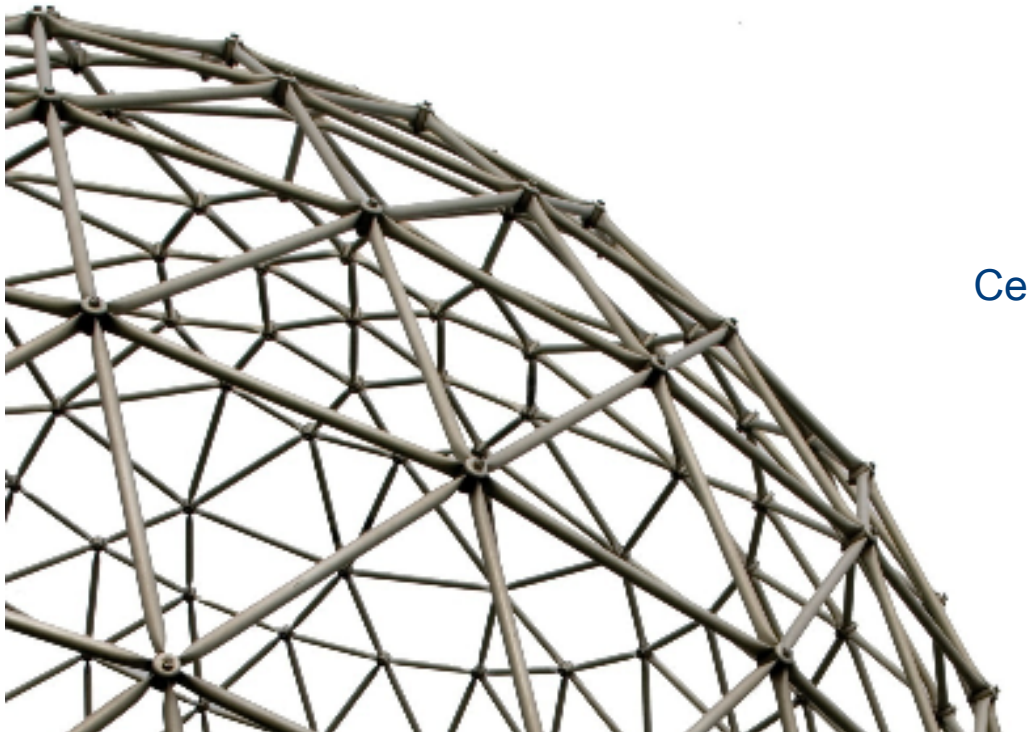
Green Communications through Integrated Network Management

UK4G Shanghai Workshop

12th September 2010



Prof. Z. Sun
Centre for Communication Systems Research
University of Surrey
Guildford, Surrey
GU2 7XH
Email: Z.Sun@surrey.ac.uk,
Tel: 01483 68 9493

www.surrey.ac.uk

Outline

- Background
- Greening Networks
- What is the problem?
- Service ON/OFF model
- Proxy Model
- Autonomic Green Network Model
- Case study
- Future Challenges

Backgroung

- Energy consumption becomes a BIG issue:
  - 3% of the world-wide energy consumed by ICT (Information and Communications Technology)
  - ICT direct contributed to 2% -10% CO2 footprint
  - $15 Billion cost for network equipment and networks

  (USA 2006 from LBNL, 1kWh=$0.10 in US,

  1TWh = $100 millions)

- A mission to save energy
- Lower CO2, and save $$$

Greening Networks

- Towards reducing energy consumption in network systems
  - ON/OFF model
  - Proxy model
- Towards reducing energy consumption in networks
  - Green services
  - Visulization, smart grid, etc.

"Desktop computing accounts for 45 percent of global carbon emissions from information technology."
- govtech.com

What is the problem?

“The problem is network presence” – Key Christensen

• “Today, billions of dollars’ worth of electricity are used to keep Ethernet (and other) connected devices fully powered on at all times only for the purpose of maintaining this connectivity.” (Bruce Nordman, 2007)

•The network presence drives PCs to be left fully powered-on at all times.

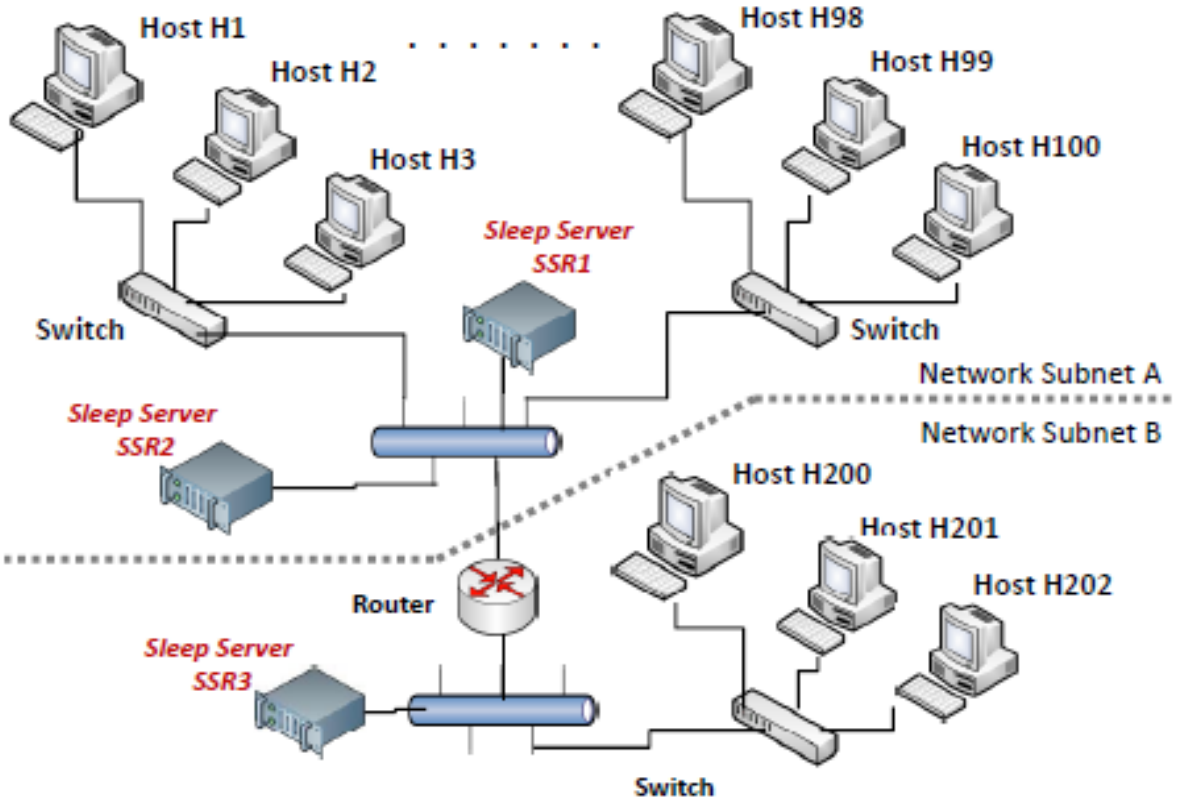
Service On/OFF model

State	Power
Normal Idle State	102.1W
Lowest CPU frequency	97.4W
Disable Multiple cores	93.1W
Base Power	93.1W
Suspend state (S3)	**1.2W**

Desktop PC	
Active State	**>140W**
Idle State	100W
Sleep Mode	1.2W

- Large Differences between ON & OFF

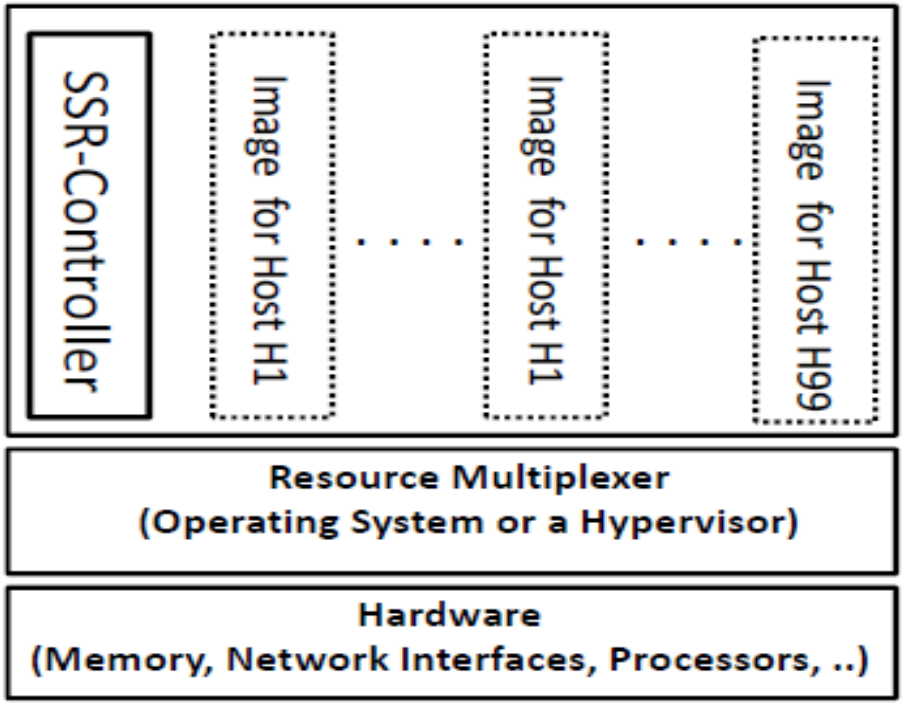
SleepServer — proxy model



- An enterprise setting

- From Y. Agarwal, S. Savage, and R. Gupta, ”SleepServer: Energy Savings for Enterprise PCs by Allowing them to Sleep,” *Proceedings of the USENIX Annual Technical Conference, June 2010.*

