Title
Software Augmented Buildings: Exploiting Existing Infrastructure to Improve Energy Efficiency and Comfort in Commercial Buildings

Author
Balaji, Bharathan

Publication Date
2016

UNIVERSITY OF CALIFORNIA, SAN DIEGO

Software Augmented Buildings: Exploiting Existing Infrastructure to Improve Energy Efficiency and Comfort in Commercial Buildings

A dissertation submitted in partial satisfaction of the requirements for the degree of Doctor of Philosophy

in

Computer Science (Computer Engineering)

by

Bharathan Balaji

Committee in charge:

Professor Yuvraj Agarwal, Co-Chair
Professor Rajesh Gupta, Co-Chair
Professor Ryan Kastner
Professor Jan Kleissl
Professor Alex Snoeren

2016

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

Co-Chair

Co-Chair

University of California, San Diego

2016

DEDICATION

To my father, who wanted me to become a doctor.

EPIGRAPH

It’s not who you are underneath,
it’s what you do that defines you.

Batman Begins

‘Can a man still be brave if he’s afraid?’
‘That is the only time a man can be brave.’

A Game of Thrones

The people who are crazy enough to think
they can change the world are the ones who do.

Steve Jobs

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACKNOWLEDGEMENTS

I would like to acknowledge Professor Yuvraj Agarwal and Professor Rajesh Gupta for their support as the chairs of my committee. Yuvraj Agarwal has supported and advised me throughout my graduate education, mentoring me closely in initial research projects and giving me independence to pursue my own ideas later on. He has encouraged me to come up with ideas that will have an impact on the society and advocated real world experiments that quantifies the effects of these ideas. He has also helped me learn how to communicate technical content and has made me confident in both writing research papers as well as presentations for various agenda.

Rajesh Gupta has encouraged me to pursue forward looking ideas and I have learnt much from his approach towards ambitious research projects with multi-year time scales. His teachings of precise writing and a single thought line in presentations has helped me compose many of my papers and presentations. He has always provided me with enough resources and provided opportunities for collaboration with both fellow students and research colleagues. I thank him for supporting me through many challenges that come up in graduate school.

Special thanks to my PhD committee members Ryan Kastner, Alex Snoeren and Jan Kleissl for serving in my committee and providing invaluable feedback to improve my research work.

I would like to thank Nadir Weibel, who introduced me to the research area of Human Computer Interaction, and has fueled my passion towards developing solutions that can improve the experience of people in society. He has encouraged me to pursue my ideas independently, and helped me understand as well as study the human aspect of the research challenges.

I would like to thank Balakrishnan Narayanaswamy (Murali) who helped me with pursuing machine learning in my research. He guided me through many experimental data analysis and introduced me to tools that brought me up to speed with state of the art methods. He has inspired me to respect machine learning and mathematical analysis in general, and I will continue to apply them in my future research projects.

Much of my work has been supported by the university Facilities Management, and in particular, I would like to thank Anna Levitt, John Dilliot, Robert Austin and Bob Caldwell for their many insightful discussions that helped me get an in-depth view of buildings as well as helping us to set up the infrastructure for smart buildings research. Also, special thanks to the Computer Science and Engineering building manager Dave Wargo for supporting us throughout our experiments in the building. Many thanks to Virginia McIlwain, Glenn Little and Steve Hopper for assisting me with many requests that have transpired in the past 5 years.

I would like to thank Alex Snoeren, George Varghese, YY Zhou for their inspiring lessons in computer systems and Scott Klemmer for his inspirational teachings in developing research ideas as well as Human Computer Interaction. In addition, I would like to thank Alon Orlitsky, Steve Swanson, Geoff Voelker, Ryan Kastner, Sorin Lerner, William Griswold, Donald Norman, Sanjoy Dasgupta for their lessons and discussions throughout graduate school. I also thank Alberto Cerpa, David Culler and Mani Srivastava who have inspired me with their research ideas and provided constructive criticism for my research work.

I thank Hidetoshi Teraoka, who visited UC San Diego as a research scholar from Hitachi, Japan. Hidetoshi helped me build the foundation of my thesis work, and his hard work and perseverance has inspired me to pursue ambitious projects. I thank Jian Xu, who partnered me with my first independent research idea, and together we implemented many building wide experiments and wrote a successful research paper. I thank Anthony Nwokafor who was our lab system administrator. Anthony and I had many fruitful research discussions on system implementation, and he has helped me maintain our lab infrastructure through many critical deadlines in my research projects. I also thank Thomas Weng, Seemanta Dutta, Michael Wei, Jacob Lyles, Sathyanarayanan Kuppuswamy and John McCullough, who were instrumental to start off my research career and I learnt many hands-on skills with their help.

Many thanks for Dr. Du Li who was my mentor at Ericsson Research and provided me guidance on how to attack new ideas, and the importance of precise goals and diving deep into a single project. I also thank my internship colleagues - Xiying Wang, Kshitiz Singh, Yifei Jiang - for both their collaboration and friendship. I also appreciate efforts of post internship collaboration by Xiying Wang.

I thank Alessandro Nacci who visited San Diego as an exchange student from Milano, Italy. Alessandro provided me a different perspective towards approaching research ideas, and I realized meaningful discussions and proper setting up of experiments play a critical part in making a project successful.

I have had the pleasure of mentoring many fellow students throughout my graduate program, and I thank all of them for providing me that opportunity and thank them for working with me. I thank Jason Koh, who has already co-authored multiple papers with me and is going to take over my smart buildings research. I thank Francesco Fraternali, who has worked with me closely on the hardware projects in our research group, and will be taking over the sensor deployment research. Special thanks for Master of Science students Rizhen Zhang and Huaipeng Zhang who made influential contributions to the web services infrastructure in our research lab. Also, thanks to Yan Zhang, Charlie Chen, Qi Li and Eugene Kolinko for their respective contributions in the form of MS projects. Thanks to Michelle Tang, Yung Nguyen, Wendy Cheung, Sze wun Wong and Sandeep Dey for their work towards Better Buildings Case Competition where we won the Most Innovative Award. Another vote of thanks to Michelle Tang who continued to work in our lab to create detailed 3D models of buildings. I also thank the undergraduate students who gave me mentorship opportunity – Danny Zhang, Andrew Yi, Macklin Lee, Xueyang Li, Mike Lara, Kevin Chavez, Jayon Huh, Watson Yim.

Many thanks to Chetan Verma for his friendship, many in depth discussion on both research as well as personal issues and for chipping in to co-author in one of my research papers. My heartfelt thanks to Ashish Tawari, Aman Khosa, Manish Gupta, Rishi Kapoor, Minu Jacob, Aman Bhatia, Swati Wagh, Arpit Gupta, Paula Anzenberg, Peter Edwards, Mark Srethabhakti, Tongqiu Jia, Tiffany Chan, Elena Menyaylenko, Or Sagie, Patil Varvarian, Ilean Chai, Mohammad Adnan, Tejaswi Iyyanki, Sana Ahmed, Deepti Pant, Pradeep Dhananjay, Shalmali Joshi, Siddharth Joshi, Dipen Parikh, Shams Pirani, Panagiotis Verkis for their friendship and many discussions that have shaped my ideas and direction in graduate school. It is not possible to list everyone who has helped me through my graduate school, but I thank them from the bottom of my heart.

Most of all, I thank my parents, Balaji Ramalingam and Yamuna Balaji for encouraging me to pursue the doctorate program and emphasizing the importance of knowledge to improve my quality of life. I cannot thank them enough for their support, love and confidence they instill in me. Last, but never the least, I thank my sister, Vidhya Balaji, who has been my confidant and a delightful companion through many adventures.

Chapter 3, in part, is a reprint of the material as it appears in Proceedings of ACM Conference on Embedded Networked Sensor Systems (SenSys 13), 2013 by authors Bharathan Balaji, Jian Xu, Rajesh Gupta and Yuvraj Agarwal with the title Sentinel: An Occupancy Based HVAC Actuation System using existing WiFi Infrastructure in Commercial Buildings. The dissertation author is the primary investigator and author of this paper.

Chapter 4, in part, is a reprint of the material as it appears in Proceedings of ACM Workshop on Embedded Systems For Energy-Efficient Buildings (BuildSys 13), 2013 by authors Bharathan Balaji, Hidetoshi Teraoka, Rajesh Gupta and Yuvraj Agarwal with the

title “ZonePAC: Zonal Power Estimation and Control via HVAC Metering and Occupant Feedback”. The dissertation author is the primary investigator and author of this paper.

Chapter 5, in part, has been submitted for publication of the material as it may appear in SIGCHI Conference on Human Factors in Computing Systems (CHI ’15), 2015 by authors by Bharathan Balaji, Jason Koh, Nadir Weibel, Yuvraj Agarwal with the title “Genie: A Longitudinal Study Comparing Physical and Software-augmented Thermostats in Office Buildings”. The dissertation author was the primary investigator and author of this paper.

Chapter 6, in part is currently being prepared for submission for publication in the Journal of Energy and Buildings by authors Bharathan Balaji, Nadir Weibel, Rizhen Zhang and Yuvraj Agarwal with the title “Understanding Building Operators to Improve Building Management Software”. The dissertation author was the primary investigator and author of this material.

Chapter 7, in part, is a reprint of the material as it appears in Proceedings of ACM Conference on Embedded Systems For Energy-Efficient Built Environments (BuildSys ’15), 2015 by authors Bharathan Balaji, Chetan Verma, Balakrishnan Narayanaswamy and Yuvraj Agarwal with the title “Zodiac: Organizing Large Deployment of Sensors to Create Reusable Applications for Buildings”. The dissertation author was the primary investigator and author of this paper.

VITA

2005 – 2009 Bachelor of Technology, Electronics and Communication Engineering, Visvesvaraya National Institute of Technology, Nagpur, India

2009 – 2011 Master of Science in Electrical Engineering, University of California, San Diego

2011 – 2016 Doctor of Philosophy in Computer Science, University of California, San Diego

PUBLICATIONS

Bharathan Balaji, Chetan Verma, Balakrishnan Narayanaswamy and Yuvraj Agarwal, "Zodiac: Organizing Large Deployment of Sensors to Create Reusable Applications for Buildings." In Proceedings of ACM Conference on Embedded Systems For Energy-Efficient Built Environments **(BuildSys '15)**, 2015.

Bharathan Balaji, Mohammad Abdullah Al Faruque, Nikil Dutt, Rajesh Gupta and Yuvraj Agarwal, "Models, Abstractions, and Architectures: The Missing Links in Cyber-Physical Systems" In Proceedings of the 52nd Design Automation Conference **(DAC '15)**, June 2015 Invited Paper.

Balakrishnan Narayanaswamy, Bharathan Balaji, Rajesh Gupta and Yuvraj Agarwal, "Data Driven Investigation of Faults in HVAC Systems with Model, Cluster and Compare (MCC)" In Proceedings of ACM Conference on Embedded Systems for Energy-Efficient Buildings **(BuildSys '14)**, 2014.

Bharathan Balaji, Jian Xu, Rajesh Gupta and Yuvraj Agarwal, "Sentinel: An Occupancy Based HVAC Actuation System using existing WiFi Infrastructure in Commercial Buildings" In Proceedings of ACM Conference on Embedded Networked Sensor Systems **(SenSys '13)**, 2013.

Bharathan Balaji, Hidetoshi Teraoka, Rajesh Gupta and Yuvraj Agarwal, "ZonePAC: Zonal Power Estimation and Control via HVAC Metering and Occupant Feedback" In Proceedings of ACM Workshop on Embedded Systems For Energy-Efficient Buildings **(BuildSys '13)**, 2013.

Bharathan Balaji, John McCullough, Rajesh Gupta and Yuvraj Agarwal, "Accurate Characterization of Variability in Power Consumption in Modern Computing Platforms" In Proceedings of USENIX Conference on Power-Aware Computing and Systems **(Hot-Power '12)**, 2012.

Thomas Weng, Bharathan Balaji, Seemanta Dutta, Rajesh Gupta and Yuvraj Agarwal, "Managing Plug-Loads for Demand Response within Buildings" In Proceedings of ACM Workshop on Embedded Sensing Systems for Energy-Efficiency in Buildings **(BuildSys '11)**, 2011.

Yuvraj Agarwal, Bharathan Balaji, Seemanta Dutta, Rajesh Gupta and Thomas Weng, "Duty-Cycling Buildings Aggressively: The Next Frontier in HVAC Control" In Proceedings of ACM Conference on Information Processing in Sensor Networks **(IPSN '11)**, 2011.

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

Bharathan Balaji, Bheemarjuna Reddy Tamma, and B. S. Manoj, "A Novel Power Saving Strategy for Greening IEEE 802.11 based Wireless Networks" In Proceedings of IEEE Global Telecommunications Conference **(GLOBECOM '10)**, 2010.

ABSTRACT OF THE DISSERTATION

Software Augmented Buildings: Exploiting Existing Infrastructure to Improve Energy Efficiency and Comfort in Commercial Buildings

by

Bharathan Balaji

Doctor of Philosophy in Computer Science (Computer Engineering)

University of California, San Diego, 2016

Professor Yuvraj Agarwal, Co-Chair
Professor Rajesh Gupta, Co-Chair

Commercial buildings consume 19% of energy in the US as of 2010, and traditionally, their energy use has been optimized through improved equipment efficiency and retrofits. Beyond improved hardware and infrastructure, there exists a tremendous potential in reducing energy use through better monitoring and operation. We present several applications that we developed and deployed to support our thesis that *building energy use can be reduced through sensing, monitoring and optimization software that modulates use of building subsystems including HVAC.* We focus on HVAC systems as

these constitute 48-55% of building energy use.

Specifically, in case of sensing, we describe an energy apportionment system that enables us to estimate real-time zonal HVAC power consumption by analyzing existing sensor information. With this energy breakdown, we can measure effectiveness of optimization solutions and identify inefficiencies. Central to energy efficiency improvement is determination of human occupancy in buildings. But this information is often unavailable or expensive to obtain using wide scale sensor deployment. We present our system that infers room level occupancy inexpensively by leveraging existing WiFi infrastructure. Occupancy information can be used not only to directly control HVAC but also to infer state of the building for predictive control.

Building energy use is strongly influenced by human behaviors, and timely feedback mechanisms can encourage energy saving behavior. Occupants interact with HVAC using thermostats which has shown to be inadequate for thermal comfort. Building managers are responsible for incorporating energy efficiency measures, but our interviews reveal that they struggle to maintain efficiency due to lack of analytical tools and contextual information. We present our software services that provide energy feedback to occupants and building managers, improves comfort with personalized control and identifies energy wasting faults.

For wide scale deployment of such energy saving software, they need to be portable across multiple buildings. However, buildings consist of heterogeneous equipment and use inconsistent naming schema, and developers need extensive domain knowledge to map sensor information to a standard format. To enable portability, we present an active learning algorithm that automates mapping of building sensor metadata to a standard naming schema.

Chapter 1

Introduction

It has been estimated that people spend 87% of their time indoors [104], and thus, buildings shape the experience of the populace in many ways. Modern buildings have evolved to meet growing requirements and comprise of a plethora of systems such as lighting, security, water, fire safety and air conditioning. Operating these systems requires considerable amount of equipment and consumes substantial amount of energy. The United States Department of Energy estimates that buildings constitute 40% of the total energy and 70% of the total electricity in the country [5].

With growing concerns on global warming and climate change, there is a movement to reduce the reliance on petroleum and coal based energy consumption. Renewable energy sources such as solar and wind have been growing steadily to reduce societal dependence on carbon based fuels [144]. Automobiles are slowly being converted to hybrid or fully electric vehicles, reducing the petroleum usage by the transportation sector [169]. These solutions are often referred to as the *supply side innovations* in smart grid terminology. Another approach to reduce fuel consumption is to reduce wastage and improve the efficiency of operation [177]. The energy efficiency solutions are referred to as *demand side management* [119]. As buildings are a large consumer of energy [5], reducing their energy consumption will have a large impact on the overall carbon emissions. Figure 1.1 shows the energy flows within the US as of 2014 and how buildings play a

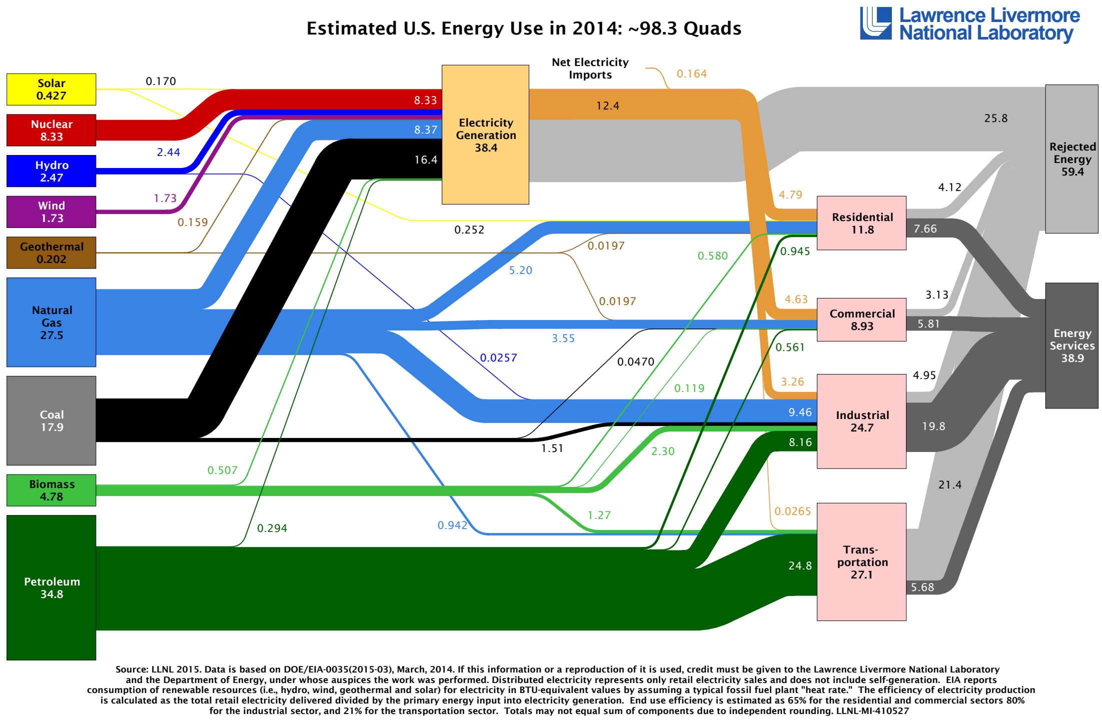

Figure 1.1. Sankey diagram showing the energy flow from generation to consumption in the US as of 2014 [15]

central role in overall energy consumption. The long term plan of the US Department of Energy is to reduce residential building energy consumption by 50% compared to 2010 baseline consumption [16].

A variety of efforts are being undertaken to reduce building energy consumption. Incandescent lights are being replaced with CFL and LED lights [132]. Motion sensors and photo sensors are installed to turn on the lighting only when needed. Building windows are carefully engineered to reduce solar radiation during hot weather. Energy Star appliances are being developed that are more efficient than traditional appliances. Digital devices such as TVs and computers now go to deep sleep when not in use. Constant speed fans used for ventilation are being replaced with variable speed fans whose power consumption is proportional to demand. Construction of buildings are carefully monitored to satisfy certifications such as California Title 24 requirements and obtain LEED certifications that ensure high energy efficient standards. Despite these efforts, building energy demand has been growing steadily for several decades due to increased economic activity inside buildings. Buildings contributed to only 26% of overall energy consumption in 1980 compared to 40% in 2010, and they are expected to contribute 45% by 2035 [5]. We need effective measures that can curtail building energy consumption further, and this dissertation explores innovative software based solutions towards improving energy efficiency.

To provide a concrete example of energy consumption profile of a building, we examine the Computer Science and Engineering (CSE) building at University of California, San Diego (UCSD) as an example modern building where many energy efficiency measures have been implemented. The building was constructed in 2004 and has a gross area of 150,000 sq ft. Figure 1.2 shows an image of the building. The building consists of faculty, administrative staff and student offices as well as lecture halls, conference rooms and a server room that consumes about 150KW on average, although its largely constant. Energy efficiency measures include motion sensor based lighting, glazed windows to reduce solar radiation, central air conditioning with variable speed drives, and a dedicated cooling system specially designed for the server room.

Figure 1.3 shows a breakdown of the electric power consumption by various subsystems in the building - lighting, plug loads, server room and mechanical load such as elevators and air conditioning [20]. We specially installed meters to obtain subsystem level power measurements to analyze the building power profile. As can be observed, the lighting system consumes around 10% of the overall power consumption, showing that the use of CFL bulbs as well as motion sensor based lighting has effectively curbed lighting power. The building power consumption is dominated by the plug loads, the server room and the mechanical load. The mechanical load consists of elevators, domestic water system as well as Heating, Ventilation and Air Conditioning (HVAC)

Figure 1.2. The Computer Science and Engineering building at University of California, San Diego. It was constructed in 2004 and has an area of 150,000 sq ft. The building contains office space, lecture halls and a server room.

system. Another observation is that the building power consumption is still high during nights and weekends when there is relatively low occupancy and activity. HVAC system power consumption also remain consistent during weekdays as they run on a static schedule from 6am to 10pm. In addition to electricity, the HVAC system consumes considerable amount of hot and cold water for heating/cooling the building, which is accounted as thermal power consumption. The combined energy consumption of HVAC system constitutes about 50% of the overall building consumption. Our findings are in line with observations made in prior work which estimated that HVAC systems consume 48% - 55% of the total building energy across several countries worldwide [142].

