Research Project
A study on energy consumption of cluster nodes toward Green Clouds

Vesselin Hadjitodorov
Master education System and Network Engineering

Supervisors:
Dr. Paola Grosso
Drs. Cosmin Dumitru
Drs. Ralph Koning
System and Network Engineering Research Group

Science Park 904,
1098XH Amsterdam, The Netherlands

February 2011

Contents

1 Introduction

1.1 Energy and environmental problems

Currently over 50% of the electricity worldwide is produced from burning fossil fuels. This trend is expected to continue in the next 25 years and even after [1]. A lot of effort is spent on developing renewable energy sources, which have a significantly lower CO_2 footprint, but energy produced with these technologies is still more expensive compared to burning fossil fuels or nuclear energy.

The amount of energy consumed by the ICT industry is growing constantly and it's estimated to double every 6 years [2]. Currently it is comparable to the aviation industry and it's expected to overtake it. This effects are due to the growth in quantity, scale and power consumption of the computing and site equipment worldwide. We are close to the point, where the cost of energy to operate computing equipment will exceed the cost of acquiring it, even over its relatively short (3 to 5 year) exploitation period.

If the energy efficiency of ICT equipment is improved, the rate of growth of the power consumption can be decreased. This will lower energy costs and the CO_2 emissions. In this research project I examine the power consumption of system components in nodes. The results allow analyzing the energy needs of the nodes and propose improvements that will increase their efficiency.

1.2 Research question

What is the influence of system components in cluster nodes on the total energy consumption?

Although a lot of information is available about the amout of power consumed by each system component, in most cases it is too generic. When someone is improving energy efficiency, he should take into account system specific details, which can only be obtained by tests and measurements. Understanding the details will help in choosing the best way to lower the power consumption without sacrificing performance. If the results are satisfying the same practices can be implemented on other clusters and data centers, which will lower CO_2 footprint as well as power bills.

Sub questions:

How does one measure the energy consumption of cluster nodes, during the execution of various testing scenarios?

Estimating the energy consumption of a system is important step for making it more energy efficient. This allows one to make a comparison between the before and after state of the system, when changes are introduced as well as give idea on how much power it required in practice. During the measurements I did on the cluster nodes I generated load on specific components in order to determine how they contribute to the total power consumption. This allowed me to find the answer to my research question and provide background information for future research on this topic.

What improvements can be made without sacrificing performance?

Although energy efficiency is important, the performance is the main feature of most super computers. If performance drops after optimizing the power consumption, the system might not be able to fulfill its tasks any more. Some of the solutions I propose are software based, other require replacement or addition of hardware components. Part of the hardware solutions are not economically feasible, but in the near future the prices of components will most certainly drop and they will become real alternative.

2 Energy efficient computing

The price of energy is a important factor for a company to decide where to construct a data center or to move their equipment. Energy produced by burning fossil fuels is cheaper compared to renewable and nuclear sources. If datacenters move to locations where energy is cheaper, they will have actually larger CO_2 footprint. It is important to find an other solution, which will lower the expences of ICT organizations.

Fortunately, many opportunities exist to reduce High-performance computing (HPC) energy consumption. Scientists and developers are looking for new technologies, which will make the ICT products more energy efficient. This will reduce environmental impact, energy costs and also acts as a decision taking factor for organizations planning to purchase a large scale computer. These ideas are welcome for the operators of data centers, because higher energy consumption of computers requires more expensive site equipment - air conditioning, UPSs, power grid. More efficient servers will lower their bills and will also save space, because they will have less energy dissipation, this will allow integrating more processing power in a single design. This can be achieved with energy aware components and by introducing accelerators, which perform certain tasks with greater efficiency.

Major companies (Amazon, Google, IBM, Sun, Microsoft) are working on cloud computing infrastructures and experts agree clouds will change the way computing is done [3] [4]. Grid and cloud applications are becoming more flexible and malleable, because these environments are inherently dynamic. Scalability of resources and virtualization will allow balancing between the energy consumption and performance. Reducing energy consumption of clouds is expected to have significant impact.

2.1 GreenClouds

GreenClouds is a project by VU and UvA, which studies methods to reduce the energy footprint of modern High Performance Computing systems (like clouds). Clouds are designed to be distributed, elastically scalable, and contain a variety of hardware. The project takes a system-level approach and studies the problem of how to map high-performance applications onto these distributed systems, taking both performance and energy consumption into account. GreenClouds will make extensive use of the DAS-4 infrastructure, which is a wide-area test bed for computer scientists [8]. The results of the project will be utilized by SARA, the Dutch national HPC center that operates a supercomputer, clusters, accelerator systems, and an HPC cloud.

2.2 Shortcomings

The DAS-4 infrastructure has been delivered in September 2010. There is no exact estimate on the energy consumption of the hardware. Measurement equipment, which should be used to obtain the necessary data, was supplied along with the cluster. However the measurement equipment was also a new product and should be tested and requires software tools to be created in order the data to be extracted from the devices. One of the goals of this research project was to create the application, which reads and records the power consumption of the nodes, perform additional measurements on the nodes and to analyze of the acquired data. The output of the research will be useful for the GreenClouds project, because it will provide information about the power consumption of the nodes and the capabilities of the measurement equipment.

2.3 Related work

There are more projects active in this area. Here I summarize two of them.

2.3.1 Green500

The Green500 project provides a ranking of the most energy efficient super computer in the world. In order to raise awareness to metrics of interest like performance per watt and energy efficiency for improved reliability, the Green500 offers lists to encourage supercomputing stakeholders to ensure that supercomputers are only simulating climate change, but not creating a climate changes. The performance of the super computer is measured in MFLOPS and is benchmarked with Linpack. The website of the project provides news and resources on the topic of green ICT [5].

2.3.2 GreenLight

GreenLight enables scientific researchers to make ”green” data decisions by offering a suite of physical-layer architectures that leverage advanced middleware. A wide range of hardware and software based optimization solutions as well as system monitoring tools are presented on the web site of the project. They are being tested in the infrastructure of the University of California, San Diego campus innovative energy and cooling sources and employing middleware that automates and optimizes computing and power strategies [6]. The GreenLight instrument is involved in variety of research projects that include the use of accelerators, DC powered infrastrucure and virtualization techniques.

3 Test infrastrucure

3.1 Overview

For the research I used two nodes from the DAS-4 cluster. The nodes were connected through power distribution units (PDU) with measuring capabilities. The voltage, current and power factor read from them are recorded in a database. Various load generating applications were run on the nodes most energy consuming component. Some of the applications were well known and available to the public and other were created especially for this research project. The parameters of the system during the test were also collected and saved in the same database along with the power measurements. This allowed examination and making conclusions about the energy efficiency of the system and how it can be improved.

Figure 1: DAS-4 twin node

3.2 Cluster nodes

The DAS-4 cluster nodes used for this research have identical hardware components and are assembled two machines in a single chassis. No resources are shared between the systems except the two power supplies. If one of the PSUs fails, the nodes can continue working using the other one. In case single node is running, both PSUs are used and energy is distributed between them. Because of this the measurement of the power consumption should always be performed on two PSUs.

