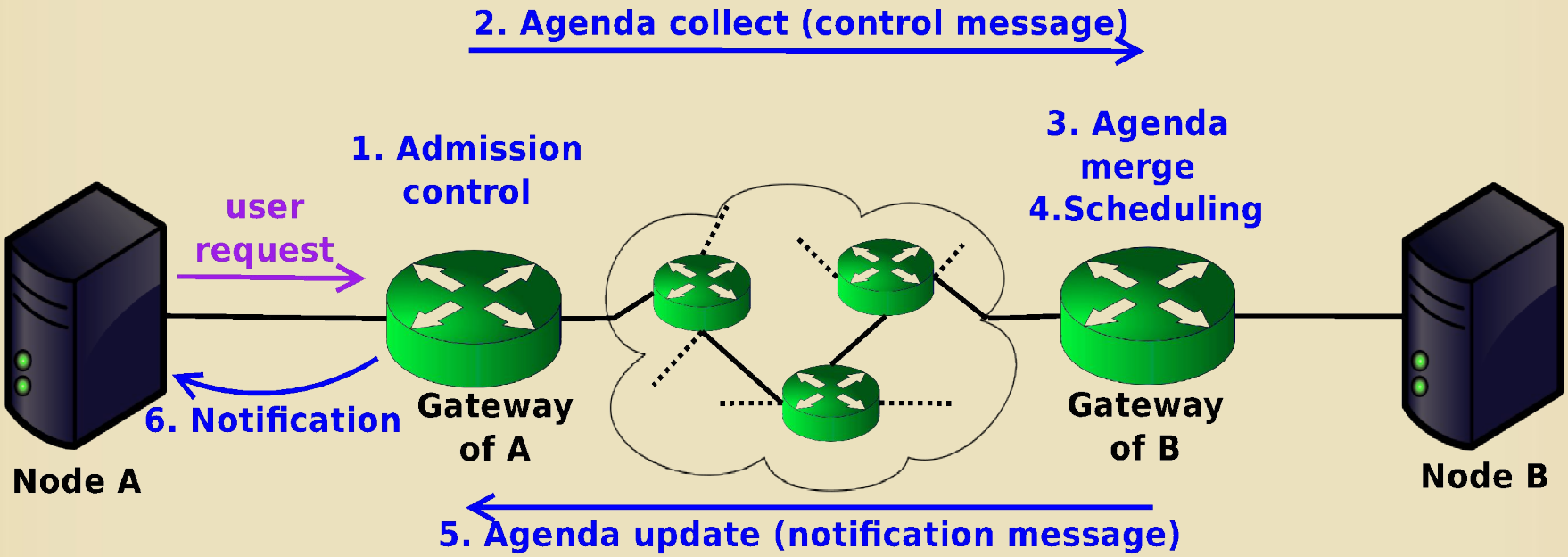
High-level Energy-awaRe Model for bandwidth reservation in End-to-end networkS



HERMES

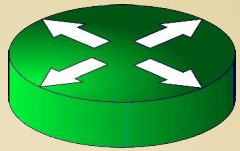
- Switching off unused nodes
- Distributed network management
- Energy-efficient scheduling with reservation aggregation
- Usage prediction to avoid on/off cycles
- Minimization of the management messages
- Usage of DTN (Disruptive-Tolerant Network) for network management purpose

