

EnergyAware

Mobile-based Architecture for Maximizing Energy Savings during Computer Idle Times

Miguel Jimeno¹, Martín Díaz¹, John Candelo² and Pedro Wightman¹

¹*Systems Engineering Department, Universidad del Norte, Km 5 Via Pto Colombia, Barranquilla, Colombia*

²*Electrical Engineering Department, Universidad del Norte, Km 5 Via Pto Colombia, Barranquilla, Colombia*

Keywords: Energy Efficiency, Energy Saving Policies, Computer Networking, Smart-Phones.

Abstract: Energy consumption of computers in office buildings remains an issue of concern for companies. For example, for a university campus, they might represent between 50% and 80% of all the power consumed. Besides that, computer usage behavior sometimes makes the application of energy saving policies a difficult task, as many users prefer not to be annoyed by waiting for computers to wake up. We present in this paper a novel software architecture to enable dynamic and static appliance of energy saving policies in office computers while observing and taking into account user behavior. The key is taking advantage of idle periods for the computers and the localization of the user to determine when to turn Off or On the PC and maximize the savings. We ran a study at our campus where we found that idle periods represented between 30% and 50% of the On periods. We simulated the savings for our policies and could potentially achieve savings between 32% and 38% of current power consumption numbers. With even more tuning, we believe savings could be higher and similar techniques could be used for other devices in buildings.

1 INTRODUCTION

Energy consumption in personal computers remains a problem of concern especially for office buildings managers. An example is an empirical study in (Agarwal, 2009) which found that between 50% and 80% of all electrical power consumption in the buildings at the University of California San Diego campus comes from IT equipment, from which most of it would be desktops. Different energy saving techniques have been applied with relative success during the past decade. The fact that the user still has the last word about how and when such techniques are applied might prevent such techniques to be applied in most of the computers or during most of the time. In developed countries users might be switching desktop use to laptops, tablets and smartphones, but annual shipments of PCs are still high, about 80 million per year, counting desktops and laptops, according to an IDC report. And in emerging and developing countries, office setups still have desktops as the main computing device. Companies, according to the report, still consider replacing PCs which might even help at least maintaining sales in upcoming years.

As a result of this, different PC managing solutions have emerged to help offices to administer their energy consumption and focus especially on the computers. The most popular is Verdiem, a company that offers a system to manage energy consumption of IT equipment in an office. Their advertising promotes that default energy saving options are not enough to achieve maximum savings. In the personal level, there are many applications that can be used to monitor the energy usage and apply static energy saving policies. One of the missing features in the available software is the possibility of creating user-centered policies that could be changed on the go as the situation changes with the least intervention from the user. In this paper we propose architecture composed of a server, a software application for PC and a software application for a mobile phone that combined could increase energy savings when compared with the use of typical energy saving policies. The architecture relies on a very detailed monitoring of the computers usage to detect the exact moments at which the user is not present and put the PCs to sleep during those periods. The architecture also relies on the constant monitoring of the location of the user to achieve a

better experience by decreasing the annoyance and also increasing energy savings at the same time. Thus, the contributions of this paper are: 1) A software architecture to monitor PC usage, energy consumption and apply savings policies. 2) A behavior-centered dynamic energy saving policy. 3) A user location-based energy saving policy that accommodates to user location to improve both savings and user satisfaction.

The rest of the paper is divided as follows. Section 2 summarizes the related work, Section 3 describes the EnergyAware architecture. In Section 4 the location-based energy saving policy is designed. In Section 5 the implementation and evaluation is shown, and the paper is concluded with a discussion of future work.

2 RELATED WORK

Computers in offices can achieve substantial savings in energy consumption. In a report from EMA, 2008, the authors conducted a survey among hundreds of professionals where they responded that their computers were kept on during 43% of the total non working hours or the equivalent of 55 hours of consumption during a week. This equates to an additional cost and unnecessary \$73 per computer annually. Although computers have the ability to be configured to sleep or hibernate, it does not always take advantage of it. In the survey it was found that from the people who did the survey including teachers, secretaries, assistants and administrative staff, only 12% of them have ever changed their energy savings on their computers (specifically suspend and hibernate). This shows that even if companies have the option, energy saving policies are not mandatory in all companies, and users are left with the choice of saving energy or not.