Each of the examined nodes consists of the following hardware:

Chassis:	Supermicro CSE-827HD-R1400B 2U Twin Chassis
Motherboard:	Supermicro X8DTT-HIBQF+
CPU:	2 x Intel Xeon E5620 2.4 GHz
Memory:	6 x 4GB Kingston KVR1333D3D4R9S/8GED
Storage:	1 x 1 TB Western Digital WD1002FBYS

There were no additional peripheries connected to the nodes during the measurements, except for the network connection, which is required in order system monitoring data to be transmitted.

The operating system on the nodes is CentOS 5.5 Final.

3.3 Measuring power

3.3.1 Power metrics

Before I go into details about this research, I think it is important to mention what the main metrics for power are and what is the difference between them.

Power in an electric circuit is the rate that energy flows past a given point of the circuit. In alternating current (AC) circuits, energy storage properties such as inductance and capacitance may result in periodic reversals of the direction of the flow. The portion of power averaged over a complete cycle of the AC waveform, results in the transfer of energy in one direction, thus is known as real power. The portion of power due to stored energy, which returns to the source in each cycle, is known as reactive power [7].

The **real power** is the current multiplied by voltage and is measured in watt (W). This is the power, which is utilized by the electric devices and does the useful work.

The **reactive power** is the product of the root mean square (RMS) voltage and current multiplied by the sine of the phase angle between the voltage and the current. It is measured in volt-amperes reactive (Var).

The **apparent power** is the vector sum of real and reactive power. Figure 2 shows the relation between these quantities.

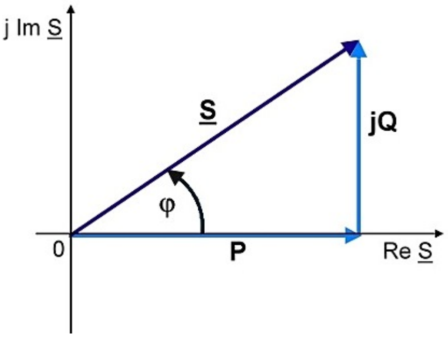

Figure 2: real power (P), reactive power (Q), apparent Power (|S|)

Apparent power is measured in volt-amperes (VA). It is important for engineers, because the infrastructure should be designed with sufficient reserve to handle it. The energy difference between real and apparent power is lost due to heating the cables and is considered wasted energy.

Power factor is the ratio between real power and apparent power. It's a practical measure for the efficiency of a power distribution system. For two systems transmitting the same amount of real power, the system with the lower power factor will have higher circulating currents due to energy that returns to the source.

In this research the power measurements are presented in real power, although the power factor is always collected along with the voltage and current. Real power is used because the PSUs have varying power factor, which is a function of the total power consumption. Using apparent power will make comparison between different tests more dependent on the efficiency of the power supplies, rather than the tested components.

3.3.2 Measuring infrastrucure

The DAS-4 cluster is equipped with Schleifenbauer manageable PDUs. The functions it provides along with power distribution are control, protection and metering. Each outlet can be switched on and off, which provides opportunity for remote restart of unresponsive node. The PDU has integrated surge protector and fuses, that lower the chance of damage to the equipment. For this research I used the PDU to determine the power consumption of each node.

The PDU is managed via a PDU gateway, a standalone device with several connection interfaces for administration (RS232 port, USB and Ethernet). The gateway has a web based control panel for configuring the connected PDUs and extracting data from them. The gateway provides information per PDU or per power outlet (channel). This information includes internal and external temperature, voltage, current, power factor and total power consumed. To extract the data from the PDU gateway, without using the web interface, APIs are provided in several popular programming languages Perl, PHP, .NET, and SNMP.

After testing the PHP API I found out it is not behaving as expected - it provided constantly the same information from all the read parameters, which is wrong.

The Perl API was working better and I decided to perform the measurements using this API. It requires a .pm file to be copied in the module folder of PERL. The APIs are able to read or write one parameter at a time using a uniquely named registers, containing the desired data. There is no register for power consumption and it has to be calculated by multiplying current and voltage. Although it is trivial to implement, this requires 2 instead of 1 read operations from the PDU. The higher the metering density, the more accurate model of the power consumption that can be created. With the current version of the API about 4.5 values per second can be retrieved, which is slow when more than one output should be monitored. The PDU doesn't support several simultaneous connections and this is the highest metering density possible. In order to obtain the voltage, current and power factor for 4 outlets of 9, it will take the Perl API roughly 2,7 seconds - which will be the minimal refresh rate of a power output. Using this interval in time measurement values will almost certainly mean that sudden short peaks in the parameters will not be recorded.

Since the Perl API is an early version, it is expected that there will be improvements in the code which will make it faster. This problem might also be fixed in newer versions in the firmware of the PDU or gateway.

The measured data from the PDU and the monitored system show that there is a time delay between the load change and corresponding peak change in the power measurements, when the system load changes suddenly. Figure 3 shows the offset between power consumption and CPU load.

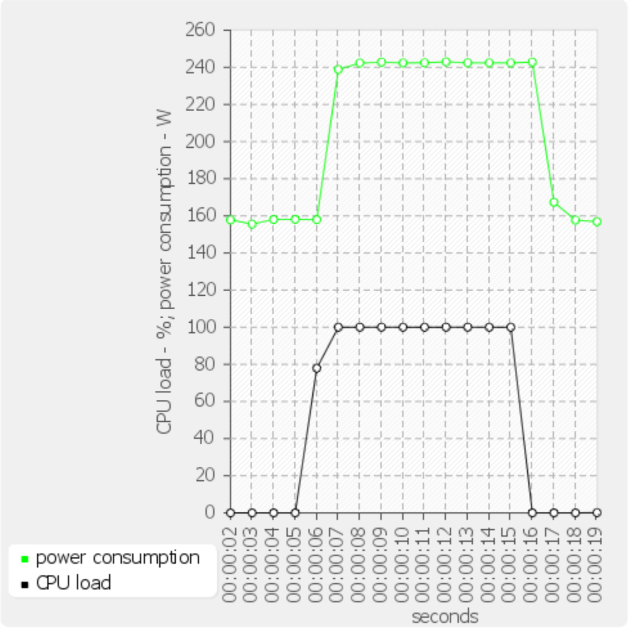

Figure 3: Offset between power consumption and CPU load

This delay can be caused by slow response time of the measuring equipment or because of the capacitors in the PSU, which for a moment compensate the need of more power from the grid. Its duration is less than the time required to obtain the next set of values from the PDU and was not examined, because it requires additional measuring equipment and time.

According to the PDU specification the deviation of the voltage and the current is under 0.5% [9]. The gateway returns the data with precision of 1%. During measurements the power consumption on idle node was changing per outlet with around 2 W, because of variations in voltage and current. Even a 0.01A increase in the current leads to 2.3W increase at power consumption at 230V. This makes the equipment not suitable for monitoring low consumption components, because the changes in current are close to the variations of an idle system.

4 Measurement software

Data from the tested nodes and the PDU are obtained separately and stored on a server. Figure 4 shows an illustration of the measurement setup.