SleepServer model



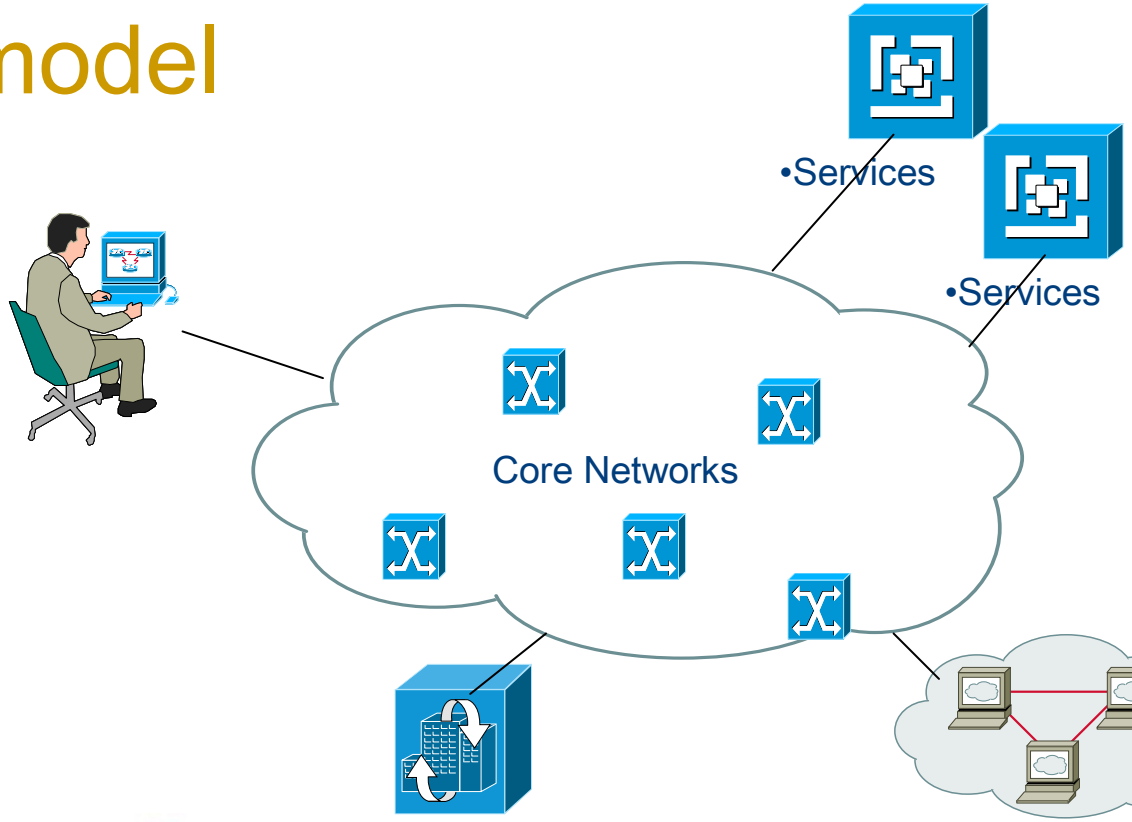
• Energy savings ranging from 27% to 86% with an average savings of 60% for heterogamous PCs

A SleepServer serves a collection of host PCs (H1, ...H99).

All resource sharing and access to the hardware is mediated by the SleepServer controller software module running on the SleepServer.

www.ee.surrey.ac.uk/CCSR

Autonomic Green Network model



•Network Measurement → Energy Model → •Policy → •Service •Provision

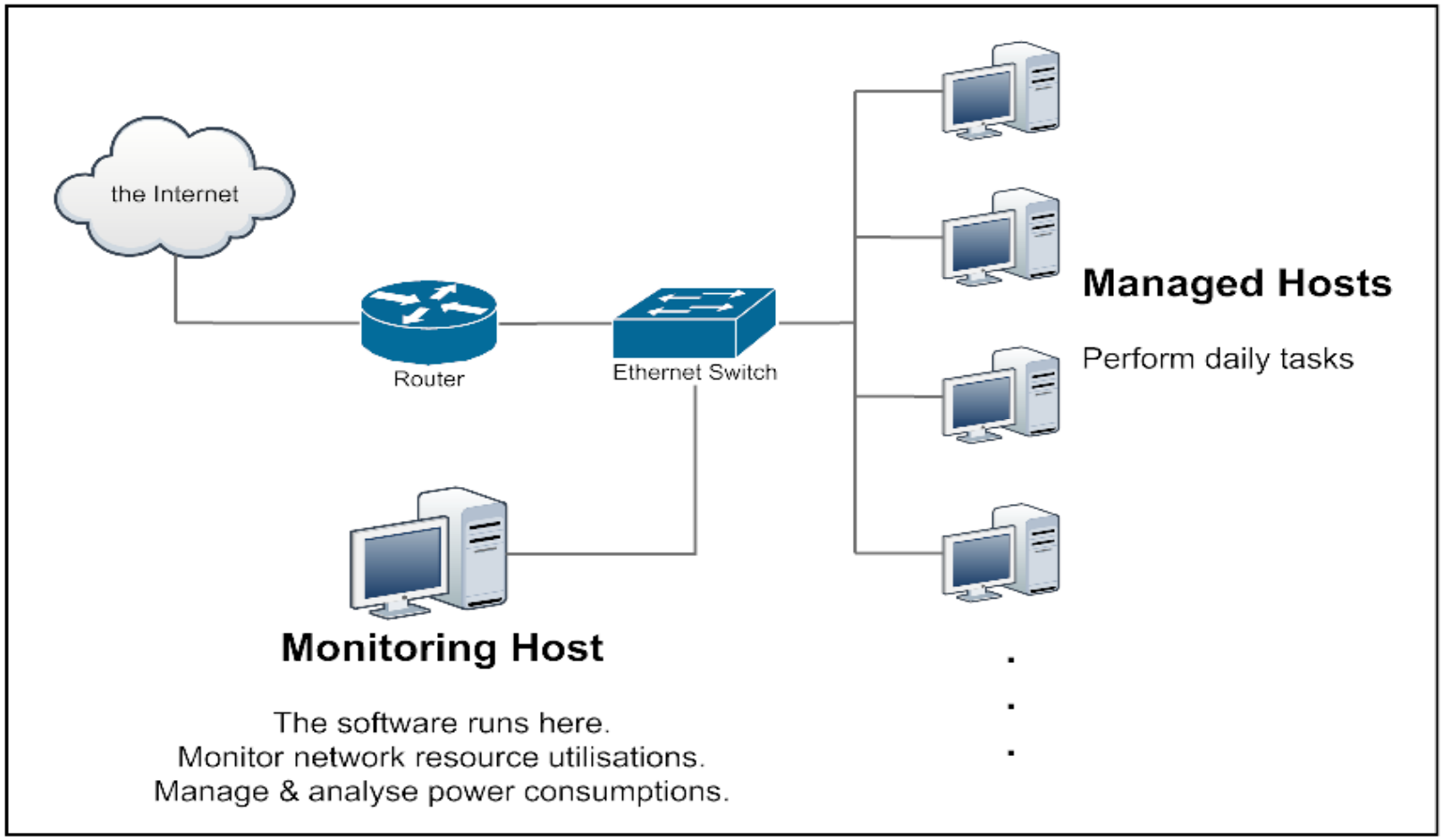
Case studies

- Network resource
  - Hardware: hosts energy consumption, CPU load
  - Link: bandwidth occupation
- Improve utilisation efficiency
  - Focus on: energy consumption management
- Trend analysis
  - More effective network management

