UNIVERSITY OF CALIFORNIA, SAN DIEGO

# Using Occupancy Information to Reduce Energy Consumption within Buildings

A thesis submitted in partial satisfaction of the
requirements for the degree
Master of Science

in

Electrical Engineering

by

Bharathan Balaji

Committee in charge:

Professor Rajesh Gupta, Chair
Professor Bill Lin, Co-Chair
Professor Yuvraj Agarwal
Professor Ramesh Rao
Professor Mohan Trivedi

2011

The thesis of Bharathan Balaji is approved, and it is acceptable in quality and form for publication on microfilm and electronically:

Co-Chair

Chair

University of California, San Diego

2011

DEDICATION

To SYH and TJ.

# EPIGRAPH

*But it's not who you are underneath,*
*it's what you do that defines you.*
—Batman

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACKNOWLEDGEMENTS

I am thankful to Prof. Rajesh Gupta and Dr. Yuvraj Agarwal for giving me an opportunity to work in their lab and work on exciting projects. I would like to acknowledge Thomas Weng, with whom I have worked with on all of my projects and without his contribution, these projects would not have been possible. I would also like to thank Seemanta Dutta, Sathyanarayan Kuppuswamy, Michael Wei and Jacob Lyles who have contributed to the projects in various ways with their ideas, continuous feedback and inputs on its different facets. I would like to thank my Mother and Father who have supported me throughout and will always continue to do so. Lastly, I would like to thank my friends and family, who have supported me through this phase of life.

## VITA

| | |
|---|---|
| 2005-2009 | B. Tech. in Electronics and Communication *cum laude*, Visvesvaraya National Institute of Technology, Nagpur, India |
| 2009-2011 | Graduate Research Assistant, University of California, San Diego |
| 2009-2011 | M. S. in Electrical Engineering, University of California, San Diego |

## PUBLICATIONS

Yuvraj Agarwal, Bharathan Balaji, Seemanta Dutta, Rajesk K. Gupta, Thomas Weng - "Duty-Cycling Buildings Aggressively: The Next Frontier in HVAC Control" *In Proceedings of the 10th Conference on Information Processing in Sensor Networks: Sensor Platforms, Tools and Design Methods (IPSN/SPOTS '11)*, April 2011.

Yuvraj Agarwal, Bharathan Balaji, Rajesh Gupta, Jacob Lyles, Michael Wei, Thomas Weng - "Occupancy-Driven Energy Management for Smart Building Automation" *In Proceedings of the ACM Workshop on Embedded Sensing Systems for Energy-Efficiency in Buildings (BuildSys '10)*, November 2010.

Bharathan Balaji, Tamma Bheemarjuna Reddy, B. S. Manoj - "A Novel Power Saving Strategy for Greening IEEE 802.11 Based Wireless Networks" *In Proceedings of the 53rd IEEE Global Communications Conference (Globecom '10)*, December 2010.

ABSTRACT OF THE THESIS

**Using Occupancy Information to Reduce Energy Consumption within Buildings**

by

Bharathan Balaji

Master of Science in Electrical Engineering

University of California, San Diego, 2011

Professor Rajesh Gupta, Chair
Professor Bill Lin, Co-Chair

Buildings account for 73% of the total electricity consumption in the US. To get an in depth view of where this energy is consumed within buildings, we instrument and monitor the buildings at UCSD to study their power consumption patterns. We observe that the energy consumed is not proportional to the occupancy levels of these buildings, thus indicating energy waste. In order to make the power consumption more proportional to its actual usage, we build an occupancy detection system for the CSE building at UCSD. Using this occupancy information as an input, we duty-cycle the different subsystems of the building to save energy. We show that by careful scheduling of the HVAC system based on the occupancy

levels, we can reduce their energy consumption by as much as 40%. Further, we have developed the Smart Energy Meter to monitor and actuate plug loads in the building. This allows us to study the energy consumption patterns on a per device basis. Based upon our smart energy meter, we have developed the Energy Auditor, an analysis engine. It provides feedback to the users and building managers by visualizing the energy consumption data, shows them the opportunity to save energy based on the occupancy patterns and also allows the building managers to actuate the plug loads in case of a demand response event.

# Chapter 1

# Introduction

According to the US Department of Energy (DOE), buildings account for almost 40% of primary energy consumption, 73% of total electricity and 40% of the total carbon emissions in the United States as of 2008. Commercial buildings consume 46.2% of this primary energy usage[9]. Given their relatively long lifespans, buildings constitute a major opportunity for reductions in energy use.

Given their large energy footprint, there has been a plethora of work trying to characterize the energy consumption within buildings. The US Department of Energy has a detailed report characterizing buildings in both the residential and the industrial sectors. The report authored in 2008, provides a detailed breakdown of the contributors to the energy consumption, the chief sources of energy generation, the environmental impact of the buildings and other significant statistics[9]. At a smaller scale, plug energy meter called ACme has been developed by UC Berkeley and deployed massively to trace out the energy patterns within a building[18]. Google Power Meter and Microsoft Hohm are other examples of commercial efforts, providing visualization of electricity consumption provided the Smart Meter installed can send data in their standard API[13, 27]. The Energy Dashboard project from our group provides a visualization of the power consumption of the entire UCSD campus[3].

As a part of our Energy Dashboard project, we have instrumented the Computer Science and Engineering building to obtain a breakdown of its power consumption. Analysis of the data obtained showed that there are three major subsystems contributing to the electricity consumption of the building – the HVAC system, individual plug loads, lighting and the computing loads in the machine room. This is in accordance with the data shown by the DOE[9] and Perez-Lombard et al[30]. The HVAC system in CSE consumed 25% to 40% of the total power consumption. Thus, we studied the HVAC system in further detail.

Most of the HVAC used in commercial buildings runs on a static schedule, from morning to evening on weekdays. It does not take into account the actual occupancy levels of the building. Modern HVAC systems divide the building into different thermal zones and are capable of controlling them independently. If occupancy information is available, we can duty-cycle them to save energy. We developed our own occupancy sensors and deployed them across one out of four floors of the CSE building at UCSD. The building energy managers allowed us to control the HVAC system based on the occupancy information received. We actuated the air-conditioning for each zone based on real time occupancy information for one floor out of the four floors of the CSE building. Our results show that we saved 10% of the HVAC power consumption during our testing. If the system is deployed across all the floors in the building, we estimate savings of up to 40%.

Another important contributor to the total energy consumption in buildings is plug loads. To monitor the power consumed at each power point, we developed our own plug load energy meter called the Smart Energy Meter (SEM). This allows us to study the energy consumption of individual devices, individual rooms and identify how much energy each person in the building is using. Combining this information with the occupancy information that we are collecting, we were able to estimate the amount of energy being wasted in each room. The energy waste information gives feedback to both the occupants and building manager on where they can save energy by changing their usage patterns. Actuating the devices is tricky though, as some loads like desktop PCs have to be gracefully turned off. To alleviate this concern, we attempt to identify the type of the device that is connected to our energy meter. Using this type information gives the user opportunity to automatically turn off their devices based on a set of expressive policies. We have implemented this system and shown some preliminary results.

The rest of the thesis is organized as follows. Chapter 2 gives a detailed view on energy use by buildings and its subsystems. Chapter 3 explains our occupancy detection system and how it is used to control the HVAC of the CSE building. Chapter 4 explains our energy metering solution and the accompanying analysis engine with actuation mechanisms. Chapter 5 concludes the thesis.

# Chapter 2

# Trends in Building Consumption

Buildings are diverse systems in terms of their energy use modality. To optimize its energy consumption, its important to know which subsystems are major contributors to the total energy usage. Then, we can study these subsystems and try to come up with solutions to make them more efficient. Recent work has tried to solve this problem by providing a visualization of the electricity consumption of buildings, and use it as a feedback to the users[3, 13, 27]. The Energy Dashboard project from our lab provides real-time plots of energy consumption of the UC - San Diego(UCSD) campus - from its overall consumption, to individual buildings, to subsystems within buildings and even some individual plug loads[3]. Google Power Meter [13] allows any Energy Meter compatible with their API to store the data in Google's servers and provides real-time visualization of the consumption patterns. The Green Soda project from UC - Berkeley [18] also provides a similar visualization using the ACme nodes as their energy metering solution.

## 2.1 UCSD as a Testbed

UCSD acts as an ideal testbed for such a study. The campus sprawls over an area of 1200 acres, with a daily population of more than 45,000 people, of which 29,000 are students. 10,000 students reside on-campus in UCSD housing. There are a total of over 450 buildings, with ages varying from a few years(CSE) to almost a century(Scripps), and usage patterns varying from residential apartments to theaters to departments. It would not be an exaggeration to call it a city in itself.

UCSD has taken an ambitious goal of taking the campus off the electricity grid. As a part of this effort, the campus has an extensive energy generation, storage and management system in place to deliver both electricity and thermal energy. The thermal energy is delivered in the form of hot and chilled water loop to buildings across the campus. The centralized Energy Management Systems(EMS), by Johnson Controls, manages HVAC systems of 60 of the largest buildings on campus. The electricity is generated using a 30MW natural gas co-generation plant, 2.8MW Fuel Cells and 1.2MW of solar panels. This amounts to 82% of the annual electricity demand from the 42MW (peak) microgrid. The remainder is imported from SDG&E.

To monitor and manage this enormous system, the buildings have been instrumented to provide real-time energy usage data. The electrical energy is measured using three-phase high accuracy PowerLogic meters from Schneider electric, communicating with the central campus servers using a wired network. The thermal energy demand of the building is calculated using various measured parameters like the rate of the chilled water passing through the building and the temperature of the water as it enters and exits the building.

The Computer Science and Engineering(CSE) building has been further instrumented for fine grained measurements. There are several interesting characteristics of this building. Constructed in 2004, its a fairly modern building, consisting of low-E glass windows, a zonal and floor-by-floor control of the HVAC system, and motion sensor based lighting. Cooling for the building is provided by the campus hot and cold water loop with the HVAC system running on electricity. CSE has a total of about 1200 occupants, 750 Desktop computers, one machine room for servers and six instructional computer labs. A total of 15 separate circuits were metered so that the building energy consumption can be broken down by lighting, plug loads, HVAC and machine room.

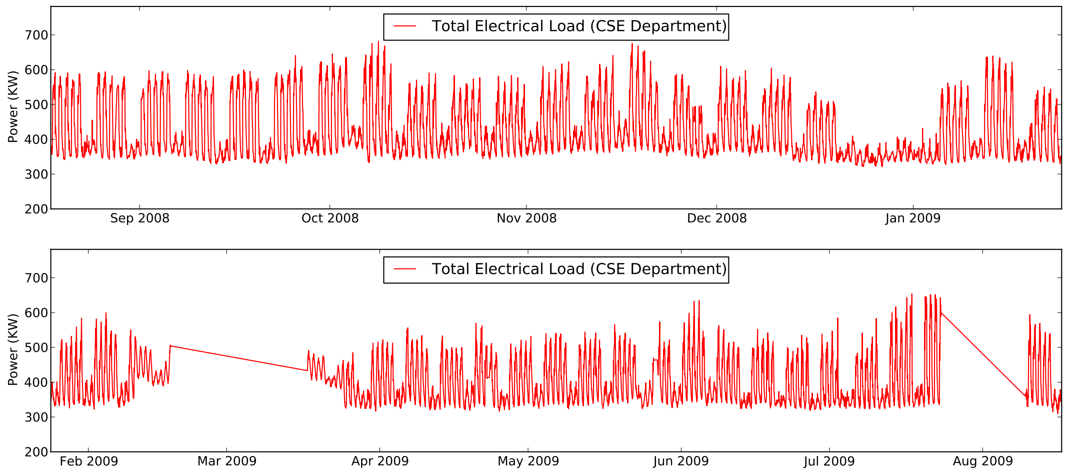


**Figure 2.1**: CSE Mixed-use Building: Total electrical load for a year (August 18th, 2008 through August 16th, 2009). While the daily load varies by as much as 250KW, it never goes below 325KW. This is the **base load** of the CSE building.

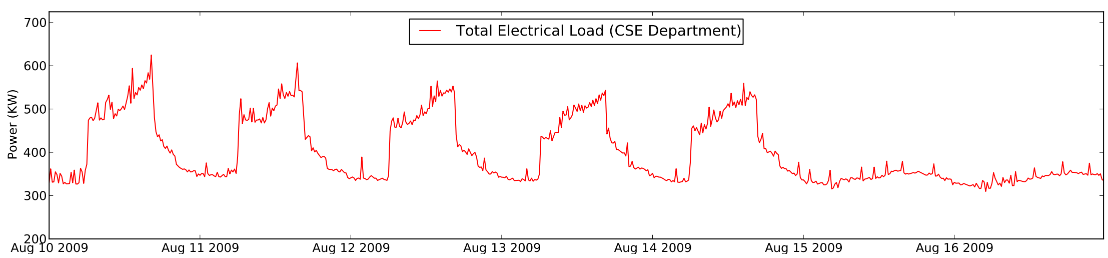


**Figure 2.2**: Variation of the electrical load in the CSE building at UCSD during a week in August.

## 2.2 Building Level Trends

Figure 2.1 shows the electricity consumption of the CSE building through the year 2009. We can see that there is not much variation in the pattern from season to season. This is because San Diego enjoys a temperate climate throughout the year. The peak power is consumed during the summer months, when more cooling is required. However, we can observe that there is always a “baseline” consumption, irrespective of the season. This can be clearly seen in Figure 2.1 during the winter break from December 18, 2008 to Jan 5, 2009. The minimum power consumption is as high as 325kW. During this period, no classes are held,

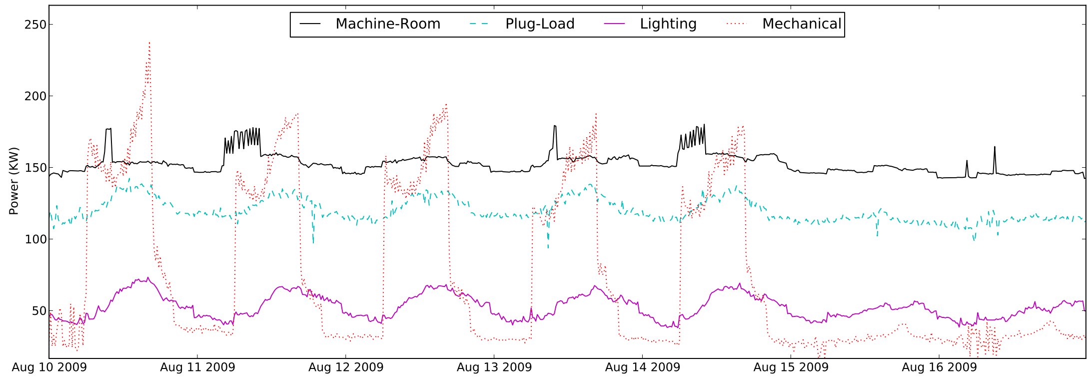


**Figure 2.3**: Detailed breakdown of the energy consumption of the CSE mixed-use building. The data presented is for a week in August.

there is very little activity in the labs and most of the faculty are on vacation. A similar trend can be seen in almost all the metered buildings across UCSD.

Figure 2.2 shows the energy consumption pattern for a week in August, 2009 in CSE. Its clear that the energy consumption goes high during the day and comes down again in the evening. As pointed out in Figure 2.1, there is a significant baseline consumption. This pattern can be split up as baseline power and dynamic power. The CSE building has a baseline power consumption of 62.2% of the total power consumed. For different buildings across the campus, a similar pattern can be observed. However, this percentage of baseline to total power varies with its usage patterns, architecture, surrounding environment (buildings in shadows experience lesser temperatures than those exposed to sunlight) and numerous other factors. To site a few examples, the Hopkings Parking Structure has a baseline power of 85.3% of the total power, and in the Cognitive Sciences building it is 81.3%. Clearly, to make buildings more energy efficient we need to make this baseline consumption as low as possible. The submetering in the CSE building gives us further insight into the consumption patterns by providing the breakdown of the total power consumption in the next section.