Such subsystem level measurement studies point us towards opportunities that

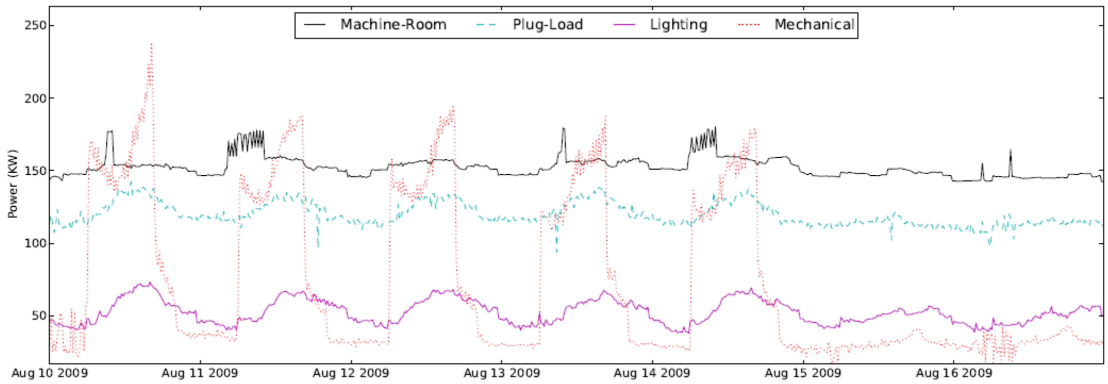

Figure 1.3. Breakdown of electrical power consumption in the CSE building into lighting, plug loads, mechanical and server room loads. The power consumed by the building is not proportional to actual activity and HVAC operates on a static schedule [20].

can lead to significant reduction in building energy consumption. For example, plug loads could be turned off when not needed and HVAC systems can be tuned to reduce wastage without sacrificing occupant comfort. To improve energy efficiency, we can continue using the same strategy developed over the years – upgrade inefficient equipment, install sensors that increase observability and improve operating efficiency. Many innovative solutions are being developed – plug level meters that measure appliance power and allow remote actuation [94, 175], accurate occupancy sensors that can be used for HVAC and plug load control [64, 18], thermal comfort sensors [75], smart blind systems [107], etc. Although these solutions can lead to significant energy savings, they invariably involve equipment or hardware installation which incurs high upfront cost and requires regular maintenance after deployment. Such solutions deters building owners and institution facilities management from adopting them. As an example, installing a single motion sensor in an existing building can cost \$900 in our university as per discussions with our university facilities management and majority of the cost is attributed to manual labor. Another example is where our university spent ~\$10,000,000 to upgrade HVAC equipment in a 200,000 sq ft building, called Pacific Hall, and estimates savings of ~\$900,000 per year. The expenses involved is high not necessarily because of hardware cost but also because of manual labor involved in incorporating upgrades in an *existing* building in active use.

Modern buildings consist of individual subsystem infrastructure that operate independent of each other. A working thesis here is that buildings can be made significantly more energy efficient using software solutions that leverage these existing infrastructure by exploiting information flow across disparate subsystems and creating holistic solutions that encompass various aspects of energy flow within the building. Since software only solutions do not require extensive hardware upgrades or additions they can be installed easily and at lower costs than those solutions that require major retrofits. Buildings deploy multiple systems to satisfy current requirements – power and water meters for billing purposes, WiFi network for connectivity, sensors and actuators for operation of HVAC system, etc. Software systems can be built that analyze the information from these existing infrastructure to not only increase energy efficiency but also improving occupant comfort, maintainability of existing systems. Such software solutions enable exploration of a different design space than those already addressed with hardware upgrades. Although software solutions already exist in many of these systems deployed in buildings today, they are stand alone solutions and do not communicate well with each other. Furthermore, energy consumption of a building is an aggregate function of how different building subsystem operate as a whole and the energy flow within a building is interdependent on the different systems deployed.

1.1 Related Work

Building energy efficiency has been identified as an important area of research. Many academic conference and journals focus on the topic of building energy - ACM International Conference on Embedded Systems For Energy-Efficient Built Environments, Journal of Energy and Buildings, Journal of Building and Environment, Sustainability track in Human Computer Interaction (CHI) conference and other venues where energy efficiency of buildings is actively discussed. We focus on a few pertinent related work here that focus on software enhancements to buildings for saving energy.

There have been several research projects which exploit existing properties of an infrastructure to gain insights about a system. Prior work has focused on installing a single sensing point in a system in homes to gain insights into resource usage. For example, Hydrosense detects water usage at each faucet by measuring pressure in one of the central pipes in the house [73]. Electrisense detects use of individual electric appliances within a home by careful analysis of EMI interference measured by electrical noise [139]. There is an entire community of people researching ways to disaggregate energy usage to each appliance within homes using smart meter data [106]. The ideas presented in this dissertation builds on top of these works, and instead of installing new sensors, we exploit existing sensors already installed in buildings for other purposes. Further, we specifically focus on developing systems that provide actionable insights and means to save energy in real building deployments.

Modern buildings consists of large number of networked sensors and actuators for regular operation and maintenance. The building systems are maintained using Building Management Systems (BMS) which visualize sensor information and assist building managers implement various control policies. However, these BMSes are provided as vertically integrated systems, and are not optimized for implementing energy efficiency measures. To overcome the limitation of BMSes, the Energy Information Systems (EIS) have been recognized to provide insights into building energy efficiency [80]. These systems collect energy information from disparate sources – power meters, water flow meters, heating/cooling thermal meters, etc. and mitigate the interoperability problem caused by vertically integrated systems. As building maintenance personnel can now view information across the system, they can better analyze the building energy performance.

However, these systems do not integrate actuation mechanisms, and rely on human actuation to enact efficiency measures. In the systems presented in this dissertation, we not only integrate data from different sources, but also develop systems that actuate building systems automatically to improve energy efficiency and develop energy data analysis methods to reduce cognitive load on the maintenance personnel.

Energy consumption of a building is heavily influenced by actions and behavior of individuals - occupants, building managers, maintenance personnel [118]. It is important to engage all the stakeholders involved to improve the effectiveness of energy efficiency measures. Several studies have focused on design of feedback systems that encourage users to save energy [83, 72]. In this dissertation, we present an in-depth user study of building managers and maintenance personnel to understand their perspective on energy efficiency measures. We also present software applications for occupants to interact with building components. With analysis of their usage patterns in real building deployments, we present an analysis of interaction design mechanisms can lead to both improved comfort and energy savings.

Many papers have developed data analytic and control optimization techniques for various aspects of building systems such as fault detection, improving control system efficiency and reducing peak demand. Fault detection and diagnosis in large complex systems is an active area of research and many sophisticated algorithms have been proposed and evaluated [99, 100]. During peak electricity demand, the generation of electricity is expensive, while also causing higher levels of pollution, due to use of peaker plants. During periods of such high demand, the electric grid issues *Demand Response* events to buildings to reduce the demand on the grid [23]. Algorithms for integration of thermal energy storage and demand response have been developed [160]. Further, systematic protocols for automated demand response is being implemented [145]. Other works include building energy simulation [52] and prediction of building energy [179].

However, principled approach to integrate these numerous control and analysis methods in a holistic manner is missing. We will present the software frameworks we have developed that integrate such systems in a systematic and scalable manner.

This short literature review presents the prior works related to the thesis of this dissertation. Many contemporary research efforts parallel the solutions proposed in this dissertation. For example, sMAP [57] integrates information from various data sources and BAS [109] creates APIs that eases software development on top of existing buildings. Thermovote [63] and Comfy [14] allow occupants to interact with the HVAC system using software applications. In addition, there is other prior work that is more closely related to the various systems we will present in this dissertation. We will examine them in detail in the respective chapters.

1.2 Contributions

We present the design, implementation and evaluation of several software systems that exploit the existing building infrastructure to improve energy efficiency, maintainability and occupant comfort. We focus on the commercial buildings sector, particularly operation of HVAC systems as they are energy intensive and large improvements can be obtained by incorporating innovative software solutions. Each of these solutions have been deployed on a real testbed and evaluated extensively to validate the solution as well as quantify the actual benefits. We also present the challenges and solutions for deploying software systems across multiple buildings so that software applications once developed can be deployed on a large scale without incurring significant expenses.

Detecting occupancy within buildings at a fine spatial granularity is key for building management, security, asset tracking, etc. It is also essential if we want to do occupancy based HVAC control for energy efficiency. Most prior solutions require additional hardware sensors to be deployed across buildings. The first system that we will discuss exploits existing information available for enterprise WiFi network deployments, coupled with other readily available metadata, to provide robust occupancy detection at a room level within a building. The occupancy inference algorithm makes use of metadata such as the building floorplans, office spaces assigned to occupants and location of access points. *This system shows how information flow from WiFi network, office space management to the HVAC system is exploited to improve HVAC efficiency.*

The second system confronts the apportionment of HVAC energy consumption using existing sensors deployed. Most buildings consists of aggregate building power meters and it is difficult to identify areas where energy efficiency measures can be implemented. We analyze the data from HVAC sensors, study the mechanical design of the CSE building at UCSD and apply heat transfer equations to obtain energy consumed by each thermal zone in the building. We also develop a web service that provides a real-time feedback of energy consumption to the building occupants. The user interface, called Genie, also allows occupants to check HVAC status and change their temperature controls. *Thus, by using information from HVAC sensors, building level power meters, building architectural diagrams, we analyzed energy flows in depth as well as provided occupants with this key information with a web application.*

The thermostats already installed in the building was not easy to use, did not provide sufficient control, occupants did not whom to contact in case of discomfort and as the thermostats were shared between offices, some occupants could not access the thermostat. After we deployed Genie in the building, many occupants appreciated the information available, and made regular use of temperature control provided as well as sent feedbacks when HVAC did not function correctly. Users did not abuse the control available, and the energy consumption changes were minimal. The facilities management appreciated that problems such as use of space heaters and blocking of thermostats got reduced because of feedback mechanism available in Genie. *Thus, our software system* *exploited existing HVAC infrastructure to bridge the communication gap with building occcupants, and provided them transparent access, control and feedback mechanism for using the HVAC system.*

We interviewed building managers and personnel from facilities management to identify energy efficiency implementation challenges. Our usage study showed that maintenance staff have overwhelming work load and would often prioritize occupant discomfort over efficiency measures. The maintenance personnel struggled to infer insights due to lack of contextual data and analysis tools. We designed our fault management system that integrates information from diverse sources, stores historical data, and enables third party apps using RESTful APIs. We detected 88 faults in the CSE building, and many of these faults were causing energy wastage which were not found using traditional methods. *Thus, we show another instance where integration of information across existing resources and analyzing the data using modern algorithms can lead to new insights that can save energy, and in this case, improve maintainability of buildings.*

Buildings are heterogeneous entities and the infrastructure present as well as their usage characteristics vary widely depending on age, usage model and vendors used. The naming convention used for sensors vary widely even though most of the equipment are similar. Mapping of these sensors to a machine readable format requires considerable manual effort and domain expertise, and increases the cost and time of deployment. With the help of unsupervised clustering methods and active learning based labeling of sensors we utilize available sensor information and domain expert labeled examples to accurately map sensors to the standard naming convention. *Thus, with our algorithm we can map existing building sensors to a machine readable format that can be used for porting software applications across multiple buildings.*

1.3 Organization

This dissertation is organized as follows. In Chapter 2 of this dissertation, we present the challenges involved in deploying hardware based solutions in buildings with the example of wireless occupancy and plug meters developed by our Synergy Labs research group. In addition, we introduce the principle working of HVAC systems and their associated software management system which is the focus of all of the software systems described in this dissertation. We also present the data storage and dissemination solution developed by our research group that allows us to communicate with various systems present in the building and build various applications on top of them. The software systems presented in this dissertation build on top of this data storage service.

In Chapter 3, we present our software system that exploits existing enterprise WiFi network to detect presence of occupants in buildings and uses the information to actuate HVAC system accordingly. It is common for commercial buildings to install an enterprise WiFi network, and occupants tend to connect to this network either using their laptop or smartphones. We use the WiFi connectivity information along with building occupant metadata to infer office occupancy and connect this information to the HVAC system for occupancy based control.7

In Chapter 4, we present our energy apportionment system for HVAC systems which estimates heating, cooling and electrical power consumption of individual thermal zones in a building. Such an apportionment can be used to understand the energy flows in a building, assign energy bills as per individual use and identify methods to reduce consumption of energy intensive areas. We also provide energy information as a feedback to the occupants of the building along with useful features such as control of temperature setting and ability to send thermal feedback.

In Chapter 5, we present the analysis of the software interface usage by occupants of the CSE building across 21 months. Over 220 users registered for our service across 21 months, and we compare the usage of the software interface with the traditional thermostats to examine the pros and cons of the software system. We provide design guidelines towards the end for improvement of future software interfaces.

In Chapter 6, we present an analysis of software interface design for building managers and other maintenance staff. Current maintenance personnel are overwhelmed with the number of tasks they need to do for maintaining HVAC systems. We interviewed various building management software users across 5 institutions to understand their point of view. We also present our own solution that may alleviate the concerns expressed by the building managers, and show the challenges in implementing similar systems for other existing buildings.

In Chapter 7, we present an automated way to organize sensor metadata information to a standard format to enable reusable, portable applications across buildings. Building vendors use inconsistent naming conventions and it becomes difficult to deploy software applications in buildings due to initial work involved in mapping existing sensor metadata to a standard format. We take four example buildings at our university and show machine learning methods can be used to reduce manual labor involved significantly.

Chapter 8 discusses future work and Chapter 9 concludes the dissertation.

Chapter 2

Building Testbed and Sensor Data Organization

We describe here our testbed, including sensor nodes and data architecture used for this research.

2.1 Embedded Sensing for Occupancy

We built occupancy sensing by combining sensing data from Passive Infrared (PIR) based motion sensors as well as magnetic door sensors [19]. These sensors improved accuracy of traditionally used motion only sensors which failed to detect people when they were relatively motionless. By incorporating door sensors, the direction of motion can be inferred, and accuracy of occupancy detection in single person offices was found to be 97% in an experimental deployment. We designed the sensors to be wireless, low cost and low power so that they could be deployed in existing buildings without extensive wiring. We showed that the occupancy sensor was effective in curtailing HVAC power consumption as we could now turn On HVAC in offices only when they are occupied [18]. In addition, we developed a wireless plug level power meter, which allowed monitoring of appliance power consumption as well as remotely switch the power outlet [175]. The plug meter was envisioned for detailed power monitoring and

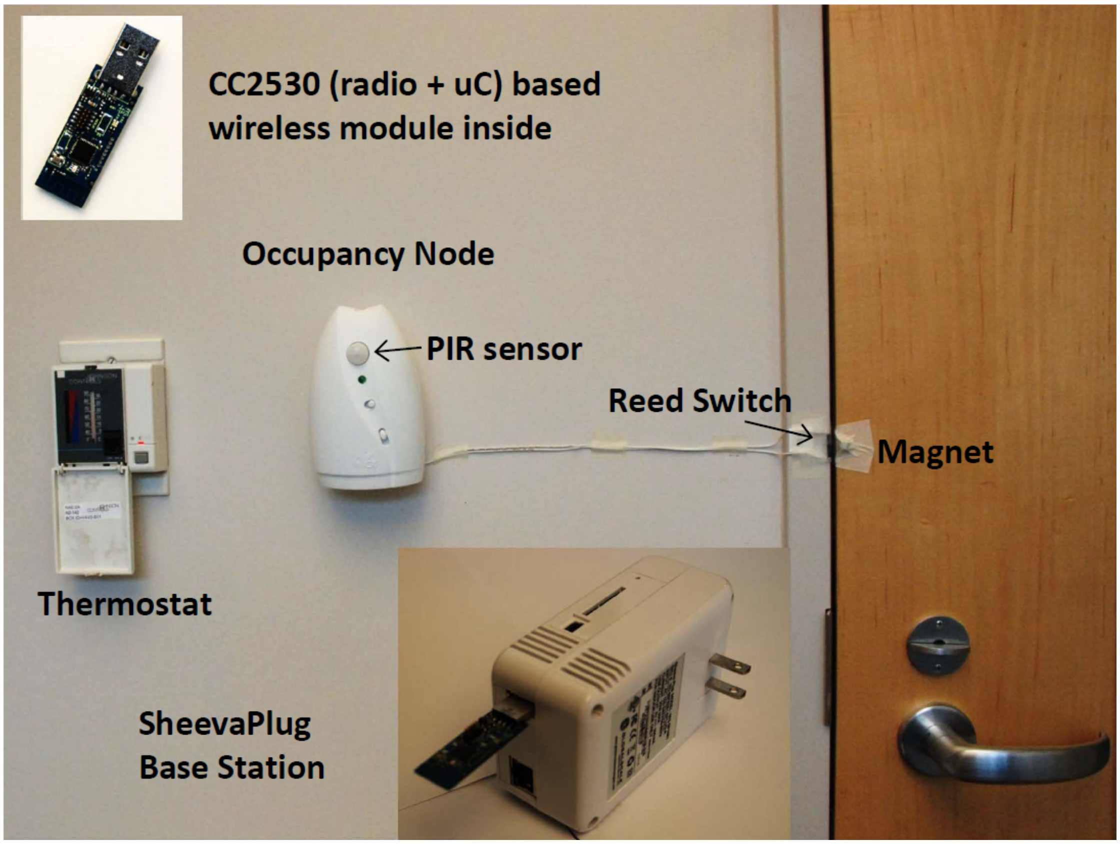

Figure 2.1. Occupancy node deployed on the wall of an office. The reed switch, PIR sensor and our CC2530 based radio module inside the occupancy node are also shown [18].

for effective management of devices during demand response events. Figure 2.1 shows the occupancy sensor and Figure 2.2 shows our plug meter. In addition, we draw upon deployment experiences published by Hnat et al. [86] and Dawson-Haggerty et al [56].

2.1.1 Design Challenges

To develop sensors that are low cost and easy to deploy, several aspects need to be considered. The sensors needs to be networked wirelessly as wiring of sensors in existing buildings is cost prohibitive. The sensor also needs to be low power so that it does not need power lines and can be perpetually powered using energy harvesting methods [60] or has a multi-year battery life time. In addition to meeting these stringent hardware

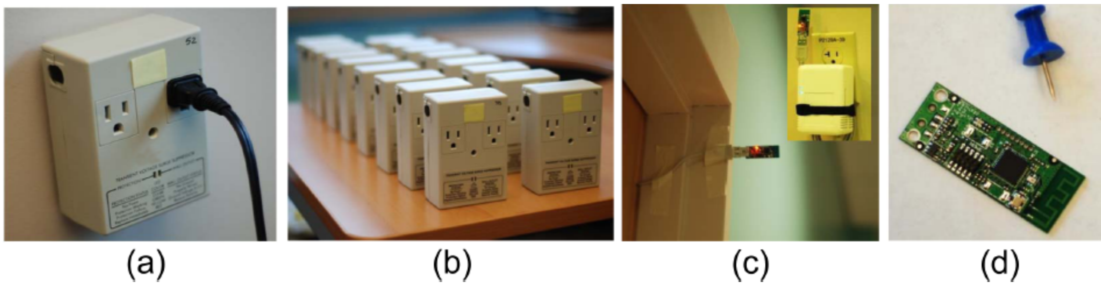

Figure 2.2. Picture of our energy meter (a, b), our SheevaPlug base station (c) that is deployed in the hallways, and the CC2530-based wireless module (d) used in our base station and energy meters. [175].

requirements, the final product needs to satisfy requirements of certification bodies such as FCC and UL as well as meet building codes such as those for fire safety. Aesthetics of the product also makes a large impact in adopting of such sensors. For example, prior deployments have noted that research study participants would unplug sensors because the LED lights were annoying [86].

As the sensor needs to be low power and wireless, it needs to adopt low data rate wireless protocol such as ZigBee or 6lowPAN. Thus, just like the WiFi network infrastructure, a separate network infrastructure needs to be created for these wireless sensors. Creating this additional layer of network infrastructure is challenging as well. The wireless network needs to be robust to interference from devices such as WiFi or microwave. At least minimal wiring is required for gateway devices that connect the sensor network to the Internet. And finally, the sensors may be placed in challenging RF environments such as behind a metal desk which reduces their range considerably. Hence, adequate repeaters need to be deployed to provide coverage to such sensors.

2.1.2 Deployment Challenges

Even though wireless sensors mitigate extensive rewiring in the building, there are number of concerns that needs to be addressed for a smooth deployment. For instance, placement of sensor is constrained by the type of the sensor and its sensitivity. For example, our occupancy sensor needs to be close to the door and should not be blocked for proper motion detection (Figure 2.1). But, this is sometimes not possible because of furniture or other equipment that the occupant needs to have at the same place. Such impediments reduce sensor accuracy and data validity.

Typically, the same binary program is flashed onto the sensors and metadata information such as room number needs to be entered at the time sensor is installed. To reduce deployment time, a specialized application needs to be developed that configures the sensor with minimal manual effort. Another time consuming step in the deployment of a sensor node is the calibration of the sensors to the particular indoor environment. The deployment time itself needs to be coordinated with occupant requirements. Hnat et al. [86] report that appointment coordination was a major hurdle in their deployment efforts.