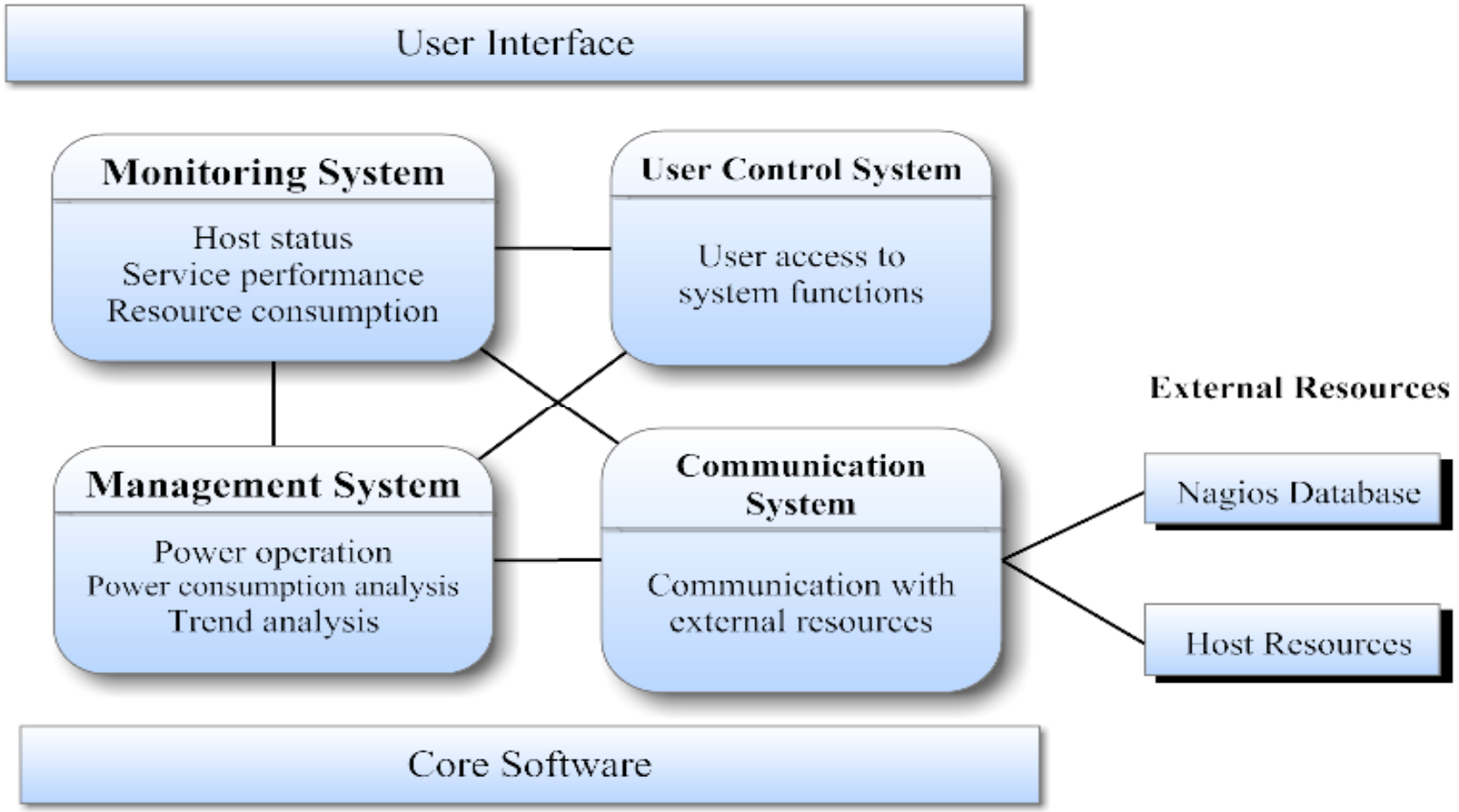
Objectives

- Network management system
  - Deployed in small-scale LAN
- Monitor hosts:
  - CPU load, hard disk usage
  - Traffic flow rates
  - Energy consumptions
- Manage network resources:
  - Energy-saving operations
  - Trend analysis (CPU load, traffic rate, energy consumption)

Deployment Scenario



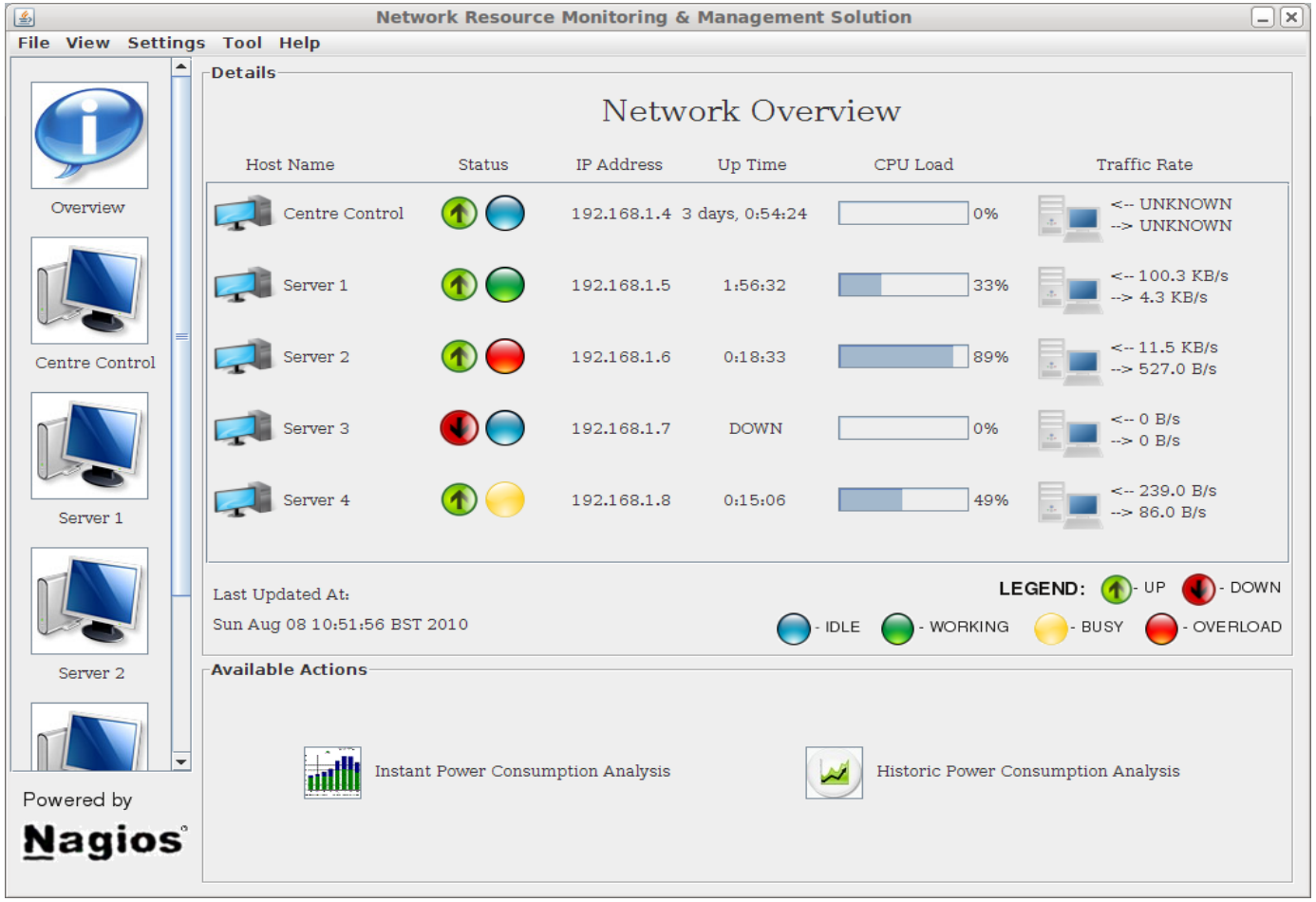
System Design



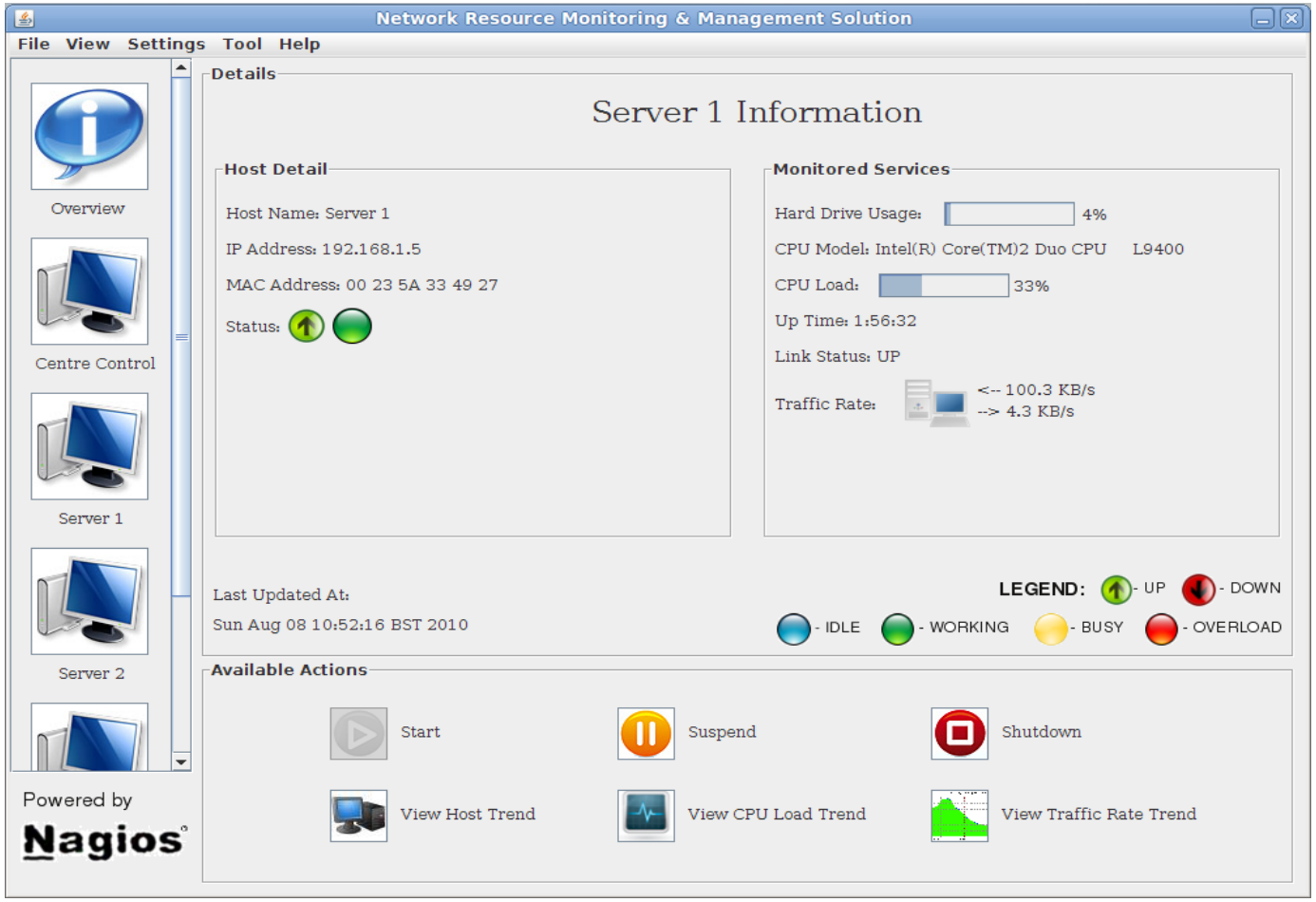
Functional Implementations

1. Network monitoring
   - Monitor host details, status, performances, energy consumptions.
   - Updated per minute.
2. Energy-saving operations
   - Put idle hosts to sleep.
   - Auto-detect host activity to avoid data loss.
3. Trend analysis
   - Analyse host's status, CPU load, traffic rate, and energy consumption's trends.
4. Customisation features
   - Parameters are adjustable according to network needs.