In (Lin et al., 2011) the authors explained several methods with which they could generate energy savings in computers. The most interesting was RFID. RFID is used to save energy in the computers that are used in classrooms. When the computer is not in use, it turns Off or suspends automatically, avoiding energy waste. To implement this technology, a sensor that identifies when a person leaves should be installed at the entrance of the office or room. It will automatically send a signal to the computer and thus start sleeping. The authors used a server that retrieves data using an RFID reader. It allows knowing when a user is close to or away from the computer. The server decides to put the computer to sleep if the user is away for 10

minutes, or decides to turn it off if the user spends more than 30 minutes away from the computer.

In (Korn et al., 2004) the authors show a framework for energy savings on computers. They proposed the creation of profiles for each computer or group of computers and in turn make them have Off-time schedules depending on the profile assigned to it. The authors did however not take into account how the users will handle the decision of turning off the computer in the case he/she needs to leave it on to complete long-running tasks.

The Engineering Department of the Samsung Institute of Technology has a system that allows you to suspend the device when not in use. They managed to implement it using a hardware that recognizes when the user gets up from his chair. It orders the device to go to a suspended state, and when the user returns to his/her desk, it resumes the operating system (Reilly et al., 2011). This however restricts its use by requiring a piece of hardware. The Department of Electrical Engineering and Computer Science at the University of Fukuoka in Japan proposed a technology similar to the one Samsung proposed, but differs in the presence verification process. Instead of verifying if the user is present or not in his/her desk, the University of Fukuoka checks whether the user is currently looking at the computer screen (Kim, J.-Y., 2011) or not. This is done by means of a webcam, a presence sensor/vision and an algorithm called EYE-TRACKING. In this way if the person comes completely out of the camera range, it means that it is not in front of the computer and therefore is not using it. The implementation turns off the screen and suspends the computer to achieve energy savings. This also depends on hardware.

Mobile phones are used as an authentication mechanism and could be used to determine if the computer owner is close to it. One of the authentication methods most difficult to control is that which uses tokens or any mechanism created to store security passwords. An example is the proposal made in (Jaros and Kuchta, 2010), where the authors performed a token-based authentication through mobile devices. The communication between the mobile and the computer is done via Bluetooth. Another proposal tries to overcome the problems with GPS by using Bluetooth and Wi-Fi to improve location calculation accuracy (Almuairfi et al., 2013).

In this article, we like to combine several of the technologies used in other applications. We will use the mobile phone as an authentication token. We will propose an energy saving mechanism that relies

on the knowledge about the location of the user and its phone at any given time. To achieve confident levels of accuracy about the location of the person, we will combine GPS and Bluetooth to achieve more precision. We considered that it differentiates from previous propositions in the granularity of the computer usage statistics and the dynamism of the energy saving policies based on the behavior of the user.

3 THE EnergyAware ARCHITECTURE

The main purpose of the EnergyAware architecture is to achieve improved energy savings in a group of computers, when compared to existing energy savings strategies. This is achieved by giving to the user certain remote management capabilities and statistical information about the group of monitored computers. At the same time, the PC could be put in energy saving states during idle times to maximize savings. Figure 1 shows the 3 main parts of the architecture: the managed PC, on which we ran software agents to monitor activity, the web server to store the usage information from the managed PCs and an administrator that could be either an IT staff or a regular user. We will explain in detail the different components.