2.1.3 Maintenance Challenges

Maintenance of sensor networks can be surprisingly difficult due to the scale of deployment. At UC San Diego, even with wired sensors already deployed in buildings, many sensors fail and get miscalibrated. But, facilities management do not have sufficient manual labour to fix all of the problems that arise. Even minor maintenance such as replacement of batteries becomes manually intensive at the scale of hundreds of sensors. Many such minor problems such as sensor drift or miscalibration are not fixed for months at a time.

Wireless sensor networks make maintenance even more challenging due to network interference caused by other WiFi devices, metallic furniture, building metal infrastructure and other radio devices. Both Dawson-Haggerty et al. [56] and Hnat et al. [86] report challenges in maintaining a functional network because of lack of reliable connections, unexpected link changes, etc. We find similar challenges in our sensor network as well.

Similar challenges are encountered in other types of retrofit solutions. For example, the facilities management identified that the economizer in our CSE building was not functional. But the economizer dampers were large and expensive to replace, and the management needed to identify both the personnel as well as the funding for undertaking the retrofit. The economizer was eventually repaired over 18 months later with total expenses of $26000.

Such challenges in adoption of hardware based solutions motivates my thesis to look towards software solutions that can lead to energy efficiency improvements. Although hardware retrofit solutions cannot be avoided altogether, we will present several examples of software systems which augment or improve upon existing energy saving solutions. Software solutions can provide insights into building operation with data analytics, can provide easy to use building interaction systems and incorporate control optimizations that improve efficiency. In the rest of the chapter, we provide a brief background to help the reader navigate the rest of the dissertation.

2.2 CSE Building Testbed

Our building testbed, the Computer Science and Engineering (also called EBU3B) building at University of California - San Diego, was built in 2004 and consists of 466 rooms with 150,000 sqft of floor space.

2.2.1 CSE HVAC System

The HVAC system within our building uses a combination of hot and cold water pipes in conjunction with air-handler units(AHU) to maintain the appropriate thermal environment within the building. Given the size of our university, we employ a central utility plant for producing the hot ($\sim 325\,^{\circ}$F) and cold ($\sim 45\,^{\circ}$F) water distributed campus

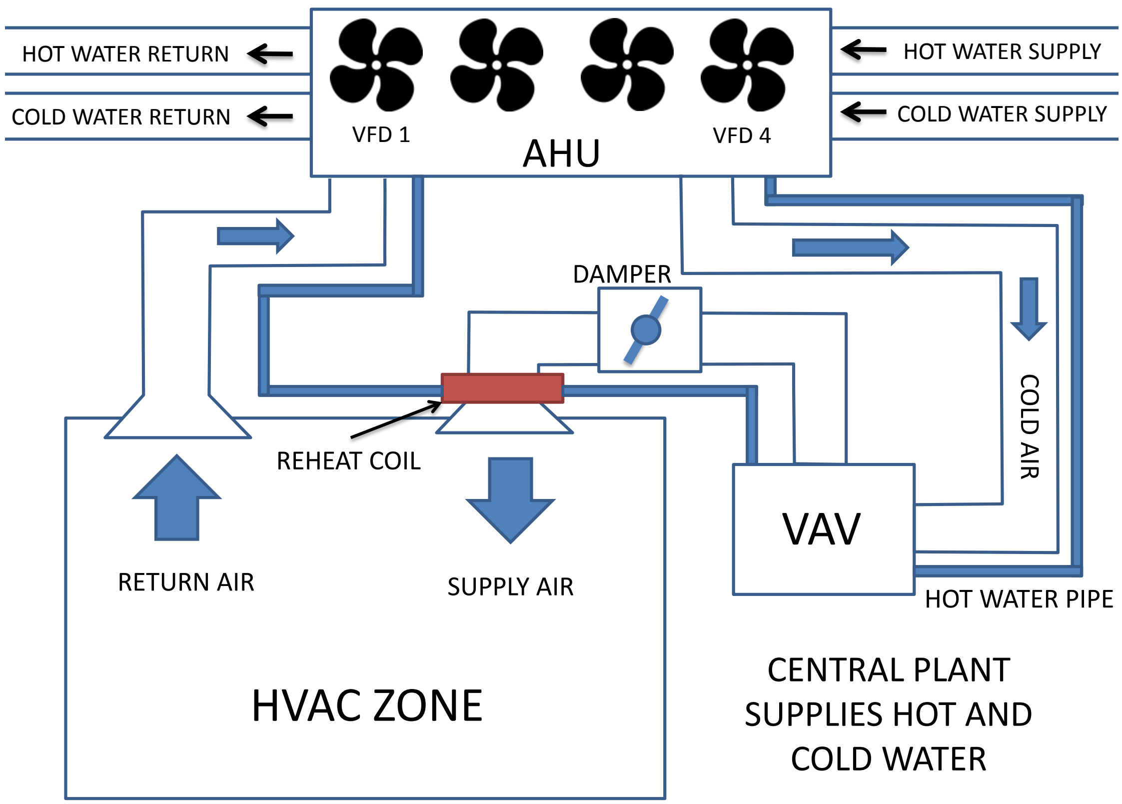

Figure 2.3. Overview of the HVAC System in commercial building on our campus. Hot water and cold air is pumped to different VAV boxes by AHU. VAV boxes provide independent control in each HVAC zone.

wide using separate loops as shown in Figure 2.3. Our building uses Variable Air Volume (VAV) boxes that allow local temperature control, which is estimated to cover 20% of cooling systems and are commonplace since 1990s [90]. The AHU in our building consists of variable speed drives which supply cold air (converted from the supplied cold water) using ducts to VAV boxes distributed throughout the building. The hot water loop is also connected to these VAV boxes using separate pipes. Each VAV box controls the amount of cold air to be let into an HVAC zone using dampers. A reheat coil, which uses supplied hot water, is used to heat the cooled air to meet the appropriate HVAC settings for each zone.

Figure 2.4 shows the system design of the centralized part of the HVAC system.

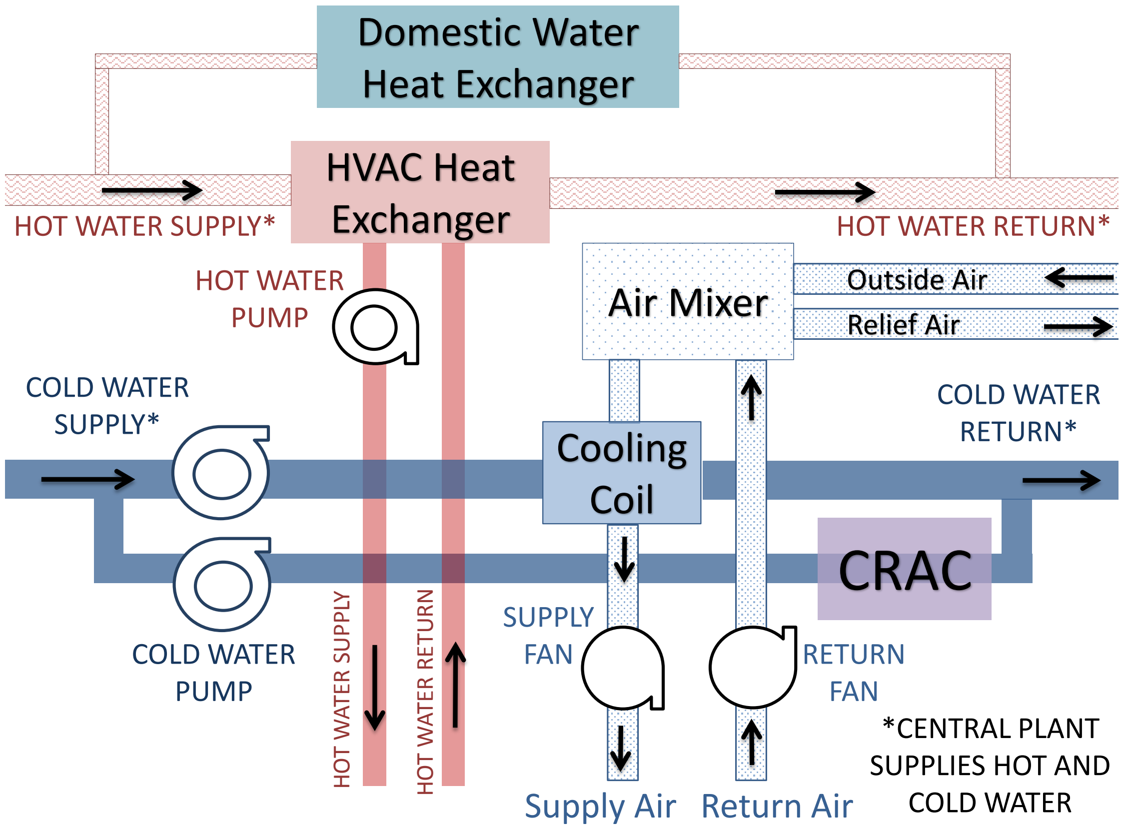

Figure 2.4. Centralized part of HVAC system showing details of air handler unit - cooling coils, heat exchanger, water pumps, supply/return fans, domestic water and CRAC units.

UCSD has a Central Utility Plant (CUP) that operates gas driven turbines with hot and chilled water by products. Chilled water is used for large scale energy storage that accounts for nearly 14% of daily energy use of the entire campus. As a result, we do not have chillers, cooling towers or boilers in the building normally associated with commercial HVAC systems. As the weather is temperate throughout the year in San Diego, the HVAC system was designed without any humidity control. Thus, we do not have any humidifiers or de-humidifiers found in buildings with harsher outdoor environment. The centralized part of the system supplies cold air and hot water to the VAV boxes for providing conditioned air to building spaces.

The cold water from CUP, supplied at $\sim 44\,^{\circ}$F, is distributed to the rest of the

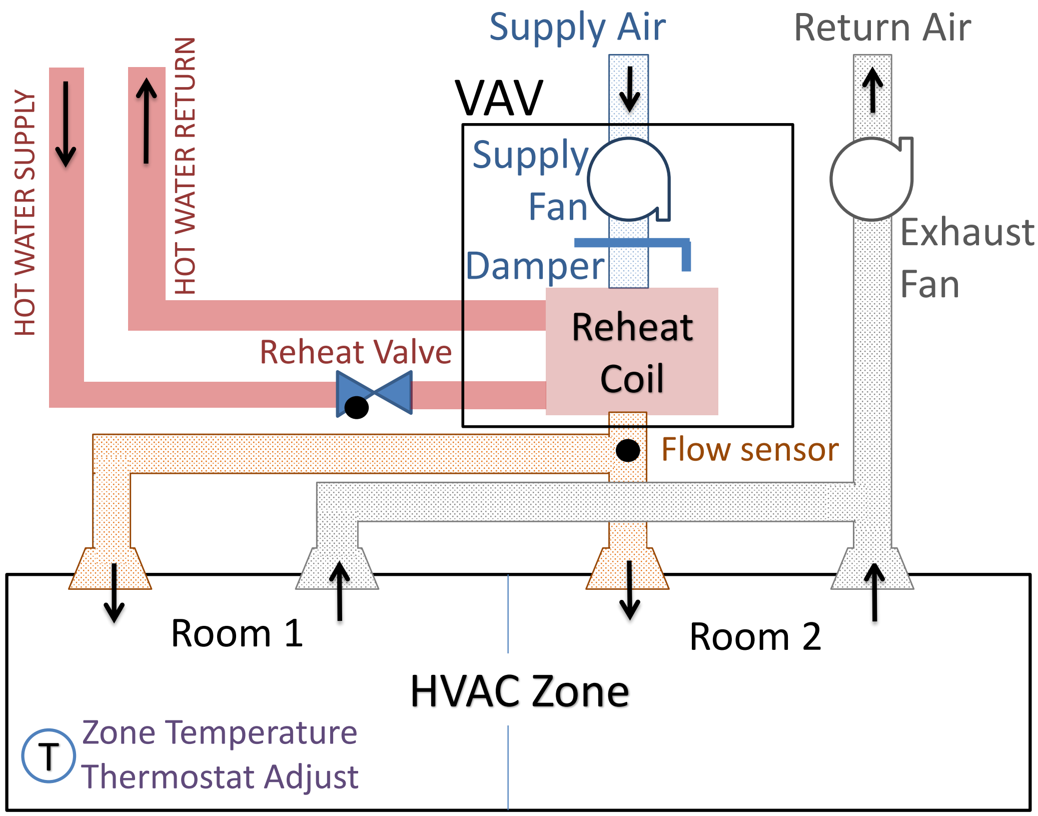

Figure 2.5. VAV with reheat system used for controlling the temperature of discharge air in each HVAC zone.

system by four cold water pumps, two of which are dedicated for supplying water to the Computer Room Air Conditioning (CRAC) unit for the server room in CSE. The cold water is passed through cooling coils, which cool the mixture of outside and return air to the appropriate setpoint ($\sim 55\,^{\circ}$F) to provide supply air to all the zones in the building. The supply air is dispensed to the VAV boxes via ductwork using supply fans, and the flow of return air is facilitated using return fans. The air mixer uses economizers (not shown) to increase the proportion of outside air if outdoor conditions are favorable for reducing energy usage.

The hot water from CUP, supplied in the form of pressurized steam at $\sim 325\,^{\circ}$F, passes through heat exchangers for heating up the hot water returned by the VAV boxes.

Part of the hot water is used to heat the domestic water. The hot water from the HVAC heat exchanger is supplied via pipes to the VAV boxes using hot water pumps. All the pumps and the fans used in the centralized part of the system employ Variable Frequency Drives(VFDs). The CRAC units do not use the hot water, and have an electric reheat system for environment control.

Figure 2.5 shows the HVAC design of the VAV boxes in CSE. The amount of cold air supplied to each zone is modulated using a damper, and a flow sensor measures the airflow rate. The zonal temperature is controlled by modulating the amount of cold air and by using the hot water coil to reheat the air. The amount of hot water used in heating the air is modulated using an electronically controlled 2-way valve. Every zone has a thermostat which measures the current temperature, and acts as the feedback for the VAV control system. Occupants are allowed to change their temperature setpoints by $\pm 1\,^{\circ}$F using the thermostat dial.

2.2.2 Building Management System

A central Building Management System (BMS), managed by Johnson Controls, has supervisory control over the HVAC system and the various HVAC components are connected to the BMS via BACnet - a standard protocol for Building Automation and Control networks [41]. Each VAV box has sensors for measurement (zone temperature, air flow, damper position), virtual sensors for monitoring (occupancy status, heating and cooling temperature set points) and control (change set point, change minimum air flow, change occupancy status). Figure 2.6 gives an overview of BACnet connecting different HVAC subsystems.

The BMS operates the HVAC system on a weekly schedule. On weekdays, the HVAC system is put to “Occupied” from 6am to 6pm, then changed to “Standby” mode till 10pm, and switched to “Unoccupied” for the rest of the night. In the “Occupied” mode,

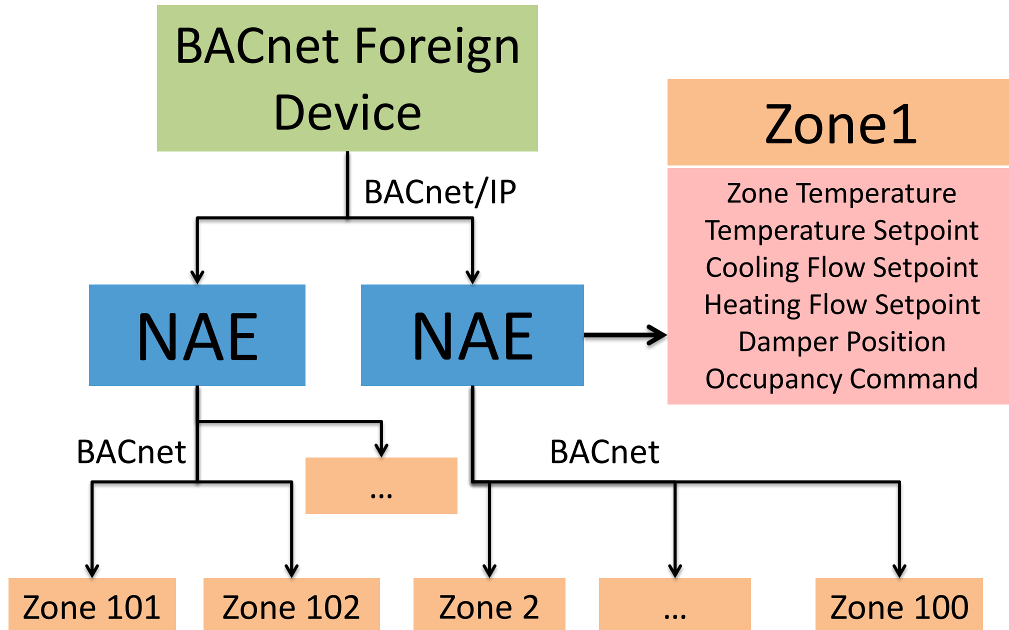

Figure 2.6. Overview of control system of HVAC using BACnet. We connect to the BACnet as a Foreign Device and sends commands using BACnet Read Property and Write Property for control of HVAC system.

a minimum amount of airflow is maintained for ventilation, and the minimum airflow setpoint for each zone is determined based on its maximum capacity. The temperature of the zone is maintained within 4 °F range, and the exact range is determined by the temperature setpoint set by the BMS as well as the thermostat adjustment set by the occupant. In the “Standby” mode, the airflow is reduced to minimal as per safety standards, and the temperature range is increased to 8 °F, and in the “Unoccupied” mode, the temperature range is further increased to 12 °F. The HVAC system remains in the “Unoccupied” on weekends and holidays, and if an occupant were to use a zone during that time, she can express her occupancy by pressing a button on the thermostat. The basement student labs and public circulation area are an exception to the schedule, and are always set to “Occupied” mode.

2.3 BuildingDepot - Data Management Service

Our software applications builds on top of a building information management web service, BuildingDepot (BD) [22], an open source web service framework that integrates information from various sources in a building. A connector for each type of sensor protocol translates vendor specific information to a uniform format, and a RESTful API provides access to sensor data and metadata. Sensor data is associated with tags and metadata such as such as location, type and sensor ID, that describe the properties of the sensor as well as its function in the system. BD supports standardized naming convention, enabling applications to be reused across different buildings [176]. Currently our naming convention is the standard imposed by our university, and it can be easily extended to support standards such as Haystack [1]. BD uses a fast timeseries database using Cassandra and a metadata cache using Redis. MySQL is used for organizing metadata as well as relationships among different sensors. In addition, BD supports services such as authentication, access control, sensor groups and subscription service. Unlike traditional information management systems, BDDepot also supports actuation of control systems.

Each application has to register with BD to gain access to the system. Depending on the permissions provided by the administrator, an application can create/delete sensors, read/write to specific sensors or sensor groups and subscribe to sensor changes. BD has been designed for enterprise level management of buildings, and can be implemented in a distributed manner. For the CSE building, we have implemented BD in a virtual machine running on top of the Xen VMM. The HTTP server is implemented using Nginx as the web server, and uWSGI is used as the interface between the web server and python application, which is implemented with the Flask framework.

2.3.1 BACnet Connector

The BACnet Connector(BC) creates a virtual sensor in BD for each BACnet datapoint in the building. The metadata for the sensors are gathered from BACnet object properties, which include sensor type, location, and BACnet specific ID. The connector polls the sensors which are relevant for HVAC zone control, and posts the value to the BD.

For actuation, the BACnet protocol provides a priority array to resolve contention between applications which send actuation commands to BACnet objects. Our BC is assigned a higher priority over the default BMS schedule for actuation of HVAC zones, and any commands sent by the BC will override the default schedule being used by BMS. BACnet also provides a way to relinquish control, so the system switches back to the default schedule when BC does not control the HVAC system.

We have implemented our BC on a desktop machine, which is registered to the BACnet network as a Foreign Device. The BC server is added to the VLAN dedicated to BMS for controlled access to the BACnet/IP network. The connector has been implemented in C, on top of the open source BACnet Stack [7].

2.4 Data Collection

We collect data from HVAC sensors using the BACnet Connector, and store the timeseries data in BuildingDepot. We have been collecting data for the CSE building since August 2013, and have expanded the collection to all buildings on our university campus since January 2015.

The power meter data at UCSD is maintained on a separate network, using a Schneider Electric proprietary protocol, called ION. We poll the meter data using a separate service. Our building testbed has specially instrumented meters, measuring the temperature and flow of hot and cold water used, accounting for thermal power consumption. Thus, we can accurately measure the savings obtained from various efficiency measures. Power meters at the subcircuit level allows us to measure electric power consumption of the entire HVAC system as well as the lighting, computer room and plug loads subsystems. We also collect data from the local weather stations from Weather Underground.

In the Metasys BMS, only the sensor data and its associated metadata is provided. We studied the architectural plans of our building, and manually extracted useful information for analysis. For each VAV box, we extracted which rooms belong to a particular zone, in which of these rooms the thermostat was located, the area and the usage model (office, kitchen, etc.) of these rooms. We also extracted design specifications for each VAV box, corrected misnamed sensors, informed FM about missing sensors and zones. Further, we created a graphical representation of the floor plans and the system diagrams of the VAV and AHU.

Chapter 3

Occupancy Based HVAC Control Using WiFi Infrastructure

In this Chapter, we present an occupancy inference system that relies on existing WiFi infrastructure already deployed in many commercial buildings. This occupancy information is then used to control the HVAC system for saving energy. This shows that with appropriate software infrastructure, it is possible to leverage existing buiding systems to improve its energy efficiency.