Reservation process



DTN usage

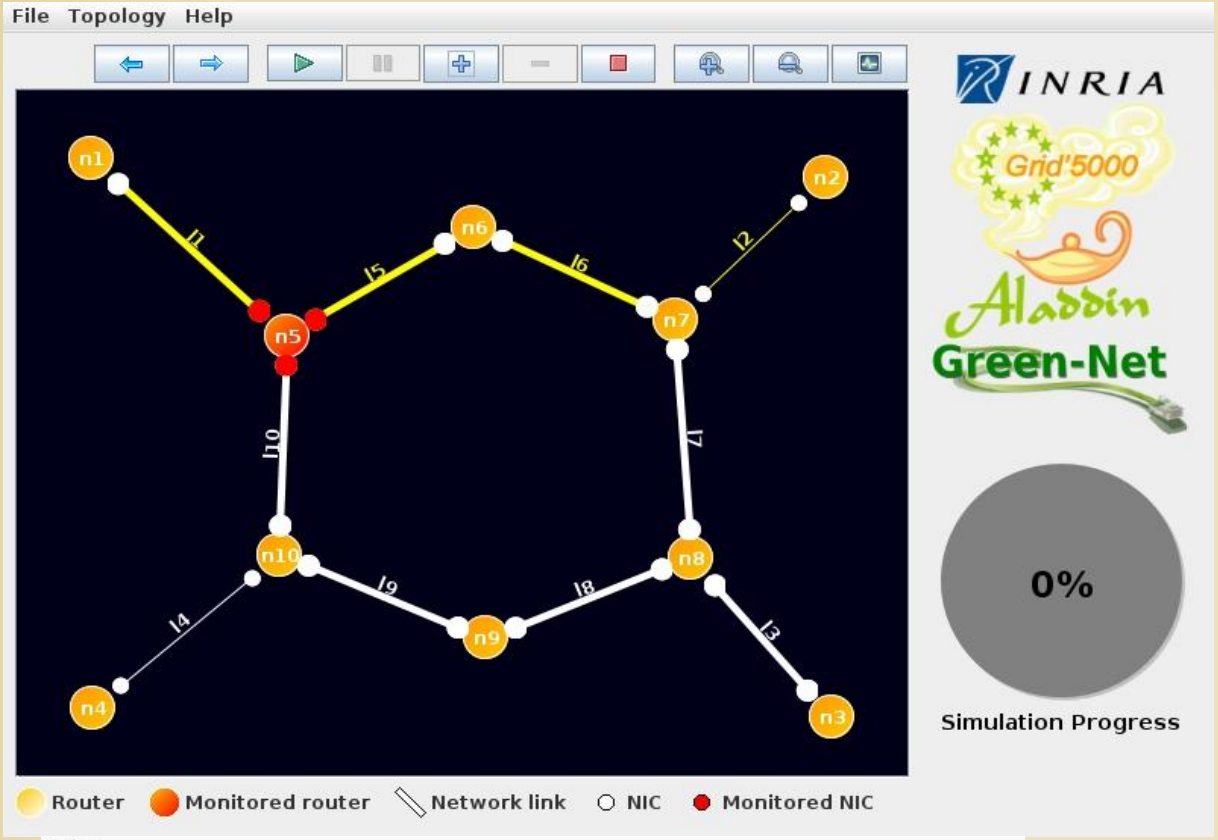
- Each reservation request has a TTL
  - if TTL = 0 → request to compute now, answer to give as soon as possible
  - otherwise, users can wait for the answer. The request moves forward into the network hop-by-hop waiting for the nodes to wake up. If the TTL is expired, the whole path is awaken.