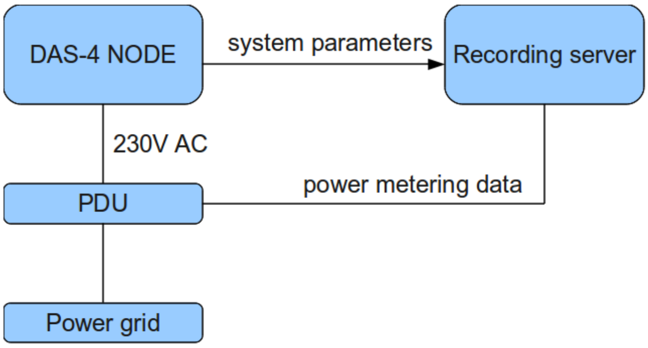

Figure 4: Measurement setup

4.1 System information

In order for someone to examine the nodes state during tests, the one should be able to obtain information about the current parameters of the system e.g. number of CPU cores, their frequency, load, memory and storage devices. Part of the information is available in the */proc/cpuinfo* virtual file system and the other information has to be extracted with the help of tools. For this purpose I used software which is available in the repository of most UNIX and Linux distributions, because it will make script migration easier. The main package I used is *sysstat*, which consist of several system monitoring applications for CPU performance, memory usage, storage devices and other. Part of this package is *sar* - a program that writes the content of selected cumulative activity counters in the operating system to standard output. The minimum interval is 1 second. The output contains additional information about the fields and data, which is not needed. Currently the script collects the total percentage of CPU usage for user and kernel, number CPU cores, individual percentages of usage for user and kernel per each core and frequency of each core. For example 8 core systems have to collect a total of 26 parameters on each loop. Next the output is sent to *awk* to format it and then a GET request is sent to a web and a database server using *curl*. The whole process is automated using a bash script.

4.2 Power measurements

To read the data from the PDU I created a Perl script, which uses the API provided by the vendors of the device. The script can retrieve one parameter at a time from the PDU. For example if I want to obtain the power consumption of channel 3, I have to extract the voltage and the current and then multiply values. If I want to monitor several outputs I have to loop through all of them.

After the voltage, current and power factor of one channel are extracted, I sent them to a web server, where they are stored and processed.

It is important to note that the measured node should not run additional software without a solid reason. This can introduce unwanted and unpredictable usage of the resources of the node and increase the power consumption. To prevent this, the script reading data from the PDU is located on machine different from the two measured nodes. In my case it was running on the same system as the web server used for recording the measurements.

4.3 Database

The readings from the PDU and the measured nodes are stored on a web server running Apache, PHP and MySQL database. With minor modifications the script, which records the data, can be adapted to use Oracle, XML or CSV files. In my current implementation it uses PHP with MYSQL, because the technologies are Open Source and provide an easy way to create web based applications for monitoring the cluster nodes.

When node parameters are monitored, the data is sent to a PHP script on the web server along with a unique identifier (name of the node) as GET requests. This PHP script checks if there is a table with a name corresponding to the identifier of the node and if it doesn't exist, it creates it. Next it inserts the received parameters in the table. Since the data from the node and the PDU have to be synchronized in the database for the purpose of analyzing, it is important how time is recorded. Synchronizing the clocks of the PDU and node is a hard and unnecessary task, because it will require additional resources and connection. The PDU currently supports only one connection at a time, so it would not be possible to adjust the time, while reading the power consumption. The problem is solved using the current time of the machine, which records the information from both nodes and PDU. In this way it is not required to synchronize time of the node and the PDU.

The database is automatically created by the PHP script. The system load of each node is recorded in a tabled named "loadNodeId", where NodeId is the unique identifier of the system. The number of fields in the table is dependent on the number of cores of the node. This is simplified version of the structure "load" table:

- loadid - unique identifuer of measurement - primary key
- loadtime - timestamp of the measured value - NOW()
- ucpuall - percentage of the total CPU usage generated by user
- kcpuall - percentage of the total CPU usage generated kernel
- ucpuN - percentage of core number N usage generated by user

- kcpuN - percentage of core number N usage generated by kernel

- cpuclN - frequency (in MHz) of core number N

The information from the PDU is stored in table ”energy”, shared among all nodes. It contains a fixed number of fields:

- energyid - unique identifier of measurement - primary key

- energytime - timestamp when the value was measured - NOW()

- channel - the number of the PDU outlet, from which the data is extracted

- volts - voltage of the outlet

- current - the current through the outlet

- factor - power factor of the outlet

The table ”records” contains the time intervals of the performed measurements, the ID of the node, the name of the test performed and information on what component was the target of the test. The structure:

- recordid - unique identifier of record - primary key

- starttime - timestamp of the measurement start

- stoptime - timestamp of the measurement was stopped

- sysid - unique system identifier of the node (name)

- recordname - name of the experiment

- object - the focus of the test, could be the number of the core, power supply or address of storage device

A table ”status” is used for recording the current state of each measured node:

- statusid - unique identifier of status - primary key
- sysid - unique system identifier of the node (name)
- value - enables and disables recording data, can be set to ”on” and ”off”

4.4 Generating load

The applications, which generate load on the tested nodes, are executed using bash scripts. Each of them sends a GET request to the web server, which starts recording at the beginning and stops at the end of the experiment. This avoids constantly writing information in the database. Between the sending of GET request and the actual load generation, there is a 5 second sleep period, during which the system activity is expected to normalize. Most of the bash scripts have a loop, which repeats the test to verify the measuments.

Before running the bash script it is a good idea to check the system identifier and experiment name, in case they were copied from another system. Otherwise the information will be written in the wrong database table.

Most of the measurements are performed in the following way:

1. The Perl script, which measures the power consumption, is executed on the background.
2. The bash script measuring the load and frequency of the cores is executed on the tested node.
3. The bash script, which generates activity on the tested node, is executed. It sends the system identifier, the name of the experiment and a start command to the web server using *curl*. This makes the web server change the ”value” in the node ”status” table to ”on”. In this way the data received from the PDU and the script, which measures the load of the system, will be inserted in the database.
4. Depending on the experiment performed the tests are looped several times. Meanwhile data from the PDU and node is written in the database.
5. A stop command is send to the web server using *curl*. It sets back the ”value” field in ”status” to ”off” in the database.
6. Although the measurement of parameters of PDU and node are still running and sending data, they are no longer recorded.