Prior research has shown that most modern buildings use static schedules to run HVAC systems, thereby wasting considerable energy in conditioning unoccupied spaces [18, 58, 64, 65]. Also, as detailed in Chapter 2, HVAC systems in our university run on a static schedule from 6am to 10pm. Furthermore, while modern building HVAC systems use Variable Air Volume(VAV) control, which allows independent control of thermal zones [90], it is not leveraged effectively by facility managers in practice due to the absence of accurate occupancy information within physical spaces. As smart phones, laptops and other WiFi enabled devices are common place today, they can be potentially leveraged to detect occupancy within buildings.

Using occupancy information for HVAC control has in fact been studied extensively [18, 62, 64, 65, 68, 69]. While CO_2 sensors are used to detect occupant density in large spaces [6, 10], the detection times for changes in concentration of CO_2 with occupancy were found to be too slow for use within commercial buildings [69]. Motion sensors used for lighting control in modern buildings are inadequate for HVAC control as they fail to detect relatively stationary occupants [19]. Recent works from Erickson et al. [64, 65] and Agarwal et al. [18] have therefore focused on deploying more accurate occupancy sensors within commercial environments, as well as actuating the HVAC system based on the near real-time occupancy information collected. They estimate that the energy use of HVAC systems can be reduced by 30% to 42% effectively in enterprise-scale buildings.

While these occupancy based HVAC actuation systems are indeed effective in terms of reducing HVAC energy usage, they require deployment of additional occupancy sensors and the design, setup and maintenance of the associated data collection network. To examine the upfront installation cost, Erickson et al. [64] report an expense of $147k for just the hardware for a three floor building, and even simple wireless motion sensors would cost over $120k for our five floor building testbed. Most importantly, the deployment and maintenance hurdles are particularly daunting in case of existing buildings with occupants already inhabiting them. Although wireless sensors help reduce the deployment costs to some extent, recent research has shown that it can be very difficult to deploy and maintain a large-scale wireless sensor network in reality [56, 86]. Challenges on deployment of wireless sensor networks in existing buildings has been elaborated in Chapter 2.

We show that it is possible to implement occupancy based control of HVAC systems by leveraging the information already available in commercial buildings. There is a tradeoff between accuracy of detection, cost of deployment and energy savings. This paper presents one such design point whose effectiveness we have quantified. Specifically, we present the design and implementation of *Sentinel*, a system that utilizes a building's existing WiFi network along with WiFi enabled smartphones carried by occupants of that building to infer occupancy and use that information to actuate the HVAC system. We show that even coarse grained information readily available from enterprise WiFi systems such as the Authentication, Authorization and Accounting (AAA) logs of WiFi clients is sufficient in most cases to determine occupancy of office spaces. In contrast to recent infrastructure based occupancy solutions [78, 125], Sentinel augments the information collected from the AAA WiFi logs with metadata information such as occupant identity, WiFi MAC address and AP location within the building to improve the accuracy of occupancy detection further.

We have implemented Sentinel on top of BuildingDepot [22], a RESTful webservice that interfaces with legacy building management systems, and show that it is scalable and can actuate the HVAC system in our building effectively. We have deployed Sentinel in the Computer Science and Engineering(CSE) building, a 145,000 sqft enterprise-scale building at UC San Diego(UCSD). We show that Sentinel can effectively determine occupancy in office spaces, covering ∼40% of floor space in the CSE building. We demonstrate the feasibility of using WiFi as a sensing solution by observing the usage pattern of smartphones in CSE and studying the WiFi implementation in modern smartphone operating systems. We find that the requirement for continuous WiFi connectivity contradicts the aggressive WiFi sleep algorithms implemented in smartphones, and provide provisional solutions to maintain WiFi connectivity without significant affect on battery life. Based on ground truth occupancy collected for over 10 days we show that Sentinel accurately infers occupancy 86% of the time, with only 6.2% false negative occupancy detections in personal spaces (Actual=Occupied, Inferred=Unoccupied). We highlight the reasons for the inaccuracy, mostly attributed to aggressive power management by smartphones. Finally, we control 23% of the HVAC zones of our test building using Sentinel in a single day experiment, and measure savings of 17.8% in the HVAC electrical energy consumption.

3.1 CSE Building Testbed

Sentinel utilizes several key infrastructures prevalent in modern buildings for occupancy based HVAC control - a Building Management System that allows remote actuation of HVAC system, an enterprise WiFi network and metadata such as occupant office space assignments. We use the CSE building at UCSD as our building testbed whose infrastructure and HVAC system operation are described in detail in Chapter 2.

For occupancy based control, we set the zone to *Occupied* mode when we detect a zone to be occupied, and set it to *Standby* mode otherwise. We chose a shallow setback temperature for our control to reduce any discomfort to the occupants due to misdetection by Sentinel. Prior research has shown that increased energy savings can be achieved by deeper setback temperature and modulation of ventilation rate based on the number of people in a zone [65, 79, 165]. Thus, the energy savings we demonstrate is a conservative estimate of the savings that could be obtained using advanced control methods. In the rest of the paper, we refer to an HVAC zone being turned *On* and *Off*, which is equivalent to the HVAC zone being set to *Occupied* and *Standby* modes respectively.

3.1.1 WiFi Infrastructure

UCSD employs a modern enterprise-class WiFi system to support the 48,000 strong community. The enterprise WiFi network in UCSD consists of three SSIDs, one open network - `UCSD-GUEST`, and two secured networks - `eduroam, UCSD-PROTECTED`. The two secured networks are mostly identical, and henceforth, we refer to them as the *protected* network. The protected network employs WPA2-E/802.1x for encryption, and authorized users login using their Active Directory username and password. It is common in our building, as we will show in Section 3.4.1, for occupants to connect to the *protected* network for regular usage. `UCSD-GUEST`, on the other hand, is generally used by visitors of the campus and is insecure with limited access. We describe the specific details of the WiFi logs collected and used by Sentinel in Section 3.2.2.

3.2 Sentinel: System Design

Our initial goal was to determine the occupancy of each zone in our building using existing infrastructure without requiring additional sensors or installing any software on our occupants phones. Although we do not achieve this goal completely, we show that it is indeed possible to infer occupancy information for approximately half the zones in our building using WiFi network logs with minimal functionality on client devices.

3.2.1 Occupancy Inference Algorithm

The idea of localization using wireless radios is well known [29]. Turner et al. [170] studied the performance of established self calibrating WiFi localization algorithms within the CSE building and found that the median and the 95th-percentile error distance of the algorithms to be worse than simple nearest access point location algorithm. The errors were attributed to signal reflection and RSSI variations with time. The accuracy of indoor localization could be improved with fingerprinting algorithms at the cost of significant manual effort [180] or with use of compute intensive algorithms [45]. For our application, we need to localize up to a thousand people in our building for real-time actuation of HVAC zones. Furthermore, we want to develop an occupancy detection solution that relies on minimal information from the network infrastructure. Therefore, for simplicity and scalability, we concentrate on easily obtainable coarse-grained location of client devices, without employing complex localization techniques that may be more accurate. Thus, when a client sends a packet to an access point(AP), we assume that the client is located in a zone within the range of the AP. We show, with the occupancy

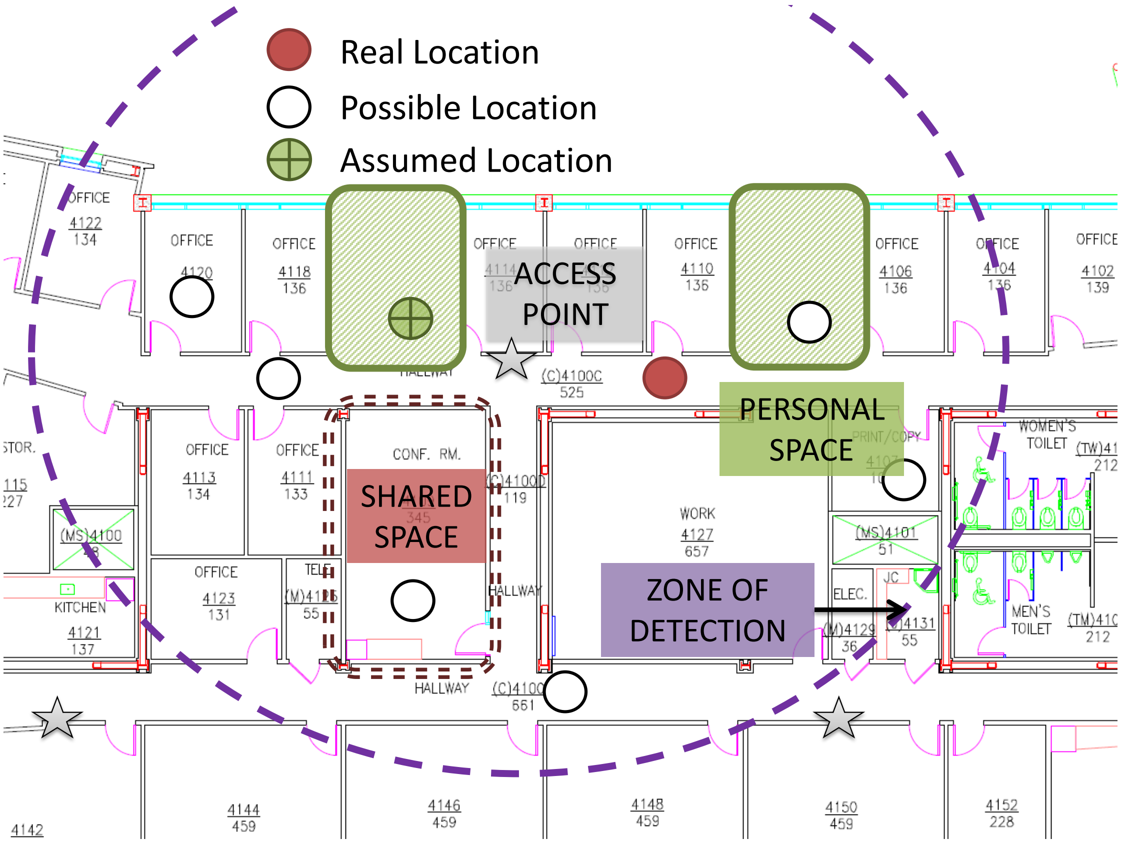

Figure 3.1. Example of occupancy inference using WiFi connectivity. The occupant is assumed to be in her personal space whenever she is within the associated AP's zone of detection, as denoted by "Assumed Location".

model described below, that it is possible to make inferences about the occupancy of users in the building even with such coarse-grained information.

Personal and Shared Spaces

We classify physical spaces into two categories: *personal spaces* and *shared spaces*. We define a *personal space* as an area with a designated owner such as individual offices assigned to faculty, or desks assigned to students in a lab. There is no restriction on the size or type of a personal space, so it includes single person offices, cubicle spaces and rooms shared by multiple people. *Shared spaces* on the other hand includes the rest of the building, which essentially have no designated occupant or owner such as restrooms, conference rooms, cafeteria, etc.

Consider an occupant with a WiFi enabled device located within the building as depicted in Figure 3.1. As the device is associated with one of the access points(APs) in the building, it can be located anywhere in the range of the AP. The occupant could be in her office, or visiting a colleague's office, or in a shared space. We assume that the occupant does not visit a colleague's office unless the colleague herself is present in the office. Thus, a personal space cannot be occupied unless the owner is present in the space. If we can detect the presence of owners in their respective personal spaces, then we can effectively monitor the occupancy of all personal spaces in the building.

Shared spaces, on the other hand, can be visited by anyone in the building without restriction. Thus, for inferring occupancy of shared spaces correctly, we would need to detect the entry and exit of **each** person in the shared space accurately. Since we do not employ fine-grained localization information, we **do not** aim to detect occupancy in shared spaces and assume that they are always occupied.

Note that one person can be allocated to more than one personal space, and any number of personal spaces can exist within an HVAC zone. Thus, the personal and shared space division can be applied to a wide variety of buildings and occupancy patterns.

Zone of Detection

We refer to the physical area covered by a WiFi Access Point (AP) as its *zone of detection*. An AP is *affiliated* with a personal space, if the personal space falls within an AP's zone of detection. There can be multiple APs affiliated to a personal space. If the owner of the personal space is connected to an affiliated AP, then she is considered to be present in the personal space.

Smaller zones of detection will naturally lead to more precise occupancy inferences, while larger zones of detection causes loss in accuracy. In our building, we found that the zone of detection of an AP typically covers up to 10 HVAC zones. This lack of precision means that we sacrifice potential energy savings when an occupant is just outside their personal space but inside the zone of detection. Note that even WiFi localization methods will not help as the 95th-percentile error distance from AP was found to be **worse** than the nearest AP algorithm [170]. Figure 3.1 gives an example of occupancy inference of an occupant who is within the zone of detection of an AP. In this case, the occupant is assumed to be in her office irrespective of their actual location within that zone. This assumption resolves the discrepancy between the area covered by zone of detection of APs and HVAC zones.

We conservatively mark the boundaries of zone of detection of each AP as well beyond the points at which a typical client handoff takes place. We also assume there is no cross floor interference between the AP coverage as it was never observed in practice. For our building, each personal space was associated with at the most four APs. Figure 3.2 shows an example of the personal spaces associated with one of the APs in the building.

Identity

When the WiFi logs indicate that a client device is connected to a particular AP, we infer that the client is within the AP's zone of detection. In order to make a relation with the personal spaces within the zone of detection of the AP, the client needs to be mapped to the owner of her personal space. Therefore, an accurate mapping of owners to personal space, i.e. occupant to office number, has to be maintained by our system. Further, information of all wireless capable devices used by a building occupant also has to be maintained. As we are using the AAA logs from the WiFi network for inferring occupancy, the wireless device to actual building occupant mapping is available to us.

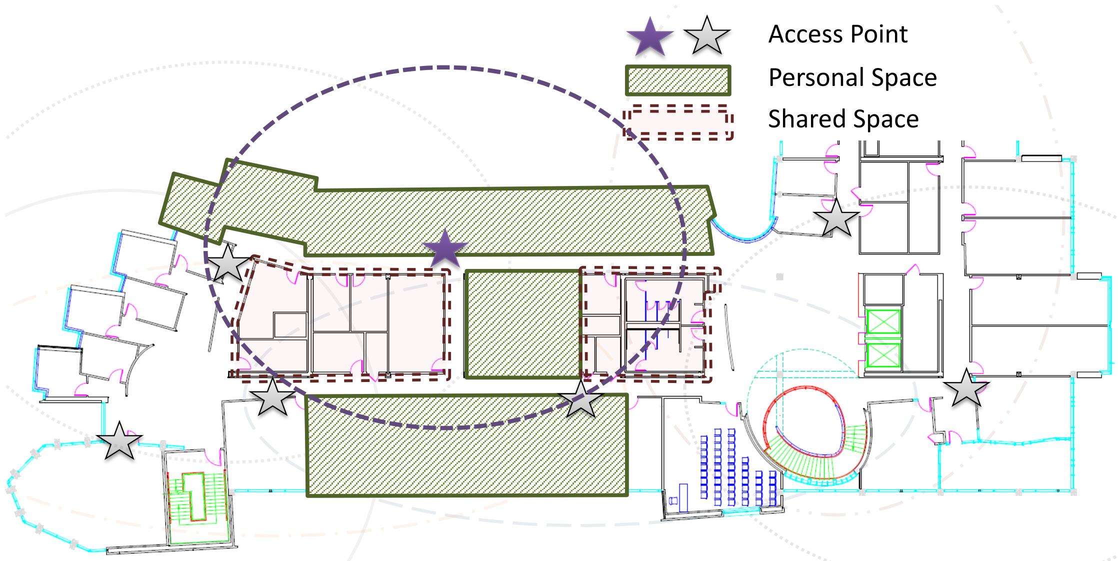

Figure 3.2. Example of an AP with its associated personal spaces. The network coverage of APs are marked conservatively to reduce false negative errors.

3.2.2 Capturing WiFi Data

We use AAA logs from the WiFi network to collect relevant information from the occupant devices. AAA logs only collect the connection, disconnection and periodic live packets from the client devices, which provides us with enough information for occupancy inference. An alternative is to collect data at the AP level and process each packet sent by the device. However, the additional information does not help to improve the accuracy of detection as we show in Section 3.2.4, but increases the burden of data processing by several orders of magnitude and also intrudes on the privacy of the occupants.

We use the requests received by the RADIUS server as part of the WPA2/802.1x protocol for acquiring information on the WiFi devices in CSE. A WiFi device sends an authentication request to the AP when it first tries to make a connection. The AP forwards the request to the RADIUS server, which has information on the client MAC address, the AP MAC address, the SSID to which connection was requested for, as well as the client username and password. After successful authentication, the AP sends an accounting packet indicating the "Start" of the connection to the server.

Similar authentication and accounting packets are sent to the RADIUS server when a client migrates from one AP to another in the same network, and when the client disconnects from the network. In addition, the AP sends "Alive" accounting packets to indicate the client is still connected to the network. If the AP does not hear from the client for a fixed period of time (1000 seconds in our network), it terminates the connection with the client and sends a "Stop" accounting packet to the RADIUS server.

When the RADIUS packets indicate that the client has connected to one of the APs near the personal space of the occupant, then Sentinel marks that personal space as occupied. When the client migrates to APs in other areas of the building, or gets disconnected from the network, Sentinel marks that personal space as unoccupied.

3.2.3 Phone Detection Algorithm

There are many WiFi enabled devices popular today - laptops, smartphones, tablets, and it is possible that a building occupant owns more than one WiFi device. When the occupant is moving in and out of her personal space, she may not carry all her WiFi devices. For accurate inference of occupancy, it is important that the system knows the MAC address of the device which is representative of the current location of the occupant. For most occupants in our building, this WiFi device was their smartphone, and henceforth, we refer to the *phone* as the location representative device.

The RADIUS server gets a packet when a client migrates from one AP to another. When an occupant is moving inside the building, the phone gets handed-off between many APs. Over a period of time, the phone would send more number of requests of authentication to the RADIUS server than other devices. Thus, we mark the device with the highest number of requests to be the occupant's phone.

The algorithm fails when an occupant buys a new phone. As the new phone starts off with zero requests, it would be ignored even if it best represents the location of the user. Such an event cannot be ignored at the scale of a thousand occupants, as there could always be a few occupants who have a new device.If we do not see any access request from the device with highest number of requests for 48 hours, we reset the number of requests of all device owned by the occupant. The 48 hour resets also increased the robustness of the system to the changing usage patterns of the occupant.

We verified the accuracy of the algorithm by identifying the MAC addresses used by Sentinel for changing the occupancy status of a personal space. 44 occupants were chosen at random for manual verification, and for 40 of them, the phones were identified correctly. The algorithm worked well for all types of devices despite the aggressive WiFi sleep policies employed (Section 3.2.4).

We found that Mac OS X devices connected and disconnected from the WiFi network despite being put to sleep mode. Thus, when a Mac OS X computer is left in sleep mode over a weekend, the number of access requests of the computer exceeds those of the occupant's phone, and our system detects the room as occupied. We observed this on four occasions during our experiments, and it can be avoided by incorporating the unique number of access points connected to by a device into the algorithm.

3.2.4 Perpetual WiFi Connectivity

Sentinel assumes that the phone is continuously connected to the protected network when the occupant is in the building. However, this may not happen in practice because of various reasons - the occupant may not own a smartphone, the occupant may have forgotten her phone at home, the phone may run out of battery, WiFi network coverage may be poor within the office, or there may be a network outage. These problems are associated with any system which seeks to use WiFi clients as a sensor, and we do not handle them as part of this work. If the entire building is affected, Sentinel falls back to the default schedule. If an individual occupant is affected, alternate means of informing the occupancy of an HVAC zone can be provided. In Sentinel, the occupants indicate their presence by pressing a button on the thermostat. We also provide a web interface for indicating user occupancy and preference, similar to the personalized building control system developed by Krioukov et al. [108].

With smart devices permeating every part of our lives, we hope that WiFi connectivity will become part of the essential infrastructure provided in commercial buildings, and the connectivity issues would become a rare event in a few years. Further, as offices typically have abundant power supply, we assume that the occupant would connect the phone to a charger once it indicates low battery. However, battery powered smartphones employ a number of power saving strategies, and the specifics of WiFi sleep algorithm depend on the type of operating system and the model of the device.

We consider three popular variants of smartphones - Android, iOS and Windows Phone. Both Android and Windows Phone provide options for WiFi power management when the device is in sleep mode, and the user can opt to keep the WiFi radio awake even when the device is not in active use. iOS, on the other hand, employs aggressive sleep algorithm as soon as the screen is locked. On studying the network traces of a WiFi-only iPad2 using iOS v6.1.2, we observed that when the device screen is locked, it only keeps the TCP port to Apple Push Notification Service open, and does not respond to other network packets. When the device does not get a push notification for a period of time, the WiFi radio is turned off and woken up at 30 minute intervals. In order to avoid errors in occupancy detection, we request the occupants of the building to change their settings to fetch mail every 15 minutes, thereby ensuring that we get some information coming from them over WiFi.