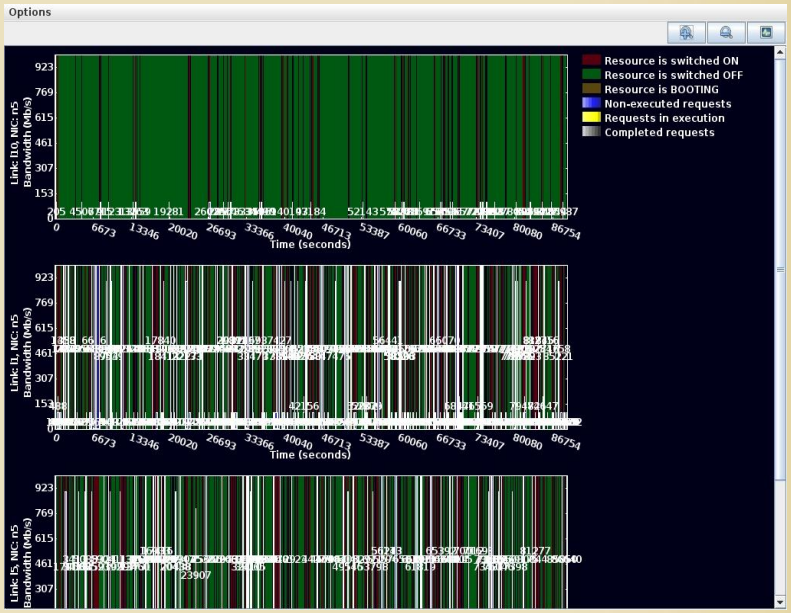


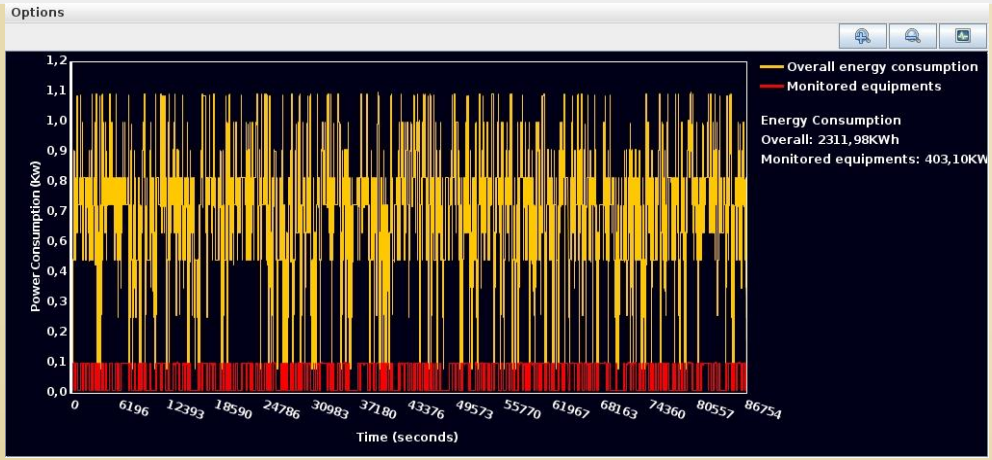
Simulation results

- BoNeS (Bookable Network Simulator)
- Written in Python (6,000 lines)
- Generates random network with the Molloy & Reed method or uses configuration file
- Generates traffic according to statistical laws:
  - submission times (log-normal distribution)
  - data volumes (negative exponential)
  - sources and destinations (equiprobability)
  - deadlines (Poisson distribution)

Replayer







2010 SuperComputing demo, Marcos Dias de Assunção

Comparison with other schedulings

- **First**: the reservation is scheduled at the earliest possible place;
- **First green**: the reservation is aggregated with the first possible reservation already accepted;
- **Last**: the reservation is scheduled at the latest possible place;
- **Last green**: the reservation is aggregated with the latest possible reservation already accepted;
- **Green**: HERMES scheduling;
- **No-off**: first scheduling without any energy management.
  → always before deadline

Simulations

- Network simulated: 500 nodes, 2 462 links.
- Random Network (Molloy & Reed method)
- All the nodes can be sources and destinations.
- Time to boot: 30 s.; time to shutdown: 1 s.
- 1 Gbps per port routers

Component	State	Power
Chassis	ON	150 W
	OFF	10 W
Port	1 Gbps	5 W
	100 Mbps	3 W
	idle, 10 Mbps	1 W

Results with a 30% workload

- 80 experiments for each value
- Four hour period of simulated time for each experiment
- Energy consumption in Wh