## 2.3 Building Subsystem Trends

The CSE building has been specially metered to provide electrical power consumed by different subsystems - subsystems in the machine room, lighting circuits to individual floors, the electrical load from the HVAC system, the elevator load from all the plug loads. Figure 2.3 combines these power values into four different types of loads - the Machine Room, Lighting, Plug Loads and Mechanical loads. The power consumption pattern from August 10, 2009 to August 16, 2009 is shown.

The power consumed by lighting varies from 46kW to 63kW on a typical day. Surprisingly, it consumes only 11.6% of the total electricity. It should be noted that the building has employed Compact fluorescent lamps(CFL) and tubelights for its lighting. Also, they are directly connected to the locally installed motion sensors, which switch off the lights automatically after 30 minutes of inactivity. The power consumed by the Machine Room is a whopping 150kW on an average during the week, and remains almost constant (12% variation) irrespective of the workload. A look at the power consumption levels of the Machine Room throughout the year shows that the minimum power consumption is as high as 120kW (through the months of January to April). This indicates that the servers are not optimized to consume less power during periods of inactivity. Recent work from Google have shown similar results [4]. The authors show that Google servers are only at 30% utilization on average.

As a department of computers, most of the plug loads in the building consist of laptops and PCs. As a part of the Somniloquy project[1], more than 750 desktop PCs were accounted for in the building. On average, these PCs consume 100W, accounting for close to 100kW of electricity consumption. The Plug Loads graph in Figure 2.3 shows that their consumption varies between 110kW to 140kW. The high baseline for the plug loads can be attributed to PCs which are not put to sleep. Recent studies have shown that people do not put their computers to sleep because of numerous reasons – they want to remote login to their computers, have background processes running, need quick availability, etc.[1]

The Mechanical load takes into account the power consumed by air handlers, the climate control units and the elevator. The elevator only adds a marginal load (maximum of 8%) on this subsystem, and most of the energy can be attributed to the HVAC system. The HVAC system follows a static schedule, switching ON at 6AM, providing air flow to the entire building, and going to standby mode at 6.30PM. In standby mode, the users in the building have to indicate their presence manually using the installed thermostats and only those zones will be ventilated by the system. A sharp increase in the power consumption can be seen when the HVAC system starts up at 6AM. The electricity demand slowly increases throughout the day, as more occupants pour in and the day temperature increases. There is a sharp fall at 6.30PM as the HVAC goes into standby mode. However, the consumption is still significantly higher than that during early morning hours. As more and more occupants leave the building, the demand decreases to the baseline level (at about 8PM).

## 2.4 Discussion

In order to make buildings energy efficient, we will have to tackle the problem from several different directions. Modern buildings are being built with thermally efficient materials, low-E window glazing, solar reflecting roofs, overhangs to block solar radiation on windows, and even building shape and orientation are designed to save energy[12, 23]. The equipments installed in the buildings strive to be increasingly energy efficient - programmable HVAC systems, low-flow plumbing and motion-sensitive lighting. The newer buildings at UCSD go even a step further, with natural ventilation, reclaiming waste water for irrigation, rooftop solar panels and using green sustainable flooring, cabinets and paint.

Most of these techniques are useful to lower the overall energy consumption of the building and help in reducing its carbon footprint. However, very few techniques, like motion sensitive lighting, concentrate on reducing the energy being wasted in daily usage. A detailed energy monitoring system in place allows us to examine this aspect of the problem and helps us come up with solutions to tackle them. The following chapters go on to show that the HVAC systems and the plug loads energy consumption are not proportional to the occupancy levels of the buildings. We come up with solutions to identify such situations and actuate the building systems to help reduce the wasted energy.

# Chapter 3

# Occupancy Controlled HVAC System

The power consumption patterns from the Energy Dashboard show that heating, ventilation and air conditioning (HVAC system) consume significant amount of electrical energy. DOE states that HVAC takes up to 45% of the primary energy consumption in residential buildings[9]. Perez-Lombard et al [30] state that HVAC systems in office buildings consume 48% in the US, 55% in the UK and 52% in Spain. The HVAC electrical load for the CSE building was between 25% to 40% of the total load. This does not account for the thermal load that goes in to cooling and heating the water that goes in and out of the system. The energy consumed for this purpose ranges from 100kW to 200kW.

All the buildings in UCSD are managed by a central Energy Management System(EMS) and are operated on a static occupancy schedule of 5.15AM to 10.00PM on weekdays. This is common practice in commercial buildings to accommodate the standard working hours. At other times, the occupants have to manually indicate their presence and turn on the air conditioning for their room. However, not all the occupants come in as early as 5.15AM or leave as late as 10.00PM. The staff typically arrive at 8.00AM and leave at 4.30PM. The graduate students arrive at around 10.00AM and leave at varying times in the evening. Further, occupants leave their offices for lunch and meetings for extended periods of time, there are offices which are empty, conference rooms which are occupied for only a few hours in a day and storage rooms which are seldom visited.

Using occupancy information to drive energy management within buildings has been explored extensively both in commercial products and by the research community. The most common use is to detect presence in physical spaces using Passive Infra Red (PIR) based motion sensors to drive lighting systems[8]. In most cases these PIR sensors are hard wired to the buildings and are not a compelling solution for existing buildings. Recognizing this limitation, commercial wireless occupancy solutions [10, 17] have emerged that allow direct control of a single HVAC unit in a home or a light fixture. Recent work has even proposed the use of coarse grained information such as occupants entering or leaving their houses to drive a smart thermostat[26]. Researchers have even proposed the use of ultrasonic sensors to detect height differences between occupants[33] as well as network access information[22] to identify individuals within shared spaces. We have developed binary occupancy sensors and collected the information in a centralized server to manage the HVAC system of the CSE building.

The CSE building is divided into different thermal zones, and each thermal zone can be independently controlled by the HVAC system. Such a system was constructed so that the ventilation is provided to only those zones which are currently occupied. However, this facility is only used when the HVAC system is in standby mode and the occupants indicate their presence manually. If the occupancy of each of these zones could be detected automatically and reliably, the HVAC system can then schedule the ventilation according to this information. This ideal HVAC system would then have a baseline energy usage to cool the common areas like lobbies and hallways, and the dynamic part will depend on the zones which are currently occupied, making the HVAC energy demands proportional to the occupants in the building.

The rest of the chapter explains our system in detail and is organized as follows. Section 3.1 explains the HVAC system of the CSE building. Section 3.2 shows the potential energy savings that can be achieved if a fine grained occupancy driven HVAC system is in place. Section 3.3 provides the characteristics of the occupancy sensing node developed by us. Section 3.4 explains the actuation mechanisms employed based on the occupancy information. Section 3.5 gives the results obtained and Section 3.6 discuss the related work and future challenges.

## 3.1 The HVAC System

Modern buildings typically have centralized HVAC systems. Cold air is supplied by huge air handlers in the basement of the building, which reaches the individual offices using the air-conditioning ducts. The building is divided into different thermal zones, each of which is directly controlled by a Variable Air Volume(VAV) box. The VAV boxes control the flow of air into the rooms using dampers. The whole system is managed using a centralized Energy Management System(EMS).

The Computer Science and Engineering building at UCSD consists of four stories and a basement. The cold and hot water loop from the Central Utilities Plant(CUP) provides the chilled water that acts as a heat exchanger to cool the air passing through the air handlers. There are four large air handlers in the basement of the building and provide the cool air using duct framework at 55°F. The whole building is divided into more than 300 thermal zones, each of which can be controlled independently by the Energy Management System(EMS). As in a typical system, each thermal zone is served by a VAV unit to control the amount of air flow. As the air temperature at 55°F might be too cold for some regions, a facility for reheating the air is provided before it can enter the room. Each of the rooms is configured with cooling and heating setpoint temperatures within which the temperature of the room should be kept in. The cooling setpoint for the room are statically set and vary from 71°F and 78°F (for storage rooms). Most rooms are set to 72°F. The heating setpoint varies from 66°F to 68°F, depending on the room. The required air flow that the air handlers need to provide is controlled by a PID controller using a pressure sensor in the main trunk of the air duct as its feedback element. As additional VAV units release more cold air into the building, the pressure drops in the air ducts forcing the air handlers to throttle up. Conversely, as VAV dampers close, the air pressure builds up, signaling the air handlers to slow down.

All the buildings are managed by the Physical Plant Services(PPS) at UCSD using the EMS. Each thermal zone in every building on the campus can be controlled using a BACNet control network. This control network is centrally managed by a Metasys ADX server which provides access to all of the elements in the network. Thermostats have been installed in each zone, which give feedback on the current temperature in the zone. There are three basic command modes for each of these zones to the EMS - **on**, where air is released to allow the room to satisfy its temperature requirements; **stand-by**, where a minimum amount of air flow is maintained, and **off**, where the dampers are completely closed.

The building HVAC is scheduled statically with occupied mode starting at 5.15AM for the fourth floor, 5.30AM for the third floor, 5.45AM for the second, and 6.00AM for the first floor. The start times of the floors are staggered to avoid causing too much stress on the air handler units. The occupied mode lasts till 6.30PM, with stand-by mode set from 6.30PM to 10.00PM, and unoccupied mode set from 10.00PM to 5.15AM. When the building is in unoccupied or stand-by mode, occupants can turn on the HVAC by pressing a button on the thermostat. A zone can consist of one to three rooms, and there is only one thermostat for each zone. If an occupant does not have the access to the room that has the thermostat for that zone, they will not be able to turn it on. The system is set to unoccupied mode during the weekends.

The PPS people responsible for the HVAC systems have revealed interesting details on how the system is optimized and can be improved to be more energy efficient. The chilled water provided by the CUP is produced partly as a byproduct of our campus natural gas fired plant, and is relatively inexpensive. So, the EMS optimize the building HVAC system for cooling. Further, experience has shown that over cooling, in general, generates less complaints than a hot building. The air that flows from the air handlers through the ductwork is close to 55°F, and the VAV minimally reheats the air to save energy. The weather in our location is mild and warm year-round, and the heating component of the HVAC network is only put to real use during the colder days. The static schedule practiced has been put in place to best accommodate the diverse work times of the occupants, since real-time reliable occupancy information is not available. This type of schedule is common among modern buildings.

For our deployment, we were allowed to control only the 2nd floor of the CSE building. This floor contains a total of 81 rooms and has a mixture of faculty, graduate students and staff workers. There are 11 graduate students labs, housing over 100 graduate students, which become occupied when the first graduate student comes in (usually between 8.30AM and 10.30AM). There are 23 faculty offices, of which four are unoccupied at present and eight are occupied only occasionally. The occupancy of the remaining offices varies greatly depending on their schedule. There are 11 affiliate offices that are occasionally occupied when affiliates come in or when there is an event in the building. A total of 17 staff offices are present, of which 5 are currently empty. Staff workers start entering between 8.30AM to 9.30AM and leave between 4.30PM to 6.00PM. The rest of the rooms include 1 classroom, 5 hardware labs that are occupied as needed, 2 computer labs, 4 conference rooms, 3 kitchenettes, and several equipment and storage rooms that do not contain occupants. Each of these 81 rooms falls under one of 55 thermal zones on this floor. The hallways are divided in to two zones, each of the labs and conference rooms are in their own zone, several of the office rooms are also in their own zone, and the remaining zones represent two to three office rooms.

## 3.2 Motivational Experiment

In the previous chapter, we saw that the CSE building has been specially instrumented to provide us the breakdown of electricity consumption from different type of loads in the building. Using this facility, we could measure the electrical energy consumed by different components such as air handlers, pumps, fans and actuators. To measure the thermal energy consumed by the HVAC system, we have installed thermal meters on the inlets and the outlets of the chilled water loop provided by the CUP. This thermal energy is measured in MMBTUs (Million British Thermal Units) that can be converted to equivalent kW based on the

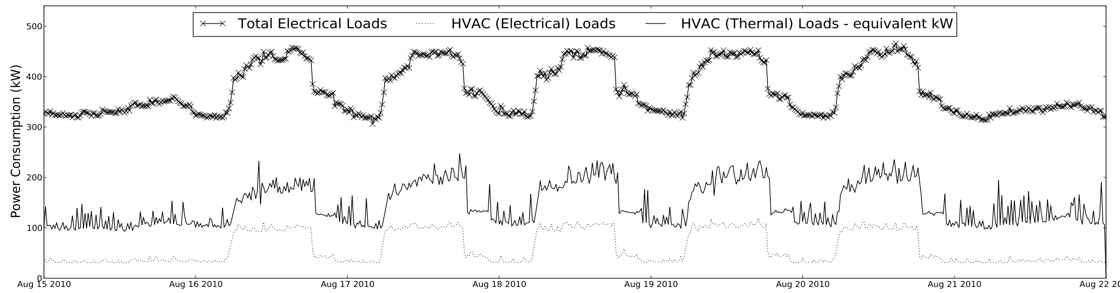


**Figure 3.1**: Power consumption breakdown for our building. The HVAC electrical load is between 25% to 33% of the total electrical load. The HVAC thermal load, as expressed in kW equivalent of cooling, is also significant.

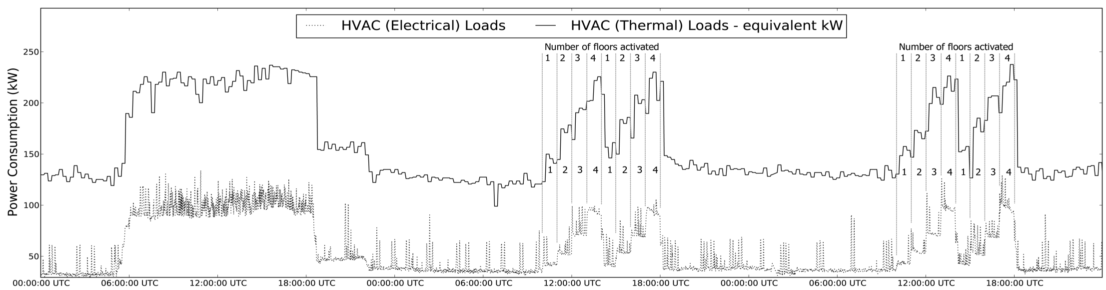


**Figure 3.2**: HVAC power consumption - Friday Oct 22nd to Sunday Oct 24th. The first day is a regular day (Friday).The next 2 days we turn on the HVAC one floor at a time – starting from 1st floor to all four floors at intervals of one hour, repeating twice a day.

aggregate energy sourcing architecture as supplied by our campus energy managers. Note that since the chilled water loop is produced partly as a byproduct of our campus natural gas fired plant, the conversion factor into kW is an approximation and should not be taken literally as the energy it would take to chill water using electric power.

Figure 3.1 illustrates the power consumption of the HVAC system with respect to the total power consumption. As we have seen from the trends in power consumption from the previous chapter, the HVAC electrical loads rise rapidly in the morning, stay high during the day and reduce gradually in the evening. During this particular week, the HVAC electrical loads account for 25% to 33% of the total electrical load of the building. The HVAC Thermal equivalent load (in kW) is also significant and follows a similar pattern as the HVAC electrical load. The combined HVAC electrical and thermal load during nights and weekends is due to the fact that a minimum air flow must be maintained for the basement laboratories at all times. Further, the building has a small server room which must be air conditioned at all times.

Figure 3.1 provides insight into the energy consumed by the HVAC system in our building, giving us opportunity to study its contribution to the total energy consumption and where energy can be possibly saved. To save energy, we planned to install accurate occupancy sensors in the individual offices and turn off the air conditioning to the unoccupied zones. However, before undertaking the effort of designing and deploying the system, we needed to know the amount of energy savings that can be achieved by the system. If the amount of energy saved is minimal compared to the total cost of the system, then the solution is not economically feasible. We designed an experiment to find out the potential impact the variations in occupancy have on the energy consumption of the HVAC system. This would provide a bound on the potential energy savings a detailed occupancy driven HVAC system may have.

The basic plan behind the experiment was to initially turn off HVAC in all the four floors of the building, and then incrementally turn on the HVAC system on each floor to see the change in energy consumption. We conducted this experiment spanning three days - from October 22-24, 2010. Friday, October 22nd, was our baseline day, and we actuated the HVAC on both Saturday and Sunday. We turned on the HVAC on the first floor from 10.00AM to 11.00AM, the first two floors from 11.00AM to 12.00PM, the first three floors from 12.00PM to 1.00PM, and all four floors from 1.00PM to 2.00PM. We repeated the same pattern from 2.00PM to 6.00PM. The weather for the two days were mild and typical of San Diego, with a high of 77°F.