4.5 Data parsing

After the data collection is completed the data should be formatted and filtered in order to read results of the experiments. This requires writing separate SQL queries depending on the desired result. Because the nodes are connected with two power supplies it is necessary to obtain data from both of them separately. Data from both channels is not received by the web server in the same time, because the PDU doesn't support parallel reading of the registers. Because of this, the power consumption is calculated using two consequent entries from the database, one for each channel. For example:

```
Energyid: 102; channel: 7; energytime:  2011-01-18 21:49:23;
powerusage: 105W

Energyid: 103; channel: 8; energytime:  2011-01-18 21:49:24;
powerusage: 115W
```

The total power is the sum of these values and it is used for the time of the first record, because it is not possible to put an intermediate record between them:

```
Tempid: 51; channel: 7+8; energytime:  2011-01-18 21:49:23;
powerusage: 220W
```

The next record for total consumption will include the second row from the previous values and the row after it:

```
Energyid: 103; channel: 8; energytime:  2011-01-18 21:49:24;
powerusage: 115W

Energyid: 104; channel: 7; energytime:  2011-01-18 21:49:25;
powerusage: 107W
```

The result will be:

```
Tempid: 52; channel: 7+8; energytime:  2011-01-18 21:49:24;
powerusage: 222W
```

The total consumption is inserted in a temporary table, which is JOINed to the load table of the currently examined node. A sample query looks like this:

```
SELECT TIMEDIFF(loadtime , '2011-01-18 23:18:37') AS loadtime2,
loadtime, AVG(watts) AS watts, AVG(ucpuall) AS ucpuval,
AVG(kcpuall) AS kcpuval FROM ‘tempenergy‘
RIGHT JOIN ‘loadnode1‘ ON ‘temptime‘=‘loadtime‘
WHERE loadtime >= '2011-01-18 23:18:37' &&
loadtime <= '2011-01-18 23:25:38' && tempid > 2
GROUP BY ‘loadtime‘ ORDER BY ‘loadtime‘ DESC
```

Although kernel and user CPU usage are retrieved as separate values, they are always summed together and called ”CPU usage” or ”core usage”.

The result of the query can outputted to a variety of tools to present it in more readable way to the user e.g. charts generated with ”R” [10] or a web based system displaying in almost real time the power consumption and usage of the nodes.

System parameters and power measurements should be analyzed in order to create a relation between the load of the components and power consumption. This can be done with automated scripts, which calculate the ratio in predefined moments of time and events or using visual analysis of charts. During the research project I used the ”pChart” [11] PHP class to create PNG images with charts.

5 Examined components and results

In this section is presented information about the examined components, details about the performed measurements and the results from them.

5.1 Power supply unit

The PSU is the component, which converts 110V - 220V AC energy from the power grid to 12V, 5V and 3.3V DC energy. In rare cases, when the energy is provided by a UPS, DC to DC power supplies are used. The maximum output power of the PSU should always be greater than the sum of the energy consumption of all connected components in order to provide adequate power during peak loads of the system. Choosing an efficient power supply is a major step to increase the efficiency of the whole system, since all the components are powered through it. If a PSU is rated with 500W maximum output and 80% efficiency, it is expected to consume 600W from the power grid, converting the additional 100W into heat. The efficiency is not a constant and it drops significantly during low loads. Because of that the efficiency should be measured not only during full load, but also when the system is idle and shut down. The 80 PLUS initiative issues several levels of certificates, based on the efficiency measured during 20%, 50% and 100% of the maximum output of the PSU. This creates a unique market opportunity for power supply and computer manufacturers [12].

Power factor in AC powered PSUs have become a recent issue of concern for computer manufacturers. Many power supplies built in the last years now include power factor correction (PFC). The PFC can be passive or active. The passive PFC power supplies are cheaper to manufacture, but require more energy and space compared to the active. The minimum requirement for PSU to obtain 80 PLUS certificate is power factor over 80%. Currently both active and passive PFCs achieve this, with some examples of 99% for active PFC [13].

Tests on PSU

The PSUs used in the tested DAS-4 twin node are rated "Gold" by 80 PLUS [14]. With the current configuration of the system they have a big power reserve - their combined maximum output is 2800W. The two nodes with fully loaded CPUs consume around 500W. This provides opportunity for installing additional components with high power demand such as GPUs, without replacing the power supplies.

The 80 PLUS test results don't provide information about the consumption of the power supply, when the system is shut down. Although the nodes are turned off, the power supplies still provides power to the motherboard. This might be the reason the power usage of a shut down system is not included in the 80 PLUS report. I measured the consumption of the twin nodes and the average value is 71W for both power supplies, which makes 35.5W for each. In my opinion this consumption is significant for node that is shut down.

When the DAS-4 twin node has its both PSUs connected, the ratio of power consumption and power factor between the different PSUs is not distributed equally. For example this data is measured while the system is idle:

```
channel: 8; energytime:  2011-01-18 19:09:14;
powerusage: 110.4406W; power factor: 81.67%

channel: 7; energytime:  2011-01-18 19:09:15;
powerusage: 95.9933W; power factor: 85.02%
```

I changed the power outlet of each power supply to see if this ratio difference is caused by the measuring equipment or the power supplies. The results were similar to the previous measurements - the consumed energy and power factor for the identical power supplies can differ at the same moment. I started monitoring over a longer period and I noticed that when a PSU is disconnected and reconnected from the grid or the nodes are rebooted, the observed ratio changes. In some cases the two power supplies were consuming almost the same power and had the same power factor, but in sometimes they had difference of several percent like the above example. My assumption was that a power regulator inside the nodes is responsible for the distribution of the power load.

I recorded the power factor and consumed power of each node from idle to 100% load of the CPUs for the purpose of analyzing the relation between power factor and power consumption. Although the load of the two power supplies is not distributed equally in some moments, the tendency to increase the power factor while increasing the consumption is shown in figure 5.

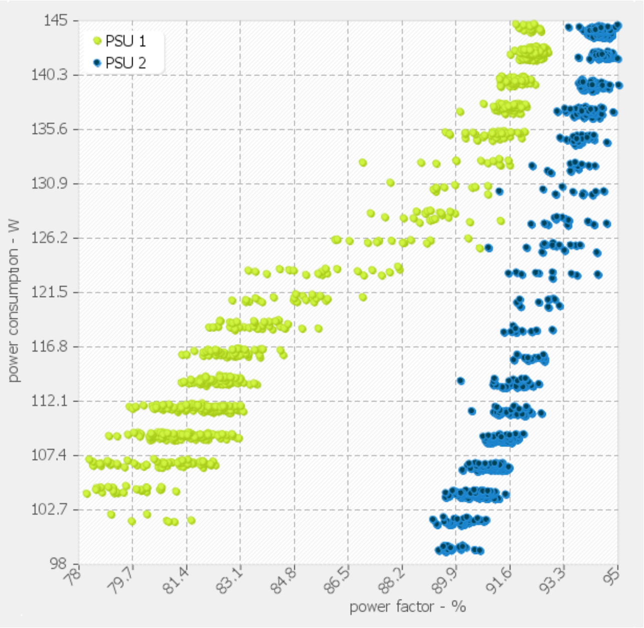

Figure 5: Power factor observation

The next performed test was to measure the difference between running on a single or dual power supplies. For this purpose I disabled one of the outputs on the PDU. I measured the power consumption and compared it to the previous measurement done with two power supplies. Again I increased the load of the CPUs from idle to 100% gradually. For this purpose I use a script, described in the CPU section of the report.