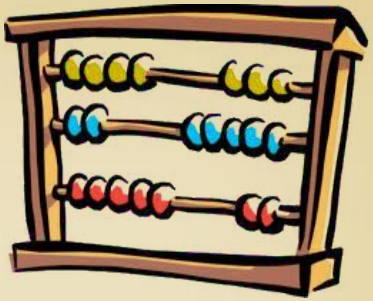
Scheduling	No off	First	First green	Last	Last green	Green
Average	412 306	205 270	203 844	204 949	196 260	203 342
Standard deviation	2 685	2 477	1 938	2 375	2 695	2 145
Accepted volume (Tb)	2 148	2 148	2 128	2 014	1 853	2 149
Cost in Wh per Tb	191.92	95.55	95.78	101.74	105.92	94.60

Different workloads

- 30%, 45% and 60%
- Average occupancy per link
- Compared to current case (no-off), HERMES could save 51%, 46% and 43% of the energy consumed depending on the workload

Workload	No off	First	First green	Last	Last green	Green
31%	191.92	95.55	95.78	101.74	105.92	94.60
46%	149.84	81.61	81.95	87.74	92.40	80.63
61%	130.45	74.73	74.91	80.09	84.63	73.79

Summary



- Complete and energy-efficient bandwidth reservation framework for data transfers including scheduling, prediction and on/off algorithms
- Validation of HERMES through simulations
- Perspective: to encourage network equipment manufacturers to design new equipments able to switch on and off and to boot rapidly.

Conclusions

Conclusions

- Proposition of ERIDIS, an energy-efficient reservation framework for large-scale distributed systems
- Proposition of EARI for data centers and Grids and validation on traces with measured consumptions
- Proposition of GOC for Clouds and validation on real nodes
- Proposition of HERMES for dedicated wired networks and validation through simulations

To use in production environments?

- HERMES : validation through simulations
- GOC : validation through prototype implementation with tool scenario
- EARI : validation through replay of real traces

→ ideas of EARI applied to OAR (batch scheduler)

→ currently under test on Grid'5000

http://wiki-oar.imag.fr/index.php/Green_OAR

Thank you for your attention!

Questions?

annececile.orgerie@ens-lyon.fr

http://perso.ens-lyon.fr/annececile.orgerie

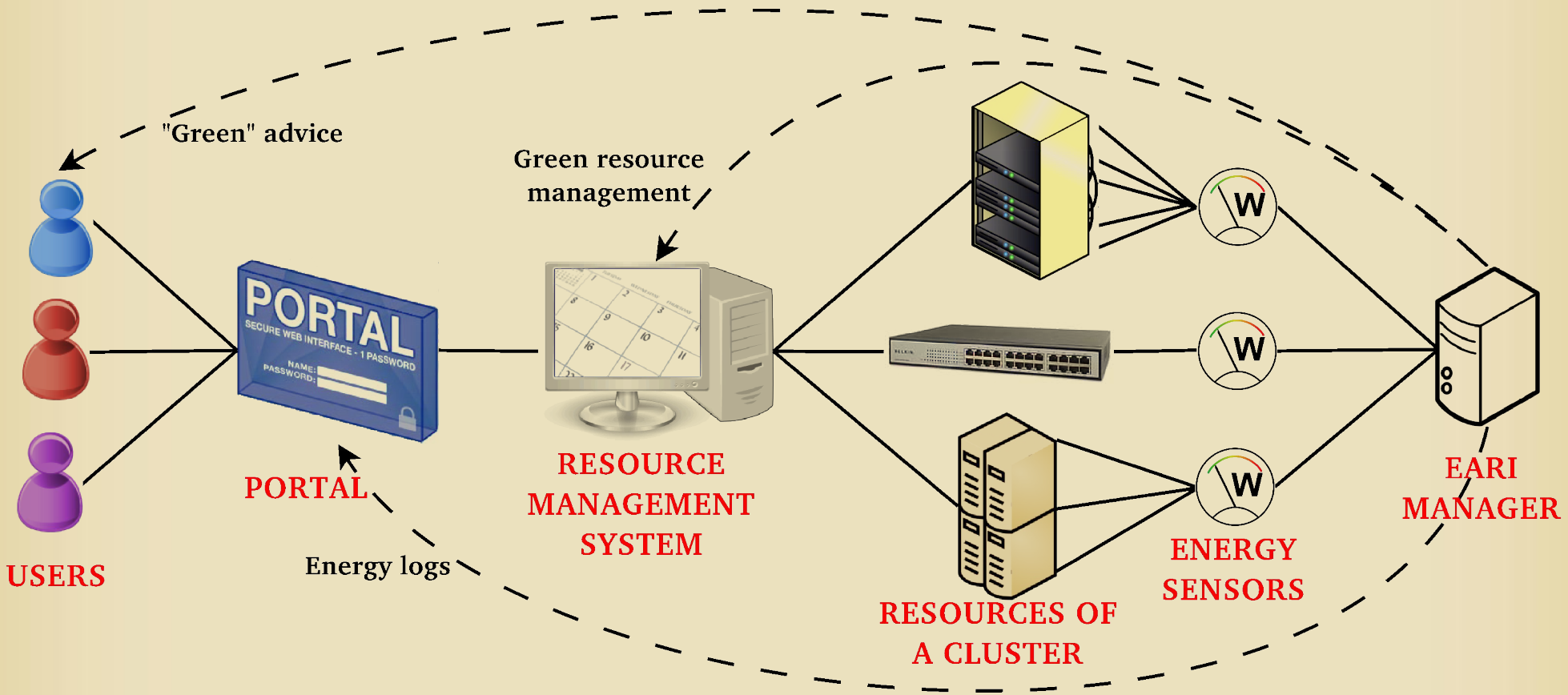
ENS DE LYON RESO INRIA RHÔNE-ALPES LIP UCBL CNRS CENTRE NATIONAL DE LA RECHERCHE SCIENTIFIQUE UNIVERSITÉ DE LYON