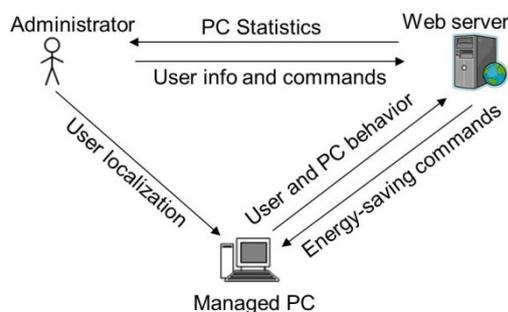


Figure 1: Components of the EnergyAware architecture.

3.1 PC Software Agents

The PCs that will be managed need a software agent running on them. This agent will record user activity with a granularity that could be modulated by the administrator. The purpose of this is to have real time information about how the PC is being used, for example how often is actually used, applications running, and bandwidth usage. The data to be recorded is as follows:

- Keyboard and mouse activity. A 1 is stored if there is input, otherwise a 0 is stored. We would like to know actual time that the PC is used as a fraction of the time it was kept in an On state.
- Current network bandwidth usage.
- List of processes running. We store a short list of processes ordered by amount of CPU used. This information could be later crossed with keyboard and mouse activity to detect idle periods during which certain programs could be using CPU time without user interaction.
- PC Energy State: This is state is not recorded directly but it can be calculated in the server by comparing timestamps of previous logs of user activity.

The variables listed above are retrieved at the same time but this could be tuned according to the needs.

3.2 Web Server

The main purpose of the web server is to act as bridge between the manager and the managed PC itself. It can be considered as a proxy of the commands the manager will send to the computers. The list of tasks to complete is the following:

- Store usage and profile information from each monitored computer.
- Send Hibernate, Sleep or Turn-off commands on behalf of the user to one or more computers
- Serve as web platform for managing application.

The information is stored in a database and the communication is achieved through web services. The reason for choosing web services is that it does not depend on open ports in the server and the computer. The server could be running inside of the company or in the Internet. Every time the computer needs to upload information, it will call a method in the web service. There is a method for creating a new entry (associating a computer to a user in the database) and another method for inserting collected usage data as often as needed.

The commands from the server to the computers can be sent as responses to the web services requests from the computers. The information uploaded from the computers needs no response from the server. If the user needs to send a command to the computer, the web server will get the command and wait for the next upload from the computer. Because the updates are done often, no more than a minute should pass before the PC receives the command. At this time, the server will respond with the command sent from the user. The computer then will interpret it and execute the necessary steps to go to the

requested energy state. Commands can also be sent automatically whenever an energy saving policy is activated.

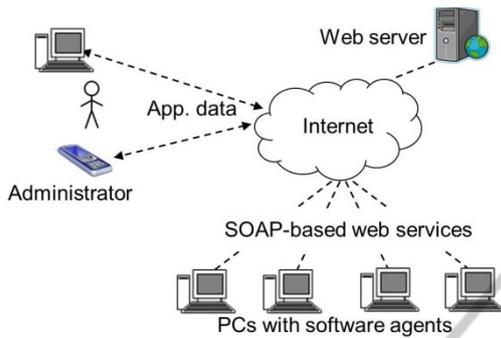


Figure 2: Architecture while used by an administrator.

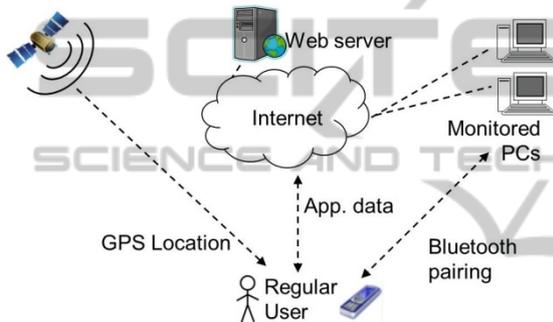


Figure 3: Architecture while used by a regular user.

3.3 Administrator Software

This part of the architecture gives control to the user. It has the expected functionality of showing reports about the different monitored variables in the managed computers. The most important function is to actually control the energy consumption of computers, either automatically or by manually sending the command.