Figure 3.2 illustrates the results of our experiment spanning the three days. Both HVAC electrical and thermal loads have been shown for comparison. Friday, our baseline day, was running the normal static schedule imposed by the EMS. We can clearly see that the amount of energy consumed in both electrical and thermal

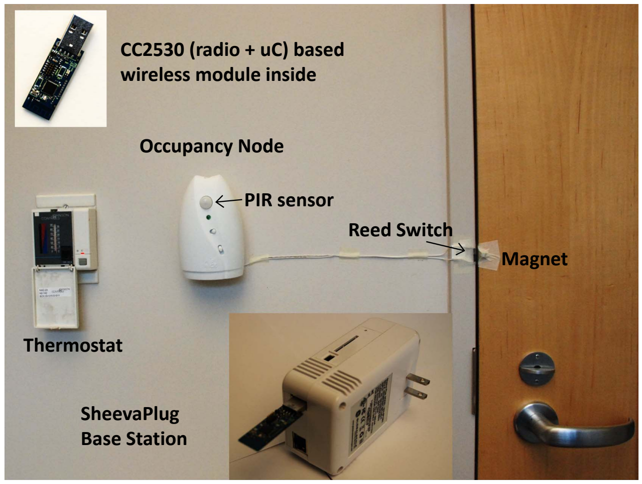


**Figure 3.3**: Occupancy node deployed on the wall of an office. The reed switch, PIR sensor and our CC2530 based radio module inside the occupancy node are also shown.

load is monotonously increasing as the HVAC in each additional floor is being turned on. It is interesting to note that the HVAC electrical load is increasing in a step wise manner, and the step size increases as each additional floor is being activated for cooling. This means that the closer the HVAC system is to maximum cooling, the more expensive cooling each additional zone becomes. One possible explanation for this is that the power consumed by the fans in the air handlers is cubic with its fan speed. What this implies is that reducing HVAC loads during the workdays for even a few zones can potentially have significant savings in energy. In other words, the sensitivity of energy consumption to occupancy increases at higher occupancy levels, making a very strong case for dynamic HVAC control.

## 3.3 Development of Sensor Nodes

In order to have a fine-grained occupancy based HVAC control system, it is critical to have accurate, reliable and real-time occupancy information of each zone in the building. Many modern buildings use Passive InfraRed sensors(PIR) for sensing motion. However, these PIR sensors detect only motion, and can give inaccurate results. Thus, these are connected directly to the local lighting fixtures and are rarely used for intelligent HVAC management. Other methods for detecting occupancy that have been studied in the research area include sonar-based methods [34] or camera based systems [35] that bring up concerns relating to cost, deployment and privacy issues. Carbon dioxide sensors have also been examined - the main limitations of these systems are that they are very slow to respond in detecting the changes in occupancy and need to be calibrated for every environment[37].

**Objectives**: A practical occupancy detection system needs to meet several key design objectives. First, sensor nodes have to be as inexpensive as possible to make it economically feasible to deploy them on a building wide scale. If the cost of the sensors is high compared to the energy savings accrued by the system, building managers will be reluctant to adopt the system, and the idea will remain as a research project. Second, we wanted the system to be incrementally deployable within existing buildings, without requiring large scale modifications such as new wiring. Not only does wiring increase the cost of deploying the system, it involves several hours of manual labor, requires design effort in retrofitting the wires into the existent system and is expensive, if not prohibitive, to modify or repair the system. Hence, it is imperative that the sensor nodes transmit the information wirelessly. Further, wireless sensors allow us to easily experiment with the system during the prototyping stage. Finally, the occupancy detection algorithms should be very accurate since it is critical to minimize the errors when controlling the HVAC system.

**Sensors**: We chose to use a combination of two sensors to deduce the occupancy in a room - a magnetic reed switch to detect if the door is closed or open and a PIR sensor module to detect motion. Figure 3.3 shows our sensor module. The reed switch we have used is a normally open switch consisting of two metallic plates that are close to each other. When a magnetic field is present in the right orientation, the two metallic plates make contact, turning on the switch. This switch is inexpensive and draws no current when the switch is open. When the switch is closed, the amount of current flowing through the circuit can be controlled using a series resistor. We detect the status of the door by placing the reed switch on the wall near the door and a small magnet on the door itself, as shown in Figure 3.3. The PIR sensor is based on a pyroelectric sensor which converts incoming infrared radiation to electrical signals. Two such sensors are placed close to each other so that a difference in infrared radiation in the environment can be detected. A heat emanating body motion can be detected up to 10 meters away from this sensor. A Fresnel lens is placed on top of this arrangement to increase the angles of incidence captured by the sensor. The lens helps capture motion up to 120° in the horizontal and 90° in the vertical directions. The exact angles depend on the design of the lens. The differential signal is then amplified and processed to give a digital signal.

**Microcontroller**: We chose Texas Instrument's (TI) CC2530 as our microcontroller. This chip is a System-on-Chip (SoC) solution consisting of 2.4GHz IEEE 802.15.4 Compliant RF Transceiver, an industry-standard enhanced 8051 microcontroller core, in-system-programmable flash, dedicated hardware for AES encryption and TI's software solution for Zigbee, called Z-Stack. The single package reduces both cost and form factor of the overall module. The 8051 core allows us to easily detect the signals provided by the sensors using GPIOs (General Purpose input/output) and apply our algorithms on them. CC2530 comes with various sleep modes, which allows to optimize the performance for our battery operated sensor nodes. We modified TI's CC2531 USB Dongle reference design for CC2530 and manufactured them for our sensor nodes. We chose this design as it was compact and has a printed PCB antenna rather than a separate external antenna. While this choice allowed us to reduce cost, it reduced the radio range of the nodes. However, our experience with deploying them showed that the range was sufficiently large for meeting our requirements.

**Case**: To house the entire module, we needed a case. To reduce the effort of designing our own case during the prototyping stage, we chose to use readily available Airwick motion sensing air fresheners. They have a PIR based motion sensor in them and squirt the scent on detecting motion. We retrofitted our module into this product by removing the liquid scent and stuffing it with our module instead. The PIR that came with the Airwick had good sensitivity compared to some of the more expensive ones available in the market, so we re-purposed the PIR circuitry to be used with the rest of our system. The case also came with space for 3 AA batteries, which is perfect to power up the CC2530 module. The total estimated cost of a single sensor node in quantities of 1000 is less than $15. This includes the cost of the parts, casing, PCB fabrication and assembly.

**Algorithm**: The algorithm for detecting occupancy tries to use the sensors judiciously. The reed switch is able to sense when the door is open or closed. Based on the typical occupancy modalities of our building and other buildings around our campus, we observed that almost everyone closes their office door when they are either leaving for the day, or when they are going to be out for more than a few minutes. Thus, our occupancy detection works as follows. When the door is open, we mark the room as occupied. When a door closes, there are two possibilities. Either the person closed the door and headed out (room unoccupied), or the person just closed the door and is still inside the room (room occupied). If the PIR sensor goes high, it means that there is still a person inside and we mark the room as occupied. If the PIR sensor does not detect motion, then we decide there is no occupant in the room. However, the PIR can be triggered by the motion of the closing door itself, or from the rush of air that accompanies it. To compensate, we have developed a simple algorithm that will ignore the first six seconds of pulses from the PIR sensor and sample constantly for two seconds after that. There is one scenario where we will incorrectly declare a closed room to be unoccupied. If a visitor closes the door while the main occupant of a room is sitting relatively still at his desk (like reading a book, or typing on his computer), the PIR sensor will likely not detect movement and thus determine the room to be empty. To account for this, we turn on the PIR interrupt whenever we mark a room as closed and unoccupied. If we detect movement in the future, the CC2530 will wake up, poll the PIR and check the resultant pulse pattern for occupancy. If it passes, we determine the room to be actually occupied. Thus, there are three types of messages the node can send with respect to occupancy - **open-door occupied**, **closed-door occupied**, and **closed-door unoccupied**.

**Wireless Network**: We have chosen the ZigBee protocol as our choice of the wireless network stack. We made this choice because it is the only protocol which has been standardized by the industry, and is designed with the Smart Home applications in mind. Further, the CC2530 chip has been designed with ZigBee as an intended application. The Z-Stack implementation of Zigbee provided by Texas Instruments has been ported to this hardware, reducing our design effort in the process.

We have followed a star topology for our network. This topology serves our purposes much better than a multi-hop mesh network. First, we intend to use the same Zigbee network for all the smart building solutions. For this we would like to make full use of the narrow bandwidth (256 kbps) provided by ZigBee. For each hop, the throughput of the network decreases roughly by half. Second, since we have building-wide Ethernet and WiFi, we connect the Coordinators of each ZigBee network to this infrastructure, making the data collection and integration relatively simple. In buildings without this infrastructure, a multi-hop network might be the only choice. Third, because each star network is logically separate from the other, we can reuse the channels as in cellular networks. We can also have multiple star networks overlapping each other as long as they are on different channels. Finally, it is much easier to deploy and maintain a star network rather than a mesh network.

Each of the occupancy sensor nodes acts as an End Device in a ZigBee network and communicates directly with its ZigBee Coordinator. The ZigBee Coordinator is a CC2531 USB Dongle, similar to the end devices, with a small form factor and a printed PCB antenna. The only difference is that CC2531 has a USB controller and can communicate with a USB Host directly. Each of these USB Dongles are connected to inexpensive $100 Linux based plug computers called the GuruPlug and SheevaPlug, both of which have a 1.2GHz ARM class processor, 512MB of memory, flash storage, Ethernet and USB ports. We chose the plug computers for their low cost, small form factor, low power (typically less than 5W), the availability of several expansion ports and safety certification from UL. We call the ZigBee coordinator + Plug computer as our "basestation". Each basestation is connected to the Ethernet infrastructure of the building.

Our preliminary experiments showed that the range of the sensor nodes was close to 20 meters line-of-sight and spanned close to 10 rooms in one corridor of the building. We deployed a total of nine basestations to cover all the dead spots in the network and span the entire 2nd floor of the building. We configured our occupancy nodes to automatically connect to the nearest basestation with the correct extended PAN ID and to start sending data to its parent. Using the Zigbee stack allows us to leverage many of the features of the stack, such as authentication and AES encryption for security. Once the wireless nodes are connected, the nodes will send even messages whenever an occupancy event happens. In addition, the nodes transmit a heartbeat message every 15 minutes so that the base station and central server can determine if a node has fallen of the network. This heartbeat period can be made even longer to conserve energy and further increase node lifetime.

**Power Characterization**: An important design goal for our wireless occupancy node was to make it battery powered, necessitating aggressive energy management. Our choice of using the CC2530 was in part because of its low power consumption. However, since our occupancy node combines several sensors with the CC2530 we wanted to accurately measure its power draw in different modes and estimate total battery lifetime. We use a high sample rate Data Acquisition card from National Instruments USB-6210 to measure the total current draw across a sense resistor, and consequently calculate the power draw. The maximum current draw is 30mA during data transmission. The CC2530 supports multiple sleep states, the lowest of which consumes less than 0.045mA. All current measurements are at 3.6V. The average daily energy consumption depends on various factors that determine which power state the occupancy node is in. Assuming a periodic heartbeat of once every 15 minutes and over 100 occupancy events transmitted per day, we calculate the total energy drain to be 3.37mAh. Assuming standard alkaline batteries with a life of 2850mAh, this current draw translates to a lifetime of over 2 years. The low power draw of our occupancy sensor makes it perhaps possible to employ energy harvesting using indoor solar cells, and use super-capacitor based

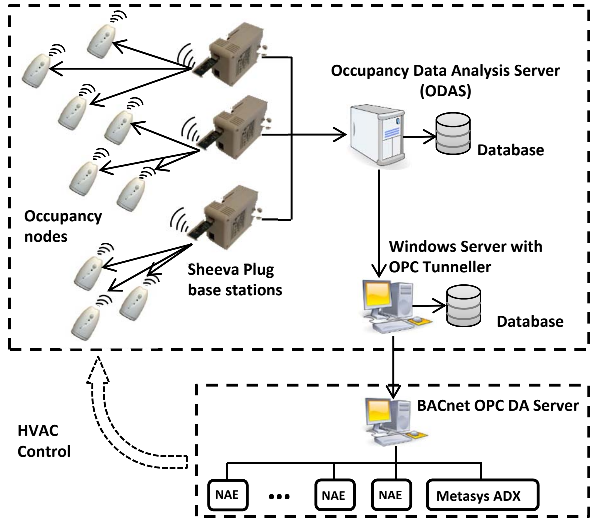


**Figure 3.4**: Implementation Diagram.

designs to makes these nodes almost perpetually powered[5, 31].

## 3.4 Development of Actuation System

The base stations send the wireless sensor data along with the status messages to the central server, which we call the Occupancy Data Analysis server (ODAS). The core component of the server software is the database that stores the information and a collection of Python programs that read from the database and perform actuations based on the data.

The ODAS is connected to a Windows server machine that runs an OPC tunneler. OPC is a common standard that allows for process control and communication between industrial devices. This machine is connected directly to two OPC Data Access servers managed by the facilities group on campus, with one providing real time energy usage data in our building, and the other providing access to the BACNet network. We developed OPC client applications to interface with the OPC servers and obtain the data points that we are interested in, which include temperatures for each thermal zone and energy usage of the HVAC system. In addition, we were given write access privileges for setting the occupancy command for each zone in the building.

Our ODAS runs a process that retrieves this data (zone temperatures) from the Windows server and stores it in the ODAS database. It is important to note that the temperature readings can be delayed for up to 10 minutes. To control the building HVAC, the ODAS sends a zone HVAC command (e.g. turn zone 2121 to unoccupied) back to the Windows server. The *actuator* OPC client application will scan incoming commands at a rate of once a second to see if a new one has arrived. If so, the actuator client will write the appropriate value for the OPC item to the BACNet OPC server. The BACNet OPC server has a higher priority than the static schedules, which allows it to override any previous command. (Higher priorities do exist, such as for emergencies). This set up provides us with an ideal test bed to experiment with different HVAC control algorithms revolving around using occupancy information along with other information sources. Figure 3.4 shows our overall system design with the individual components marked.

The ODAS database comprises of several tables for our HVAC control including ones that contain all of the rooms on each floor, the thermal zones, mappings between rooms and thermal zones, and temperature of each zone. Using this information, our HVAC control algorithms improve upon the static schedules set by the facilities management. Due to our location in a mild climate zone, typical days generally do not exceed a high of 76°F (except in a few weeks during the summer). Temperatures do not get very low either, with lows of the mid 50s. When we looked at the indoor temperatures for each zone during a normal warm weekend, we noticed that even with HVAC off, most of the rooms would not go above 75°F. The exception was one set of sparsely occupied offices on the wing facing the sun containing some IT equipment. These offices would climb up to 77°F, even on mild days, due to the solar effects and the heat-generating computers.

Our HVAC control system implementation comprises of several programs. The first program checks the occupancy of every zone and will turn-off (or put into stand-by) zones that are currently unoccupied. We rate-limit this to once a minute in order to prevent thrashing for the dampers. Damper power consumption is, however, quite low measuring around 100W over 20 seconds. In addition, we have a program running that will check the temperature for every occupied zone

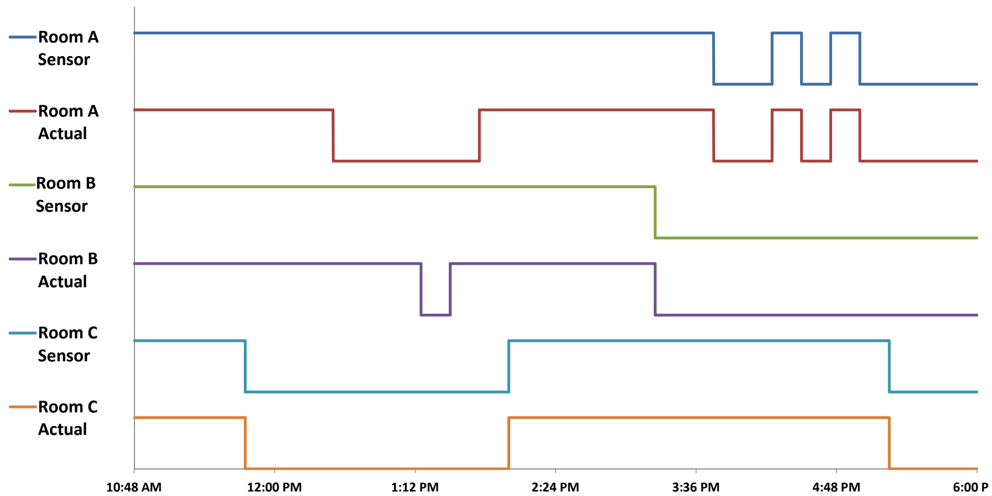


**Figure 3.5**: Accuracy test of three representative rooms over seven hours.

and will turn on the HVAC if the temperature goes over 76°F or under 66°F.