Energy-Aware Reservation Infrastructure (EARI)

The main features are:

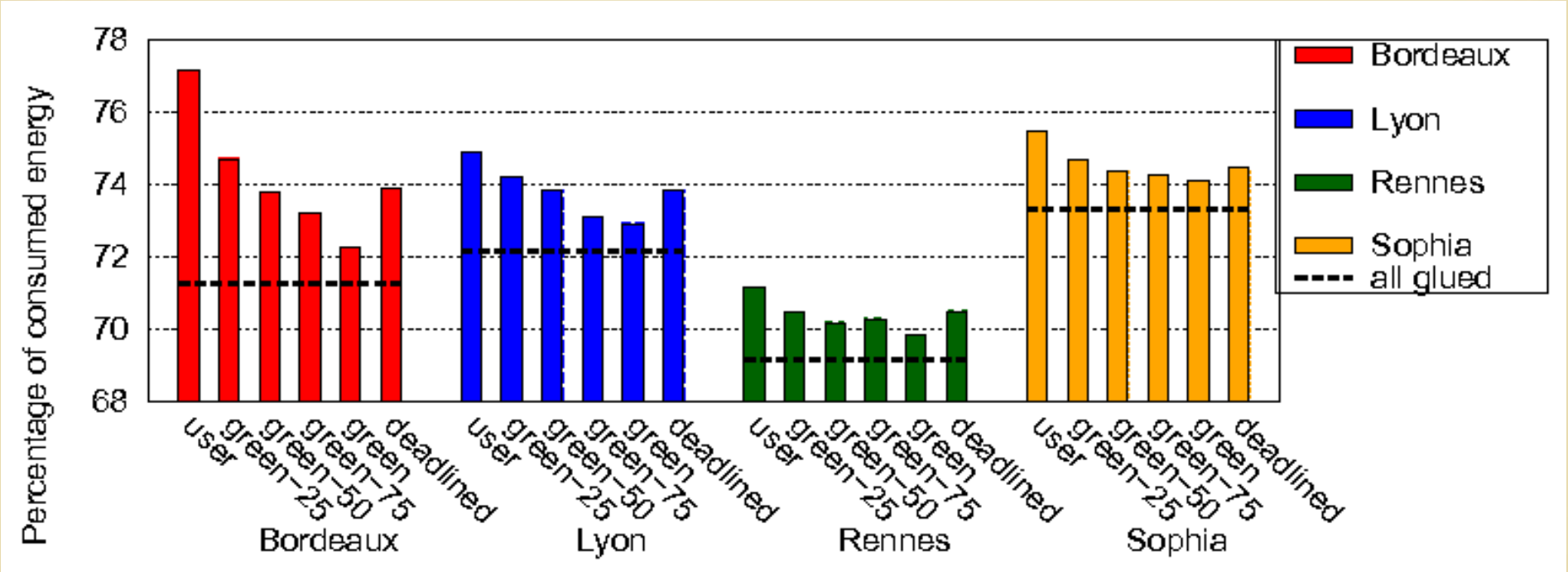
- **Switch off** unused computing resources;
- **Predict** next use;
- **Aggregate** the reservations by giving green advice to the users.