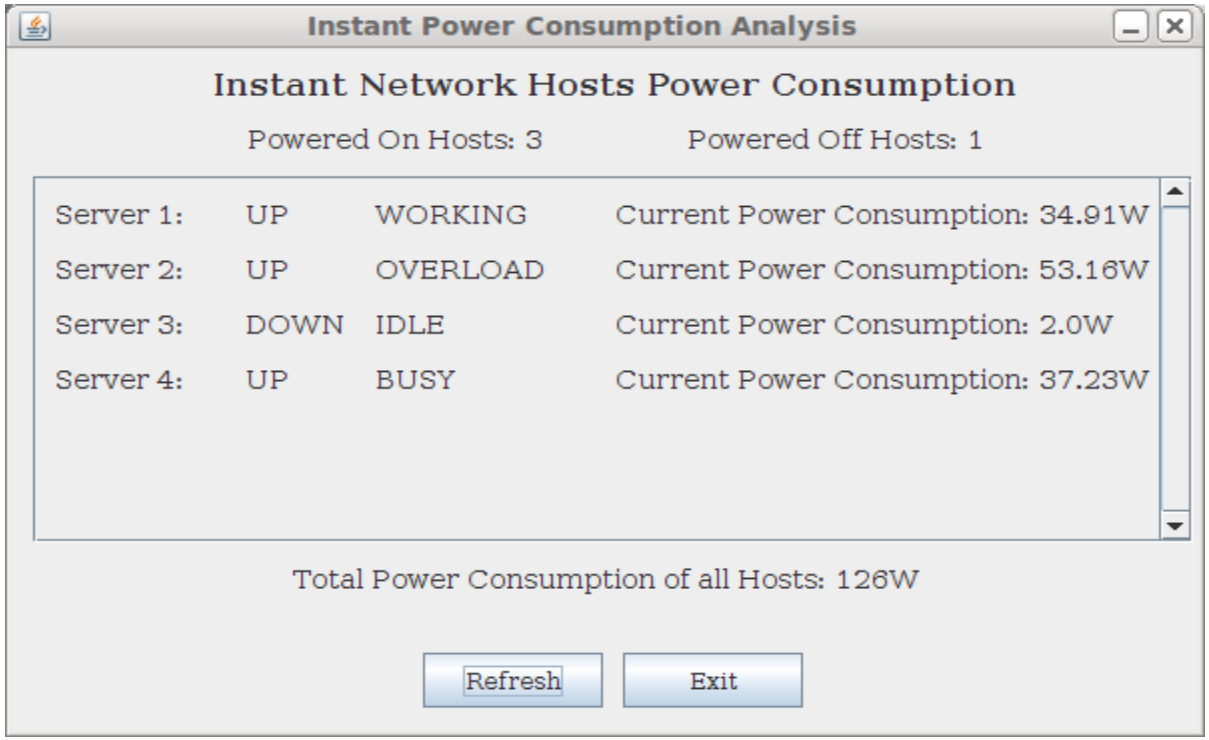
Network Monitoring: Overview



Network Monitoring: Host Detail

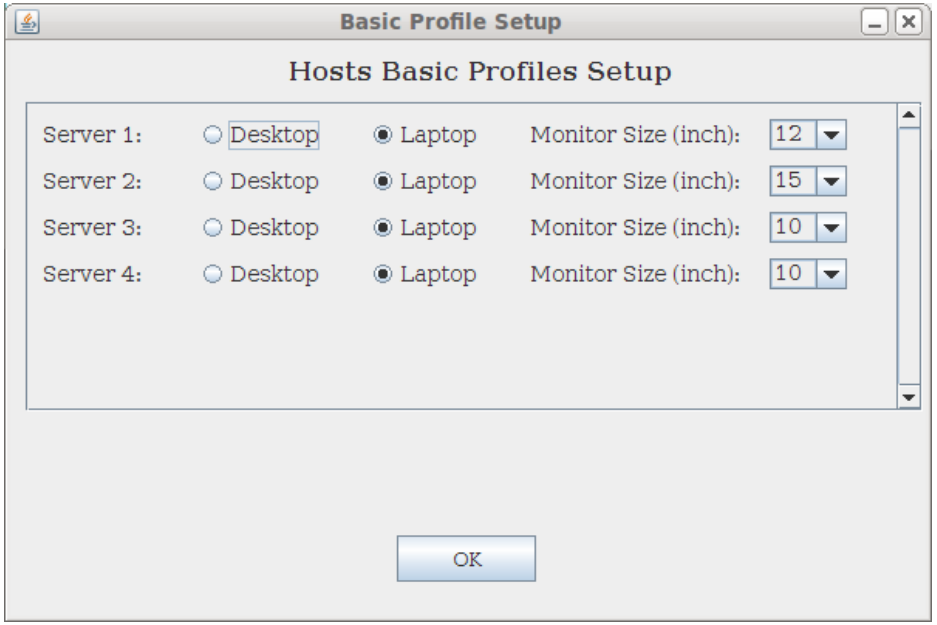


Network Monitoring: Energy Consumptions



Energy Consumption Calculation

- Based on:
  - Manually set by user:
    - Host type (desktop/laptop)
    - Monitor size (inch)
  - Dynamically monitored:
    - CPU utilisation (%)



Energy Consumption Calculation (cont'd)

- Host Down:
  - Desktop: 5W,
  - Laptop: 2W
- Host Up:

Reference values: $I = 1.0$

Desktop: $P_{typical-monitor} = 20W, S_{typical-monitor} = 17(inches)$

$P_{typical-CPU} = 65W.$

Laptop: $P_{typical-monitor} = 10W, S_{typical-monitor} = 14(inches)$

$P_{typical-CPU} = 45W.$

$$P_{monitor} = P_{typical-monitor} + I \times (S_{monitor} - S_{typical-monitor})$$

where $P_{monitor}$ is monitor energy consumption, $S_{monitor}$ is monitor size (inch). I is index.

$$P_{CPU} = P_{typical-CPU} \times (0.4 + 0.6L_{CPU})$$

where P_{CPU} is CPU energy consumption, L_{CPU} is CPU utilisation in %.

$$P_{host} = P_{monitor} + P_{CPU}$$

where P_{host} is host's total energy consumption.

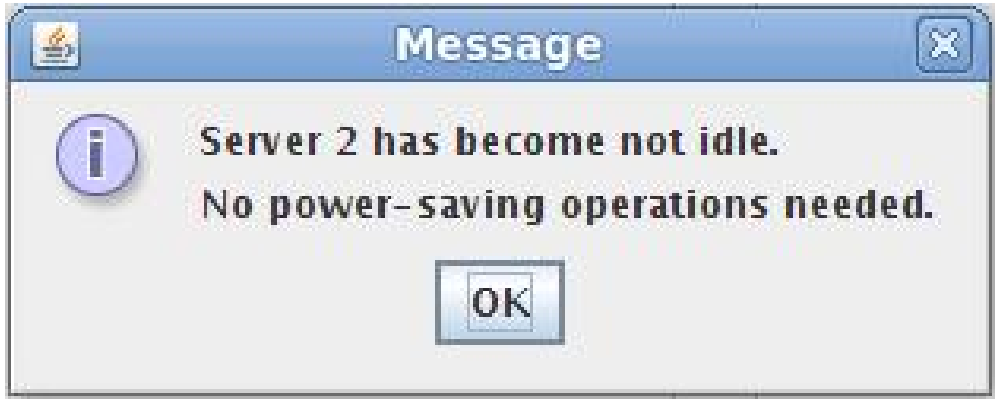
Energy saving operations?

- Monitor host load status
- If a host becomes idle, start its timer
- When reaches the thresholds, prompt:

  (2-level idle timers) (default timeout: 70 seconds)