- Automatic control of energy consumption: The localization of the user is used to guarantee that the managed PC will be put in an energy saving state if the user is not close to the computer anymore. The mechanism used will be explained in more details later in the paper.
- Manual control of energy consumption: The user has a choice of disabling the automatic control and turning off the computer whenever it considers necessary.

It makes more sense to be used in a mobile phone, to give more freedom to the user. If the functionalities are accessed through a website in the server, the automatic control is not used. In this case, just the manual control would be of use.

3.4 User Specific Features

Figures 2 and 3 both show the two different scenarios: Figure 2 explains the communication between a group of PCs being managed and the IT staff in charge, and Figure 3 shows a single PC being managed by its owner or user. An administrator might use its mobile application to mostly get almost real time power consumption from the monitored computers. It could also put PCs to sleep if no user input is detected after a long period of time. It is assumed in this case that the computers are not monitored at the same time by its users. In the case of a PC monitored by its user, the options in the mobile application allow it to either manually or automatically save energy by putting its computer to sleep. To calculate power consumption, the IT staff would only need to measure power consumption levels of monitored computers and use the values for the different periods recorded.

4 ENERGY SAVING POLICIES

The proposed architecture is complemented with two energy saving policies. We will calculate later in this paper how their implementation could benefit savings in an office environment.

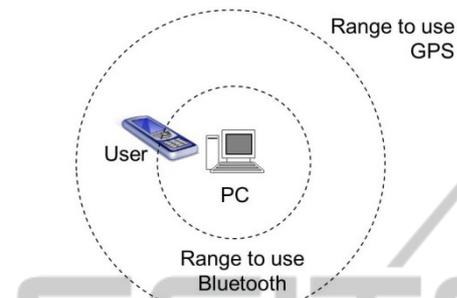
4.1 Location-based Dynamic Energy Saving Policy

With the information gathered from the mobile phone, we tried an approach to save more energy given the current position from the final user. The purpose of the policy is put the PC to sleep when the person goes away from the computer and wake it up if the user is close enough. The novelty of this mechanism relies on that we didn't need extra hardware to apply it. It can be assumed that the monitored PCs have Bluetooth interfaces. We explain now how the PC is associated with the user.

4.1.1 PC-user Binding and Location

The software agent in the computer has an option in the menu where you can register the computer in the server's database. With this, the MAC address of the computer is bound to the user's ID. The user can register as many computers as he/she wants. The same computer can be registered with a new user, but this will erase the previous registration, meaning a computer can be associated with only one person. The software in the phone has also an option to

register the location of the monitored computer. The user and its phone need to be in front of the computer to register its location. It is assumed that the GPS mechanism used by the phone would be able to find an approximate location for the computer the user wants to register. This means that the office should be relatively close to a window for the phone to be able to calculate its position.



- 1) Start with Bluetooth if possible
- 2) If connection lost, switch to GPS
- 3) Use GPS when away. If Bluetooth available, switch

Figure 4: Location mechanism to wake or put PC to sleep.

Once the computer has been associated to a user and its approximate location has been registered, the user can later use this information to turn the computer On or Off automatically. The mechanism to do this is now improved with the use of Bluetooth besides the use of GPS and is shown in Figure 4. We can merge both technologies to improve reliability of the proximity measure. GPS is used to determine initial approximate distance. If the distance is shorter than a threshold, the application will use Bluetooth instead to check if the person is within a reachable distance from the computer. The system requirements are the following:

- The monitored computer will not move from the position it was at when it was associated with the user. If it needs to be moved, the user will have to update its position.
- The phone needs to have Bluetooth activated and the application constantly running in the background to detect if it is close to the computer.
- The computer needs to have a built in Bluetooth receiver.

If the Bluetooth is Off in the phone, there is no way to associate the user's current position with the computer's energy state. The user can still manually send a power management command to the computer. The following steps are used to determine if the user is close to its registered computer:

1. Determine user location using GPS and compare it with the position saved in the server for the monitored computer.
2. If the computer is closer than a threshold then activate Bluetooth to improve accuracy.
3. Activate location algorithm and determine if user passes threshold to suspend/start computer.