EARI architecture



Experimental validation of EARI

- Real traces of an experimental Grid: Grid'5000
- 4 different sites, one year period



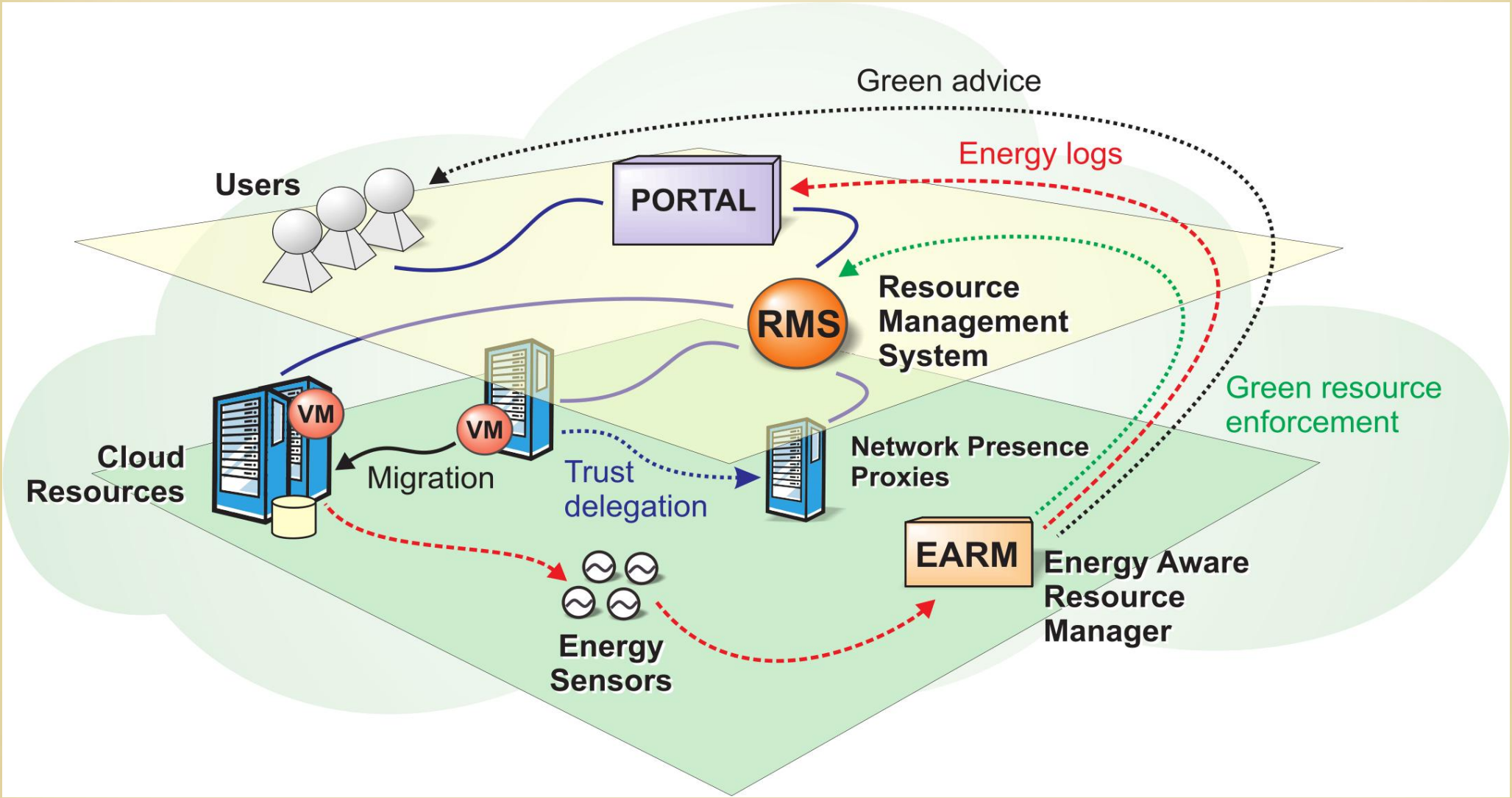
Extrapolation to the whole Grid