## 3.5 Evaluation

We now evaluate our test deployment for accuracy of occupancy detection, show how occupancy patterns vary across people, and demonstrate the potential energy savings for running our dynamic HVAC control scheme.

### 3.5.1 Occupancy - Accuracy and Patterns

For checking the statistical accuracy of our system, we compared the data given by the occupancy server to the actual occupancy in rooms. The ground truth measurements were done manually, by checking each office in the floor every fifteen minutes. The data was collected over a period of seven hours. Figure3.5 compares the ground truth and sensor measurements of three representative rooms. It should be noted that this does not cover all the cases discussed earlier. We can see that the sensor nodes capture the data fairly well. The inaccuracies occur because of the reasons discussed in Section 3.3. Statistically, of the 33 nodes we tested, 29 nodes showed an accuracy(with respect to time) of 96%. The four exceptional

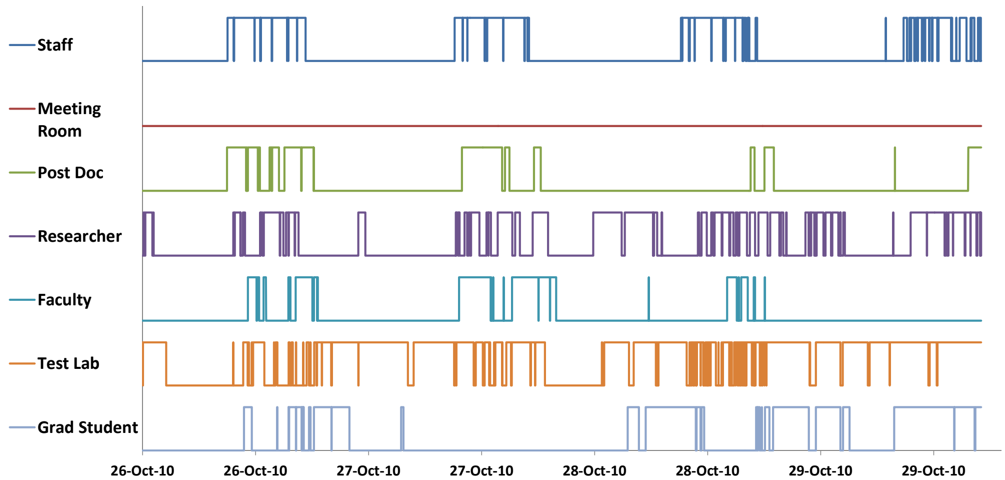


**Figure 3.6**: Occupancy for a representative set of seven occupants across four days. The data shows significant diversity in their occupancy patterns.

nodes had a high degree of inaccuracy because the sensors were placed too close to the door, causing false events to occur due to gusts of air during door events. We had calibrated the PIR sensors to be sensitive to even small movements in the room, and this resulted in no false negatives. Thus, a person is never detected as absent when he is present in the room, causing no discomfort because of sensor inaccuracies. However, we do waste energy when a vacant room is detected as occupied.

Our occupancy data reveals several interesting trends. It is important to note that as this is a building on a university campus, occupancy patterns are extremely dynamic when compared to a typical 9-5 office building. Figure 3.6 shows occupancy patterns of seven representative rooms over four days. We can observe that the staff worker has a fixed schedule from 8:30AM to 4:30PM every day. The ad hoc meeting room (that is often times not used) was empty during these 4 days. The faculty and post doc have sporadic occupancy patterns, mainly because of different commitments outside of their offices. The gaps in the occupancy of the graduate student perhaps indicate that he/she was attending classes.

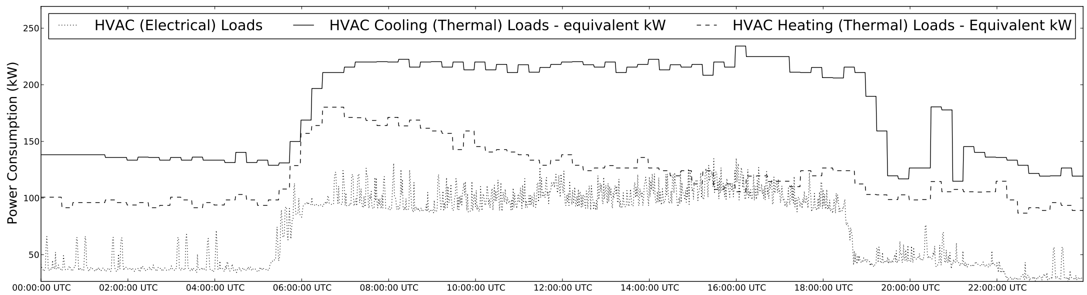


**Figure 3.7**: The energy consumption of HVAC during our baseline day. We show HVAC electrical loads as well as the HVAC thermal loads for both cooling and heating (as equivalent kW).

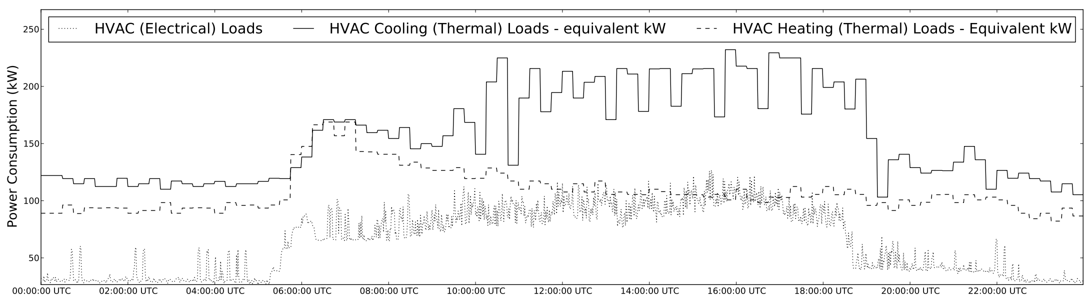


**Figure 3.8**: The energy consumption of HVAC during our first test day. The HVAC-electrical savings compared to baseline shown in Figure 3.7 are 11.59% while the HVAC-thermal savings are 12.41% and 9.59% for cooling and heating loads respectively.

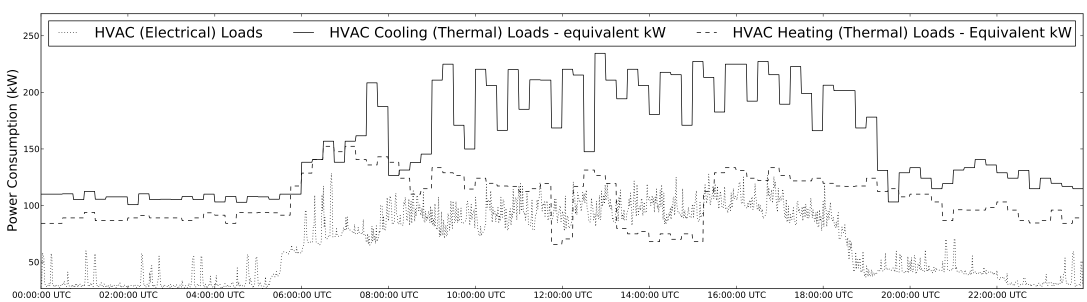


**Figure 3.9**: The energy consumption of HVAC during our second test day. The HVAC-electrical savings compared to baseline shown in Figure 3.7 are 9.54% while the HVAC-thermal savings are 12.85% and 11.51% for cooling and heating loads respectively.

### 3.5.2 Energy Savings

To study and evaluate how much energy we can save by duty-cycling the HVAC system, we controlled the thermal zones in the second floor of the CSE building for four different days, two each in fall and winter seasons. This was to study the effect across variations in the same season and across different seasons. The first set of experiments were conducted in the fall on October 27th(Wednesday) and 28th(Thursday), and the second set of experiments in winter were on February 23rd(Wednesday) and 24th(Thursday). In the first set of experiments, which we call the Fall Experiments, only half of the rooms on the second floor were covered by the occupancy nodes. These rooms were duty-cycled according to their occupancy, while the others were set on static schedules. During this set of experiments we were aggressive, and turned on the HVAC between 8.45AM and 10.00AM depending on the arrival time of the occupant. In the second set of experiments, we were able to cover 75% of the whole floor, the exceptions being unoccupied rooms and people who were uncomfortable with the sensors because of privacy concerns. For the latter rooms, we ran the HVAC on the standard static HVAC schedule from 5.45AM to 10.00PM.

Running the experiments over these two seasons also offered us the opportunity to determine how our system is affected by different outside temperatures. The fall season in San Diego is fairly warm, while the winter/spring season is fairly cool. In the former, the main mode of operation is in cooling the building, while in the latter it is in both warming and cooling the building. In both cases however, the mild temperature and modern building enclosure meant that even without HVAC, temperatures never got too extreme.

It should be noted that a strict comparison between two days is not possible, as the occupancy levels and environmental conditions vary from day to day. The measurements from two consecutive days gives us a fair idea of the energy savings across these variations to some extent.

**Fall 2010 Experiments**

Because the offices often had moderate indoor temperatures, we were extremely aggressive in reducing HVAC loads for our first test day. Of the rooms that we were unable to deploy our nodes, we either set the HVAC to on or off for the entire day, depending on whether the rooms were currently vacant (or vacant for the day). For rooms that were occupied, we set their occupied commands at a time closer to when the occupants arrived (instead of 5:45AM), usually between 8:45AM to 10:00AM depending on the room.

For the rooms that had our occupancy sensors installed and were in their own zone, we simply cycled the occupancy command based on actual room occupancy. However, larger zones that contained multiple rooms were more challenging to control, and here we opted to save energy. If the other occupants of the zone stated that they felt the cooling was too high, we simply put the HVAC to standby when one occupant was gone. Combined with the fact that cooling from other zones would seep in anyway, and the fact that the days rarely got hot, we believed that it would be enough to maintain comfort. For unoccupied rooms, we simply turned off the HVAC. After we ran our experiment on our first day, we changed our control procedures to opt for a more conservative approach on the second day.

The first test day was typical for the location, with mild temperatures and a high of 75°F. The second day was warmer, hitting a high of 82°F. Comparing energy consumption for HVAC across multiple days is difficult, as the exact weather patterns are difficult to reproduce, and HVAC loads are directly impacted by temperature and solar radiance. However, aside from this difference, a reasonable comparison across multiple days can still be made. Our test days were on October 27th (Wednesday) and October 28th (Thursday) of a typical work week, and we set as our baseline October 25th which falls on the Monday of that same week. This day has a fairly representative HVAC energy pattern for a mild day in our location, with a high of 73°F. We note that this baseline day was much cooler than our two test days, therefore our energy savings are somewhat conservative and would have been even higher had we compared with a similar warmer day. Figure 3.7 shows the HVAC energy trace of our baseline day, including both electrical consumption of the air handlers as well as the thermal loads for the building (given as equivalent kW). As mentioned earlier, equivalent kW is merely an approximation, and should not be taken as an exact conversion.

**Fall 2010 – Day 1 Results**

Figure 3.8 shows the HVAC power consumption traces for test day 1 (Oct 27). The graphs show the energy consumed by the entire building's HVAC system, not just the second floor (which we controlled). A normal static schedule day will start up the four floors in sequence starting at 5:15AM (for the fourth floor) until 6:00AM (for the first floor). A close look at the comparison day (Figure 3.7) from 5:15AM - 6:00AM shows the spikes that each floor causes. The effect on energy consumption is apparent, as the average power consumption hits past 100 kW. The rapid succession of open dampers causes the air handler to have to ramp up its fan speed, and this causes an even greater energy load.

For our test day 1 control scheme, we actually started our energy control scheme for floor 2 at 6:05AM since we wanted the normal initialization procedures to start up first. At 6:05AM our control commenced. Because it was before 6:30AM (the earliest time that we set for HVAC initialization), the system immediately set all the second floor zones to unoccupied. The energy savings this had were surprising. Rather than seeing energy consumed go up to an average of over 100kW, the energy consumed only went up to 80 kW, and settled down at an average 68kW for the morning. At 6:30AM our system went to "on" mode, which meant that it will only turn on the HVAC when the occupant arrived (for rooms that have the occupancy sensor), or at a later time that we set statically (typically 8:30AM to 9:30AM, based on when we observed the room usually becoming occupied). Starting at 8:00AM more rooms became occupied, and at 8:30AM the rest of the rooms (ones that we did not sense) were turned on.

The energy consumption rose as the day went on, but was still 5kW to 10kW less than the comparison day until the mid-afternoon (around 3PM). The effects from duty cycling (setting rooms to unoccupied mode when they were absent) had some effect, but the occupancy patterns for many rooms were quite static (long periods of occupancy). Even when multiple rooms were part of the same zone, our control scheme would opt to set the entire zone to unoccupied until it hit 75°F when we would turn the HVAC on. Many of these staff rooms however were not facing the sun in the afternoon, and therefore actually never went that high.

The late afternoon also showed a period of significant savings. Staff workers tended to leave from 4:30PM to 6PM. The normal schedule system puts all the floors into standby mode at 6:30PM, whereas our control scheme started labeling rooms as unoccupied when they left. The effects are significant, as due to our control the HVAC loads started dropping towards 80 kW at 5PM. In comparison, the static schedule averaged over 90 kW until 6:30PM.

The total HVAC electrical load for test day 1 was 1556 kW-H. The total HVAC electrical load for the baseline day was 1760 kW-H. Therefore, in terms of electricity, our HVAC control scheme saved a significant 11.59%, despite only controlling one floor in a four floor building. We also note that the thermal load consumption was less than the baseline as well, saving 12.41% in thermal cooling loads and 9.59% in thermal heating loads (results summarized in Table 3.1).

**Fall 2010 – Day 2 Results**

The first day we were very aggressive in cutting off HVAC cooling to as many rooms as possible. However, given that we did not actually detect occupancy in half of the rooms, setting them as completely unoccupied could have potentially unintended consequences in terms of higher temperatures if an occupant did happen to come into that room. Therefore, for day 2, we ran a less aggressive cooling control pattern where we would actively monitor the temperature and turn on cooling whenever the temperature would rise past 75°F, regardless of whether or not we had a sensor node in the room or not.

Another change was to start the day off with full control, as opposed to letting the normal static procedure initiate. This meant that the entire 2nd floor would not be set to occupied at 5:45AM, and instead be turned on at 6:30AM. The effects of this were immediate, as the average power consumption hovered near 80 kW for most of the morning. As the second floor started to become occupied, the

**Table 3.1**: Fall Tests - Energy consumption for electricity and thermal cooling and heating (as equivalent kW).

| **Day** | **Electricity** | **Cooling** | **Heating** |
|---|---|---|---|
| Baseline | 1760 kW-H | 4302 kW-H | 2877 kW-H |
| Day 1 | 1556 kW-H | 3768 kW-H | 2601 kW-H |
| Day 1 Savings | 11.59% | 12.41% | 9.59% |
| Day 2 | 1592 kW-H | 3749 kW-H | 2546 kW-H |
| Day 2 Savings | 9.54% | 12.85% | 11.51% |

power rose to an average of 100 kW, not entirely dissimilar to our baseline day. However, looking closely, we observe that the average power consumption of test day 2 was still slightly lower than the baseline day. Similar to test day 1, energy consumption started falling rapidly as the work day ended.