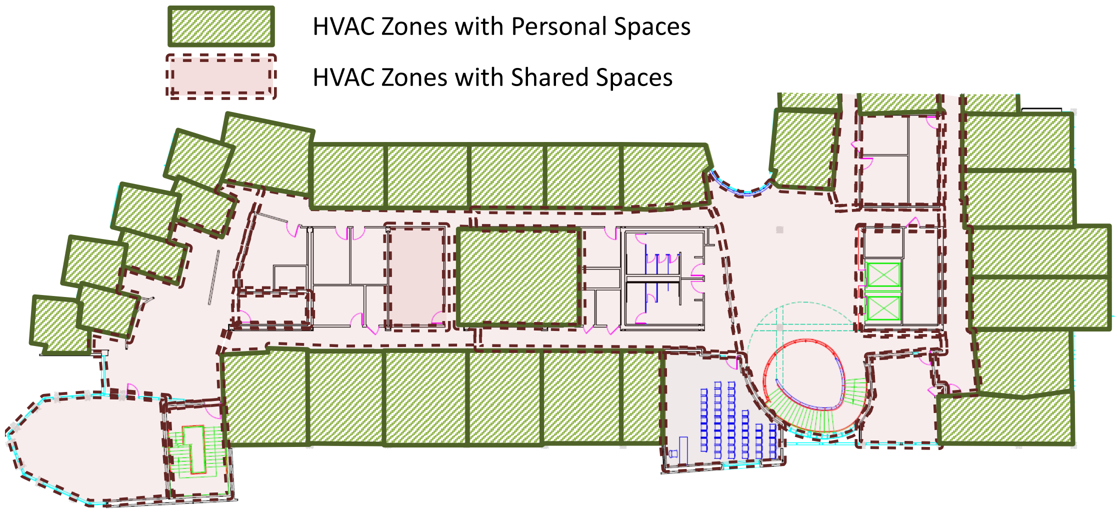

Figure 3.3. Partitioning of one wing of a floor based on shared and personal spaces. Personal spaces which have a common zone with shared spaces are marked as shared.

3.2.5 Partitioning the Building

As we explained in Section 3.2.1, we need to divide the building into personal and shared spaces. As Sentinel can only infer occupancy of personal spaces, the energy savings obtained are lower than when actuating entire building HVAC based on occupancy.

In our building, personal space consists of single room offices and multi-person shared offices. The shared space consists of computer labs, cafeteria, conference rooms, etc. In addition, there are storage rooms that are rarely visited, and we mark them as unoccupied for actuation. The HVAC zones in the building, however, do not follow the personal and shared space partitioning. For example, there are several zones which condition a personal space as well as the hallway connected to it. As Sentinel needs to run shared spaces in static schedule, the personal spaces which share its HVAC zone with a hallway or lobby are marked as shared spaces as well. Figure 3.3 shows an example of shared and personal zone mapping for a section of our building.

Table 3.1 shows the area covered by each kind of space in our building. Some

Table 3.1. Contribution of personal and shared building spaces by area and by HVAC power consumption. Actuating only personal spaces can lead to at most ~33% electricity savings.

	Area	Electrical	Cooling	Heating
Personal	37.5%	63.9%	96.0%	108.0%
Shared	58.3%	66.9%	96.4%	90.0%
Storage	4.2%	-	-	-

of the shared spaces like staircases and small hallways are not covered by HVAC zones. Hence, the HVAC power consumption of personal and shared spaces is not proportional to the area covered. To measure the contribution of each type of space to the total HVAC power consumption, we operated the HVAC system with all the zones turned on for one hour, then turned off all the personal spaces for two hours, then switched the personal spaces back on, and finally, turned off all the shared spaces for two hours. We conducted this experiment overnight, as the outdoor temperature is stable at San Diego. On the night of the experiment - March 20, 2013, the outdoor temperature was at 61±1.7°F.

Table 1 shows the electrical and thermal energy savings obtained turning off shared and personal spaces. The personal and shared spaces contributed 63.9% and 66.9% to the electrical power consumption respectively. Thus, the personal spaces contribute to roughly half of the total HVAC electricity consumption. As the shared spaces remain conditioned in our system, the electrical power savings we can obtain by occupancy based conditioning of personal spaces is ~33% for our building. The heating and cooling thermal power consumption do not follow similar trends, and we examine them in detail in Section 3.4.6.

3.3 Implementation

Sentinel's system architecture follows the principles proposed for management of sensors in commercial buildings in recent literature [26, 57, 58, 151]. Figure 3.4 provides

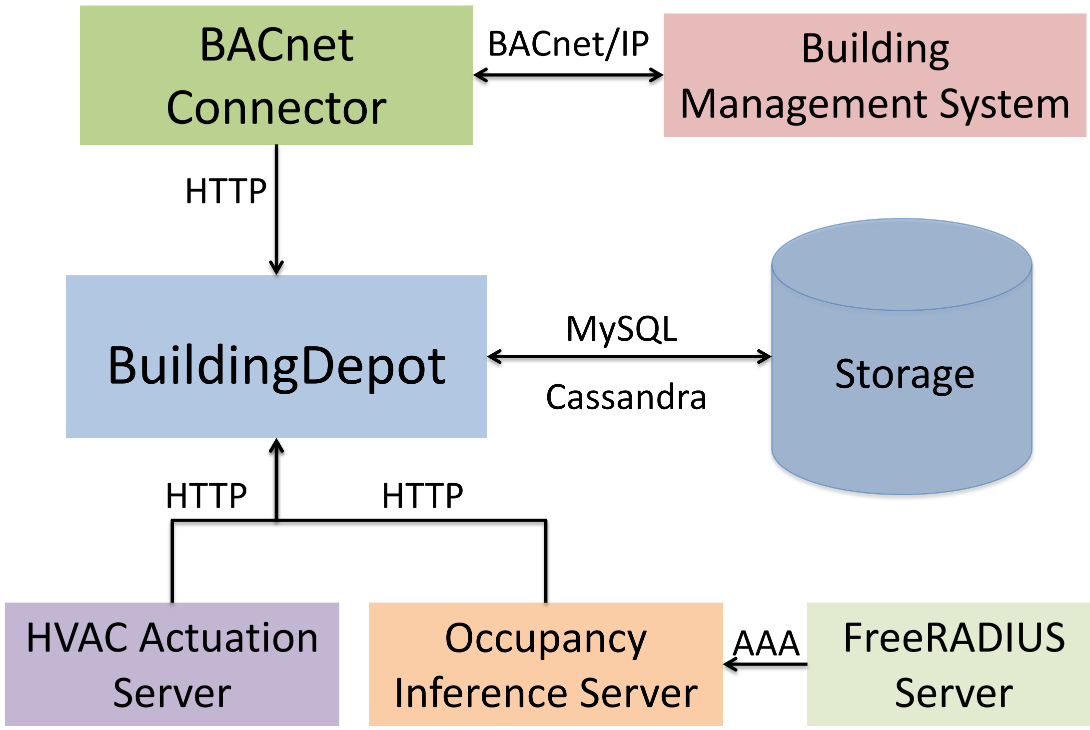

Figure 3.4. System Architecture of Sentinel

an overview of Sentinel. BuildingDepot(BD) [22] acts as a central authority for collection of sensory data of the building and provides access control to users and applications for analyzing sensor data and controlling the building actuators. The BACnet Connector acts as a gateway between the sensors which use the BACnet protocol and BD. Both BACnet Connector and BD are described in detail in Chapter 2. The Occupancy Inference Server receives a copy of the packets received at the RADIUS server, and processes the packets to infer occupancy for the various HVAC zones in the building. The HVAC Actuation Server processes this occupancy information, and actuates the HVAC system.

3.3.1 Occupancy Inference Server

The Occupancy Inference Server(OIS) receives a copy of each RADIUS packet sent by the APs in CSE building. OIS processes the incoming packets to infer personal space occupancy as described in Section 3.2.1.

For inferring occupancy, the OIS maintains several metadata information - a mapping between occupant to their phone MAC address, between the occupants and their office numbers, between offices and the APs in the building, and finally, a mapping between HVAC zones and offices. OIS creates a virtual sensor in BD for indicating occupancy of each HVAC zone in the building, and key information from each incoming packet is stored in a local MySQL database for debugging and future analysis. The usernames are anonymized in the database for preserving the privacy of the occupants.

Several levels of checks need to be made before deciding that an HVAC zone is occupied or not. The incoming packets are filtered for the registered occupants of the building, and then checked if the packets are coming from a "phone"(Section 3.2.3). If the phone is connected to an AP near the office of the owner, the corresponding personal space is marked as occupied, and otherwise, its marked as unoccupied. If all the other personal spaces in the same HVAC zone is unoccupied, the occupancy status of the zone is updated to occupied and the information is sent to BD.

We have implemented the OIS on top of an open source RADIUS client - `pyrad` [12].

3.3.2 HVAC Actuation Server

The HVAC Actuation Server(HAS) acts as a layer of abstraction between the occupancy information supplied by OIS and the HVAC control using BACnet. During normal operation, HAS converts the occupancy changes from the OIS to the appropriate commands for HVAC control. HAS was also used for experiments on HVAC control which we describe in Section 3.4.6.

Currently, we control the HVAC system in a reactive manner, i.e., we control the ventilation of a zone when its occupancy changes. Literature has shown that predictive control with deep setpoints can lead to higher energy savings in HVAC systems [27, 65, 79, 134]. However, the setback temperature setpoints allowed in our building are conservative, and the temperature of unoccupied zones is kept within the range of 70°F to 78°F. Goyal et al. [79] find that the energy savings obtained by both predictive and reactive systems are similar when the setback temperature setpoints are set as per the ASHRAE standard. They also show that reactive systems have negligible effect on the comfort of the occupants as the setback temperature setpoints are conservative. Sentinel is not restricted to reactive control, and we will explore model predictive control as part of our future work.

3.4 Evaluation

Sentinel has been operational for three weeks at the time of writing this paper, in the five floor, 145,000 sqft CSE building at UCSD. To show the feasibility of a building-wide deployment of Sentinel, we show the distribution of smartphone usage in the building. We evaluate the accuracy of occupancy detection using Sentinel over a period of 10 days. We then show the occupancy patterns of 38 smartphone users in our building across a week, and identify periods of inoccupancy which could save HVAC energy. We have run over 35 experiments on the HVAC system in our building testbed, and present the HVAC power consumption versus occupancy trends to demonstrate the potential energy savings using an occupancy based HVAC actuation system. Finally, we present the energy savings obtained by controlling 55 of the 237 HVAC zones in the building for one day.

3.4.1 User Study

We surveyed 187 of the 415 registered occupants in our building. The surveys were short, intended to garner interest in WiFi based control technology. We asked the

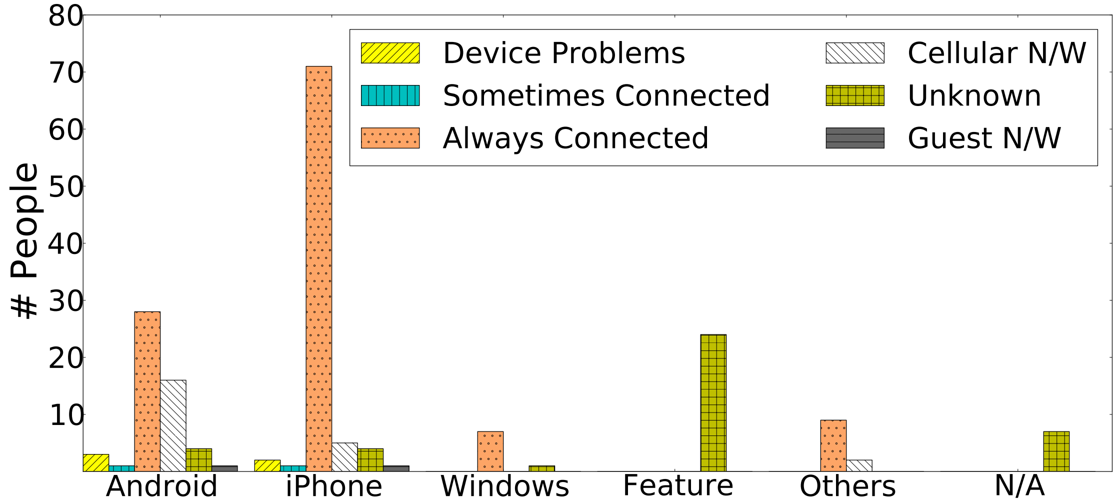

Figure 3.5. Distribution of smartphones and their WiFi usage patterns by the occupants in our building.

occupants if they would be interested in using such a technology, the kind of smartphone they use, whether they connected their smartphone to the protected WiFi network in the building on a regular basis, and if they would participate in WiFi based actuation of HVAC system in their office space.

Majority of the occupants surveyed showed interest in controlling the HVAC system based on WiFi connectivity. Over 64% of the occupants owned a smartphone, and only 10% of the occupants did not connect to the internet using WiFi. Figure 3.5 shows the usage trend of the WiFi devices in the building. Despite the prevalence of WiFi devices and network coverage across the building, many people reported that they did not connect to WiFi due to various reasons - poor WiFi coverage in their offices, adequate data capacity available from cellular network, connectivity problems with the WPA2/802.1x protocol and battery problems.

It should be noted that there is little incentive for occupants of the building to stay connected to WiFi using smartphones in an IT building. Most of the occupants have a desktop computer with ethernet, and many occupants use their laptop for internet connectivity. Several occupants indicated that they would connect to WiFi using their phone if it provided automated control of HVAC system without significant effect on battery life. Problems with network coverage can be solved by careful placement of APs within the building, and device connectivity issues would get solved over time by software/hardware updates to the smartphones. There would always be a few occupants who do not, or cannot connect to the protected network for various reasons. In our experiments, occupants need to indicate their presence by manual press of a button on the thermostat as on weekends. We later added a web based control of access to HVAC system similar to that proposed by Krioukov et al. [108] as a failsafe option.

3.4.2 Occupancy Accuracy

Accuracy of detecting occupancy using WiFi connectivity has been shown to be noisy and inaccurate in prior work [78, 125, 168]. However, by restricting the occupancy detection of Sentinel to personal spaces, and by using additional metadata information like occupant identity and AP location, Sentinel improves the overall accuracy of occupancy detection significantly. We demonstrate the accuracy of Sentinel based on data collected for 116 of the 415 building occupants over a 10 day period.

57% of the smartphones used by the building occupants are iPhones, and as explained in Section 3.2.4, iOS devices turn off the WiFi radio when it is not in active use. To participate in WiFi based HVAC control experiments, we requested occupants to keep their iOS device connected to WiFi and to change device settings to fetch emails every 15 minutes. We requested the Android and Windows Phone users to enable WiFi and to change the settings to disable the WiFi aggressive sleep option. The change in device settings were enforced for two days, and the occupants were given the option to change back to their default settings if needed.

We define an *event* as a change in occupancy of a personal space, either as detected by Sentinel, or as seen in ground truth measurements. We use the number of events correctly identified by Sentinel as a measure of the occupancy accuracy. If Sentinel incorrectly marks a personal space as occupied, we classify the error as a *false positive*, and if the system incorrectly marks a personal space to be unoccupied, we classify it as a *false negative*. On a false positive error, we incur a penalty in the energy savings obtained as the HVAC system would ventilate the personal space unnecessarily. A false negative, on the other hand, would lead to discomfort to the occupants as the HVAC system would be put to "Standby" mode. For ground truth comparison, we note the occupancy in each office across the building, and compare Sentinel logs for occupancy status at the corresponding timestamp. We also inspect the latest logs from Sentinel, and examine the occupancy status of the respective zones. In case of discrepancy, we try our best to identify the underlying cause. We ignore the errors that occur when the occupant leaves her personal space for less than five minutes, and since the timeout period in RADIUS protocol is $\sim$ 17 minutes, we accept a delay of up to 20 minutes in detection when the occupant is leaving her personal space.

We measured 436 events during the 10 test days, of which 330 events were recorded in the first two days, and Sentinel accurately identified personal space occupancy 83% of the time. The false positives and the false negatives were 9.4% and 7.5% respectively. After the first two days, the ground truth was collected only for occupants known to be still using the modified phone settings. Figure 3.6 gives a breakdown of the causes of the errors in detection.

Majority of the false positive errors by Sentinel were caused due to an error in identifying the appropriate device by the phone detection algorithm. As many of the occupants were enabling their WiFi devices for our experiments, we reset the access count request of all the recorded occupant devices. As this was done early in the morning, all the WiFi enabled devices in the building were identified as phones by Sentinel, and the errors in detection increased. The phone detection algorithm corrected itself as occupants came in, and the incorrect device errors died down by midday.

System errors constitute the errors caused due to mistakes in metadata information stored in Sentinel. Some of the errors included incorrect mapping of the occupant to their personal space, incorrect authentication username, and incorrect mapping of APs to personal spaces. We corrected the errors after the first day of ground truth data collection. If we remove the temporary errors caused due to incorrect device detection and system configuration errors, the accuracy of Sentinel improves to 86%, and the false negative errors reduce to 6.2%.

The aggressive WiFi sleep mechanism used in iOS devices led to intermittent WiFi connectivity, and was the cause of majority of the false negative errors. Although, we used various mechanisms to keep the WiFi radio active, there were still circumstances in which the connectivity was not persistent. “iOS Start” errors indicate that the occupant has entered her personal space, but Sentinel could not detect the occupant as the iOS device did not switch from the cellular data network to the WiFi network. We noticed a maximum delay of 23 minutes in iOS Start errors. “iOS Stop” errors occur when the iOS device turned off the WiFi radio when the occupant was in her personal space. This behavior was observed among phones which were not in use for a long period of time, and as much as 3 hour periods of disconnection were observed. However, on most cases, the iOS devices woke up within 10 minutes of timeout. The device errors were mainly caused due to late detection of arrival of occupants in to their personal spaces. The late detection was observed among the Android devices, as it sometimes took longer than usual to detect WiFi networks in its vicinity. The inaccuracies due to device connectivity constitute 5% of the error and can be improved by the use of an app on the phones. We provide the details of an iOS app which addresses this issue in Section 3.5.

When the occupant has left her personal space, but is still within the zone of

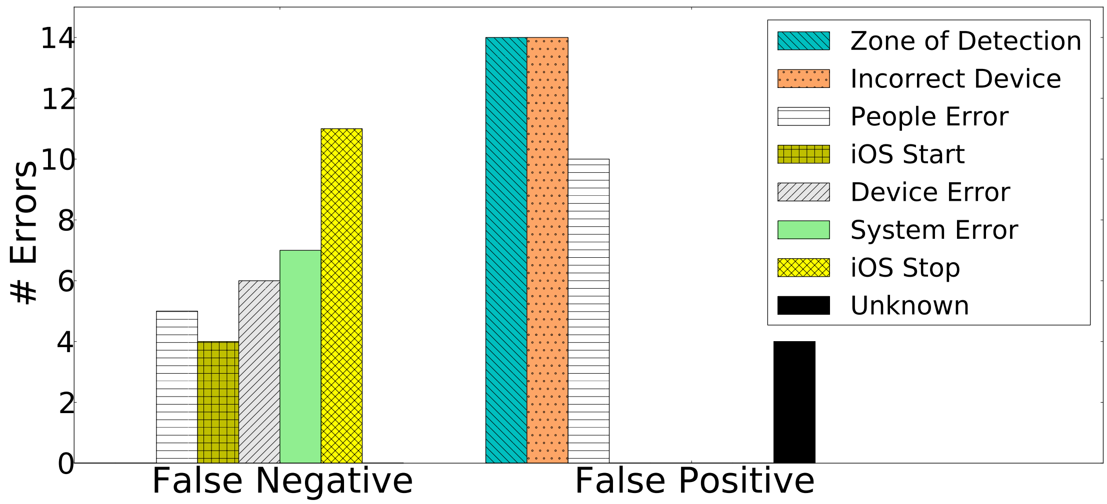

Figure 3.6. Distribution of occupancy detection errors as observed over 436 events and 10 days. The occupancy detection was accurate 83% of the time.

detection of the nearby AP, Sentinel incorrectly marks the space as occupied. We call such false positives as “zone of detection error”. A similar false positive is incurred when occupant leaves her personal space but does not carry her phone with her. We classify such error under “people error”. Occupants also sometimes forgot to enable WiFi on their phones, or connected it to the guest network, which leads to false negatives. We classify such errors as people error as well. Both zone of detection and people errors account for 6.9% error in occupancy detection, and are inherent to the occupancy inference algorithm used by Sentinel. People errors can only be reduced using wearable devices, and zone of detection errors can be reduced using accurate localization methods.

3.4.3 Occupancy Trends

We have collected the occupancy information inferred from the RADIUS logs for all the occupants for three weeks at the time of writing this paper. Figure 3.7 shows the occupancy of the 38 users who are always connected to the protected network, and have disabled WiFi sleep by default. The occupancy trend is shown for the week of March 18 to March 24, 2013 - one of the busier weeks in our building due to exams. Note that

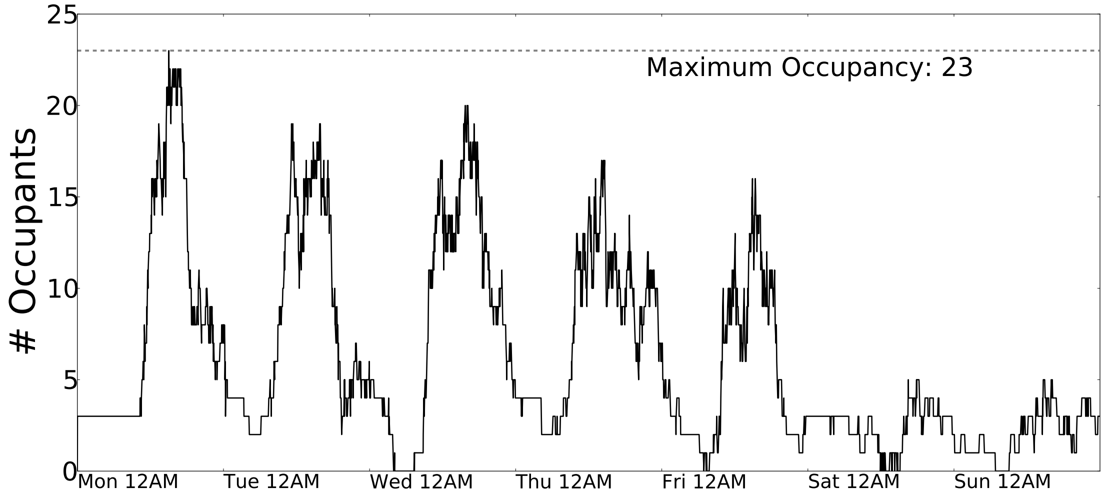

Figure 3.7. Occupancy trends of 38 occupants in our building who keep their smartphones always connected to WiFi as measured by Sentinel for the week of March 18-24, 2013

occupancy here refers to occupancy of personal spaces, rather than the whole building.