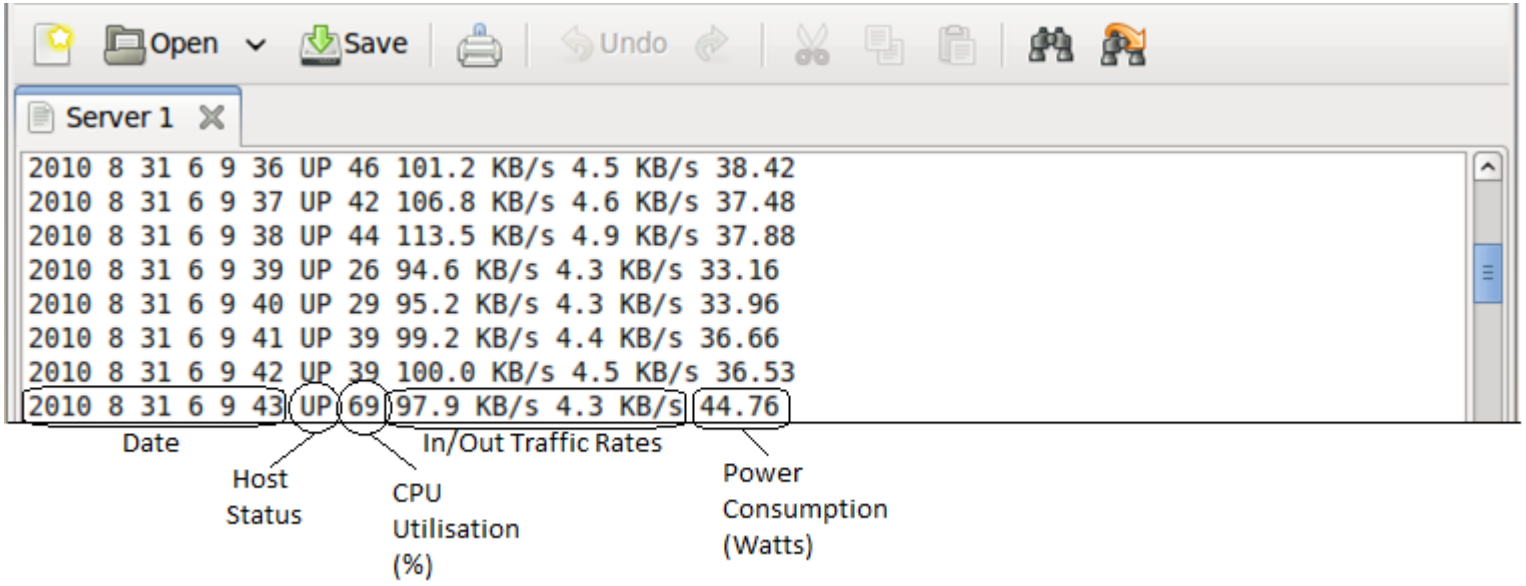
Automatic Host Activity Detection

- What if the host becomes busy during the timeout?
- System will halt the timer.

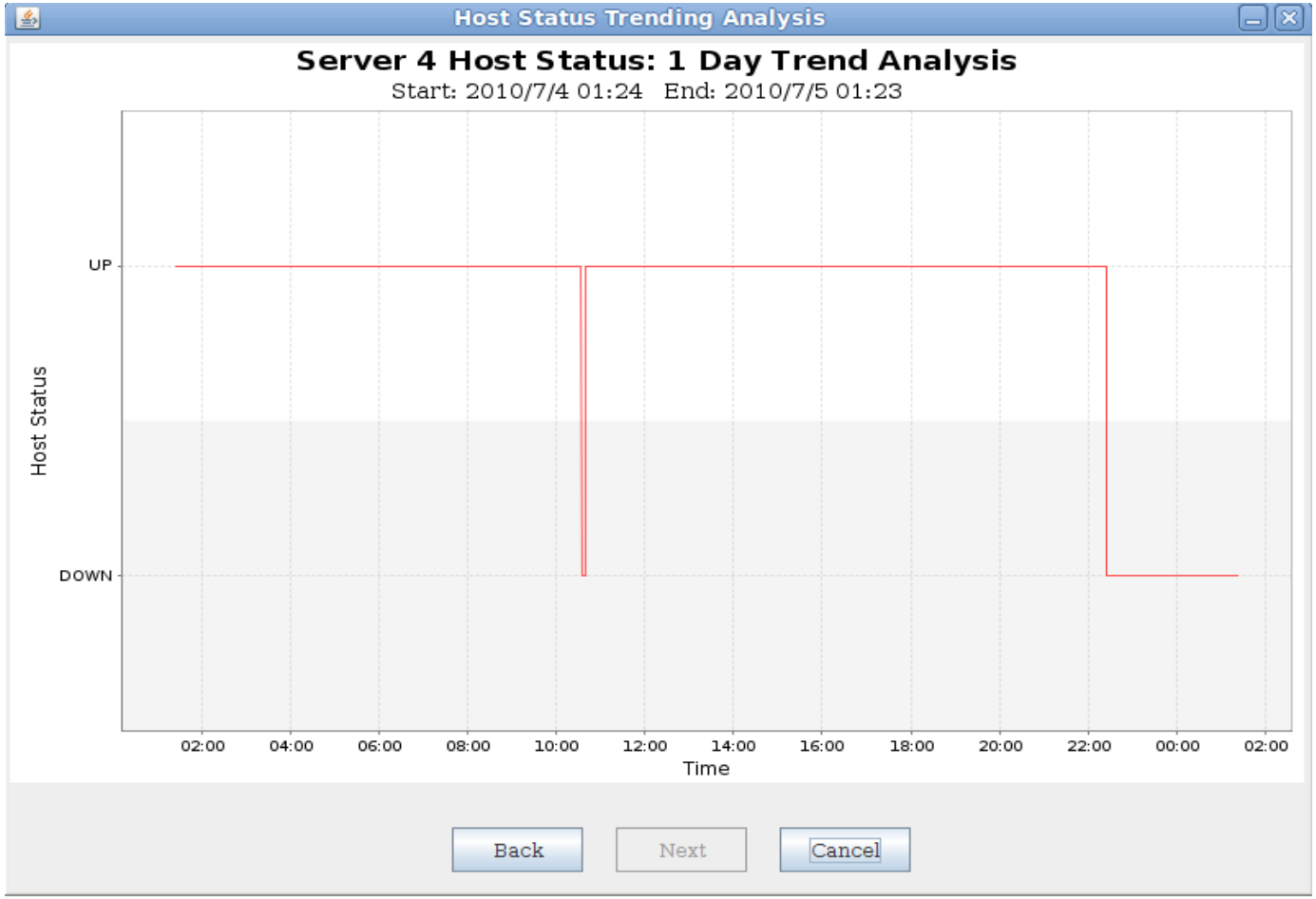


How to keep records?

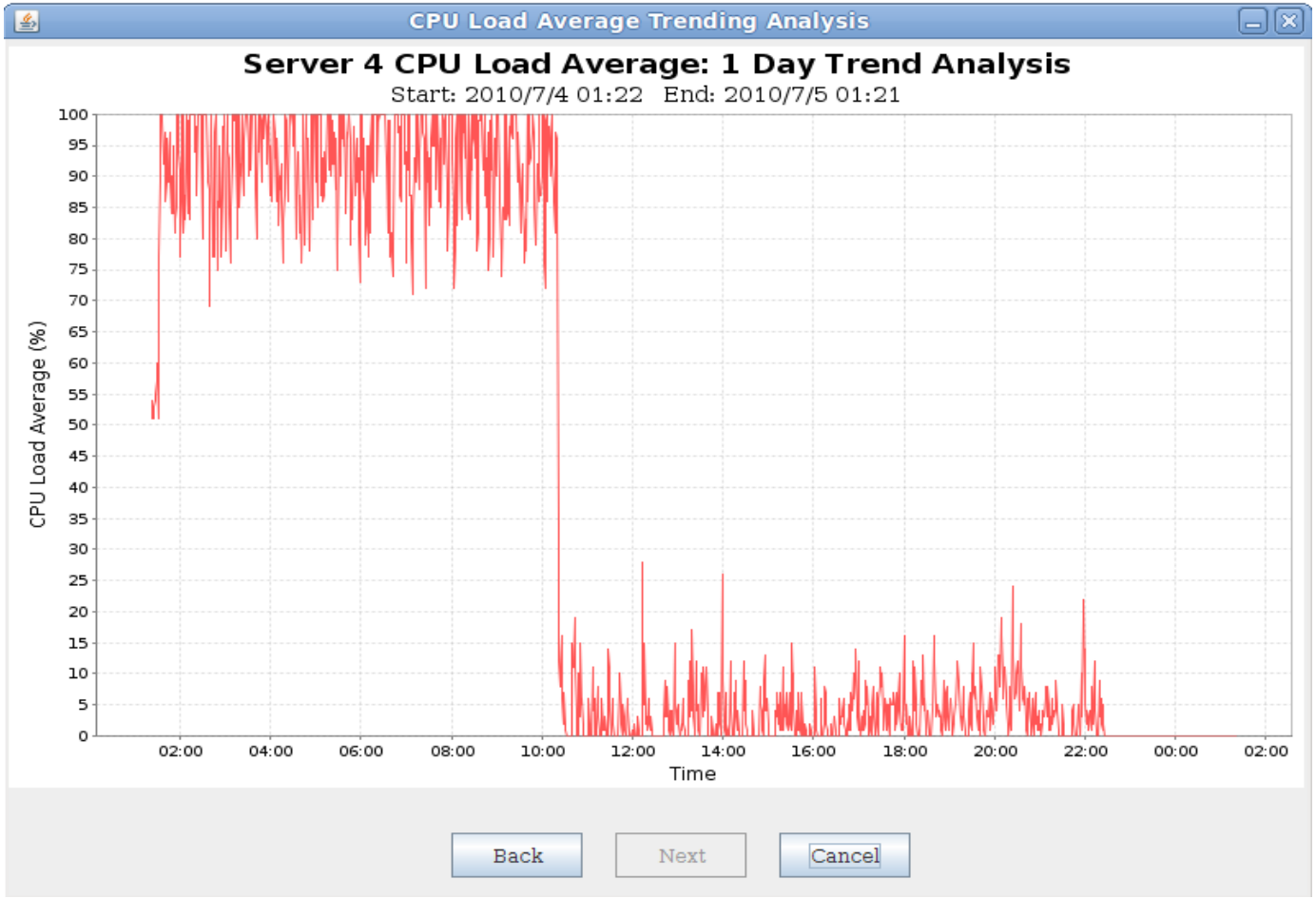
- Log files
  - One for each host
  - In plain text form to save disk space
  - Record all information in one entry / minute