209,159 kWh for the full Grid'5000 platform (without aircooling and network equipments) on a 12 month periods (2007)

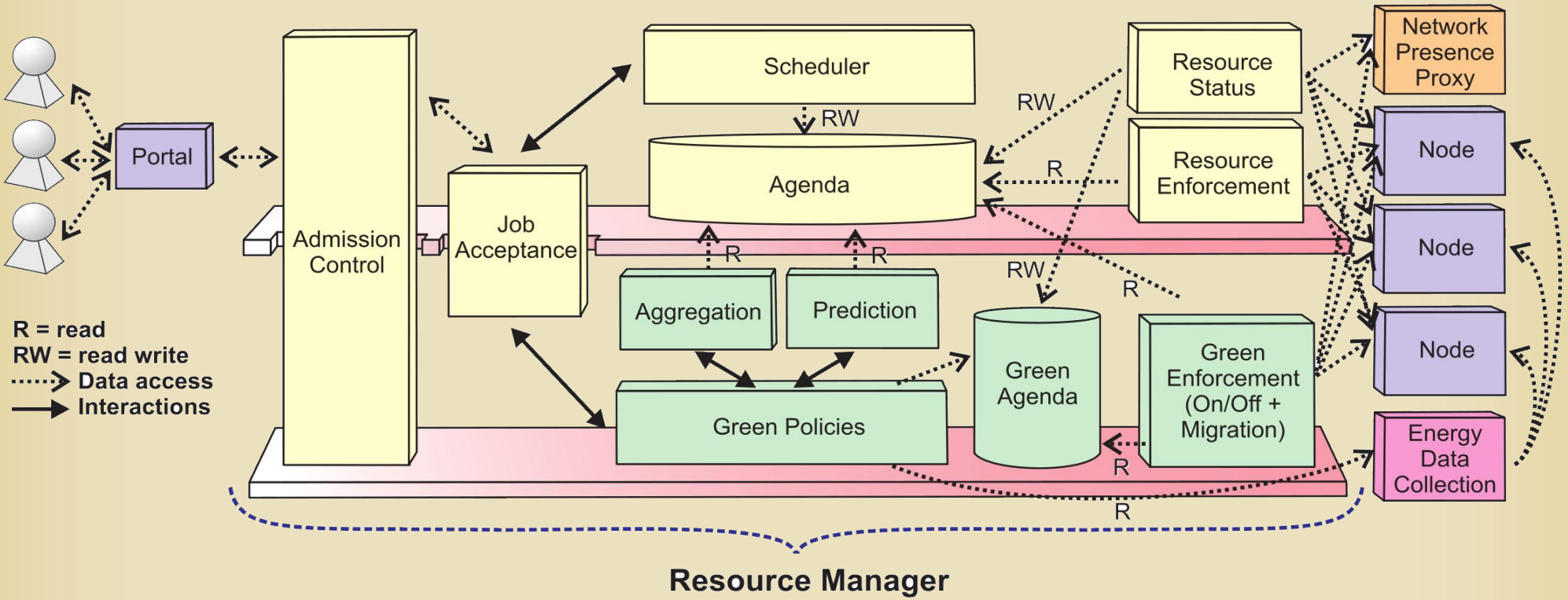
It represents the consumption of a french village of 600 inhabitants.