The most interesting part of Figure 3.7 is that the peak of the graph is at 23 people, only 57% of the maximum 38. Another point of interest is that the general occupancy decreases as the week progresses, indicating peak of productivity on Monday, and a maximum of just 15 people on Friday.

On most days, there is a fall in the occupancy during the middle of the day, indicating people leaving their offices for lunch, meetings and discussions. The graph clearly demonstrates the opportunity of energy savings that could be obtained by controlling the HVAC system based on occupancy.

On nights and weekends, the occupancy is understandably low, however it is not zero, as assumed by the static schedules used for HVAC control. The occupants are left to manually indicate their presence if they are in the building during off hours. WiFi based occupancy detection can easily detect the presence within an HVAC zone, and provide automated thermal comfort to the occupants.

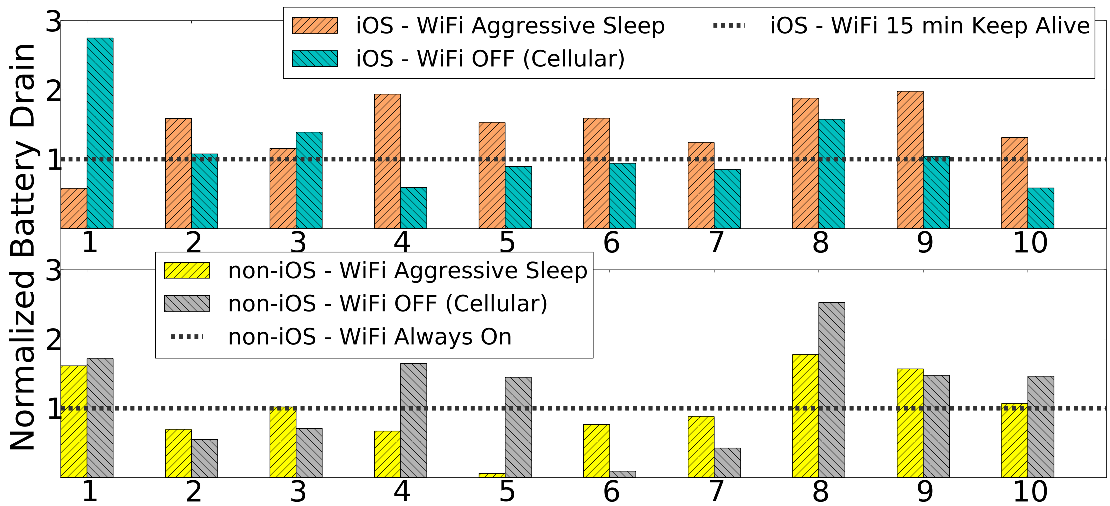

Figure 3.8. Distribution of smartphone battery consumption of 20 participants over 3 days with WiFi "always on", with WiFi aggressive sleep enabled and with WiFi off.

3.4.4 Impact on Device Battery Life

Battery life of a device is dependent on the WiFi radio chip, the network coverage, the applications using WiFi, the usage pattern and potentially other factors. Prior works on WiFi and cellular radio power measurements [32, 43, 182] indicate that WiFi sleep power is about 2x the sleep power of cellular technologies such as 3G, the data transmission in WiFi is about an order of magnitude more efficient than cellular, the energy spent by WiFi radio to scan and associate to an AP is 5x the energy spent for 50KB data transfer, and the energy consumption of cellular radio varies significantly with signal strength. To reduce the impact of higher WiFi sleep power, Rahmati et al. [147] and Agarwal et al. [21] suggest waking up WiFi only when data transfer is required, and iOS follows a similar model based on our observations (Section 3.2.4). However, this strategy may not lead to power savings if the phone keeps switching between WiFi and cellular radios frequently or if the apps installed on the phone require frequent data transfer. Thus, the impact on battery life would actually depend on the usage pattern of the phone.

Instead of measuring battery consumption in a controlled environment, we measure battery drain as seen by phone owners during their regular usage. We choose 20 participants, not necessarily building occupants, and measure their smartphone battery performance over three days. There were 10 iPhones, 9 Androids and 1 Windows Phone in the collection. On the first day, the smartphones were put to WiFi "always on" mode, by disabling the sleep mode in non-iOS phones, and fetching email every 15 minutes in iOS phones. On the second day, WiFi was enabled, with aggressive sleep mode enabled. On the third day, WiFi was switched off completely. The participants were requested to try and keep similar usage pattern across these three days and report any significant differences in usage. We normalize the battery drain during three days by the battery drain observed with WiFi "always on" option, and the combined result is shown in Figure 3.8.

As can be observed from Figure 3.8, there are no clear trends across the three WiFi modes for these devices. However, we do make several observations. First, in many cases (particularly for iOS devices) WiFi aggressive sleep leads to lower battery lifetime than keeping WiFi on probably due to the constant mode switches. Second, turning the WiFi off completely to use only 3G does not lead to significantly better battery life as compared to keeping WiFi on, or the 15-min Keep Alives mode for iOS. The Android device for which this is not the case (Device 6) were verified to be an anomaly since the user reported that they don't use the 3G data radio. Therefore, based on our current data, we have not seen conclusive evidence whether using the aggressive sleep modes for WiFi actually provides significant battery life improvements than the less aggressive WiFi on settings. However, given the variations we observed in battery consumption more extensive data collection would provide better insight into the effect on battery life due to continuous WiFi connectivity.

Table 3.2. Breakdown of latency of Sentinel from the time of reception of RADIUS packet from WiFi device to the time of sending actuation commands to HVAC.

Operation	**Latency (in ms)**
OIS $\rightarrow$ BD	194.26 ± 50.6
BD $\rightarrow$ HAS	67.18 ± 13.6
HAS $\rightarrow$ BD	158.25 ± 61.3
BD $\leftrightarrow$ BC	185.35 ± 113.4
BD $\rightarrow$ HAS	126.35 ± 30.6
Total	$\mathbf{731.57 \pm 125.4}$

3.4.5 Actuation Latency

Unlike prior occupancy based control systems [18, 64], we have implemented Sentinel on top of RESTful web services as recommended in recent literature [58, 26] using our BuildingDepot(BD) system [22]. BD is designed to support different types of building applications, is compatible with existing building management solutions and scales well with number of users, applications and sensors. Similar RESTful frameworks are also being adopted by industry and academia for building automation applications such as plug level energy meter [9, 94] and wireless lighting system [11]. Sentinel is one of the first RESTful systems to be deployed at the scale of an enterprise-scale commercial building, and actuaion latencies for such systems have not been measured in the literature so far.

Table 3.2 provides a detailed breakdown of latency to send an actuation command to the HVAC zones, from the time of detection of occupancy to the time to get the acknowledgment of the command completion. We have an actuation latency of $\sim$750ms, which is fast enough for actuating HVAC systems. However, when the access controls extend to plug loads and lighting systems, the actuation latency would need to be reduced further so that the occupants do not notice the delay.

3.4.6 Potential Energy Savings

Prior work has focused on estimating the energy savings obtained by occupancy based actuation of HVAC system using simulations on calibrated EnergyPlus building models [52], and it has been shown that significant savings can be obtained across different seasons and geographical locations [64, 65, 79]. Goyal et al. [79] show that the amount of energy savings obtained remain almost the same for both reactive and predictive strategies for different outdoor conditions if the set back temperatures are conservative as per ASHRAE standards.

Instead of simulations, we measure the actual energy savings obtained at different levels of occupancy by conducting experiments directly on our building. We perform our experiments during night time, as there are only a few people present in the building, and the night temperature at San Diego was relatively stable at the time of our experiments. All the experiments were conducted during the month of March, 2013, when the night temperature was recorded between 55 °F and 60 °F. Note that compared to the day, the load on the HVAC system during night is lower due to reduced outdoor temperature and lesser number of people and machines in operation. The energy savings measured represent a constant load HVAC system, and is a conservative estimate of actual energy savings possible.

To determine the energy savings obtained with change in occupancy in the building, we randomly choose a fixed percentage of HVAC zones, and turn them off for a period of two hours. To allow for variations with respect to outdoor conditions, we choose the same set of zones, and repeat the experiment. Figure 3.9 shows the electricity consumption of the HVAC system when we actuated 25% of the zones in the building. The experiment was started at $\sim$10pm, and all the zones in the building were gradually turned on with an interval of 10 seconds between each actuation command. The zones

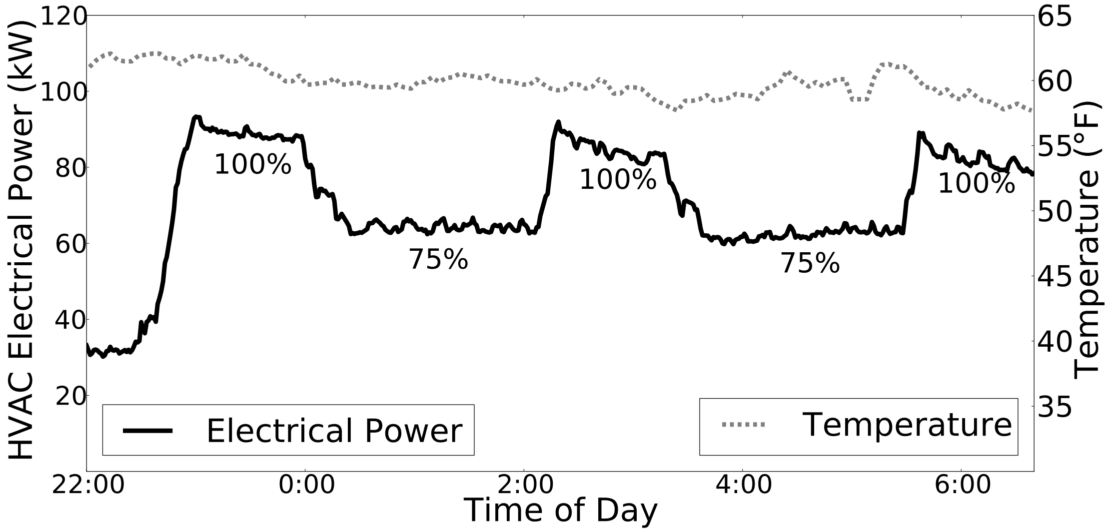

Figure 3.9. Measurement of HVAC electrical power consumption with 25% of the HVAC zones randomly chosen to be alternatively turned on and off on the night of March 16, 2013.

were allowed to stabilize for an hour, and then 25% of the zones were gradually turned off for a period of two hours. We turn back on the switched off zones after two hours, and repeat the process once more. We repeat the experiment for at least two nights for each level of occupancy.

Figure 3.10 shows the changes in electrical power consumption of the HVAC system with increase in occupancy of the building. There is a clear increase in the electrical power consumption as the occupancy of the building increases. Although its not prominent in the figure shown, the drop in electrical energy is not directly proportional to the fall in occupancy within the building. The electrical power consumption is dominated by the fans in the Air Handler Unit(AHU) of the building, and power consumption of the fans are proportional to the cube of the fan rotation speed. Thus, as the occupancy of the building increases, the fan rotates at a higher speed, leading to disproportional increase in power. Thus, the energy savings are maximum when the occupancy of the building drops from 100%, and follows the pattern of diminishing returns as the occupancy further

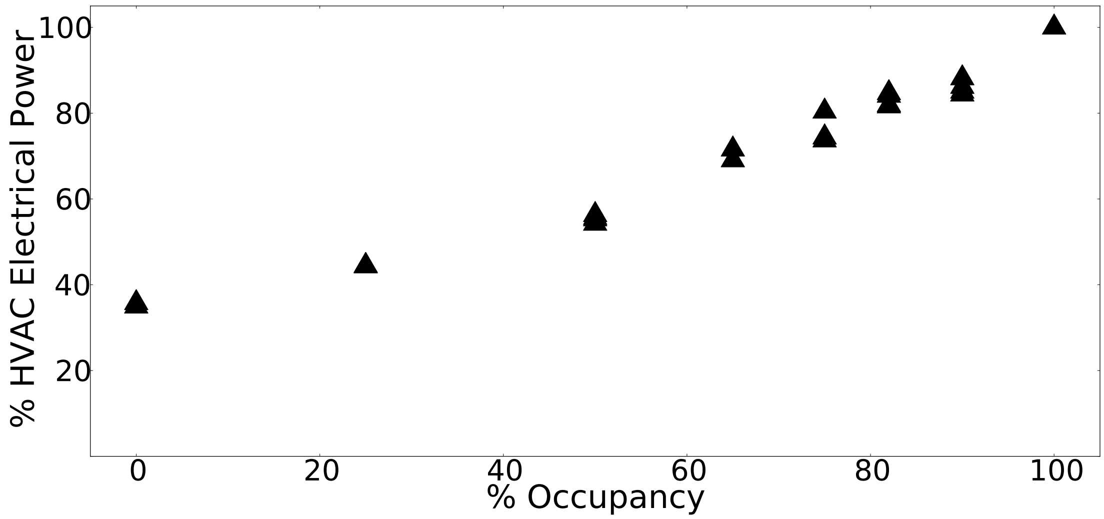

Figure 3.10. HVAC electrical power consumption with change in occupancy levels in the building.

reduces.

Figure 3.11 shows the thermal power consumption of the HVAC system with increase in occupancy. Both cooling and heating thermal power decrease gradually with decrease in occupancy of the building. The trends in heating thermal power is not as clear as cooling thermal power or electrical power because the supplied hot water is not in continuous use by the HVAC system. The cold water is converted to cold air, and is used for ventilation by the VAV boxes. The amount of cold air is regulated by the VAV box using a damper, but a minimum amount of ventilation is maintained by the VAV even when the zone is unoccupied. Hot water, on the other hand, is used intermittently by the VAV box to reheat the cold air when needed. The intermittent usage of hot water translates to different heating thermal power consumption from day to day, and thus, we do not see any clear trends with change in occupancy.

Even when the building is completely unoccupied, electrical power consumption is $\sim$35% of the power consumption at full occupancy, and heating and thermal power is at $\sim$70%. As the building is put in to “Standby” mode when it is unoccupied, the

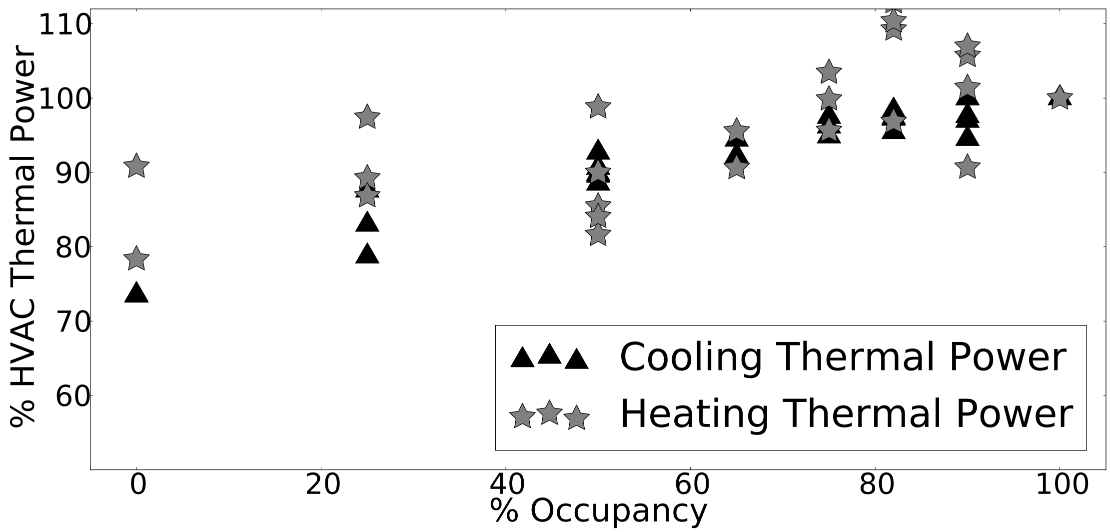

Figure 3.11. HVAC thermal power consumption with change in occupancy levels in the building

HVAC system still tries to maintain minimum thermal comfort within the building. For our building, the temperature guardband is increased by 2 °F on both cooling and heating setpoints with respect to the setpoints in "Occupied" mode.

The thermal power consumption is still high compared to electrical power when the building is fully unoccupied. This is because the cold water is used for cooling the server room in CSE, and the hot water is used for domestic water heating. Also, recall that our building receives its hot and cold water from a central utility plant(Section 2.2), and thus, the reduction in thermal energy observed is due to the decrease in the demand for hot and cold water. However, as the hot water and cold air still circulate through the building, there is still a drop in temperature in the returned hot and cold water. The thermal power consumption is measured as the energy spent due to the loss in the temperature difference between the supply and return water. Hence, even at zero percent occupancy, significant amount of energy is spent for thermal needs.

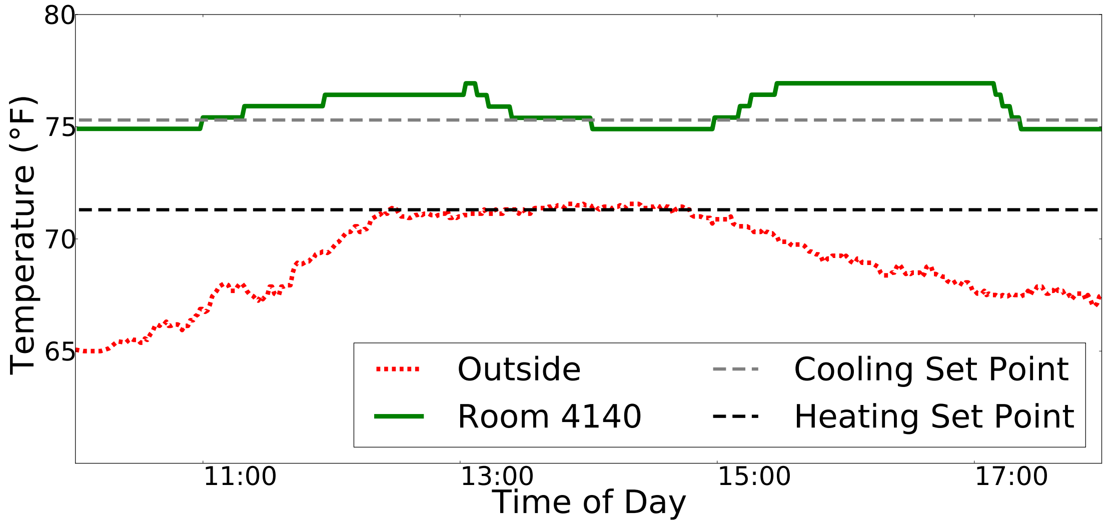

Figure 3.12. Temperature profile of an HVAC zone during daytime when it was turned on and off every two hours. Heating and cooling setpoints are 71 °F and 75 °F respectively.

3.4.7 Thermal Comfort

Prior work suggests that reactive control of HVAC system does not lead to occupant discomfort when the setback temperature is conservative [18, 79]. To test this in our building, we performed a controlled experiment on a subset of HVAC zones. We chose 12 HVAC zones, each of them having different characteristics in terms of size, location, and number of rooms. Each of the zones were alternated between "Occupied" and "Standby" modes for two hour periods over a total period of 8 hours during the day on a weekend.

One of the HVAC zones had a faulty sensor, and we do not consider its temperature data. Figure 3.12 shows the variation in temperature of the HVAC zone which showed the *maximum* thermal discomfort among the remaining 11 zones in the experiment. The heating and cooling temperature set points of the "Occupied" mode for this zone was at 71 °F and 75 °F respectively, and the corresponding set points of the "Standby" mode was 69 °F and 77 °F respectively. Unfortunately, the outside temperature at the time of the year is temperate, and does not change the temperature of the zone significantly, even when it is in the Standby mode. It is clear from Figure 3.12 that the temperature of the HVAC zone never exceeds 77 °F, and quickly drops to 75 °F as soon as the zone is switched to “Occupied” mode. Thus, we confirm that the finding by Goyal et al. [79] by real temperature measurements that the thermal comfort is minimally effected when the setback temperature setpoints are conservative.

3.4.8 Energy Savings with Sentinel

We controlled the HVAC system of our building testbed using Sentinel for the 116 volunteers from 9am to 6pm on March 26, 2013. Of a total of 237 HVAC zones, we controlled 55 zones distributed across three of the five floors in the building.