The total energy consumed for test day 2 was 1591.68 kW-H. Test day 2 was also much warmer, resulting in higher than normal HVAC loads, but our conservative approach likely added additional energy consumption over test day 1. Compared to our baseline day, test day 2 saved 10.5% in electricity. We observed 12.85% savings in thermal cooling loads, with the savings mostly concentrated in the morning, and 11.59% in heating loads. Table 3.1 shows the results for test day 1 and test day 2 compared to our baseline day.

### Additional Observations for Fall 2010 Tests

Looking at the fall results, it is clear that a significant source of the energy savings comes from starting the HVAC when the users arrive rather than as early as 5:45AM. A side benefit of this is that the load on the air handlers is more staggered. Since our building is located in a mild climate zone, it is not necessary to aggressively pre-cool, however in other climates, we would opt for a learning algorithm to predict when users arrive and initiate cooling accordingly. Since occupancy patterns for a given individual tend to be similar, this is likely to be an effective strategy.

We also note that occupancy patterns tend to disallow a great deal of online duty cycling, as people tend to be in their offices for long periods of times. This

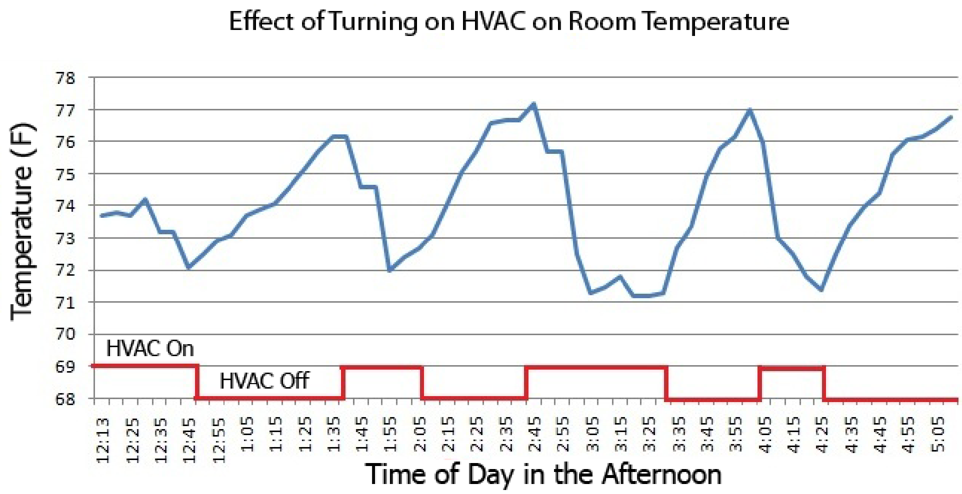


**Figure 3.10**: Effect of actuating HVAC for an IT-heavy room facing towards the sun during a warm day.

is especially true for the graduate laboratories, which always had some occupants throughout the day. The effects on air flow when one zone is being cooled and another is also significant and these thermal effects may have a non-trivial impact on overall energy consumption.

It is important to factor in the effect on temperatures through the zones. One interesting observation was that the location of the room significantly affects how warm it will be. As mentioned previously, one side of the wing of our building faces the sun during the afternoon, and thus gets much warmer than the other rooms. We noticed that the majority of the other rooms would stay constantly under 75°F even with the absence of cooling, but these rooms would rise to 77°F during the afternoons. We ran an experiment to control the HVAC in one of these unoccupied room with several computers. Over the course of a warm day, we enabled and disabled HVAC to see how temperature rises and falls, and importantly, how quickly the temperature adjusts. The temperature readings are read from the BACnet OPC DA server and thus exhibit some amount of delay and discrete jumps.

Figure 3.10 shows how this particular room reacted to the turning off and on of the HVAC over the day. The HVAC system was able to send a significant amount of cold air into the room to rapidly cool it to acceptable temperatures. We note that it takes less than ten minutes for the HVAC system to cool the room below 75°F. This suggests that allowing temperatures to float will have minimal impact on comfort levels; the effect of this scheme on energy consumption however still needs to be studied further. We do note that the cold air in the ducts is at 55°F normally, and is reheated to maintain temperatures in each zone. When a zone is warm, the reheat required is reduced. This suggests that cooling a warmer room might not significantly impact HVAC energy consumption more than maintaining a room at its setpoint.

**Spring 2011 Experiments**

We ran our HVAC control experiment again in spring 2011, and were able to achieve a much larger deployment of occupancy nodes. We also adjusted our test parameters based on our discoveries from the first set. Since we had a much better deployment, we were able to monitor most of the rooms, including some unoccupied rooms, and actuate based on that. We adjusted the timing and opted for a later start time of 8:45AM to turn on the common areas and unmonitored offices, as this was when most people came in. Rather than turn off HVAC as in our first set of tests, we opted to put unoccupied zones to stand-by instead to maintain some airflow. We also were more conservative with placing zones to unoccupied for this set of tests as we had more monitors - if a single room in a zone was occupied, we turned on the HVAC.

For this test, our baseline day was February 17th (Thursday) of a typical work week. It was mildly cloudy with temperatures ranging from 53°F to 62°F. Our test days were Feb. 23 and 24, days with similar weather to the baseline day. The temperature on Feb. 23 varied between 54°F to 63°F, with the temperatures not varying by more than 3°F between the two days. The weather on Feb. 24 was generally cloudy, with temperature varying from 51 deg F to 60 deg F. The weather was much colder than the fall tests, meaning we would be able to test how our system handles conditions where it must sometimes warm rather than cool. From observing the data over the previous weekends, we noticed that the temperatures in the building would typically range from 65°F to 73°F when the HVAC system was completely off.

**Spring 2011 Results**

Due to lack of space, we were unable to include the spring 2011 graphs, but the summary of our spring results is listed in Table 3.2. Comparing the HVAC electrical load on the test day with that on our baseline day, we see that the load increases more gradually during our experiment days. The load at 6AM on the test day is 70kW compared to 105kW. The load increases slowly to 105kW at about 8:30AM, while it remained constant during this period on the base day. The peak load on the test day was 165kW compared to 180kW on the baseline. The power consumption starts decreasing at 3:30PM compared to 4:30PM on the baseline. It remains at 120kW until 6PM compared to 140kW, mapping closer to the occupancy levels in the second floor. The total energy consumed by the HVAC electrical on the test day was 1977 kWh compared to 2187kWh on the base day. The energy saved during the period was 9.60%, with similar savings in thermal-cooling. Thermal heating energy consumption was almost the same however.

The second day had similar temperatures but was more overcast. The HVAC electrical consumption stabilized around 75W at 6AM, and stayed there until 8:15AM. The power consumption then started to increase gradually until it reached a peak of 145kW at 3:30PM. The total HVAC electrical power consumption for the day was 1843 kWh. We saved 15.73% compared to the baseline day. This day was cooler however, which meant that reducing the HVAC would save more in energy because of less heating.

As our original system was designed for warm days, running it on a colder day proved enlightening. Temperatures in the 60s is about as cold as it gets in San Diego, and thus our building was forced to warm the rooms. We did notice though that rooms got as low as 66°F in the morning. Rooms where HVAC was turned on were warmer at about 70°F-72°F, while rooms that had HVAC turned off were cooler at about 67°F-69°F in the late mornings. By the afternoons, most

**Table 3.2**: Spring 2011 Tests - Energy consumption for electricity and thermal cooling and heating (as equivalent kW).

| Day | Electricity | Cooling | Heating |
|---|---|---|---|
| Baseline | 2187 kW-H | 3137 kW-H | 2124 kW-H |
| Day 1 | 1977 kW-H | 2885 kW-H | 2128 kW-H |
| Day 1 Savings | 9.60% | 8.03% | -0.18% |
| Day 2 | 1843 kW-H | 2899 kW-H | 2021 kW-H |
| Day 2 Savings | 15.73% | 7.59% | 4.85% |

of the rooms were above the heating setpoint and some were even being cooled. In retrospect, we perhaps needed to better optimize heating strategies, as pre-heating is perhaps more important than pre-cooling for these situations. It would also have been instructive to take into consideration warm air spreading from warmer zones to non-warm zones. We did not however receive any complaints about the temperatures; the people we asked did not even notice any change.

## 3.6 Future Work

A major criticism of our work is that we do not predict the arrival of a user using his/her past patterns and preheat the room proactively. We have shown that it takes less than 10 minutes to cool the room below 75°F, and keeping the room warm may actually help in reducing the amount of reheat required by the VAV in Section 3.5.2. However, this solution may not be amenable in a harsher climate, like in Chicago or Houston, and an adaptive prediction method needs to be in place so as not to disturb occupant's comfort. Erickson et al [11] have done a detailed simulation study of effect of ventilation in different environmental conditions and how much occupancy information can help us save energy using predictive strategies. To make our system deployable at other places in the country, we plan to incorporate such machine learning mechanisms in the future.

Another important characteristic of our system is that our occupancy sensors only provide binary occupancy information. An ideal occupancy system should be able to tell how many people currently occupy the room, or atleast have a measure of the number of people in coarse granularity. Erickson et al[11] do a detailed study on the differences between the systems which have binary information and those which have a measure of the number of people in the room. In our HVAC system at CSE, we let the PID based control system to handle the amount of air to let in to the room. Thus, the amount of ventilation gets automatically adjusted. However, such a system may cause discomfort in the room if the number of people in the room changes drastically in a short amount of time. We have only one classroom in CSE that fits this pattern, and can add exception for the same. A building consisting of large number of big rooms, however, needs to have a people counting mechanism in place. We are also working on appropriate solutions for the problem.

## 3.7 Acknowledgements

I would like to thank Prof. Yuvraj Agarwal and Prof. Rajesh Gupta for giving me the opportunity to work on this project. I would also like to thank my colleagues and co-authors Thomas Weng, Seemanta Dutta, Michael Wei and Jacob Lyles for their contributions and support throughout the project. I am grateful to them for permitting me to use the following work towards my Thesis:

- Yuvraj Agarwal, Bharathan Balaji, Seemanta Dutta, Rajesk K. Gupta, Thomas Weng - "Duty-Cycling Buildings Aggressively: The Next Frontier in HVAC Control" *In Proceedings of the 10th Conference on Information Processing in Sensor Networks: Sensor Platforms, Tools and Design Methods (IPSN/SPOTS '11)*, April 2011.

- Yuvraj Agarwal, Bharathan Balaji, Rajesh Gupta, Jacob Lyles, Michael Wei, Thomas Weng - "Occupancy-Driven Energy Management for Smart Building Automation" *In Proceedings of the ACM Workshop on Embedded Sensing Systems for Energy-Efficiency in Buildings (BuildSys '10)*, November 2010.

# Chapter 4

# Plug Load Management

Plug loads of a building constitute another major chunk of the total building energy consumption. A look at the Energy Dashboard tells us that they constitute between 29% to 36% of the total electricity consumption of the CSE building on an average day. If we remove the contribution from the constant base load provided by the machine room (which is the case in a non-IT building), the energy demands of the plug loads become even more prominent. Figure 4.1 illustrates this by deleting the contribution from the servers. The plug-loads account for 41% to 67% (average 55%) of the total power consumption, over a two week period in January 2011.

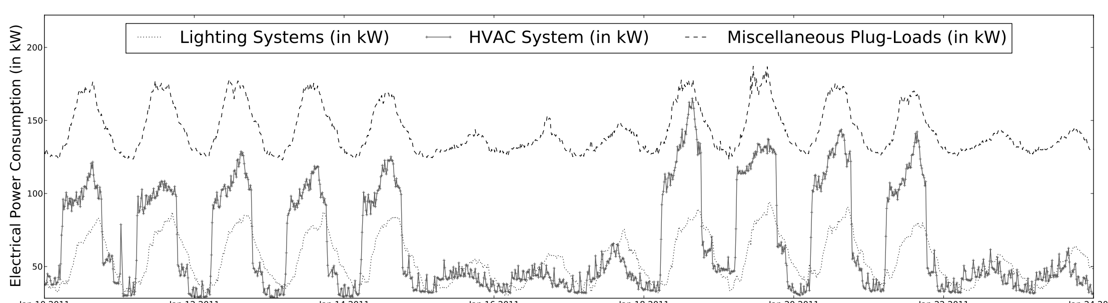


**Figure 4.1**: Electrical power usage breakdown for a typical building over two weeks in January 2011. The building has a small server cluster which is metered separately and is not shown in this graph.

Further, we can see from the Figure 4.1 that the baseline consumption of the plugloads is fairly high at 61% of the total plug load power. This indicates that majority of the machines remain on even during the night and weekends, showing that potentially we can save a lot of energy if we manage our loads effectively. Managing plug loads is a difficult problem, as there are many types of loads involved, distributed across the building and controlled by different owners. Compare this with the HVAC system, which is a single system for the whole building and centrally managed by the building managers. As the building managers can see only the overall plug load consumption, they can do little to control, or even account for, this portion of the energy consumption to the building. Moreover, individual occupants have little feedback on how much they are contributing as an individual to the total system power, and thus, have only a vague idea of the impact of using energy saving solutions. The research community has identified this need as early as 1992, and has come up with solutions to attribute energy to the individual loads[15]. This technique, called Non-Intrusive Load Monitoring(NILM) uses a single energy meter for the entire building, or a subsystem, and applies signal processing on the acquired signals to deduce the power consumed by each appliance. On the other hand, energy meters can be designed for each plug load, and deployed across the building to measure the energy consumed. Several research efforts have concentrated on this solution as well[18, 19, 21, 25]. The main argument against the plug load energy meters has been that it is expensive to deploy for all the loads in the building. Thus, we have developed our own Smart Energy Meters(SEM) which are designed to be inexpensive (BoM of less than 15$), with several improvements over previous designs.

Measuring alone, however, only allows us to observe the power consumption patterns of an individual load. The best the user can do to actually save energy is to switch to more energy efficient appliances, or even change his/her behavior to manage their devices more efficiently. This involves considerable effort from the part of the user - from putting in financial effort to changing his behavior patterns. To reduce the discomfort caused to the user, and to manage energy consumption more effectively, we need automated solutions. This will lead to wider adoptability of the solution. To achieve this, actuation of plug loads becomes a critical component of the system. However, actuation of plug loads can be difficult as every appliance has their own characteristic, and may need to be shut down gracefully. A classic example is the desktop computer, which cannot be turned off by cutting off the power as it can lead to data loss and/or can bring the system to an inconsistent state. Thus, we have integrated a relay in our SEM to actuate devices and developed a signature detection mechanism, which allows us to identify the class of device plugged in. This allows us to identify the loads which can be switched off without any repercussions, and build automated solutions on top of the energy meter.

Using the SEM nodes, we have built a comprehensive centralized system, called the "Energy Auditor" for energy analysis and management. Each of the SEM nodes are capable of wirelessly sending the data using the ZigBee protocol, and we leverage the backend system we built for our occupancy based HVAC system to collect the data. The Energy Auditor engine then analyses this data and provides a management framework for both the building manager and the end user. The Energy Auditor has a storage component to store the energy data, a visualization component to display personal and aggregate energy use to occupants and building administrators, and various analysis engines to compare energy use across time and across spaces. For individual building occupants the Energy Auditor provides actionable items on how they can be more energy efficient. On a larger scale it provides powerful knobs for energy managers to control the overall building energy consumption. We have implemented the Energy Auditor to be extensible with multiple management modules to control the energy consumption of plug loads. These management modules can represent an expressive set of policies for energy management such as reducing energy waste (turn off things that are not needed in unoccupied offices), unauthorized load detection (space heaters) and deal with demand response events (turn off non-essential loads in cases of emergencies).