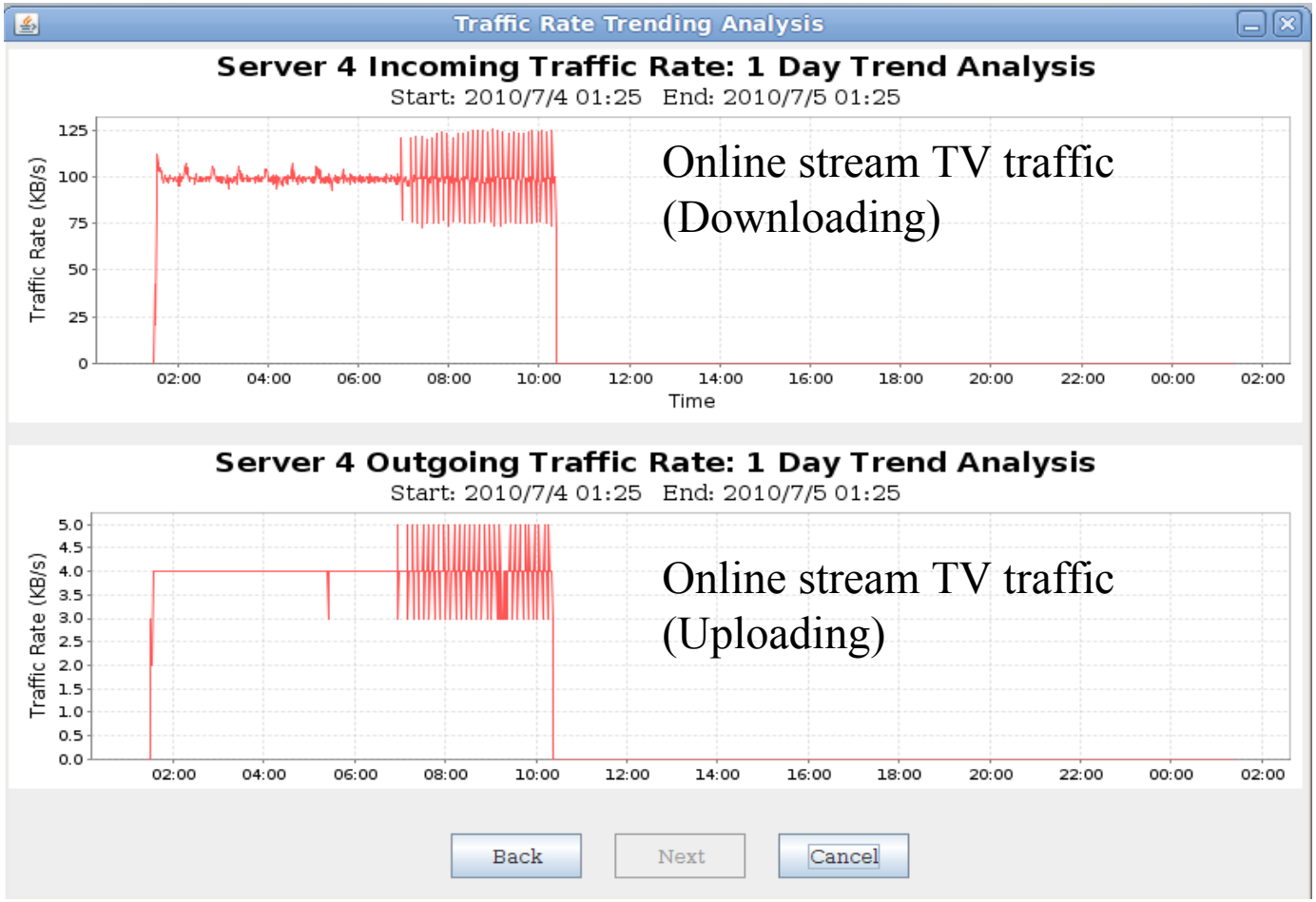
Trend Analysis:
Host Status (1-day example)



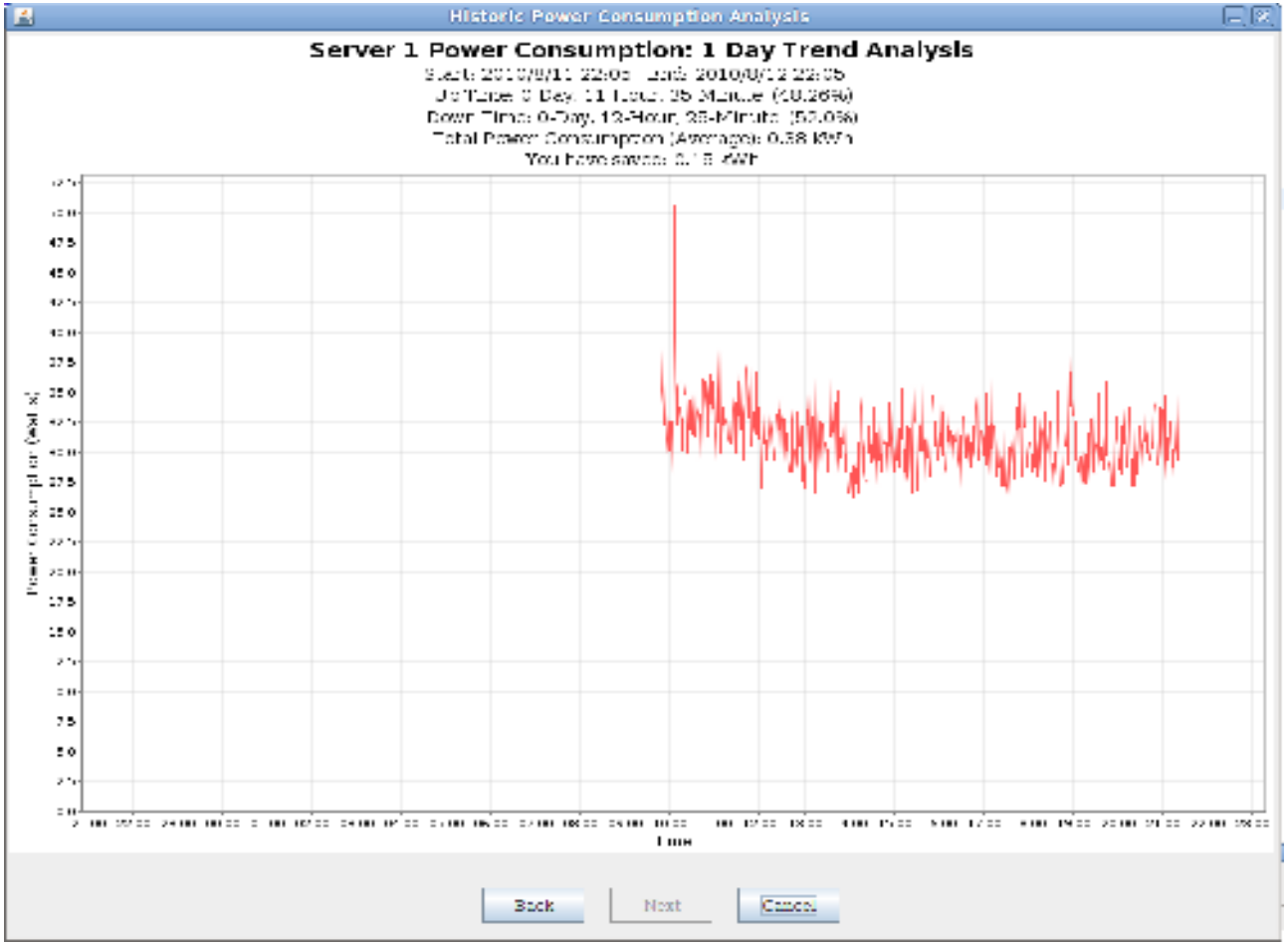
Trend Analysis:
CPU Load Average (1-day example)



Trend Analysis:
Traffic Rates (In/Out, 1-day example)