As HVAC zones are often shared between rooms, the actuation policy of the occupants located within an HVAC zone needs to be the same. As a result of this sharing, some of the personal spaces needed to be converted to shared spaces, as explained in Section 3.2.5. Similarly, the occupants who could not participate in the experiment, share their HVAC control policy with our volunteer occupants. Therefore, a single non-eligible participant in an HVAC zone forces us to treat the entire zone as a shared space. Despite this limitation, we control 55 out of 237 HVAC zones in the building for our actuation experiment. As we are requesting the occupants of the building to shift from their regular usage patterns, we had to limit our control experiment to just one day. Of the 55 zones covered by the experiment, 12 zones were known to be unoccupied apriori on the day of the experiment, and we turned them off for the duration of the experiment.

We compare the energy consumption on the day of our experiment (March 26, 2013) with the energy consumption on March 22, 2013, as the temperature profiles of the two days were similar. We refer to the day we controlled the HVAC system using Sentinel as “Experiment Day”, and refer to the day of comparison as the “Typical Day”. Other close days were cloudy, and we could not use them. Figure 3.14 show the electrical power

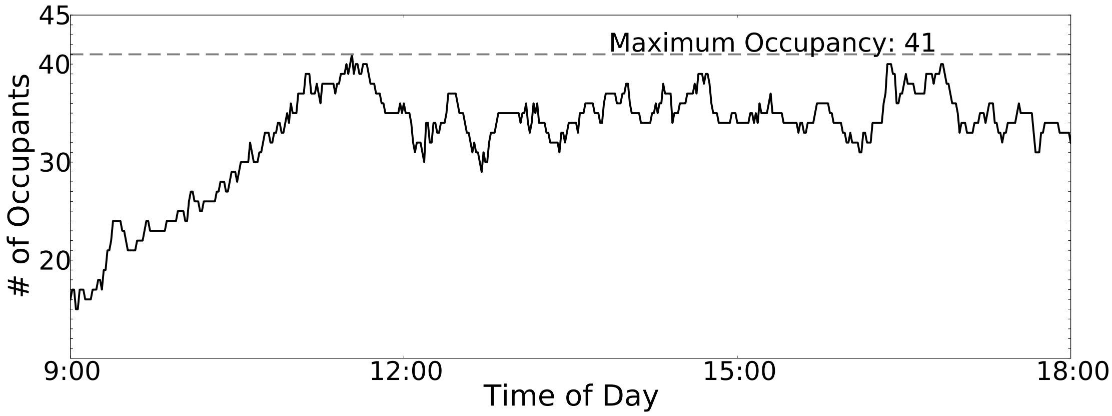

Figure 3.13. Occupancy trends of 116 volunteers on March 26, 2013, the Experiment Day.

consumption from 9am to 6pm on the Experiment and Typical days, Figure 3.15 shows the thermal power consumption of HVAC in the same time frame, and finally, Figure 3.13 shows the occupancy of the 116 volunteers on the Experiment Day as measured by Sentinel.

We saved 17.8% of electrical energy on the Experiment Day, as compared to the Typical Day. Occupancy trends from Figure 3.13 shows that the building occupancy gradually increases from 9am to 11am, and remains roughly constant till 6pm. However, the occupancy peaks at 40 people, indicating most of the volunteer occupants were not present in the building during the period of experimentation. The relative inoccupancy was expected, as the Experiment Day was the second day of the spring break at our university.

The occupancy trend is clearly reflected in the electrical power consumption of the HVAC system, as it initially starts off lower than the typical day at 9am due to the reduced number of occupants in the building. As the occupancy within the building increases, the power consumption also increases gradually until 11am. From 11am to 6pm, the electrical power consumption of both the days follow the same pattern, in accordance with the changing outdoor weather conditions. The energy savings from

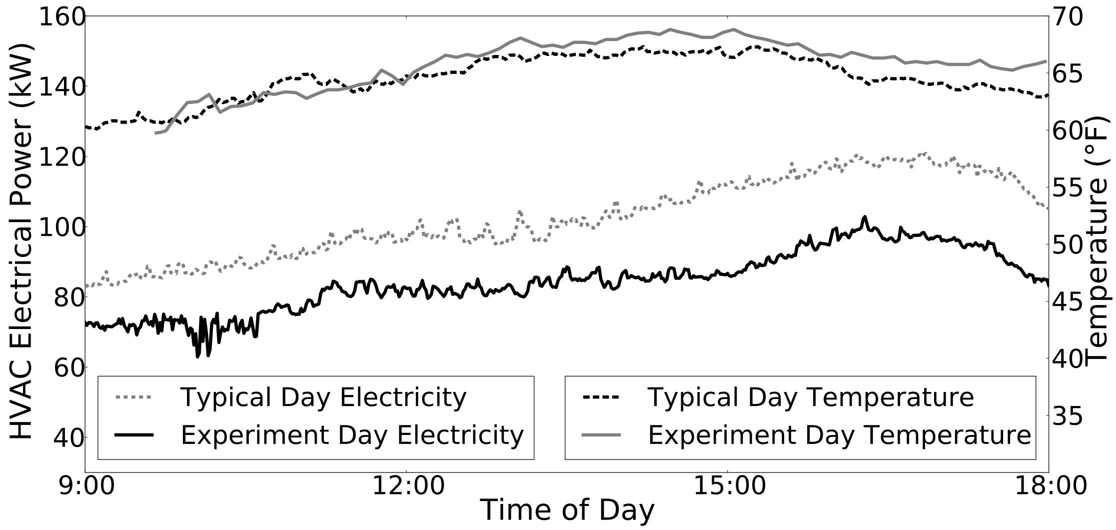

Figure 3.14. Comparison of outdoor temperature and HVAC electrical power consumption of the Typical Day and Experiment Day. Total savings of 17.8% in electrical energy was obtained for the duration of the experiment.

11am to 6pm is mainly obtained because of the occupants who did not come in to their personal spaces on the Experiment Day. The 17.8% electrical energy savings obtained is in accordance with electrical power consumption trends shown in Figure 3.10, where the corresponding building occupancy is $\sim$90%.

As our Experiment Day falls on university spring break, but our Typical Day is during exam week, it is possible that part of energy savings occur due to reduced activity in the building. We compared the HVAC electrical power consumption on Experiment Day with two other spring break days (March 27 and 28, 2013) with cloudy weather conditions when the HVAC was under static schedule based control, and still measured electrical energy savings of 7.5% and 11.8% respectively.

The trends in thermal power consumption on the Experiment Day were not as clear. Cooling thermal energy consumption decreased by 2.2%, but the heating thermal energy actually increased by 1.5%. Figure 3.11 indicates that the thermal energy consumption is also consistent with our night time trending experiments, and the heating thermal power

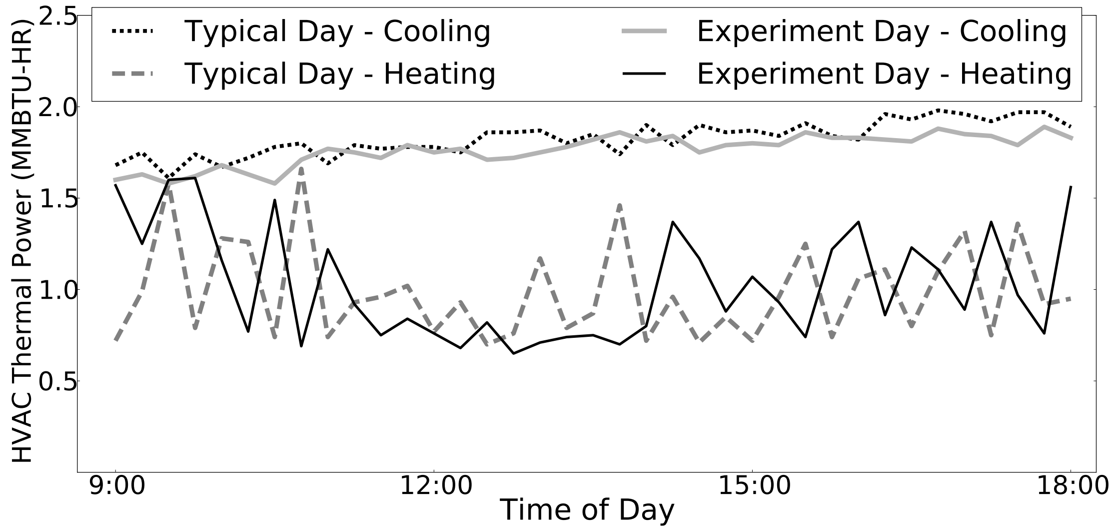

Figure 3.15. Comparison of HVAC heating and cooling thermal power comparison of the Typical Day and Experiment Day. No clear trends can be observed, and only 0.8% energy was saved for the duration of the experiment.

consumption sometimes increased despite a reduction in building occupancy.

The experiment provides an example of the energy savings that could be obtained across one particular day by controlling 23% of the HVAC zones in CSE. However, the long term energy savings will be different due to varying weather conditions or occupancy patterns. As long term occupancy patterns are not available, we do not attempt to project the energy savings obtained by simple extrapolation of trends we see for one day.

3.5 Discussion

The occupancy inference algorithm proposed in this paper uses the metadata information available and typical occupancy patterns within offices to mitigate the inaccuracies associated with locating a WiFi enabled device with respect to its AP. The algorithm can be adapted to a wide range of office spaces independent of its building topology, or usage patterns. To infer occupancy using WiFi we use key metadata relating authorized occupants with their personal office space, their WiFi device MAC address, network logs to determine the current status of network connectivity with occupant devices and the location of APs within the building.

For adoption of our solution in a commercial building, a dependable and easily accessible fallback solution needs to be provided to the building occupants. Occupants should be able to inform the BMS of their presence easily in case they forget their phones at home, or need to lend their office to a visitor. The personalized building control proposed by Krioukov et al. [108] provides a good platform for user feedback, and we have implemented a similar web based interface for CSE. Automated tools for keeping track of occupants in personal spaces, mapping of APs to personal spaces and HVAC zones to office spaces would also help in quick deployment.

Reliable WiFi connectivity from the users phones is the only requirement from the occupants of the building for the proposed algorithm. However, as we saw in Sections 3.2.4 and 3.4.4, it is difficult to maintain perpetual connectivity in iOS devices, and there may be an effect on battery life of devices when they are always connected to WiFi. IEEE 802.11ah standard [28] is being designed specifically for low power, low data rate applications, and would enable applications like Sentinel without affecting battery life. In the meantime, we plan to develop mobile apps which would maintain WiFi connectivity and still have minimal effect on battery life. The apps would break the non-intrusive model of deployment, but can be integrated with the personalized building control system [108]. Alternatively, prediction mechanisms can be used to eclipse the intermittent connectivity of WiFi devices.

As most of the false negative occupancy detection errors in Sentinel is caused by iOS devices, we have already developed an iOS app. The app creates a geofence on the building, and wakes up the device when it enters the geofenced area. The app keeps the device awake until it connects to WiFi, and then allows the device to go to sleep. Periodic push notifications from the app wake up the WiFi radio, and the notifications are turned off when the device leaves the geofenced area. However, we have not yet evaluated the app extensively to present its performance results here.

Sentinel only targets personal spaces in office buildings. To improve HVAC energy efficiency further, shared spaces should also be regulated according to occupancy. One option is to install wireless sensor network solutions [18, 64] just for the shared spaces. Use of calendars has been proposed as a proxy for occupancy [58], however it is not applicable to several kinds of shared spaces like lobby, cafeteria, etc. Indoor localization has the potential to reduce the zone of detection enough for occupancy inference in shared spaces. We plan to explore infrastructure based localization techniques as part of future work.

The HVAC zones in modern buildings are not designed for occupancy based actuation. Although VAV systems have become commonplace since the late 1990s [90], the zones normally map several individual rooms. If only one of the rooms within a zone is occupied, the remaining rooms within the zone are unnecessarily ventilated. Further, sharing of HVAC zones between shared and personal spaces, requires conditioning of personal spaces whenever the shared space is occupied. Smaller and more insulated HVAC zones would lead to more savings based on occupancy control in lieu of higher installation cost. If the architects of the HVAC system incorporate occupancy based control into their design for next generation buildings, there could be a significant reduction in the running cost of the system.

3.6 Related Work

Occupancy based HVAC control has been studied extensively for improving building energy efficiency [18, 40, 62, 68, 65, 64]. Extensive simulation studies and practical deployments in commercial buildings have shown that 15% - 42% energy savings can be obtained using occupancy based control, depending on weather conditions, building type and occupancy variation.

Several occupancy detection mechanisms have been developed over the years for HVAC control. CO_2 sensors are used for occupancy based control of high capacity spaces such as auditoriums and conference rooms [6, 10], but have been found to be too slow to respond to change in occupancy for smaller rooms found in commercial buildings [69]. Passive infrared(PIR) motion sensors have been used in modern buildings for actuation of lighting systems. PIR sensors often fail to detect occupants when they are relatively motionless, such as while reading or typing. Further, they are vulnerable to calibration errors, external triggers by sunlight or air draft and only provide binary occupancy information. These limitations make it challenging to use PIR sensors for HVAC control. Our own work improved upon these limitations with the addition of door sensors to obtain occupancy accuracy of 96% and demonstrated up to 15% savings in HVAC electrical energy for one floor deployment in CSE [18]. However, the occupancy detection mechanism is only accurate for single person offices, and depend on the occupants to close the door while exiting the office.

The POEM system [64] uses a combination of ceiling mounted camera and motion sensors to obtain 94% accuracy in occupancy detection. Erickson et al. use the near real-time occupancy information from the sensors for predictive control of 30% of the HVAC zones in an office building and demonstrate up to 26% energy savings. The cameras used in POEM exploit the hallway topology for occupancy detection. If the office spaces are located around a circular hallway, or use open cubicle spaces, the image processing algorithms would have to be modified and re-calibrated. Further, use of battery powered wireless sensor nodes in POEM involves changing batteries every 45 days.

In contrast, Sentinel provides a solution for actuation of HVAC system using near real-time occupancy derived from existing WiFi infrastructure. Sentinel neither makes any assumption regarding the topology of the building, nor requires careful calibration of sensors. Leveraging existing infrastructure allows Sentinel to be quickly deployed and easily maintained. The monetary and ease of use benefits of Sentinel comes at the cost of assumptions on usage patterns of WiFi devices by building occupants, and works only for personal spaces. Additional sensors or localization techniques still need to used for occupancy detection in shared spaces.

Numerous methods for occupancy detection have been developed that leverage existing infrastructure such as powerline [140], speakers [110], WiFi [29, 180, 45], geomagnetism [46], HVAC ductwork [138], or a combination of these [174]. However, many of these solutions do not work well for HVAC control in commercial buildings due to issues of scale [140, 45], use of specialized sensors [138, 110], extensive war driving [180, 46] or complex functionality in client devices [174].

Existing WiFi infrastructure potentially provides the scalability needed for commercial buildings, does not rely on client device functionality and eases deployment and maintenance. Ghai et al. [78] use a combination of WiFi signals, calendar schedules, personal computer activity and instant-messaging client status to infer the occupancy within cubicles with an accuracy of up to 91%. The algorithms have been evaluated for just 5 volunteers, and do not evaluate scalability. In contrast, we only use WiFi information, and show the efficacy of our algorithm over 116 occupants of our building. Melfi et al. [125] use DHCP leases within a real building for occupancy inference and found the accuracy to be low - 31% to 84%. The inaccuracies of their system were attributed to unpredictable coverage provided by APs and intermittent connectivity of the WiFi devices. We overcome the limitations of WiFi sensing by using additional known information such as occupant identity, occupant office location and focus on personal spaces. Martani et al. [123] use WiFi logs to determine the live WiFi connections within a building, and provide a breakdown of the WiFi connections on a floor and room basis.

They show that WiFi connections correspond well with the HVAC energy consumption of a building at MIT. However, they make no attempt to correlate WiFi connections with the ground truth occupancy.

3.7 Summary

We have presented the design and implementation of Sentinel- an occupancy based HVAC actuation system that leverages existing WiFi infrastructure and occupants with WiFi enabled smartphones within commercial buildings to reduce HVAC energy usage. In contrast to prior occupancy sensing solutions which required installation of additional sensors and associated wireless sensor networks, utilizing existing infrastructure for occupancy sensing reduces the costs and effort of deployment and maintenance significantly. We reduce the inaccuracies in occupancy sensing using noisy WiFi signals by using metadata information about the occupants, access points and the HVAC zones in the building. We have deployed Sentinel in a 145000 sqft commercial building, and show the accuracy of occupancy detection within office spaces to be 86%, with only 6.2% false negative errors. Furthermore, we provided a detailed analysis of the reasons for these inaccuracies, largely due to aggressive power management by smartphones. Based on our battery lifetime measurements across a number of devices we show that using less aggressive WiFi power modes, which improve accuracy of Sentinel, do not necessarily lead to significantly reduced battery life. We also discuss potential solutions, such as an App on users phones, that can increase the accuracy of WiFi based occupancy detection even further. Finally, we demonstrate occupancy based control of 23% of the HVAC zones of our building testbed using Sentinel and measure electrical energy savings of 17.8% in the HVAC system compared to the static scheduling based control used across the buildings on our campus.

Chapter 3, in part, is a reprint of the material as it appears in Proceedings of ACM

Conference on Embedded Networked Sensor Systems (SenSys 13), 2013 by authors Bharathan Balaji, Jian Xu, Rajesh Gupta and Yuvraj Agarwal with the title Sentinel: An Occupancy Based HVAC Actuation System using existing WiFi Infrastructure in Commercial Buildings. The dissertation author is the primary investigator and author of this paper.

Chapter 4

Zonal Apportionment of HVAC Energy

In this Chapter, we present a method to analyse existing sensor data to attribute HVAC energy consumption to each thermal zone in a building. This information is then presented to the building occupants using a web service application. Taking one step further from Chapter 3, here we integrate information across various sources - HVAC sensors, building level power meters, building architectural diagrams. We also draw on information available on the web about equipment installed to examine their impact on energy in detail. By using energy transfer principles and integrating all of the information, we create this HVAC energy apportionment system. Thus, this system is another example of a software based analysis of existing infrastructure to gain insights in energy flow within a building. The same information with installation of submeters in the building would be prohibitively expensive.

Several studies have shown that providing relevant energy feedback to the occupants of a building can lead to significant energy savings [54, 143]. However, the energy feedback has been limited to electricity consumption [143]and has been designed for residential buildings [24, 54]. We developed ZonePAC, an energy apportionment system that bridges this gap between building operations and the experience by individual occupants. We do this by connecting zonal monitoring and estimation that incorporates participatory occupant sensing and occupant experiential feedback to be incorporated in the building scale HVAC system.

ZonePAC estimates the heating, cooling and electrical power consumption of each zone in a Variable Air Volume (VAV) type system using existing infrastructure sensors installed as part of the Building Management System (BMS). We then provide the HVAC power consumption feedback to the occupants of the building over the web and on mobile devices along with other thermal comfort related measurements such as measured temperature and setpoint.

We have built ZonePAC on top of BuildingDepot [22] and deployed it in the CSE building at UCSD. We present the results of our data collection and its analysis regarding distribution of energy consumption across zones. We identify anomalous behavior and provide possible causes behind energy inefficiency. Since the ZonePAC system also provides the occupant with the capability to change local HVAC control settings, we provide data on user experience and the results of such individual control settings on overall building operation.

4.1 Zone Power Estimation

The goal of ZonePAC is to provide a real-time estimate of total power consumption of individual HVAC zones. We use the measurements from existing sensors, and apply first principles to estimate power consumption of an HVAC zone which consists of three parts - *cooling thermal*, *heating thermal* and *electrical*. The cooling thermal power is used for converting the warm return air from the zones to the cold supply air, the heating thermal power is used for reheating the cold supply air when the temperature setpoint of the zone is too high to be satisfied by reducing the cold air, and the electrical power is used by the fans and the pumps used for supplying air to the zones from the central HVAC equipments.

4.1.1 Cooling Thermal Power

We estimate cooling thermal power using the heat transfer equation:

$$Q_{cooling} = \rho * C * q * (T_{zone} - T_{supply}) \tag{4.1}$$

where, ρ = density of air at 20°C, C = specific heat of air, q = rate of airflow, T_{zone} = zonal air temperature, T_{supply} = supply air temperature.

In the absence of sensors to directly measure the supply air temperature of each zone, we approximate it by the supply air temperature as measured by the central air handler unit (AHU) as it exits the cooling coils. This, of course, neglects the temperature loss due to imperfect insulation and leaks in the air ducts. Similarly, we estimate the return air temperature by the zonal temperature as measured by the thermostat in the zone. Finally, the airflow rate is measured directly by the flow sensor in the VAV box.

For HVAC systems which also provide humidity control, the power consumption estimate would also have to include the latent heat transfer. The corresponding sensors measuring the supply air humidity and the return air humidity would be required for an accurate estimate.

We establish the accuracy of our estimate by comparing the total cooling power as measured by the building thermal power meter and the aggregate cooling power obtained by applying equation 4.1 to all the zones of the building. Due to implementation issues, we use an estimate of the power use by CRAC unit based on empirical measurements that showed an average use by CRAC unit in a narrow range of 0.50 to 0.60 MMBTU/hour.