4.1.2 PC Sleeping Command

We designed two ways in which the PC can be put to sleep:

- Automatic message delivery: When the mobile application detects that the computer is beyond a threshold distance, it will send a command to the web server, which will forward the message to the computer the next time it is contacted.
- Manual message delivery: The user has an option in the mobile application to put the PC to sleep when it desires. This can be used in case the automatic message delivery didn't work. It can also be used when the Bluetooth is disabled either at the phone or at the PC.

The manual delivery of the message is more intended for an administrator that could send the command to all the computers left On after work time.

4.1.3 Pc Wake-up Command

The PC Wake-up mechanism was designed but not implemented in our prototype. The ideal scenario would be that where the server and the computers are on the same network. In companies networks this is usually not possible due to security restrictions and the size of the network itself. Taking this into account we designed the mechanism with 2 approaches:

1. Using a Magic Packet sent from a Proxy Software Agent located in the same network of the monitored PCs.
2. Using the same Bluetooth interface used for calculating actual location of the user.

The Magic Packet mechanism is the most reliable but implies using a PC in the same network that is guaranteed to be On all the time, so that it can be used anytime to wake up its fellow computers. The design for this mechanism has the following requirements:

- The PC needs to be in the same network than the Proxy in charge of waking it up.
- The PC needs a Network Interface Card (NIC) with the Wake On Lan (WoL) option available

and enabled.

The mechanism that uses Bluetooth to wake up the PC needs this option to be available and enabled in the motherboard. This is not as common as the WoL option which has been available in motherboards for many years already.

4.2 Static Energy Saving Policy

The second energy saving policy is a static policy which works by putting the PCs to sleep after the idle time reaches a desired threshold. It is static because the administrator can set the threshold beforehand and all the computers should go to sleep after that threshold is reached, no matter what. Although PCs already have energy management options like this available, they are many times not enabled, and cannot be monitored or modified easily by administrators.

Our architecture leaves the opportunity of using the cellphone or not. With the phone there is potential for more savings because a dynamic policy could be fed with the information received from the phone. However, a static policy can still be applied if no phone is available.

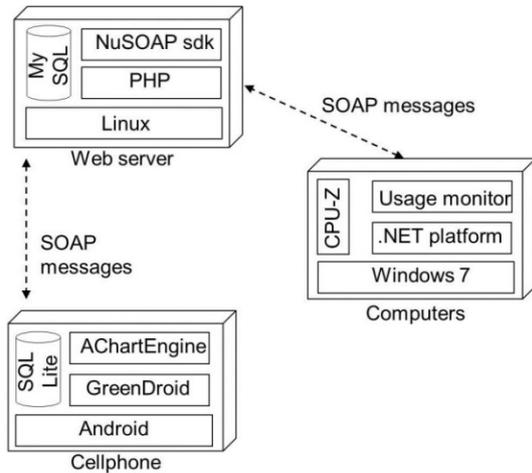


Figure 5: Implementation of the architecture.

This policy could take into account the user’s perspective about what could be annoying or not. For this purpose we interviewed 23 users to determine what could be an accepted period after which the PC could go to sleep without causing annoyance. Table 1 shows the results. We selected those intervals to compare them with typical idle periods for computer usage. We will see in the Evaluation section what we observed about the user behavior. We used the 30 minutes limit because it can be considered that after that, the person might be

away from his/her desk for a long period, e.g. taking lunch. However, in this case, the architecture could learn from each user’s behavior and improve savings. For example, at about 12:00 pm this person usually takes lunch, and it might not be necessary to wait for 30 minutes because chances are, the person will leave for 1 hour.

Table 1: Respondents for each time interval in minutes.