## 4.1 Background

At the center of every energy accounting solution is the technology to measure the power consumed by individual appliances in the building. A central approach to study the power consumption of the plug loads in a building is Non-Intrusive Load Monitoring(NILM) [15]. A high precision energy meter is placed at the main circuit of the building, and the power values are measured at a very fast sampling rate. Signal processing is done on the acquired power values to distinguish between the power consumed by the various loads in the building. The basic concept behind the idea is that as different devices are turned on at different times, their characteristic waveforms can be distinguished from the overall power consumption waveform during their startup sequence. The major drawbacks of this system is that it requires training for every device and thus becomes difficult to distinguish between appliances when multiple switching takes place and when there are many devices of similar type (for example, 30 computers in a lab). To overcome the disadvantages, researchers have come up with additional solutions which will help augment the NILM solution. Rowe et al[32] senses the electrical and magnetic flowing through the power lines of appliances to detect their state, and augment this information with the central meter. At the Flick of a switch and ElectriSense[14, 29] sense the EMI interference caused in the powerlines of the circuit whenever a device changes state. Recently, Jung et al. [20] have investigated the use of binary state sensing (on/off) of different loads combined with an incremental deployment of energy meters to perform energy disaggregation. All the above methods make the NILM methodology more accurate but still require a training phase. Furthermore, if we need to actuate appliances, we still need to have controllable switches for each of the appliances.

A more distributed and direct approach is using energy meters that are placed inline between the power cord and the power supply of the appliance. This methodology gives more accurate values and the actuation facility can be built into the meter. Academic energy meters include the ‘Plug’ from MIT[25] and ACme from UC Berkeley[18]. The Plug is a powerstrip model with additional sensors to measure the environmental context like light and sound. The project concentrates on correlating the sensor signals with the power values and device state. ACme is a plug load meter designed to scale in a building environment. It uses 6loWPAN, an IPv6 based sensor network protocol, on top of a TinyOS platform. The major drawbacks to these solutions are that the individual meters are too expensive to deploy on a wide scale and do not have reliable actuation capability. Commercial solutions include the Kill-a-Watt, WattsUp[38], Current Cost[6], the Energy Detective[36] and many others. The typical device comes with an LCD screen showing the real-time power consumption. More modern devices come with proprietary wireless solutions meant to scale for home buildings. We have not seen a commercial solution which scales to a commercial building with actuation capability.

Using these measured energy values forms another critical part of the system. Several visualization solutions have come up in the recent past to provide feedback to the user on their power consumption - Google Power Meter[13], Microsoft Hohm[27], the GreenSoda project using the ACme meters[18, 19], PowerNet from Stanford[21] and our Energy Dashboard for UCSD. Most of the above visualization tools export a standard API so that they can pull the data from any energy meter which can deliver the data in the format required[28, 7]. Hay et al[16] provide a case for why personal apportionment is necessary to save energy in a commercial environment. To identify the users automatically and to apportion the energy usage to them, researchers have developed identification of different users in the building using their height differences[33] and analysing their network activity[22]. Previous work has also tried to make certain subsystems more energy efficient. Delaney et al[8] show that 50% to 70% of energy savings can be accrued by using Wireless Sensor Networks to control the lighting system. Somniloquy and SleepServer[1, 2] from our lab have addressed the power consumption of IT loads.

## 4.2 Smart Energy Meter

At the center of our Energy Auditor is our Smart Energy meter (SEM). While there are already several commercial[38] and research[18, 19, 21] plug-load meter designs, we wanted to develop our own design with several unique characteristics. First, we wanted to make our SEM specifically low-cost and target a $15 bill of material. Second, we wanted the energy meter to utilize Zigbee since it is an industry standard protocol and interfaces with our Occupancy based HVAC system. We describe the hardware and software components of our meter and focus on how it contains the mechanisms that help facilitate the additional functionality. Low-cost and wireless communications are essential because the primary limitations against using individual plug-load meters are the cost of the meters and the deployment expense. Next, we wanted to make actuation capabilities a key component of our SEM such that electrical loads could be powered on/off. To make the actuation capabilities even more effective, our SEM has detection and classification capabilities to determine the load *type*– for example whether the load is a lamp load, a fan load, a Desktop PC, or load that is battery powered such as a laptop or a cell phone charger. By design since our SEM is single outlet which can be independently measured and actuated we do not have to deal with complex load disambiguation where multiple loads are connected to a central point of energy measurement [32, 15, 14, 29]. The SEM also stores a load *class* attribute, similar to priority levels, which is useful for policy based actuation such as demand-response – for example a policy that turns off all non-essential loads during an emergency. While previous energy meters have proposed actuation capabilities they have been utilized to a limited extent[38, 18], we show how actuation can be used in a variety of smart building scenarios. In terms of communication, the SEM conserves network bandwidth by employing multiple levels of compression. It can either average data and send periodic values, or send data only when it changes based on a set threshold, or can send data at every second for real time access. Finally, our SEM is extensible with local processing capabilities for any further enhancements.

### 4.2.1 Hardware Design

The SEM node comprises of essentially five components: voltage and current sensing circuitry, energy measurement and accounting unit, power supply unit, wireless radio, and a relay for switching the load on or off. The overall logical layout is shown in Figure 4.2. At a high level, the voltage and current sense circuitry is responsible for converting the line level voltages and current measurements to appropriate levels for sampling by the energy metering IC. The energy metering

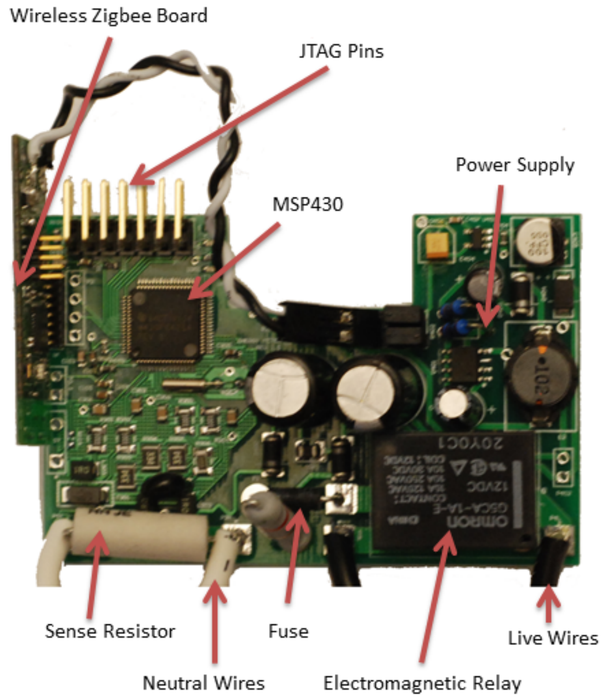


**Figure 4.2**: Picture of our energy meter with various components marked.

IC then calculates various parameters, such as power and power factor and also maintains averages over time. These average values are then transmitted to a base station over a Zigbee wireless radio. The mechanical relay is also connected to the energy metering IC and can actuate the electrical load plugged in to the energy meter.

Each of these components however needed to be designed carefully based on experimentation with different alternatives and several important considerations. Overall, we designed our SEM to be inexpensive and compact thus using the least amount of space possible while addressing safety considerations. For the current and voltage sense component we used a simple sense resistor (4 mohm/4W) based design since it is cost effective, occupies a small footprint on the PCB and can accurately help measure real and reactive power. ACme [18] provides a good comparison of various design alternatives for current and voltage sensing, with advantages and drawbacks listed. We chose to connect the sensor resistor to the neutral line as opposed to the live wire[18] for safety considerations – we tie the ground to the neutral line thereby preventing the entire ground plane from floating at a 120VAC level. The drawback of using neutral side metering is that we were not able to use the additional tamper detection functionality available in our energy metering IC. We use a 1000:1.5 voltage divider, after spike suppression circuitry, and feed that to the voltage measurement pins of our energy metering IC.

We use the MSP43042x2 as our energy metering IC (Figure 4.2). This chip from Texas Instruments provides an Analog Front End (AFE) which samples the voltage and current signals using a high-precision 16 bit ADC at 4000 samples per second. This IC internally multiplies and processes these samples to provide RMS voltage, RMS current, active power, reactive power, power factor and energy consumed into designated registers periodically. The IC also contains an MSP430 core, which is a 16 bit RISC processor with up to 32KB of flash and 1KB RAM. The MSP430 also provides an USART interface and a number of GPIO pins. An alternative was to use a dedicated analog front end IC, like the ADE7753[18, 21]. However, we chose the MSP since the additional processing capability allowed us do all the required processing in a single IC and get rid of a dedicated microcontroller for this purpose, saving us both cost and space. The USART of this IC is connected to a CC2530 radio. One of the GPIOs of the MSP430 is used for switching the load using a mechanical relay.

We experimented with different solutions for switching electrical loads. Our earlier design used a compact solid state relay (SSR) similar to the ACme design. SSR solutions dissipate a large amount of heat, which was around 15W at a large 1.5kW load thus requiring a careful design of a large heat sink. To save cost and also improve safety, our current design uses an electromagnetic relay (G5CA from Omron) instead (Figure 4.2). The part is UL recognized for switching 15/125VAC loads, general purpose usage, and consumes only 200mW at peak load. We further isolated the electromagnetic relay from the MSP430 using an opto-isolator relay (Clare CPC1017N).

As shown in Figure 4.2 we employed the Texas Instrument CC2530 as our Zigbee wireless radio. The CC2530 is a System-on-Chip solution providing both a 8051 microcontroller core and a 802.15.4 radio in a single package. We designed a small daughter board which connects using a 8 pin header onto the energy metering board. This board serves as an end device in our Zigbee network, and connects to the nearest base station available. The CC2530 communicates with the MSP430 IC using a serial port. The network structure is the same as that used by the Occupancy nodes in our HVAC control system.

The final component of the SEM is the power supply unit. We wanted the PSU to be cheap, have a small footprint on the PCB and also meet our maximum power budget of around 500mW. Our SEM design requires two voltage supplies — a +12V input for the mechanical relay and a +3.3V DC supply for the MSP430 and the CC2530 chips. While transformer based PSU designs provide isolation, we chose not to use them since transformers are expensive and bulky, thus occupying a lot of space. Another option is to use a direct rectification capacitive power supply[18], although such a PSU is usually inefficient and can supply limited current up to 40mA-50mA. Our final PSU design is based on direct-rectification with a buck boost converter IC - LNK304, providing up to 120mA at +12VDC (Figure 4.2). We use another LDO regulator to drop the +12V down to +3.3V levels. Our complete PSU design has a small PCB footprint and costs less than USD $3 (in quantities of a 1000).

Finally, we needed a safe and ergonomic casing for our SEM. To reduce the design and prototyping time, we re-purposed Leviton's duplex plug-in surge suppressor 5100-P as shown in Figure 4.3(a) and 4.3(b). We removed the surge protector components from the casing, keeping only the necessary plug contacts. The casing then provided sufficient empty space and after a careful design we were able to fit our SEM PCB and the CC2530 wireless module into it. The advantage of using the surge protector is that the casing is quite sturdy, provides ventilation and is also UL safety certified.

The bill of material (BoM) for our complete Smart Energy Meter is less than $15 (in quantities of 1000). Note that this cost does not include the PCB manufacturing and component stuffing, which we estimate at another $5 or even lower in larger quantities.

### 4.2.2 Software Design and API

The software on the MSP430 IC controls the core functionality of the energy meter. The MSP430 variant (MSP43042x2) that we use contains an ESP subprocessor that automatically calculates the energy-related data, such as power, voltage, current, reactive power, and many other power parameters. Our software comprises of three basic tasks - the energy metering task, the command task, and the serial task. The energy metering task handles retrieving values from the ESP subprocessor, doing load classification and calculating the final outputs (such as averaging or thresholding) that will be sent to the Energy Auditor. The serial task handles serial communications with the CC2350 radio. The command task controls the operation of the meter by specifying what sending mode the energy meter needs to operate as. These modes are set by commands that come from the Energy Auditor through the wireless network. There is a fourth task that monitors the temperature, and will shut down the connected load when the temperature goes above safe values.

The CC2350 runs the wireless Zigbee stack that forms our network. It's role in the energy meter is simply to handle the wireless network and act as a data pipe for the connected MSP430. Any data packets addressed to the EM automatically get sent to the MSP430 using the serial interface.

In order to support a wide variety of policies and functionality, the SEM has several parameters that can be set which control how it operates. The SEM supports different send modes, which allow the building manager to control the rate at which data messages are sent. We list the settings that determine how the EM handles demand response events and policies, all of which can be set from the Energy Auditor:

- Constant Send – Sends energy data every second.
- Threshold Send – Sends new data only when new data differs by a specified wattage. Default is .5W. The threshold value can be changed via a command.
- Average Send – Sends averaged data over specified time period. Default is 30 secs. The averaging can be changed via a command.

To support the demand response action policies, the SEM has three parameters that set when the connected device should be powered on or off.

- Day Time Priority Level – Priority level during working hours. Monitors for example might have a higher priority during day time hours.
- Night Time Priority Level – Priority level during night hours. Lights for example might have a higher priority during night time hours.
- Device Type – The type of device, which currently can be unknown, lamp, desktop computer, monitor, fan, laptop computer.

We implemented a signature detection algorithm based on simple heuristics to classify the type of load on the SEM which works as follows. In the training phase we observed the power, voltage, current and power factor of different types of loads. The power factor of the load gives the basic type of a load - resistive, inductive or capacitive. The variation in power, voltage and current values also gives an idea about the type of load. For example, a lamp will have a power factor of 1.0 with constant power draw, whereas a table fan is an inductive load, with varying power consumption depending on its speed. We have tried to classify the various types of loads seen in a typical office environment - desktops, monitors, laptops, speakers, table lamps, etc. The detection gets challenging when two devices classes have similar patterns, or when one device class has varying patterns depending upon the model and manufacturer. We observed that laptops and LCD screens, for example, have a power factor close to 0.65 and consume power in the range of 20W to 60W. We distinguished between the two with variations in the power consumption of laptop with time, and the difference in startup power draw of both devices. The detection also gets challenging with devices which have power factor correction, and many different internal states, in which case more complicated NILM approaches must be utilized[14, 15, 29, 32]. However, in Section 4.4 we show that our simple heuristic is 90% accurate in classifying common building loads and in case of incorrect detection we can always fall back to user input to specify the load type.

(a) (b) (c) (d)

**Figure 4.3**: Picture of our energy meter (a, b) along with our SheevaPlug base station (c) that is deployed in the hallways. The CC2530 based wireless module that are in both the base station and the energy meters is also shown (d).

### 4.2.3 Wireless Network

Our wireless network uses Zigbee since it is a popular standard for smart building technologies. Our network consists of our smart energy meters and base stations, both of which utilize our CC2530 based wireless module. Our base stations consist of our wireless module connected to plug computers, which are inexpensive small form factor Linux based computers. One such plug computer is the SheevaPlug which contains a 1.2Ghz ARM processor and 512MB of memory and flash memory storage. These plug computers are connected to a local Ethernet port, and communicate with our main Energy Auditor server. Figure 4.3 (c) shows a picture of our basestation along with the wireless module. We reuse the same network structure as our Occupancy based HVAC system.

One key differentiating aspects of our current deployment is that of actuation, and sending data back to the end devices. To facilitate this, we maintain a mapping of meter IDs with their Zigbee assigned 16-bit network address. When an SEM sends a data message to its parent basestation, the basestation learns and records the ID associated with that network address. It will then send a periodic mapping message to the EA server, which maintains this mapping in a database table. Whenever the EA needs to send a command message to a specific energy meter, it looks up the network address and basestation that is associated with the meter ID. It then sends the data command to that basestation, which transmits it to the desired SEM. Because of the importance for these command messages, we employ an application level acknowledgment to ensure that messages are delivered. If an acknowledgment is not received, the messages get re-sent. Zigbee operates on the 2.4Ghz ISM band and can potentially suffer from interference from other devices such as WiFi devices, which means that dropped packets are an unavoidable reality[24].

A SEM can send several different types of data messages to the server. The most common are the energy report messages which contain parameters such as watts, volts, amps, power factor, and reactive power. The actual message type depends on the send mode. Another data message is a status report, which contains all the current parameters the EM has. The messages from the server back to the EM are command messages that set the different parameters and settings on the EM, such as send mode, priority level, etc. The server can also request the status of the EM as well as actuate it.