So roughly, a village of 1200 inhabitants for the whole infrastructure (cooling, network).

GOC Architecture

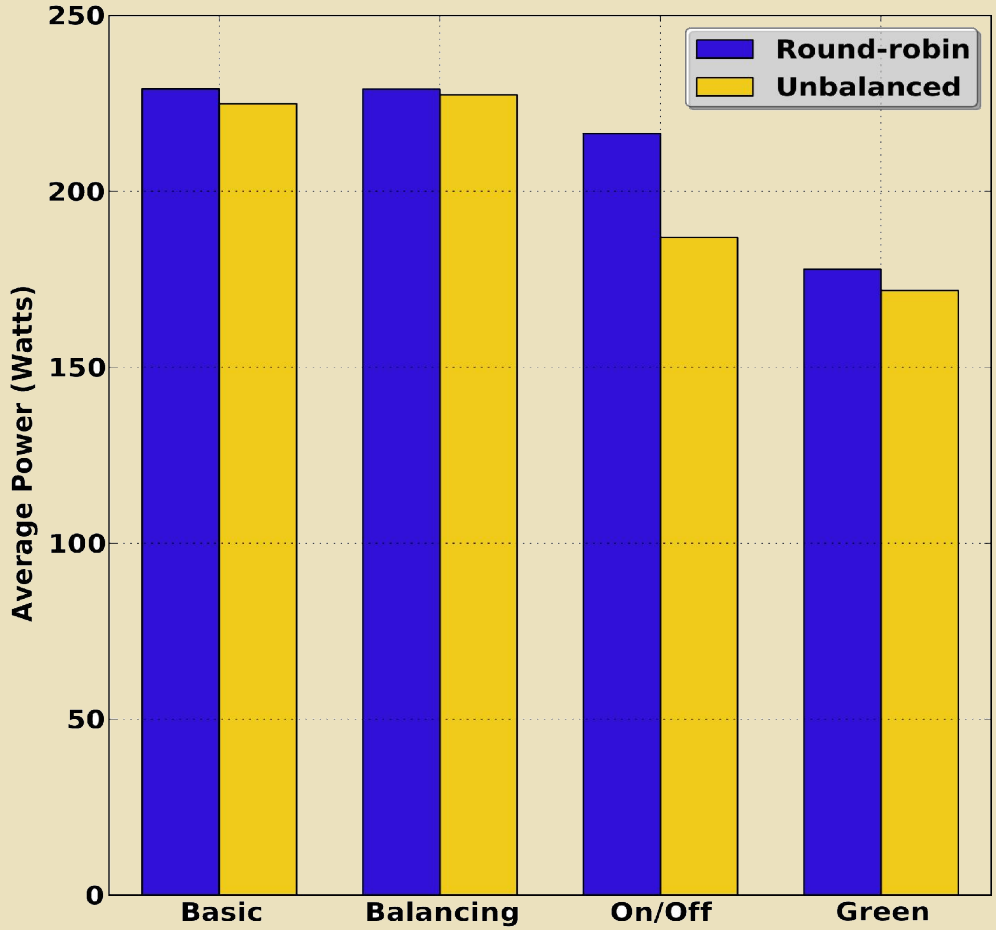


GOC Resource Manager



- Smooth integration in Cloud infrastructure

Comparison between the scenarios



Same execution time for all the experiments