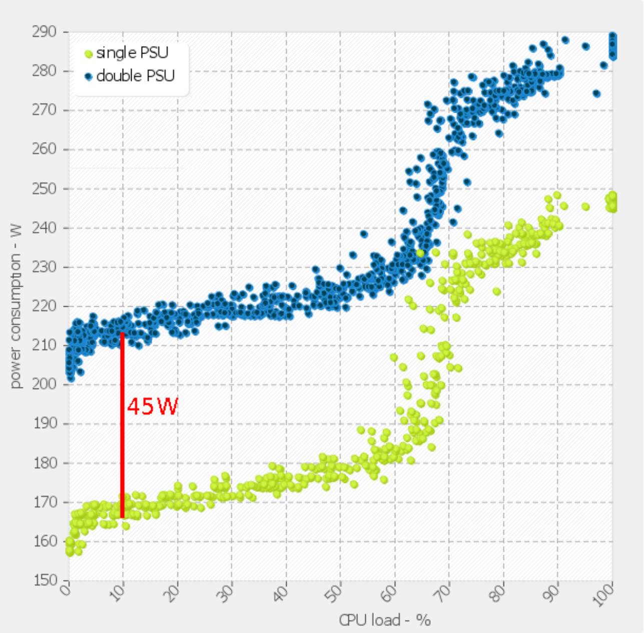

Figure 6: Comparison between the total power consumptio of a node powered by single and dual PSU

When the node is powered by a single PSU it is consuming on average between 35 and 50 W less energy, depending on the load. In the lower range the power supplies are more inefficient. In the higher range the difference between running a single and dual PSU decreases. This difference is more than 25% of the total consumption of the node while it is idle and 10% while the CPUs are loaded on 100%. The curve of power usage is not linear, because of the CPU scaling, covered in the CPU section.

This test used only 250W of the maximum power output of 1400W, which is in the range, where the power supply is still inefficient. Because of this I measured the power consumption also when the second node is performing the same task.

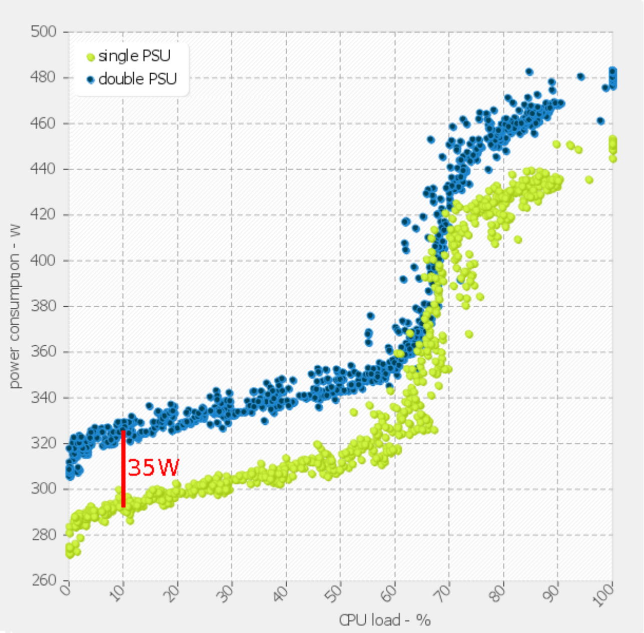

Figure 7: Comparison between the total power consumptio of two nodes powered by single and dual PSU

The difference is between 35 and 30W on average. This shows the use of second PSU adds its additional 35W power consumption measured even when the node is powered off.

From the comparison of the two charts (figures 6 and 7) it it visible that one node using 100% of its CPU consumes around for 245W single and 285W for dual PSU. When two nodes are performing the same task using again single and dual PSU the power consumption respectively is 450W and 480W. Divided by the number of nodes powered it makes 225W and 240W per node. The improvement of the energy efficiency is between 6 and 9.5%.

During certain circumstances it is relatively efficient to disconnect one of the power supplies of the system. It is not reasonable to do this, because it might cause damage to the components if done too often. This should be investigated and if proven not dangerous for the system could be implemented. Running on a single PSU should not be done, when the node is computating important tasks, because this will also cut the redundancy of the system in case of PSU failure.

5.2 Computing

5.2.1 CPU

The CPU is the primary element, which is carrying out the computer's functions. It consists of millions of transistors integrated on a die. In multi core CPU two or more individual processors are placed into one integrated circuit. The CPU is the most energy demanding component in the average server [15].

The characteristics, which have strong impact on the power consumption of modern CMOS CPUs are frequency, voltage, number of cores and architecture.

The clock speed is almost proportional to the current they draw, because every clock cycle causes many components to switch regardless of whether they are being used at that time. In order to reduce the power consumption and heat dissipated by the CPU, developers have created techniques, which dynamically scale the frequency and voltage supplied to it. In the Intel CPUs it is called Enhanced Intel SpeedStep Technology [16]. The scaling is controlled by a governor in the operating system, which monitors the CPU cores and if their parameters go beyond predifined value, the cores are tweaked. In some cases even a core can be sent to sleep mode - it is turned off [17].

Tests on CPU

The first experiment was measuring the power consumption while loading the CPU cores gradually from 0 to 100%. This is done with a script, which loops division of random numbers and sleep cycle. The loop is repeated N times and then the CPU "sleeps" for fixed amount of time. After several repetitions N is incremented. When predefined parameter values are reached, the sleep cycle is disabled and the CPUs are running on full load for 2 minutes. After that the sleep is introduced again and the load is dropped gradually from 100 to 0%. To load a particular core, the *taskset* tool is used to map the script to that specific core.

The script has to be tuned in order to work properly on slower or faster processors. When increasing the load gradually, it is possible to examine the behavior of the frequency scaling more carefully. During this measurement I found out that it is hard to keep the CPU load constantly on a certain level different from 0 and 100%. Although the CPUs performs a similar tasks, it load can vary in the range of +- 5%. This makes comparing instances of measurements not precise and requires average statistical database. The CPU scaling and the varying load can be observed on figure 8:

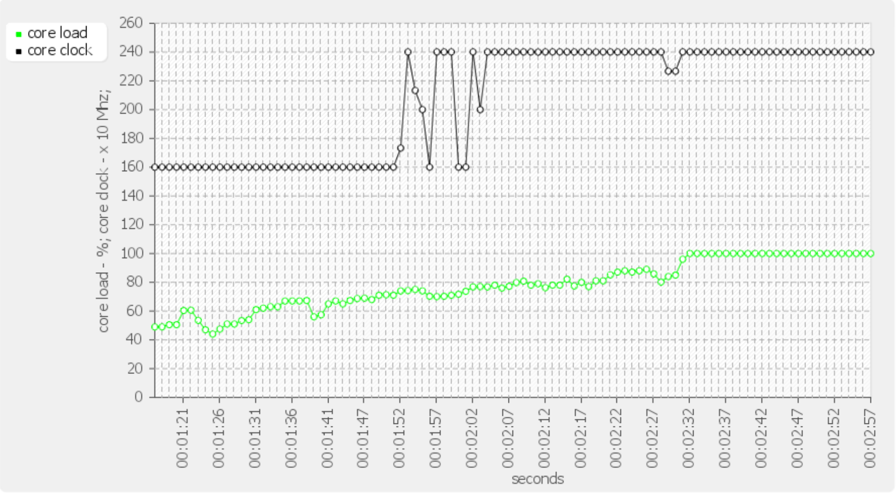

Figure 8: Frequency scaling and core usage

The frequency scaling is activated, when around 70% of the CPU core usage is reached. The next experiment was to compare the power consumption, when CPU scaling is enabled and disabled. During tests the node is simultaneously running 8 parallel instances of the script, which gradually generates CPU load from 0 to 100%. The results are shown on figure 9.