1-5	6-10	11-15	16-20	21-25	26-30	31-more
6	2	5	2	0	1	7

We can see from this, that about 30% of the users prefer the computer to remain On for at least 30 minutes. This is probably due to the annoyance caused by the resume times from the operating system (Mostly Windows 7 in our campus). It is also interesting to see that about 25% of the users are fine if the computer goes to sleep quickly (less than 5 minutes). We consider that those users are the ones with the highest level of consciousness about their impact to the energy consumption of the company. We are running a long version of this interview to detect how user’s perspective about energy consumption affects the final time spent using the computer. Although this is not a conclusive finding, we will use this empiric result to determine when we could put the PCs to sleep with the least amount of annoyance to the user.

5 IMPLEMENTATION AND EVALUATION

5.1 Implementation

Figure 5 shows the three parts of the architecture as we implemented them. The parts are the following:

- Cellphone: The application is designed for Android smart-phones. The code contains about 1000 lines.
- Server Application: it is responsible for receiving the information that the mobile application or desktop application send. The server has a web service that is called by the PC application. The service receives usage information to be stored in the database. The server is also in charge of sending Sleep commands and eventually the Wake-up commands. The code contains about 500 lines.
- PC Application: The desktop application (software agents) runs under Windows operating system and it was developed using C#. It has a

computer usage collection component, a data upload component and a power status command component. The usage collection relies on a simple C program called IdleCollect, developed by Professor Ken Christensen from USF. It outputs keyboard and mouse usage every minute. The data is uploaded every hour to the server. The power status command component applies the Sleep command received from the server. The code contains about 1000 lines.

- **DataSet:** A table stores PC usage information represented in one record per minute containing PC usage state (0, 1, or 2 if the PC was used, not used, or suspended/hibernated), a timestamp, and the identification of the user that owns the PC. Another table identifies the user with IP, MAC, and PC hardware information.

5.2 Evaluation

We ran a study with a group of employees from our university campus. We installed the software agent in their PCs to measure computer usage activity.

We picked for the study users with different job functions looking for different computer usage behavior. In this way we obtained valuable information such as how many hours was the computer Off and On during the day, and were able to corroborate expected behavior. We collected usage information from professors, administrative staff, research assistants, and secretaries. The study ran during 2 months in the summer period, thus some users were not so strict with their arrival to the office when compared to the other periods of the year. We also experienced some issues with the software agents in the computers, especially because of security restrictions from the antivirus.

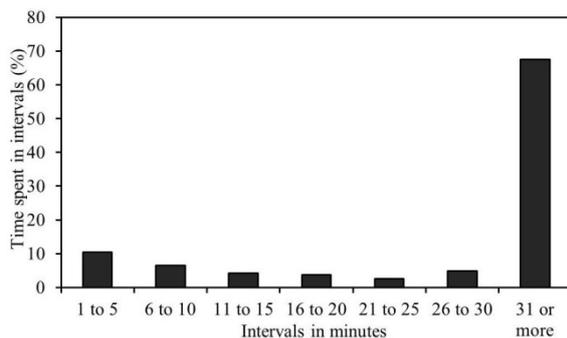


Figure 6: Time spent in idle periods for all users.

We were first interested in checking how the idle periods distribute among the intervals we created before. We can use that to apply energy saving

policies for certain intervals. With the study, we were also interested in checking how much idle periods represent when compared to actual busy times for the computers. Another important task was to observe if there was any difference in computer usage between weekdays. This could be used to establish energy saving policies that could be customized for weekdays.

5.3 Results

Figure 6 shows the percentage of idle time distributed in each of the idle periods for all the 12 users for which we collected computer usage information. We divided the idle time in intervals, which could be used to determine the energy saving policies to be applied to different intervals. We could apply a Sleep policy to the computers when their idle time reaches a threshold interval. The last interval holds all the minutes spent in periods of 31 minutes or longer. It means then that the majority of the idle time is spent in long periods. The largest amount of idle periods might be in the left part of the figure, but the total amount of time spent there is less at the end.