## 4.3 Analysis and Actuation Server

The main component of our system is the Energy Auditor engine which is responsible for the data storage, data analysis, and management modules for actuating the individual smart energy meters. The EA engine resides on a server computer, and from a conceptual view it consists of modules that provide functionality to the system. These modules perform an operation, such as checking energy waste by comparing energy usage against occupancy, and can be upgraded or extended by system programmers as needed. New modules can also be added as desired. Users do not interact directly with the modules normally however, but rather through interfaces, which provide access to the underlying modules. On top of interfaces, users can set policies and actions that give them control over the energy devices in the system. Policies are automatic commands that the users can set in order to actuate when certain conditions are met, such as turning off devices when a room becomes unoccupied. These give users the ability to control devices without have to manually actuate them. Actions are similar, but involve commands that are paired with objectives. We first describe the modules that our system currently provides, starting with the core modules of the Energy Auditor:

- **Data Server** – this module acts as a server and receives all of the data from the basestations and stores it into the storage module. It can also examine the incoming data to raise events.

- **Storage** – this module is responsible for storing the energy meter values (e.g. power consumption, voltage, current, etc.) and occupancy data for each room.

- **System Settings and Parameters** – records system information, such as the meters that are deployed, the users in our system, the basestation information, etc.

- **Visualization** – displays the data for users (via webpages), and provides several options to view the data. It also provides the ability to compare energy meters.

- **Authentication** – authenticates access to the Energy Auditor, including both end users, who have a restricted view of their meters, and the building manager, who has admin access and complete control over the system.

- **Actuation** – provides the core ability to remotely turn on and off connected devices.

Policy/Action management modules are listed next. These provide the mechanisms to support policies, which typically involve an action along with a condition. Thus users can specify that when a certain condition is met, that the following action should be taken. Actions on the other hand attempt to fulfill an objective, and are how demand response (DR) events are handled in our system. Both end users and building managers can set policies, but only building managers typically will set actions.

We designed these actions to handle DR because this gives the control to building managers, who are the ones that must make decisions when emergency events happen. These actions can be quickly setand allow managers to limit the impact that reducing energy consumption will have on the building occupants. A series of several DR modules have been created, each giving managers a different way of specifying a load-shedding action.

- **Occupancy-Based Policy** – allows users to set a policy for actuating devices based on occupancy, such as turn off a device when room is currently unoccupied.
- **DR Action (Priority Level)** – will shut down devices according to their assigned priority levels. All devices have a priority level that the building manager sets.
- **DR Action (Device Type)** – will shut down devices according to their device type. Managers can set an action to shut down all laptops (which have batteries) for example in order to reduce energy usage.

The following are the web-based interface modules, which provide the methods for end users and building managers to interact with our system:

- **MyDashboard** – the main interface that end users interact with. Upon logging in, users are able to access their meters, visualize the energy traces, and utilize the above mentioned modules, such as the energy waste module.
- **Energy Auditor Administration** – this is the administrative interface that building managers will rely on. Managers can set policies through here and also enable actions when required. Managers also have very fine-grained control over the system wide settings and can determine access levels for users, set parameters for devices, and maintain the overall mapping of devices.

While modules can be added and extended as desired, end users and building managers will typically only directly deal with the interface modules, and set policies and actions through this interface.

Our Energy Auditor (EA) is implemented on a commodity DELL PowerEdge 2950 server machine. The core of the system is a database (currently a MySQL database), which maintains all the data collected from the different modules. A large collection of tables store the data that make up our system, which include not only data from the SEM, but also meta data such as the energy meters parameters (e.g. what their current data type is and who their owners are) and the list of users in the system. The current implementation of our system uses both Python and PHP scripts that interact with the database, with each module typically consisting of a script program (and associated classes and libraries).

The data server module consists of a Python script on the EA server that acts as a network server for all of the basestations. The basestations will send all of the energy and occupancy data to the data server module. Upon reception of the incoming data, the data server module will examine and parse the packets, raise events if necessary, and store the data in the storage module (databases). Basestations also are able to receive commands from this module by requesting all commands stored in the commands database table that are meant for it.

The energy values collected from various SEM meters are essential for the analysis, visualization and actuation modules. One technique that we implemented to allow fast visualizations across different time spans is the use of materialized views. Energy data comes in at a rate of typically once a second, and attempting to view this data over long time scales (such as over a year) will result in far too much data to process in a graph. To handle this, we have four tables to store the energy meter data at different resolutions, such as once a second, averaged at once a minute, once every 15 minutes, once every hour, and once every six hours. We run a program every minute that performs this averaging and stores the data in these materialized view tables. While this means that our storage needs are increased, given that storage is extremely inexpensive, we feel this tradeoff is worthwhile. When a user visualizes the energy usage of a device across different time spans, the system selects the most appropriate time resolution to view the data. For example, to view data over 10 minutes, the 1 second data tables will be used, while to view data over one year, the 6-hour table will be used.

The interface modules are all implemented as PHP scripts that form the basis of our Energy Auditor Dashboard website. Visualization, authentication, and analysis modules all reside as PHP programs. Users log in to the system and can view their energy consumption, analyze their usage, and actuate their devices.

The action and policy modules are implemented using scripts and the servers themselves. For the demand response actions, upon receiving the objective goal, the script will send a broadcast command to the command table specifying which priority level or device type to turn off. For occupancy-based policies, a script will check for new occupancy events once a second. When it sees a change, it will actuate the devices owned by that user appropriately. The other modules are implemented in a similar way, through scripts that interface with the storage module. Through this method, we can implement more modules as desired, as well as expand on our current ones.

## 4.4 Evaluation

### 4.4.1 Data Collection Results

We show a representative set of energy traces from our deployment. Several other research efforts have also presented energy traces to demonstrate the diversity in energy consumption loads across different devices and device types[18, 21, 32]. We instead focus on a few interesting power plots. Figure 4.4 shows the traces over a month of a computer and three monitors, combined as a single load. The reduction in energy when the monitors go to standby mode can be seen clearly, while the computer remains on the entire time, thereby wasting energy. Figure 4.5 gives a view over a week for an office microwave. This particular graph demonstrates one weakness with our materialized view averaging scheme – because a microwave tends to be on for only a few minutes, being averaged at 10 minute intervals will have the effect of distorting the actual power that microwaves require (typically more than 1000 watts). The different heights are the result of different amount of times that the microwave is heating a particular food. Figure 4.6 shows what it looks like averaged for only 1 minute over a single day.

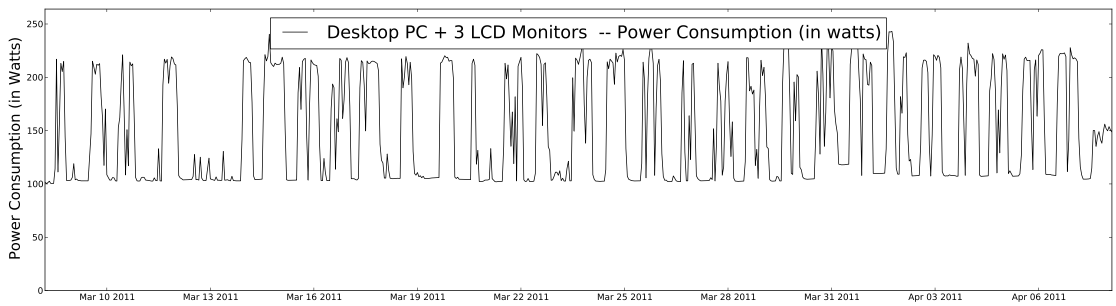


**Figure 4.4**: Power consumption of a desktop PC + 3 LCD monitors for over a week.

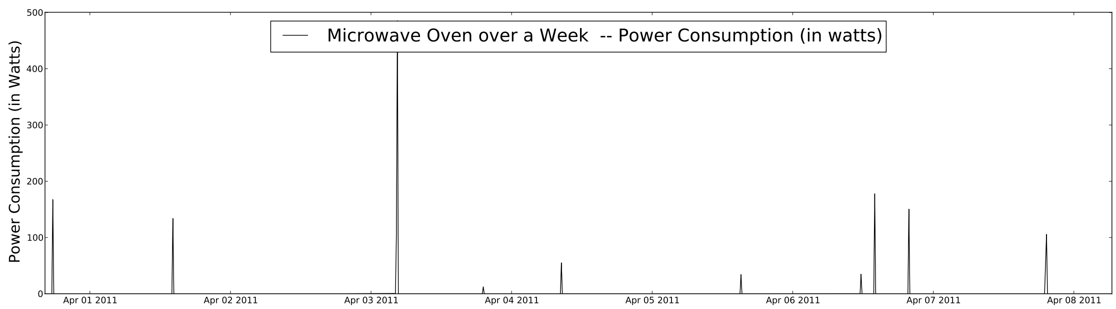


**Figure 4.5**: Power consumption for a microwave oven for a week.

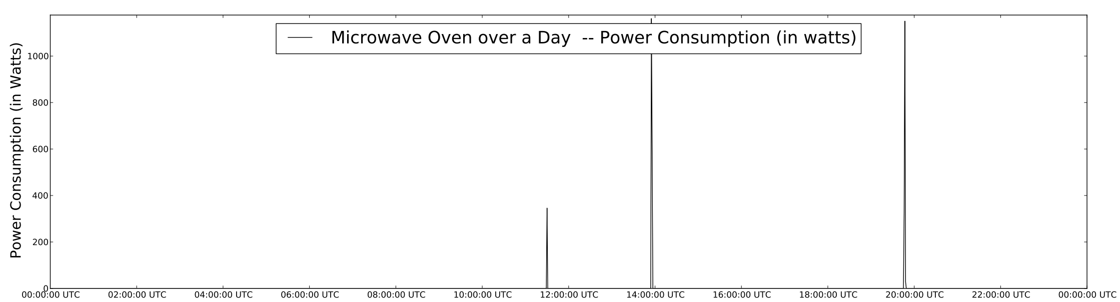


**Figure 4.6**: Power consumption for the same microwave oven for a day.

**Table 4.1**: Results of our load classification tests. For most general classes of devices, our algorithm works well and can recognize the load.

| **Load Type** | **Results** |
|---|---|
| Monitors | 14/17 (82.35%) |
| Desktop | 8/8 (100%) |
| Lamp | 4/4 (100%) |
| Laptop | 4/4 (100%) |
| Others | 6/7 (85.7%) |
| Total | 36/40 (90%) |

### 4.4.2 Meter Accuracy

We calibrated our smart energy meter using a WattsUp Pro[38] which is rated to be 1% accurate. We then tested the SEM with several loads with varying power consumption, ranging from 30W to 1kW. We also tested different type of loads - resistive and inductive (capacitive loads are a rarity). The measured values from the SEM were always within 1% of the WattsUp meter. Calibrating against a more accurate reference meter would allow us to acheive even better accuracy, and Texas Instruments claims that the particular MSP430 chip with ESP that we use has an accuracy up to 0.1%.

### 4.4.3 Load Classification Accuracy

We tested our load classification algorithm by plugging in different loads available around our building and checked the device type obtained through our load detection algorithm. The algorithm worked fairly well for loads which were similar to the ones we trained on. However, when the new load was remarkably different, like a fan with power factor correction or a 5 year old monitor, the algorithm did not converge to any class of load (returns "Not Sure"), or misclassified into a wrong device class (generally a lamp). Table 4.1 summarizes our test results with our classification algorithm on the most common device types in our building. We note that while our load detection algorithm can benefit from more complex NILM approaches [14, 29, 32] and improve detection accuracy, our simple heuristic based algorithm works well in office environments which are generally homogeneous.

### 4.4.4 Network Throughput Tests

For testing the network throughput, we connected 20 SEM nodes to a single basestation and had them report the power readings at once per second. The channel our Zigbee network uses is 20. We also actuated the device intermittently to check if both the directions of the network were working properly. We did not however, do a systematic test of switching the loads periodically, or extensively over the network. Our results showed that even with 20 nodes transmitting at once per second, the percentage of packet loss was always less than 1%, with an average of 0.05%. While we did not perform a comprehensive wide scale test, these results show that our network is able to handle at least 20 energy meters per basestation.

### 4.4.5 Demand Response

The demand response actions give building managers the ability to quickly shut off loads. We demonstrate the results of turning off devices based on priority levels. We test our priority level action in a single person office. We have seven loads, each with a different priority level (in parentheses) - a fan (1), phone charger (1), one laptop (4), a lamp (5), two monitors (6), and a desktop PC (10). The priority levels were set according to what a building manager might set - chargers and fans have low priority as they can be shut off without too much inconvenience, laptops can be shut off too because they typically have batteries, and desktop computers have the higher priorities because shutting them off can have a huge adverse effect on users (in this case, the desktop computer stays on the entire time). We stagger turning off each priority level a few minutes apart, starting from the lowest priority level and moving up to the highest priority level.

As can be seen, the fan and phone charger (with the lowest priority of one) both turn off simultaneously, and the laptop and lamp follow. We restore all the

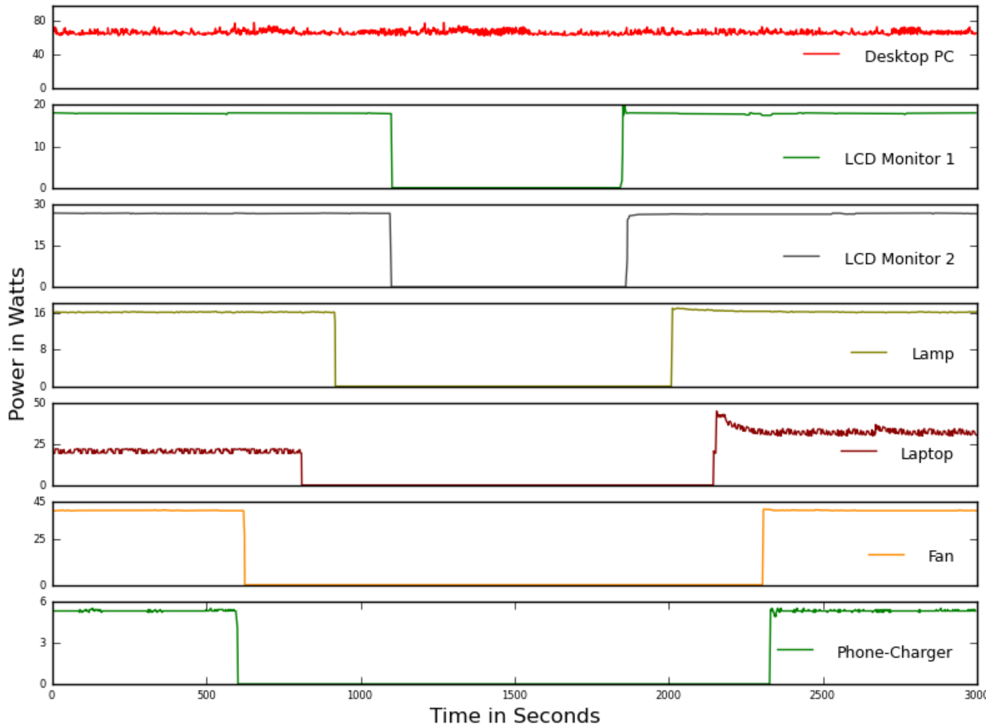


**Figure 4.7**: Our priority level actions efficiently handle demand response events. Notice how devices of the same priority level turn off and on at the same time.

devices afterwards, with highest priority first. As can be seen in the graph, the higher priority monitors turn on at the same time, followed by the laptop, lamp, and finally the fan and phone charger.

In a large deployment, with many meters and devices, the ability to shut down multiple devices with a single broadcast data message is extremely valuable. Likewise, the ability to restore devices to their previous state with one message after emergency events gives building manager options in handling energy situations.

### 4.4.6 Occupancy-based Policy

End users can set policies on their devices to actuate when certain conditions are met. One such policy is the occupancy-based policy, which will turn off devices when the user's room becomes unoccupied, and turn on devices when the room becomes occupied.