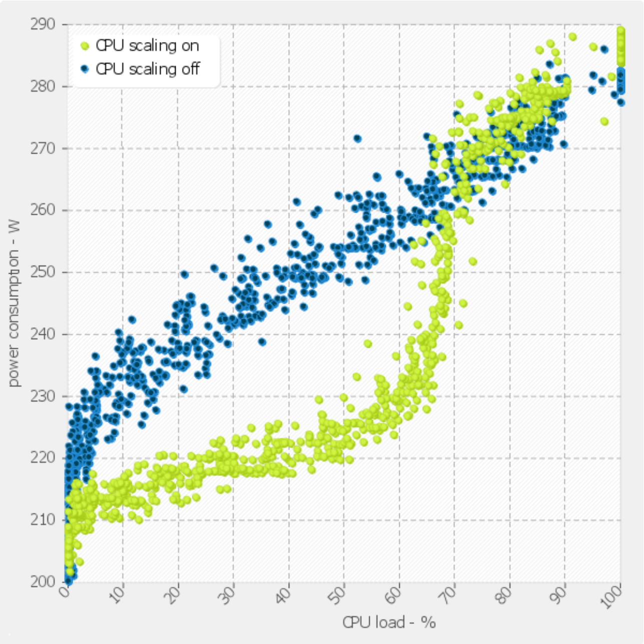

Figure 9: Comparison between the total power consumption of scaling and non scaling CPUs

The initial power consumption of the node is around 200W in both cases, although when scaling is enabled, the frequency of the cores is 1.6 GHz instead of 2.4 GHz. It is important to mention that while the CPU core is running on higher frequency it is able to perform more tasks per fixed amount of time. Because of the higher frequency, 30% load at 2.4 GHz does more computation than 30% load at 1.6 GHz. Even with slight CPU activity (around 5%) the power consumption of the system with fixed frequency jumps between 220 - 230W, while the scaling is at 210 - 216W. At 10% the increase of power usage of the non scaling system is linear until it reaches 100% using 287W. After the scaling governor switches the CPU core to maximum frequency at around 70%, the power consumption of the instances when scaling is enabled is around 3W higher than the already running on that frequency instances. This effect is around 1.2% of the total power consumption and is not further investigated.

I also examined the behavior of each core of the CPUs, utilized gradually from 0 to 100%. Although the power consumption can vary significantly compared to the amount of total consumption, the trends can be clearly distinguished. The behavior is identical to the parallel running of the script. Because of the lower total power consumption, the variations in figure 10 are more visible.

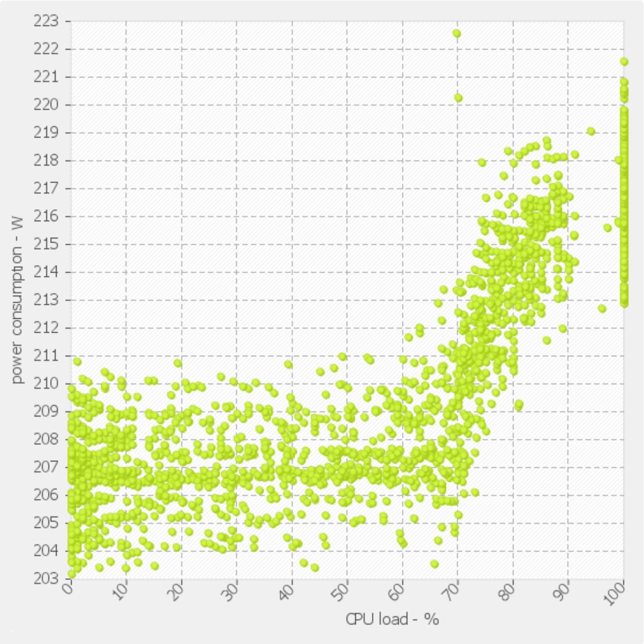

Figure 10: Total power consumption of a node with varying core usage

In the next test I use the ”cpuburn” tool [18]. Cpuburn is a program, which generates 100% load per CPU core. In order to load all the 8 cores, 8 instances of the program are run in parallel. For the experiments I started the instances with time intervals of 10 second. When 100% total CPU load is reached the script continues execution for 3 minutes and then it stops each instance with 10 second interval until load reaches 0%. This test is useful for accurate predicting of the power usage as a function of total CPU usage.

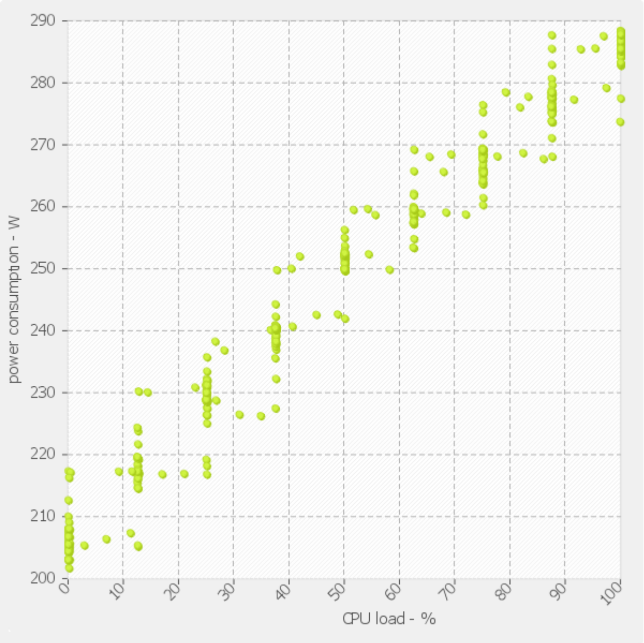

Figure 11: parallel instances of cpuburn

Figure 11 shows the power consumption during each of the 8 steps when cores are being loaded. It is similar to figure 9, because here again the cores are scaled to 2.4 GHz caused by the 100% load.

The CPUs really are the most energy consuming components of the nodes. With their 90W power addtitional power consumption, the CPUs are responsible for significant part of the consumed energy by the nodes. Keeping the load constant, where the level of energy efficiency is highest can bring improvement to the system. Since the performance should not be sacrificed, a task scheduling can be implemented, which will manages the resources optimal. This idea, along with other will be researched in GreenClouds project.

Without measuring the performance of CPUs and comparing it to other models it is not possible to determine if replacing the CPUs with an other model will provide better power efficiency for the tasks. Adding accelerators in the nodes is another way to unload the CPU and lower its consumption.