We calculated for 3 users the percentage of idle time spent during each day for a period of 3 weeks. Each idle time was the summation of all the short periods spent in all the idle periods' intervals. The averages ranged from 30% to 60%, which means for example that at a peak point, 60% of the On time was wasted. We picked the 3 users with the largest amount of data. We can however not conclude from our data about the influence of the weekday in the average idle time for the computers.

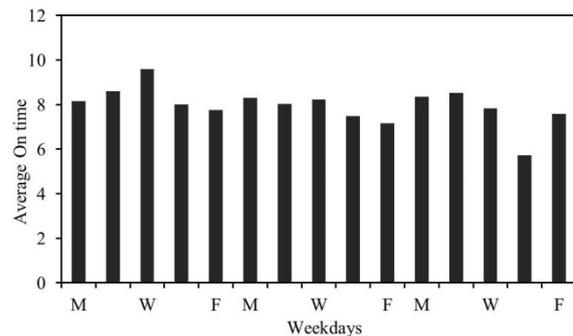


Figure 7: Average On time during weekdays.

Figure 7 shows the On time for the same 3 users during the same 3 weeks period. You can see from the figure that the average behavior from the monitored users looks like what we expected: a working day ranging from 7 to 10 hours. We will

use this information later to simulate for the whole campus the potential energy savings.

5.4 Potential Energy Savings and Discussion

With the previous observations, we gathered enough information to estimate how much energy could be saved in our university campus if policies such as the explained here were put into practice. We are using for this the following inventory of computers in our campus:

Table 2: Inventory of computers in our campus.

Location	Desktops	Laptops
Offices	1766	605
Labs	400	0
PC Rooms	300	0
Total	2466	605

Now, we will use a conservative power consumption of 80 Watts per desktop PC. For this we measured one of the HP typical desktop used in our campus and ranged between 50W and 60W when On, plus 20W for the screen. We use an average of 8 hours per day to calculate a yearly power consumption of 166 kWh per desktop PC. We have to warn here that usually related work calculate power consumption assuming PCs are left On 24 hours during the whole year. And savings are compared to that amount. We assume that at least most of the PCs are turned Off during nights and weekends. When we consider only the desktops in the offices and labs (2,166), we have a yearly power consumption of 360 MW for all the campus. We picked only offices and labs because users should have a similar behavior, i.e. users following some sort of working schedule. However, users in computer labs (PC Rooms) circulate often, meaning different people might use the same computer and there is no fixed schedule. Costs for kWh for the industry worldwide might vary. We will use US value of \$10 cents per kWh.

Now, using the results about idle periods we can apply the two energy saving policies proposed and typical ones in the following way:

- Dynamic: put the PC to sleep when user is far from the computer, assumed to be after 30 minutes.
- Static: put the PC to sleep after a threshold, of 15 minutes.
- Traditional: Putting all PCs to sleep only after working hours (8 hours according to our study).
- No Policy: No monitoring at all, which implies

some PCs might remain On during the night. It is assumed that about 10% of the PCs are left On and the rest are turned Off.

Table 3 shows yearly power consumption for the computers in our campus, and how much savings we could get when compared to regular scheduling. We made a conservative calculation of yearly savings for a company the size of our university. The most beneficial is the static policy so far. However the one with more potential is the dynamic, because it can be further be adapted more aggressively to the user behavior by applying different techniques. We plan in the future to run experiments with such techniques and explore the potential benefits of the dynamic policy. These savings could be improved if factors like the following are taken into account:

- If employees are educated about energy efficiency, their habits might change or at least they might be more open to accept aggressive energy saving policies.
- Policies could be also implemented in other types of devices like office lights.

Table 3: Power consumption and savings for energy saving policies.

	Energy saving policies			
	Dynamic	Static	Usual	No Policy
Yearly Power (MW)	262	247	360	476
Savings	28%	32%	0%	-32%
Yearly Savings	\$9,803	\$11,390	\$0	-

6 CONCLUSIONS

We built an architecture that can be used to detect computer user behavior in a company campus with energy savings in mind. The purpose of our proposal is to adapt energy saving policies to how the computers are actually being used and not just apply computer shutdowns in fixed schedules. Results showed that we can achieve important savings and we consider that our proposal can be easily applied in any company campus.