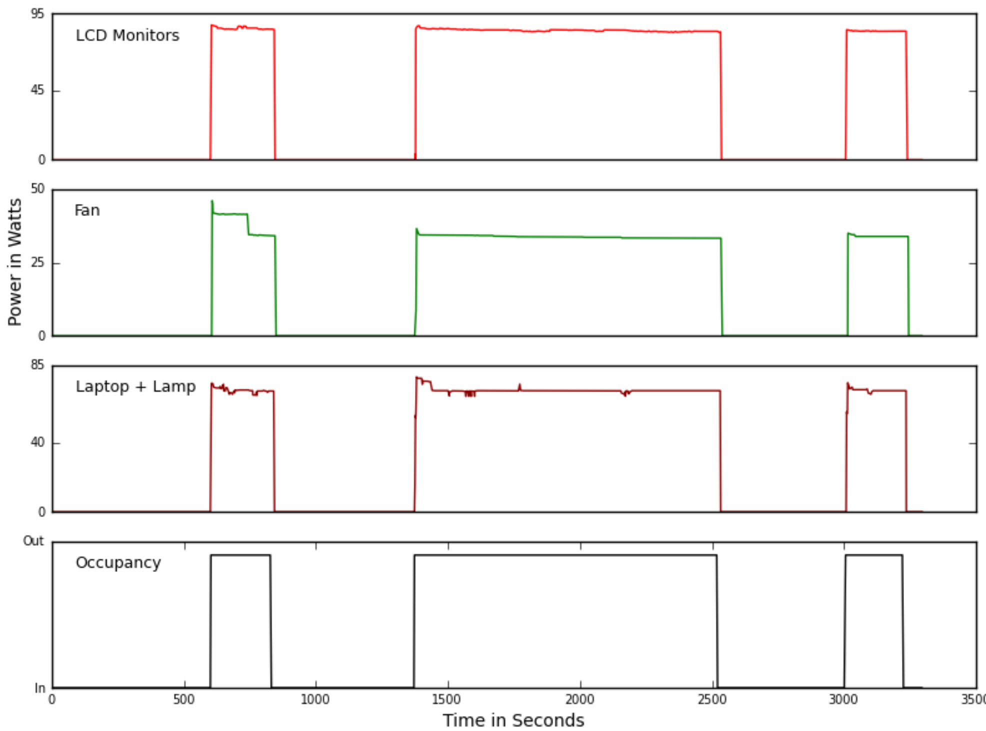


**Figure 4.8**: Results of our occupancy-based policy on a user's devices. Notice how the devices turn off and on immediately after an occupancy event.

We deployed our policy in an office room with four devices – a LCD monitor, a fan, and a laptop and lamp (on the same meter). Figure 4.8 shows the results over an hour and a half.

As the occupant leaves, the Energy Auditor detects it and sends a command to shut down all four devices, and as the occupant arrives, the Energy Auditor sends a command to turn the devices back on. Because of how our occupancy sensor works, there is a short delay of 15 seconds between a person leaving a room and our system registering that event. Therefore, it takes somewhat longer to actually turn off a device. Additionally, sometimes the data packet gets dropped and we must send a retransmission, which may further delay the actuation slightly.

## 4.5 Managing IT loads

An important contribution to the total energy consumption is IT loads. This is especially true of the IT dominated CSE building at UCSD. However, computers are becoming omnipresent and contribution of IT is slated to increase in any typical building. As we saw in Chapter 4, we cannot switch off the computers using our Smart Energy Meter as it needs to be gracefully shutdown. Previous work from our lab [1, 2] has addressed this problem. There are two solutions that we have come up with - Somniloquy and SleepServer. The basic idea behind them is that we can put the computer to sleep while maintaining its network presence. This allows the user to remote login, have small applications running in the background, wake up the computer in case of an Instant Message or VoIP call and other network related activities. Somniloquy requires attaching a USB module for this purpose to every computer interested in the product and SleepServer is a complete software solution for computers in enterprises.

Deploying this can lead to significant energy savings in IT loads. For example, had Somniloquy[1] been in use in the CSE building and using the data for the week of August presented in Figure 2.3, we can estimate the potential energy savings. Assuming all desktop PCs in the building were powered on for 45 hours during the week (8 hours a day, for 5 days per week + additional 5 hours) and were using Somniloquy at other times (during the evenings and the weekends), the direct energy savings from the reduction in plug loads alone over the entire week would have been around 20%. Additionally, if servers in the machine room were also using Somniloquy, using the same 45 hour work week, another 28% energy savings would be possible. Since this equipment will no longer be generating heat, second order effects such as reduced load on the air-conditioning and climate control system would lead to even more energy savings. Combined, the total energy savings would potentially be close to 50% of the current levels.

## 4.6 Future Work

The form factor of the Smart Energy Meter is something that can make a difference in its usage in practice. By making the energy meter so small and cheap, its feasible to embed them in the receptacles of new buildings. This would allow every plug point in the building to be monitored and actuated, giving the building managers a high amount of control on the plug loads in the building. For example, with a little bit of analytics, building managers will be able to see the power consumption in every room of a building, have an alarm raised if the power draw of a load increases beyond a threshold and so on. Another way the energy meter can be packaged is by embedding them in the plug loads themselves. This would help the energy saving appliances to show how energy efficient they are by giving a real-time feedback to the user and hence, become a selling point for them. It will also help in recognizing any anomalies in the normal working of the device. Other form factors can be in the form of a power-strip, or as a bead on the power cords to the appliances.

A primitive NILM algorithm can be embedded on the SEM to distinguish between a small number of loads, say up to 6. This would allow us to connect a power-strip to the SEM and still distinguish between the power consumed by different appliances connected to it. With this approach we can reduce the cost of deploying the energy meter on every appliance in the building. However, we will not have the control of switching each of the loads individually, and all of them will have to turned on/off as a group. Also, current NILM technologies require training, and even with a primitive implementation, this cannot be avoided.

With the capability of sending data at the rate of once per second, there will be a huge amount of packets generated by the ZigBee network in an average building. Although we have optimized the design for this high data rate, we still have to see how the network scales on a wider deployment. With ZigBee using the same 2.4GHz spectrum as WiFi commonly used in buildings, they could easily interfere with each other in a dense deployment [24]. Further, to the best of my knowledge, there has never been a sensor network deployment of nodes scaling to thousands of nodes, which is the number of nodes it would take to deploy this system across one building.

Privacy and security also become major concerns with such technologies. If a person's daily fine-grained energy consumption is exposed, one can easily interpret his regular activities. Further, as the SEM allows a person to actually switch devices on/off remotely, an unauthorized person can potentially do much more damage than just corrupting the personal computer or, obtaining confidential information. Both these concerns will need to be addressed before this technology is used in practice.

## 4.7 Acknowledgements

I would like to thank Prof. Yuvraj Agarwal and Prof. Rajesh Gupta for giving me the opportunity to work on this project. I would also like to thank my colleagues and co-authors Thomas Weng and Seemanta Dutta for their contributions and support throughout the project.

# Chapter 5

# Conclusion

With this work, we have tried to demonstrate that significant energy savings in buildings is possible if fine-grained occupancy information is available and put to good use. The Energy Dashboard gives us a good framework to study the energy consumption patterns of buildings at UCSD, and with special instrumentation, the subsystems of CSE building in particular. This allowed us to concentrate on the subsystems which consume the most energy in buildings, and also gave a good feedback on how much our solutions helped in reducing the power consumption of the building.

We designed and developed our own wireless occupancy sensor nodes for measuring the binary occupancy information of the rooms in the 2nd floor of the CSE building at UCSD. With the help of Physical Plant Services, who gave us access to the control of the BACNet system controlling the centralized HVAC system of CSE, we were able to study the effect of potential benefits of deploying the occupancy based HVAC control system on the building and also use the occupancy nodes to actually measure the performance of the deployed system on one out of four floors of the building. Our results show that we saved 9.54% to 15.73% in HVAC electrical energy and 7.59% to 12.85% in HVAC thermal energy on our test days.

To tackle the plug loads in the building, we developed our own Smart Energy Meter(SEM) and Energy Auditor system. The SEM is designed to be cheap, wireless and of small factor so that it is possible to deploy it on a wide scale even on existing buildings. The software on the SEM allows us to configure to transmit the power values at different rates, or send the data only when the values change by a certain threshold. The Energy Auditor analyses the data collected from these energy meters and provides information in a useful format to both the users and the building managers. The users can monitor all their devices, remotely actuate them and even set policies like automatically control them based on the occupancy information. The building managers can set a priority on each of the meters and actuate the meters based on this priority in case of a demand response event. We have demonstrated our system with a deployment of 15 meters in the CSE building at UCSD.

# Bibliography


[1] Y. Agarwal, S. Hodges, R. Chandra, J. Scott, P. Bahl, and R. Gupta. Somniloquy: Augmenting Network Interfaces to Reduce PC Energy Usage. In *Proceedings of USENIX Symposium on Networked Systems Design and Implementation (NSDI '09)*. USENIX Association Berkeley, CA, USA, 2009.

[2] Y. Agarwal, S. Savage, and R. Gupta. SleepServer: A Software-Only Approach for Reducing the Energy Consumption of PCs within Enterprise Environments. In *Proceedings of USENIX Annual Technical Symposium (USENIX ATC '10)*, 2010.

[3] Y. Agarwal, T. Weng, and R. Gupta. The Energy Dashboard: Improving the Visibility of Energy Consumption at a Campus-Wide Scale. In *First ACM Workshop on Embedded Sensing Systems For Energy-Efficiency In Buildings*, 2009.

[4] C. A. Barroso and U. Hlzle. The case for energy-proportional computing. In *Computer*, volume 40, pages 33–37. IEEE Computer Society, 2007.

[5] C.-Y. Chen and P. Chou. DuraCap: A Supercapacitor-based, Power-bootstrapping, Maximum Power Point Tracking Energy-Harvesting System. In *Proceedings of the 16th ACM/IEEE International Symposium on Low Power Electronics and Design (ISLPED)*, pages 313–318. ACM, 2010.

[6] CurrentCost Meters. http://www.currentcost.com/.

[7] S. Dawson-Haggerty, X. Jiang, G. Tolle, J. Ortiz, and D. Culler. sMAP: A Simple Measurement and Actuation Profile for Physical Information. In *Proceedings of the 8th ACM Conference on Embedded Networked Sensor Systems*, pages 197–210. ACM, 2010.

[8] D. T. Delaney, G. O'Hare, M. P. Gregory, and A. G. Ruzzelli. Evaluation of energy-efficiency in lighting systems using sensor networks. *Proceedings of the First ACM Workshop on Embedded Sensing Systems for Energy-Efficiency in Buildings*, pages 61–66, 2009.

[9] DOE. Buildings Energy Data Book, Department of Energy, March 2009. http://buildingsdatabook.eren.doe.gov/.

[10] Enocean: Wireless Lighting Solutions. http://www.enocean.com/en/energy-harvesting-wireless/.

[11] V. L. Erickson, M. A. Carreira-Perpinan, and A. E. Cerpa. OBSERVE: Occupancy-Based System for Efficient Reduction of HVAC Energy. In *Information Processing in Sensor Networks (IPSN '10)*, pages 258–269. IEEE, 2010.

[12] G. Florides, S. Tassou, S. Kalogirou, and L. Wrobel. Measures used to lower building energy consumption and their cost effectiveness. *Applied Energy 73*, pages 299–328, 2002.

[13] Google Power Meter. http://www.google.com/powermeter/about/.

[14] S. Gupta, M. S. Reynolds, and S. N. Patel. ElectriSense: Single-point Sensing using EMI for Electrical Event Detection and Classification in the Home. In *Proceedings of the 12th ACM international conference on Ubiquitous Computing (Ubicomp)*, pages 139–148. ACM, 2010.

[15] G. Hart. Nonintrusive Appliance Load Monitoring. *Proceedings of the IEEE*, 1992.

[16] S. Hay and A. Rice. The Case for Apportionment. In *Proceedings of the First ACM Workshop on Embedded Sensing Systems for Energy-Efficiency in Buildings (BuildSys '09)*, 2009.

[17] Honeywell: WSK-24 Wireless Occupancy Solution. http://specifyhoneywell.com/customer/techlit/pdf/63-0000s/63-4519.pdf.

[18] X. Jiang, S. Dawson-Haggerty, P. Dutta, and D. Culler. Design and Implementation of a High-Fidelity AC Metering Network. In *Proceedings of Information Processing in Sensor Networks (IPSN/SPOTS '09)*, 2009.

[19] X. Jiang, M. V. Ly, J. Taneja, P. Dutta, and D. Culler. Experiences with a High-Fidelity Wireless Building Energy Auditing Network. In *Proceedings of the 7th ACM Conference on Embedded Networked Sensor Systems*, pages 113–126. ACM, 2009.

[20] D. Jung and A. Savvides. Estimating Building Consumption Breakdowns using ON/OFF State Sensing and Incremental Sub-Meter Deployment. In *Proceedings of the 8th ACM conference on Embedded Network Sensor Systems (SenSys)*, 2010.

[21] M. A. Kazandjieva, B. Heller, P. Levis, and C. Kozyrakis. Energy Dumpster Diving. In *Proceedings of the Workshop on Power Aware Computing and Systems (HotPower '09)*, 2009.

[22] Y. Kim, R. Balani, H. Zhao, and M. B. Srivastava. Granger Causality Analysis on IP Traffic and Circuit-Level Energy Monitoring. In *Second ACM Workshop On Embedded Sensing Systems For Energy-Efficiency In Buildings (BuildSys)*, 2010.

[23] LEED Building Standards. http://www.usgbc.org/DisplayPage.aspx?CategoryID=19.

[24] C.-J. M. Liang, N. B. Priyantha, J. Liu, and A. Terzis. Surviving Wi-Fi Interference in Low Power ZigBee Networks. In *Proceedings of the 8th ACM conference on Embedded Network Sensor Systems (SenSys)*, 2010.

[25] J. Lifton, M. Feldmeier, Y. Ono, C. Lewis, and J. A. Paradiso. A Platform for Ubiquitous Sensor Deployment in Occupational and Domestic Environments. In *Proceedings of the 6th international conference on Information processing in sensor networks*, pages 119–127. ACM, 2007.

[26] J. Lu, T. Sookoor, V. Srinivasan, G. Ge, B. Holben, J. Stankovic, E. Field, and K. Whitehouse. The Smart Thermostat: Using Occupancy Sensors to Save Energy in Homes. In *Proceedings of the 8th ACM conference on Embedded Network Sensor Systems (SenSys)*, 2010.

[27] Microsoft Hohm. http://www.microsoft-hohm.com/.

[28] Pachube:. http://www.pachube.com.

[29] S. Patel, T. Robertson, J. Kientz, M. Reynolds, and G. Abowd. At the Flick of a Switch: Detecting and Classifying Unique Electrical Events on the Residential Power Line. In *Proceedings of the 9th international conference on Ubiquitous computing*, pages 271–288. Springer-Verlag, 2007.

[30] L. Perez-Lombard, J. Ortiz, and C. Pout. A review on buildings energy consumption information. *Energy and Buildings 40*, pages 394–398, 2001.

[31] V. Raghunathan, A. Kansal, J. Hsu, J. Friedman, and M. Srivastava. Design Considerations for Solar Energy Harvesting Wireless Embedded Systems. In *Information Processing in Sensor Networks (IPSN '05)*, pages 457–462. IEEE, 2005.

[32] A. Rowe, M. Berges, and R. Rajkumar. Contactless Sensing of Appliance State Transitions Through Variations in Electromagnetic Fields. In *Proceedings of the 2nd ACM Workshop on Embedded Sensing Systems for Energy-Efficiency in Building*, pages 19–24. ACM, 2010.

[33] V. Srinivasan, J. Stankovic, and K. Whitehouse. Using Height Sensors for Biometric Identification in Multi-resident Homes. *Pervasive Computing*, pages 337–354, 2010.

[34] S. P. Tarzia, R. P. Dick, P. A. Dinda, and G. Memik. Sonar-based measurement of user presence and attention. *UbiComp*, pages 89–92, 2009.

[35] T. Teixeira and A. Savvides. Lightweight people counting and localizing for easily deployable indoors wsns. *IEEE Journal of Selected Topics in Signal Processing*, 2(4):493–502, August 2008.

[36] The Energy Detective. http://www.theenergydetective.com/.

[37] S. Wang and X. Jin. Co 2-based occupancy detection for on-line outdoor air flow control. *Indoor and Built Environment*, 7(3):165–181, 1998.

[38] WattsUP. WattsUP Energy Meters. http://wattsupmeters.com.