5.2.2 GPU

The Graphic processing units were designed to accelerate certain parts of the graphics pipeline. They evolved significantly over the years and todays GPUs are general-purpose parallel processors with support for accessible programming interfaces and industry-standard languages such as C. Developers who port their applications to GPUs often achieve huge speedups vs. CPU implementations. GPUs can consume more than 200 W [19], which can be as much as the entire node. A study shows GPUs are faster and more energy efficient compared to CPUs in some applications [20].

These types of applications can be executed on the GPUs, and since GPUs are not vital component for the system, they can be disabled when not required.

5.2.3 FPGA

Field Programmable Gate Arrays are an integrated circuits, which use programmable logic resources with flexible-functionality instructions and flexible instruction sets. Systems equipped with FPGAs have increased performance over industry standard servers, while reducing energy costs associated with high-performance computing. In many cases they perform better than CPUs. A server with FPGAs consumes several times less energy compared to the same performance machine, which uses CPUs for computations. This makes FPGA an energy effective alternative which should be considered in the future [21].

5.3 Memory

Currently most of the systems use dual and triple channel architecture, which increases the data throughput from the memory to the memory controller, but requires higher amount of memory modules to function. This results that two memory modules of 4GB in dual channel mode will work faster, compared to one of 8 GB. However by increasing the amount of modules and the power consumption of the system is also increased, while having the same amount of memory [22] [23].

It is estimated that 15% of the power consumed by the servers is used by memory modules. There is significant improvement of the energy efficiency of RAM over the last years. Hardware developers achieve this by lowering the operational voltage and thus reducing the overall power [24].

Tests on memory

Initially the plan for examining the power consumption of the memory was loading it with intensive tasks, measuring the consumption and comparing the results. While running memory tests, I noticed that the CPUs are highly involved, which results in additional energy usage and inaccurate results. The load on the CPUs constantly vary, which makes the task of distinguishing between the energy consumed by the memory and CPUs hard. After examining the results from the measurements I conclude that the difference in the power consumption between intensively used and idle memory is small and insignificant compared to the energy, which the CPUs consume during the test. The data sheet of the memory modules provides good idea on the power consumption of a module[22].

I decided to remove 5 of the 6 memory modules and measure the power consumption of a idle node. In the specification of the memory is written that it consumes 3.96 W per module, while operating. This multiplied by 5 for all removed modules is 19,8 W. The measured result is 19W on average lower power consumption of the node, which shows that the memory is behaving close to the specifications. This suggests that the energy consumption can be predicted from the number and type of modules.

The improvement, which can be made to the nodes, is replacing the 6 x 4 GB memory modules with 3 x 8 GB. They will have 0.4W of additional power consumption per module. This will lead to 11W less power consumption for every node, which is around 5% of the total energy a idle node consumes [23]. However the nodes are equipped with two CPUs and they require 6 memory modules in order to have maximum memory throughput in triple channel mode. If an memory update is necessary I recommend that the modules are replaced with higher capacity ones, instead of adding additional.

5.4 Data storage

5.4.1 HDD

The DAS-4 nodes are currently equipped with single 1 TB HDD each. The common components for all HDDs are magnetic plates rotated by an electric motor, heads mounted on an arm, which is moved by voice coil and a circuit board known as a controller. Although power is consumed by the controller, the most significant amount is drawn by the electric motor, which in high performance models spins with 15 000 RPM and the voice coil for positioning the heads. Several technologies are developed to decrease the power consumption of these components. The speed at which the drive spins can be lowered to reduce the consumption with relatively small degradation in read/write speed [25]. When the disk is idle the heads can be moved near the center of the spindle, where they will produce less drag. This will lead to lower consumption of energy, because of the less resistance caused to the plates. Parts of the electronics can also be disabled when the disk is idle.

Currently several companies offer energy efficient HDDs, which make use of the above technologies. These drives are useful for storage servers that have constant read/write operations, but dont require small access times for the data [26]. For systems that require large amount of storage space it's a better practice to install one device with large capacity, instead of two more energy efficient with half the capacity. This is true in most cases, because the sum of their power consumption will be greater than of the single drive, they will require twice the physical space and the chance of failure also doubles.

Tests on HDD

To measure the consumption of the hard drives installed in the system, a specific test is required which will cause them to work on maximum load. The logical way is to make the hard drive read and write different sized data blocks and to record the performance and energy consumption. But the read/write performance of the disks depends on their physical characteristics along with other factors like the file system, usage of RAID and data distribution. Because of these factors I decided it will be too subjective to measure power consumption while doing read/write tests. Since most of the power consumed by HDDs is used by the mechanical components inside, I decided to load the hard drives with a program that causes them to seek random locations on the plates and measure the time required to relocate the heads [27]. This is independent of the file system and arrays and should provide more accurate estimation of the power consumption than read-write performance.

The result of my measurements is that the heavy used HDDs consume roughtly the same amount of energy compared to idle state - around 1W, difference which is withen the range of power variations of the whole system. With the current equipment this is difficult to examine. According to the specifications of the hard drives the difference in power consumption between idle and read/write is 0.6 W. The disks consume 7.8W while idle and 1W in standby mode [28].

5.4.2 SSD

Solid state drives use electronic chips on which information is recorded. They have no moving parts in contrast to HDDs. SSDs are divided in two types - using volatile (RAM) and non-volatile memory (NAND flash memory). RAM based solutions are not good alternative, because they can consume more power compared to a hard drive, while providing less space [29].

Flash based SSD can consume 80% less energy compared to a hard drive with the same capacity, while performing similar operations. SSDs are not limited by the speed of the moving components and can have hundreds and even thousands of times shorter data seek time times. Currently the main disadvantage of SSDs compared to HDDs is the higher price per stored gigabyte, low storage capacity of the mainstream models and shorter life expectancy [30].

If the storage space of the nodes is only required for keeping the operating system and small programs and the results of the computations done on the nodes are sent over the network, there is no need for large storage devices. The hard drive can be replaced with an alternative like a solid state drive.

6 Recommendations and future work

This section is about recommendations concerning improving the measurements and future work, which can expand the scope of the research.

6.1 Measurement software

6.1.1 System monitoring

Currently monitoring the system parameters of the DAS-4 nodes is dependent on third party open source tools. Although they provide sufficient information, the output needs to be filtered, formatted and sent to a web server using pipes. This is not efficient and introduces delays between the time the information was measured and when it was recorded. In order to better suit the needs of the project I suggest that the *sysstat* software package has to be modified. The modification should have focus on:

- voltage and frequency monitoring for CPU cores and memory
- detect memory modules and storage devices installed
- memory and storage device usage monitoring
- higher density of system measurements
- direct output to the recording software

Currently most of the information above can be extracted with the help of various tools and methods, but creating an all in one solution will make system monitoring less dependent on installed software components. The additional features listed will provide a more detailed overview of the system status during the experiments, which will result in finding more opportunities for system optimization.

6.1.2 Additional experiments

The test performed during this research are very small part of the opportunities the DAS-4 cluster provides. I suggest in the future some additional tests are performed:

- Running Linpack benchmark used in Green500 on the nodes and estimating the performance per watt metric.
- Comparing the Linpack results from the DAS-4 nodes with low power consumption nodes to determine whether the performance per watt is lower or higher in DAS-4.
- Is it possible on software level to put high energy demanding components to sleep, when they are not required and what is the impact of the total energy consumption.