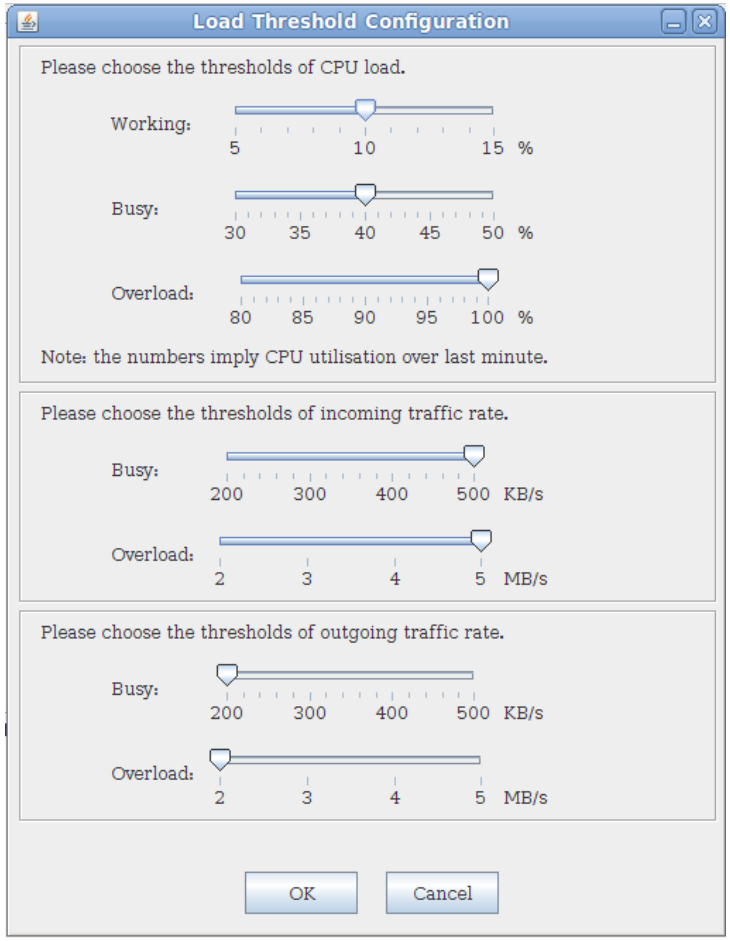


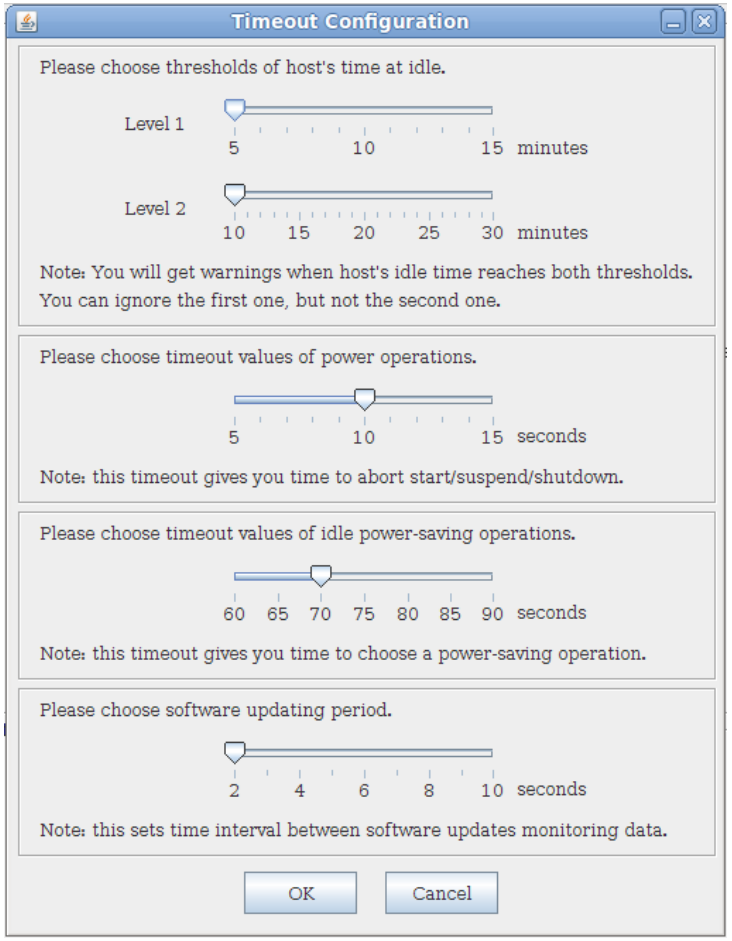
Trend Analysis:
Energy Consumption (1-day example)



Default setting: threshold & timeout

- Adjustable by specific network needs

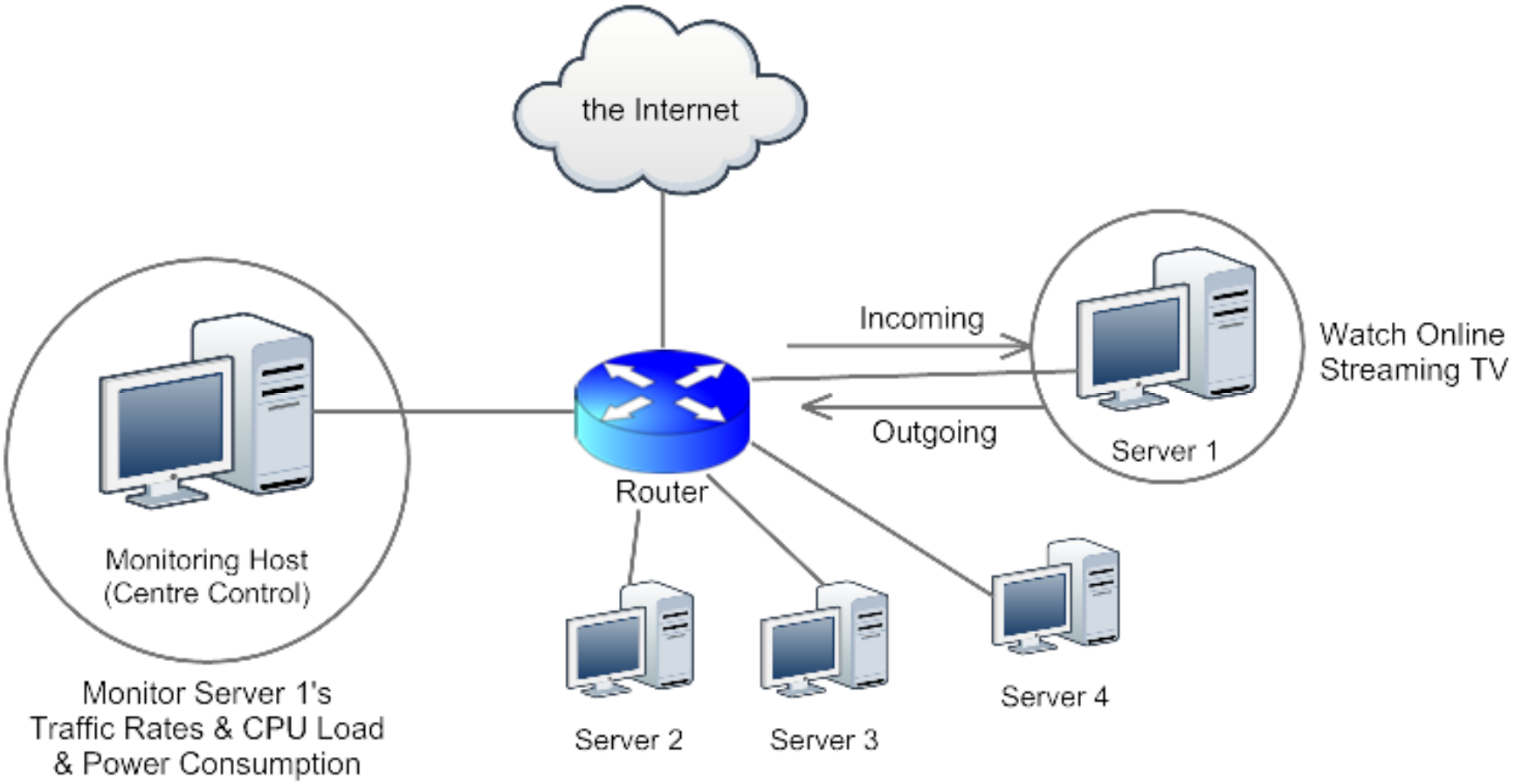




Development Considerations

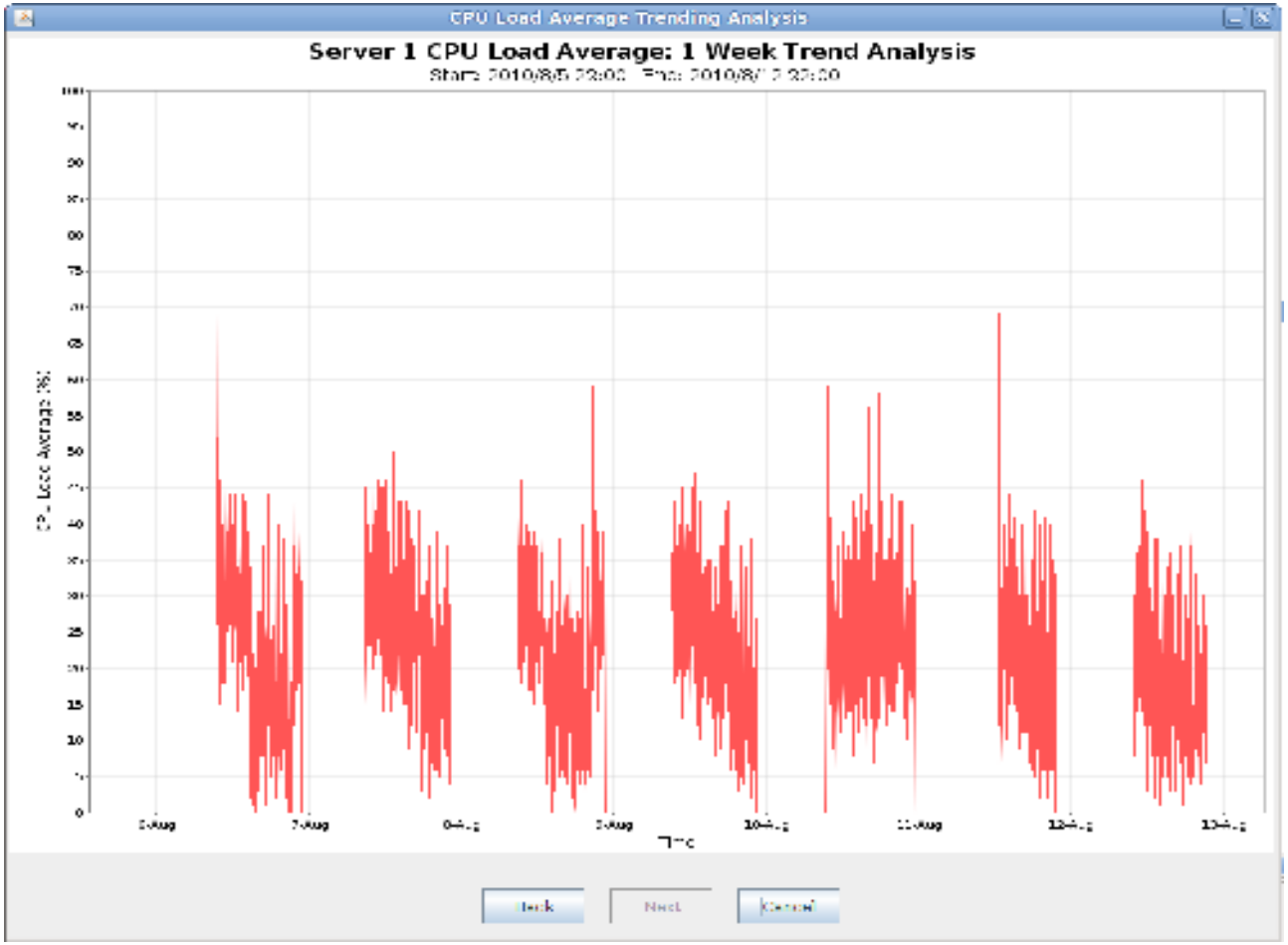
- Low algorithm complexity
  - Tested: little computing overhead
- GUI design
  - Usability, easy access to all features
  - Maximum informational
- Automatic
  - Work out-of-the-box if user followed user manual