There might be some concerns that we think our architecture handles but we can check in the future improved versions to handle the following problems better:

- Privacy of user location: the current location of the user is never stored in the server, only the location of its PC. The cellphone will get the PCs location from the server and compare it every time with its own location, without sending it to

server.

- Annoyance of being managed by someone else. This scenario applies to office environment, where usually IT department decides over each user's computer. There is still no personal information stored. In a home scenario, the administrator is the same user, so there is no privacy concern.
- Annoyance of automatically putting PCs to sleep. With the information gathered from the user, we could realize if the user is for example downloading something through the night. This will prevent the PC from being put to sleep. An option in the software interface could be put to allow the user to disable automatic energy state switches if desired.

We will conduct short interviews with computer users to acquire more knowledge about how users behave. This could help us understand how to they conceive power consumption of the devices they are using and how can we apply more efficiently energy saving policies in computers and electronic devices in general in a company campus. We would like to explore implementation alternatives in the phone that would alleviate its power consumption without decreasing energy savings.

REFERENCES

- Accenture, 2012 Consumer Electronics Report. Available at: www.accenture.com [Accessed September 25, 2013].
- Agarwal, Y., Weng, T. & Gupta, R. K., 2009. The energy dashboard: improving the visibility of energy consumption at a campus-wide scale. In *Proceedings of the First ACM Workshop on Embedded Sensing Systems for Energy-Efficiency in Buildings*. New York, NY, USA: ACM, pp. 55–60.
- Agarwal, Y., Savage, S. & Gupta, R., 2010. SleepServer: a software-only approach for reducing the energy consumption of PCs within enterprise environments. In *Proceedings of the 2010 USENIX conference*. Berkeley, CA, USA: USENIX Association, pp. 22–22.
- Almuairfi, S., Veeraraghavan, P. & Chilamkurti, N., 2013. A novel image-based implicit password authentication system (IPAS) for mobile and non-mobile devices. In *Mathematical and Computer Modelling*, 58(1–2), pp.108–116.
- Enterprise Management Associates (EMA), 2008. The True Value of Green IT – An EMA Research Report. Available at: <http://www.kaseya.com/resources/white-papers/the-true-value-of-green-it-an-ema-research-report> [Accessed September 18, 2013].
- Jaros, D. & Kuchta, R., 2010. New Location-Based Authentication Techniques in the Access Management. In *2010 6th International Conference on Wireless and Mobile Communications (ICWMC)*. pp. 426–430.
- Kim, J.-Y., Choi, J.-W. & Choi, K., 2011. Design of automatic energy saving-monitor for reducing the waste of PC electricity. In *2011 The 7th International Conference on Networked Computing (INC)*. pp. 28–31.
- Korn, D. et al., 2004. Power management of computers. In *Proceedings of the International Symposium on Electronics and the Environment. ISEE '04*. Washington, DC, USA: IEEE Computer Society, pp. 128–131.
- IDC Press Release. Available at: <http://www.idc.com/getdoc.jsp?containerId=prUS24375913>. [Accessed October 9, 2013]
- IdleCollect. Available at: <http://www.csee.usf.edu/~christen/tools/idleCollect.c>. [Accessed September 27, 2013]
- Lin, Y.-H. et al., 2011. Implement of an efficient system to reduce power consumption. In *2011 13th International Conference on Advanced Communication Technology (ICACT)*. pp. 1409–1413.
- Reilly, D. et al., 2011. Development of a Power Saving Framework for Use in Large Campus Networks. In *Developments in E-systems Engineering (DeSE)*, 2011. pp. 593–597.
- Verdiem. Available at: <http://www.verdiem.com/> [Accessed September 25, 2013]