Testing of GPU accelerators was not done in this research due to time and technical limitations. In my opinion they are important and should be performed.

6.1.3 Power measuring

The Schleifenbauer PDU provides good functionality for estimating the power consumed by servers in a certain time period, but in the current moment the capabilities of monitoring live data are not sufficient. The density and precision of the measurements are too low for the needs of the GreenClouds project. This can be improved with newer versions of the APIs and firmware of the device. If the required capabilites are not possible with this device I suggest obtaining specialized power analyzing equipment and using the PDUs for power distribution and control.

In the current setup it is possible to monitor only the total energy consumption of the nodes. In this way the estimated consumption of different components is based on predictions. I suggest introducing measurement equipment between the PSUs and the components connected to them. This provides information how efficient the PSUs are and gives more precise and complex data on individual components. It is even possible to integrate the power measurement equipment to system level by modifying the hardware of the node. This proves to provide good results in power consumption measurements [19].

7 Summary

The results of this research show that there are opportunities to lower the energy consumption, without sacrificing the performance of the system. They include changing the traditional hard drives with solid state drives and improving the efficiency of the power supplies, which waste significant amount of energy even when the nodes are turned off. The CPUs can be offloaded from some of the tasks if accelerators like GPU and FPGA are added to the system.

The energy efficiency is not limited only to the hardware of the nodes. It can be increased by dynamically scaling the resources of the cloud. Research on virtualization and task scheduling is included in the GreenClouds project and the results can provide interesting opportunities for lowering the energy needs of DAS-4. If this proves to be successful, it will be used by SARA and possibly other sites, which operate supercomputers and clouds.

References

[1] U.S. Energy Information Administration: *International Energy Outlook 2010 - World Energy Demand and Economic Outlook* `http://www.eia.doe.gov/oiaf/ieo/world.html`, July 27, 2010

[2] David J. Brown, Charles Reams: *Toward Energy-efficient Computing* `http://queue.acm.org/detail.cfm?id=1730791` ACM Queue, Vol. 8, No. 2, February 2010,

[3] Universitat Politcnica de Catalunya: *Cloud computing: changing the way we work* `http://www.upc.edu/saladepremsa/informacio/monografics/cloud-computing-changing-the-way-we-work`

[4] Laurel Delaney: *Why Cloud Computing Will Change The Way We Work?* `http://www.verio.com/resource-center/articles/cloud-computing/`

[5] *The Green 500* `http://www.green500.org`

[6] *GreenLight* `http://greenlight.calit2.net/`

[7] Wikipedia: *AC power* `http://en.wikipedia.org/wiki/AC_power` [Accessed: January 29, 2011]

[8] Henri E. Bal: *DAS-4: Prototyping Future Computing Infrastructures* `http://www.nwo.nl/projecten.nsf/pages/2300154150_Eng`

[9] Schleifenbauer: *Technical product specifications* `http://www.schleifenbauer.eu/dynamisch/bibliotheek/31_0_EN_specificaties_EN.pdf`

[10] *The R Project for Statistical Computing* `http://www.r-project.org/`

[11] *pChart* `http://www.pchart.net/`

[12] Ecos Plug Load Solutions: *80 PLUS Certified Power Supplies and Manufacturers* `http://www.plugloadsolutions.com/80PlusPowerSupplies.aspx`

[13] SilverStone Technology: *Why we need PFC in PSU* `http://www.silverstonetek.com/tech/wh_pfc.php`

[14] Super Micro Computer: *80 PLUS Verification and Testing Report* `http://www.supermicro.com/products/powersupply/80PLUS/80PLUS_PWS-1K41P-1R.pdf`

[15] Lauri Minas, Brad Ellison: *The Problem of Power Consumption in Servers* `http://www.intel.com/intelpress/articles/The_Problem_of_Power_Consumption_in_Servers.pdf` Intel Corporation, 2009

[16] Intel Corporation: *Frequently asked questions for Intel Speedstep Technology* `http://www.intel.com/support/processors/sb/cs-028855.htm`

[17] Klaus Gottschalk: *Energy Management for HPC with IBM* http://www.ena-hpc.org/2010/talks/EnA-HPC2010-Gottschalk-Energy_Management_for_HPC_with_IBM.pdf International Conference on Energy-Aware High Performance Computing, September 16 - 17, 2010

[18] *Cpuburn* http://pages.sbcglobal.net/redelm/

[19] Alexey Stepin: *Power Consumption of Contemporary Graphics Accelerators: Spring 2010* http://www.xbitlabs.com/articles/video/display/gpu-power-consumption-2010.html, March 22, 2010

[20] T. R. W. Scogland, H. Lin, W. Feng: *A first look at integrated GPUs for green high-performance computing* http://www.springerlink.com/content/5npmr03243h27523/ Computer Science - Research and Development vol. 25, First International Conference on Energy-Aware High Performance Computing, August 2010

[21] Convey Computer Corporation: *Reducing Power Requirements in HPC Datacenters with Hybrid-Core Computing* http://www.conveycomputer.com/Resources/Energy-Efficient%20Hybrid-Core%20Computers.pdf June 2010

[22] Kingston Technology Company: *Memory Module Specification KVR1333D3D4R9S/4GED* http://www.valueram.com/datasheets/KVR1333D3D4R9S_4GED.pdf

[23] Kingston Technology Company: *Memory Module Specification KVR1333D3D4R9S/8G* http://www.valueram.com/datasheets/KVR1333D3D4R9S_8G.pdf

[24] Kirstin Bordner: *Micron Continues Leadership in Energy-Efficient Memory Designs With New Low-Voltage DDR3 and Higher-Density DDR2 Parts* http://news.micron.com/releasedetail.cfm?ReleaseID=440669 Micron Technology, Inc. April 16, 2008

[25] Lawrence Webber, Michael Wallace: *Green tech: how to plan and implement sustainable IT solutions* http://books.google.com/books?id=BKTALNq5ceAC&lpg=PA62&dq=green%20disk%20drive&pg=PA62#v=onepage&q=green%20disk%20drive&f=false. p. 62. ISBN 081441446X

[26] Patrick Schmid, Achim Roos: *Energy-Saving Hard Drives* http://www.tomshardware.com/reviews/energy-disk-drive,1944.html Tom's Hardware, June 3, 2008

[27] LinuxInsight: *How fast is your disk ?* http://www.linuxinsight.com/how_fast_is_your_disk.html

[28] Western Digital Corporation: *WD RE3 1 TB SATA Hard Drives (WD1002FBYS)* http://www.wdc.com/global/products/specs/?driveID=503&language=1

[29] Tom's Hardware: *HyperDrive 4 Redefines Solid State Storage* http://www.tomshardware.com/reviews/hyperdrive-redefines-solid-state-storage,1719-4.html November 7, 2007

[30] STEC: *SSD Power Savings Render Significant Reduction to TCO* `http://www.stec-inc.com/downloads/whitepapers/Performance_Power_Advantages.pdf`