Example results of one week

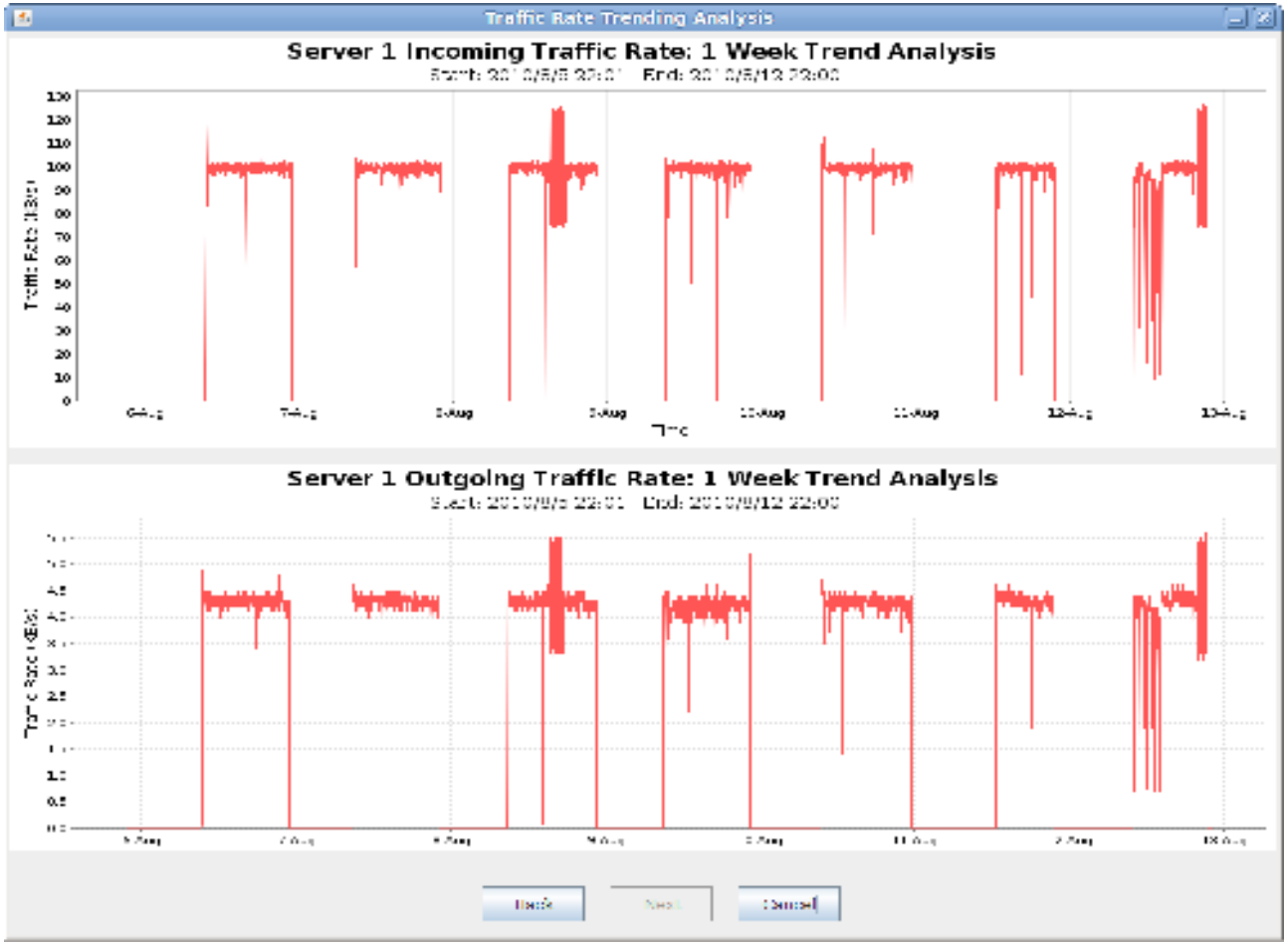


Host Status Trend

CPU Load Average Trend



Traffic Rates (In/Out) Trend



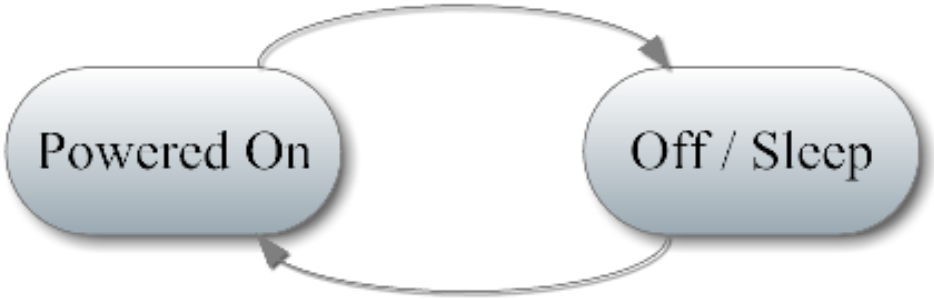
Energy Consumption Trend

Results (1/2)

- Calculated from log file of Server 1:
  - Energy consumption in active status: 3.01kWh
  - Energy consumption in idle status: 0.95kWh
- If server 1 was not managed (put to sleep), 0.95kWh will be wasted.
- Hence, $\frac{0.95}{0.95+3.01} \approx 24.0\%$ of total energy consumption have been saved by the energy management function.

Results (2/2)

- Up Time: 52.94% (3-Day, 16-Hour, 56-Minute)
- Down Time: 47.06% (3-Day, 7-Hour, 4-Minute)



- More effective management can be achieved if state transition is predictable by profiling user behaviour which is very difficult and challenging

Conclusion

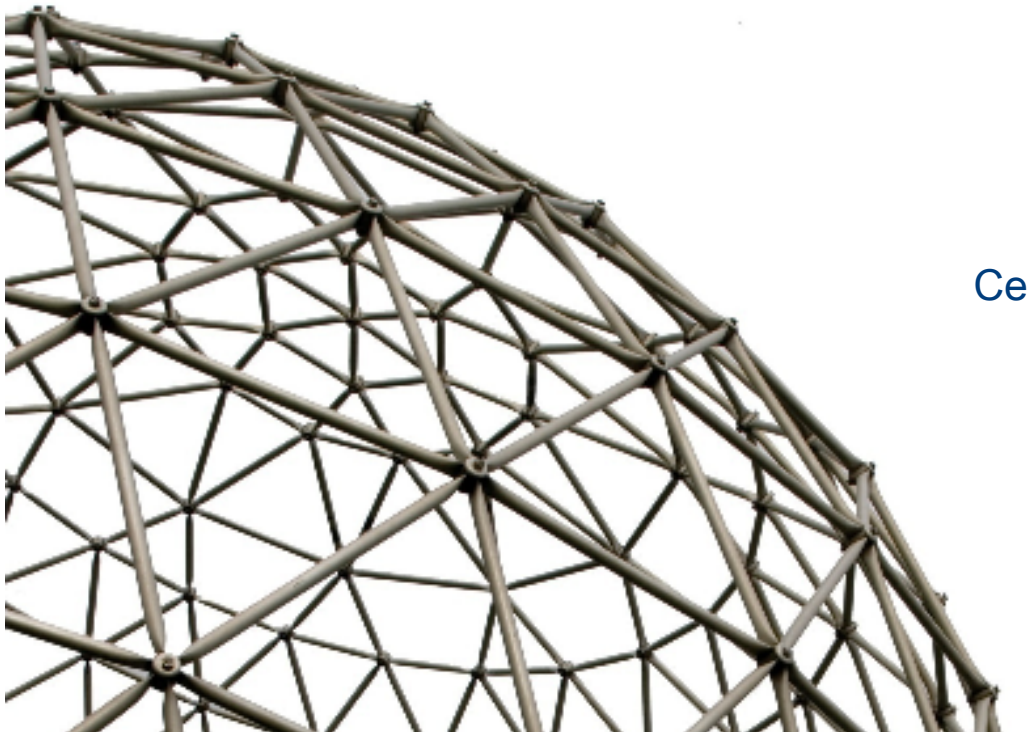
- For hosts under typical load, about a quarter of overall energy consumption can be saved.
- Host status trend prediction became possible via analysis, leading to more effective network management.
- This is one aspect of the solution for green communications through network and service management

Challenges for the future

- Network equipment
- Network hosts
- Wireless base stations
- Cloud data centres
- Distributed applications

Thank you for your attention.

Any questions?



Prof. Z. Sun
Centre for Communication Systems Research
University of Surrey
Guildford, Surrey
GU2 7XH
Email: Z.Sun@surrey.ac.uk,
Tel: 01483 68 9493

www.surrey.ac.uk