Figure 4.1 shows the comparison between cumulative estimated cooling power and the measured cooling power for July 30, 2013. The results show an average error of 12.8% across one week of measurements. We find that our estimates are accurate during the night time, but we consistently overestimate during the day. This overestimation is

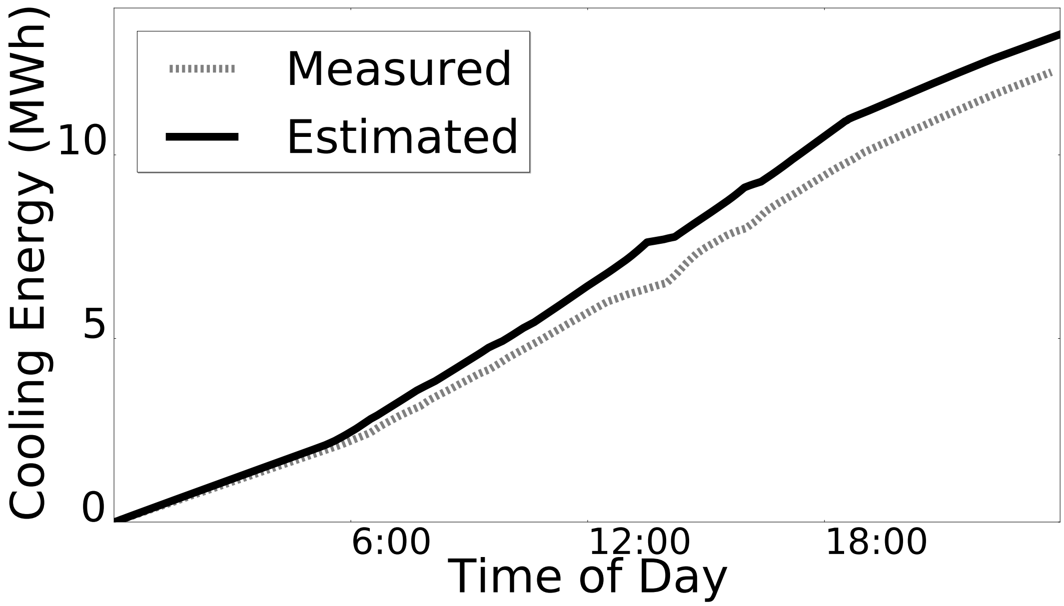

Figure 4.1. Comparison of aggregate cooling power estimated and measured cooling power from installed flow and measured sensors in the CSE building.

due to the fact that we do not have dedicated air ducts for return air, and it is directed through plenum space to the AHU. The leaks in return air reduces the air temperature when it reaches the AHU. Another reason for the overestimate is that we do not account for outside air mixed with the return air before cooling. After adjusting for both the return air losses and mixing of outdoor air using measured parameters, we found that the average error of our estimated cooling power improved to 5.1%.

4.1.2 Heating Thermal Power

The only sensor connected to BACnet related to the hot water system is the "Reheat Valve Command", which is the valve position command sent by the VAV digital control system. The reheat valve controls the amount of hot water through the heating coil, and the building plans show that our building uses a modulating 2-way electronic control valve. There are two types of modulating valves generally used in hot water coils

- linear and equal percentage, and both the types of valves are designed to provide linear heat output with change in valve position. We obtain the maximum heat output of each VAV box from the building plans, and estimate the heating thermal power as:

$$Q_{heating} = H * Q_{max} \tag{4.2}$$

where, H = reheat valve command, Q_{max} = maximum heat output of heating coil.

To evaluate the accuracy of our estimation we compared the measured heating thermal power with the aggregate estimated heating power, similar to the methodology followed in Section 4.1.1. However, the gap between measured and estimated power is much larger with an upper bound of nearly 10X the estimated heating power. There are multiple reasons why this estimate could be so far from the actual power consumption. The "Reheat Valve Command" tag indicates the position of the valve as controlled by the VAV, but there is no sensor which measures the actual position of the valve. It is possible that the value is stuck at a position different from that indicated by the "Reheat Valve Command", and causes leakage of hot water. In the centralized HVAC unit, the valve position of HVAC heat exchanger indicated significant amount of flow corresponding to the measured heating thermal power, and the domestic water heat exchanger also indicated high flow rate. Lack of flow rate sensors in the exchangers prevents determination of exact flow rate and heat exchanged. We are working with the campus facilities personnel to install flow meters to resolve the issue.

4.1.3 Electrical Power

We have added power meters in the CSE building which measures the total mechanical power, and HVAC systems account for 13% to 46% of total electric power on a typical summer day. The electricity consumption depends on the airflow demand from the HVAC zones in the building, and thus, electric power consumption needs to be attributed to each zone. The fans and pumps used in CSE are Variable Frequency Drives(VFDs), and the speed of the motor is directly proportional to the amount of airflow pumped to the rest of the building. The power consumed by the VFDs is proportional to the cube of the fan speed. Thus, to estimate the electric power attributed to each zone, we use the following equation:

$$Q_{electric} = q^3 * Q_{totalelectric}/\Sigma q^3 \tag{4.3}$$

where, q = rate of airflow, $Q_{totalelectric}$ = total electrical power measured, Σq^3 = summation of cubic airflow through all zones.

Some of the VAV boxes are equipped with additional supply fans, to maintain the required air pressure in large zones. Also, some of the zones such as restrooms and kitchenettes have exhaust fans in them. We determine the status of these terminal fans using BACnet datapoints available, and we assume they operate at their rated power provided by manufacturer as there are no power measurements available. We subtract the contribution of the terminal fans from $Q_{totalelectricity}$ in equation 4.3, and attribute their power to the corresponding HVAC zones directly. We ignore the contribution of some of the smaller equipments such as the air compressor used for operating pneumatic valves and hot water pumps, as we do not have their power measurements and their contribution to the total mechanical power is minimal.

4.2 Implementation

ZonePAC has been implemented on top of BuildingDepot [22], an open source RESTful API based building management web service. The data from BACnet sensors are collected using our BACnet connector, and the HVAC Meter Service estimates the

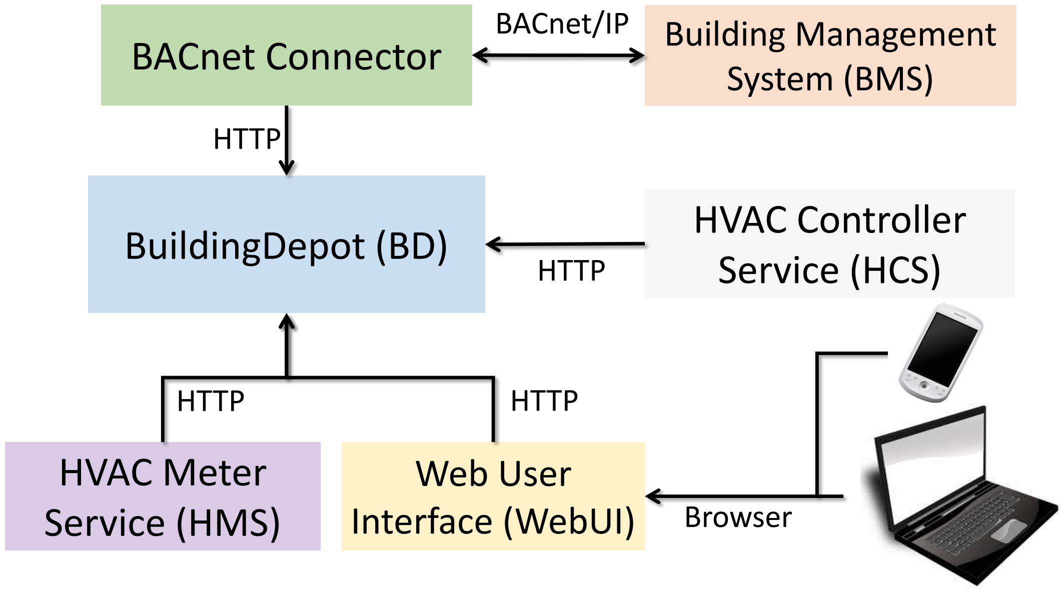

Figure 4.2. System Architecture of HVAC Meter

power consumption of each zone as explained in Section 4.1. The Web User Interface (WebUI) reads the data from virtual power sensors created by ZonePAC, and presents it to the occupants. The interface also allows for change in control of HVAC zone settings. Figure 4.2 shows the software architecture of our system.

The HVAC Meter Service (HMS) subscribes to the relevant BACnet points needed for power estimation, and BD notifies HMS as new data is posted from BACnet via a notification url. HMS estimates the power as outlined in Section 4.1 and posts the computed power back to BD as virtual sensor data. HMS also computes related useful data such as aggregate heating and cooling power, zone power consumption per unit area, average zone temperature, etc.

We implement an interactive webapp on top of BD which reads sensor data from both BACnet and ZonePAC. Interested occupants register their email address, and WebUI administrators provide permission to access the sensor information after manual verification. Access control among users is enforced by BD, and users are only provided

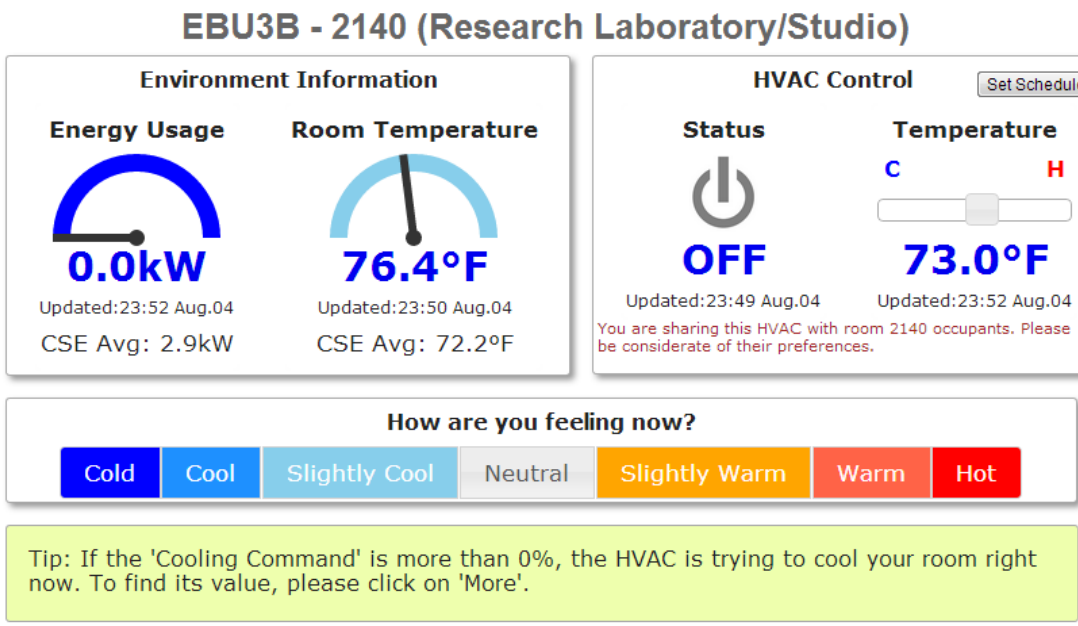

Figure 4.3. Screenshot of Web User Interface

information about zones to which they have physical access. Figure 4.3 shows a snapshot of the WebUI. It shows the room temperature and the energy consumption as estimated by ZonePAC. Users can provide feedback on their thermal comfort in 7 levels from “Cold” to “Hot” as shown in Figure 4.3 along with free form text. We allow the users to change their temperature setpoint by $\pm 3\,^{\circ}$F from the preset setpoint. When a user turns OFF the HVAC using WebUI, the occupancy mode is changed to “Standby” during weekday (6am - 10pm), and is changed to “Unoccupied” on nights and weekends. We also include a suggestion box as an experimental feature that shows personalized energy saving recommendations. For example, if the VAV is cooling a zone excessively for over an hour, a suggestion is provided to increase the setpoint by $0.5\,^{\circ}$F to save energy. The details of the WebUI design are provided in Chapter 5, where we study the effect of long term deployment of such a service to the occupants.

The HVAC Controller Service (HCS) relays the commands provided by WebUI to the corresponding BACnet points. The HCS was designed such that the control service

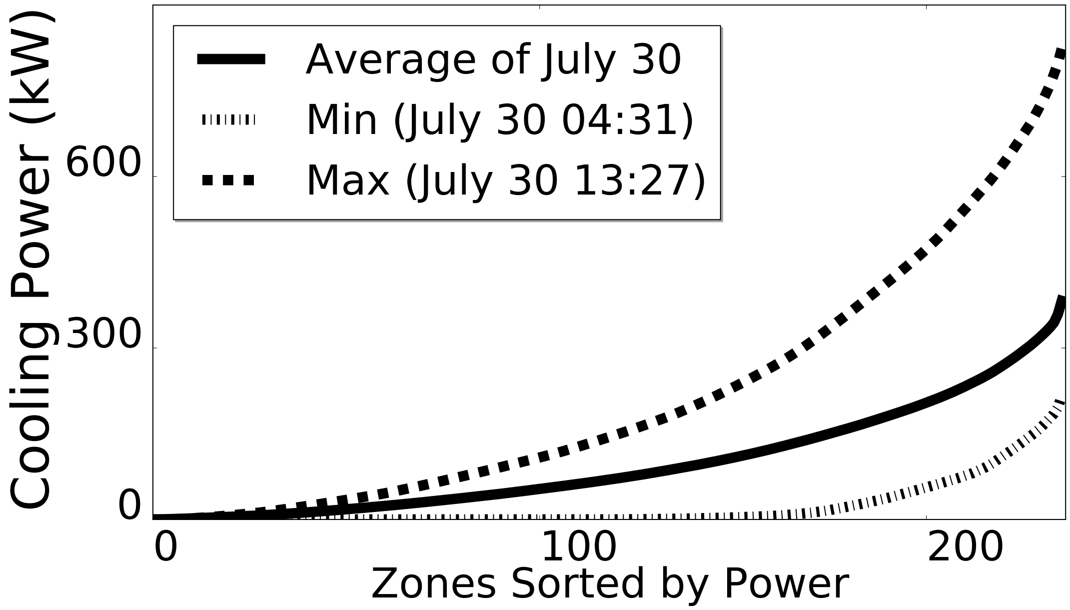

Figure 4.4. Cooling Power Distribution across HVAC Zones

could be made unavailable to the users if needed without affecting the feedback services provided by WebUI.

4.3 Results

The estimates on zone power using ZonePAC enables us to collect historical data and analyze the trends in energy consumption. We present our insights from observing ZonePAC data for 10 days across the 237 zones in CSE. Further, we deploy ZonePAC WebUI in CSE, and present the data collected for 65 registered building occupants.

4.3.1 Power Consumption Trends

In order to understand the distribution of HVAC power across the zones in the CSE building, we present the cumulative contribution of individual zones to the total power. Figure 4.4 shows the distribution for HVAC cooling for the average, the maximum and the minimum power consumption for July 30, 2013. The peak cooling power is more

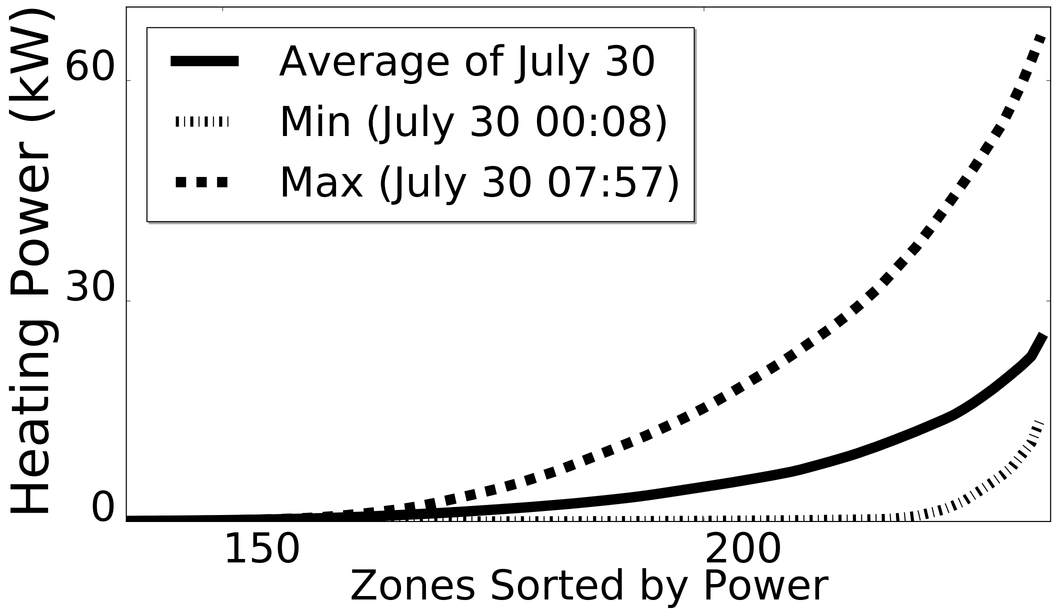

Figure 4.5. Heating Power Distribution across HVAC Zones

than double the average power consumption and which is the reason HVAC system is the dominant target for energy reduction during demand response events in our campus. On an average, 50% of the zones consume only 20% of the cooling power, and the remaining half of the zones account for 80% of the zones. Thus, with limited resources available, it will be prudent to target power intensive zones for aggressive energy saving strategies such as occupancy based HVAC control for maximum benefits. Similar trends can be observed in the distribution of heating and electrical power in Figures 4.5 and 4.6, with over 150 zones accounting for less than 2% of the total power.

Ironically, the most power intensive zones are the ones which house HVAC equipment and building substation. The equipment rooms are followed by basement computer labs. As there is no fixed schedule followed by the students, the labs are always kept conditioned. Further, as the minimum cooling air is determined by the maximum capacity of the labs, these zones are overcooled when fewer occupants are

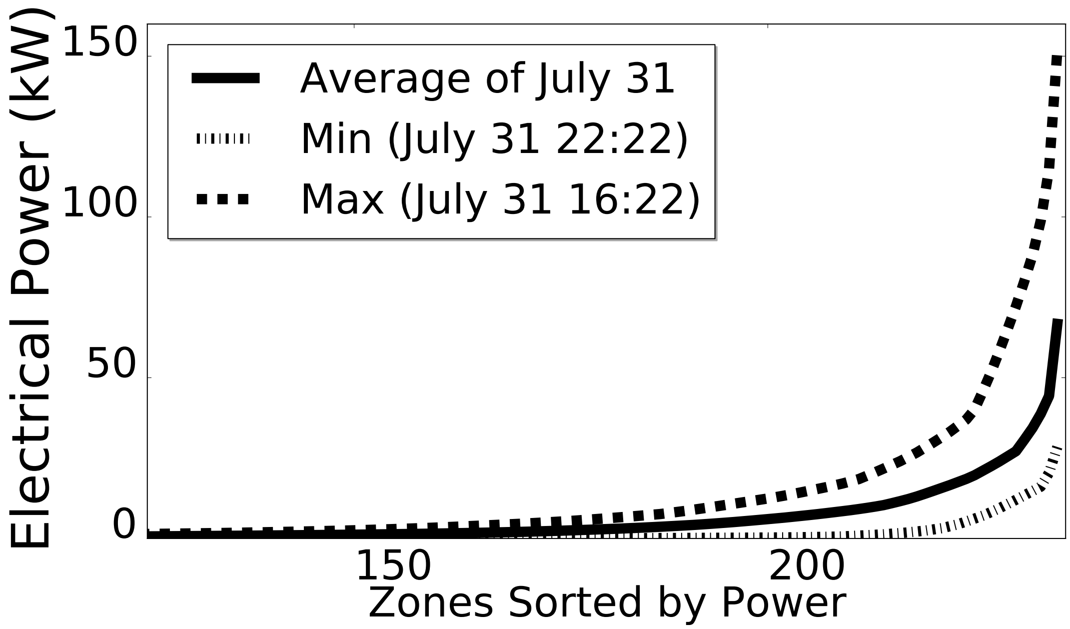

Figure 4.6. Electrical Power Distribution across HVAC Zones

present causing them discomfort. An occupancy detection system would not only save energy by reducing the airflow ventilation with change in occupancy, but also provide better thermal comfort.

The thermostat adjust control in basement labs are often kept in their extreme positions, further exacerbating the effect of over cooling. As the labs are a shared space, no one takes responsibility for temperature control, and students are often unaware of the thermostat location. When the thermostat is set to decrease the temperature setpoint, there is excessive use of cooling power, and when it is set to increase the setpoint, heating coils are used with minimum cooling air. Thus, we find basement labs to be dominant in both heating and cooling power. We observed similar thermostat settings in several spaces which are shared - student lounge, conference rooms, lobby, kitchenette, etc. The sizes of the shared zones are large as they are designed for higher capacity, and hence, the minor changes in thermostat leads to large losses in energy. An ideal energy saving strategy would be to provide temperature control to occupants only when they are physically present in the shared space, and reset to energy efficient settings once the occupant leaves.

To examine energy inefficiency in smaller zones, we plot the trends in zone power consumption per unit area, as shown in Figure 4.7. We find that aberrant thermostat settings cause energy inefficiencies even in smaller zones. Although the thermostat could have been set according to occupant comfort preference, the feedback from our WebUI (Section 4.3.2) indicate that many occupants are unaware of the thermostat location and are uncomfortable with the current temperature settings. This is not unreasonable as a single zone can constitute multiple office rooms and the thermostat is located in only one of the rooms. By providing the WebUI, the occupants were both informed of the measured temperature, and could change their settings if they were not comfortable.

To further investigate the relation between thermostat setpoint and the zone power consumption, we manually inspect thermostats in the zones which required abnormally high heating. Although the facilities management mandates a range of $\pm 1\,^{\circ}$F from the predetermined setpoint, we found several thermostats allowed deviation of over $3\,^{\circ}$F. Further, the change in the thermostat dial did not lead to a linear change in the temperature setpoint, and each thermostat had its unique mapping to actual changes in setpoint. For instance, the sensitivity of one of the thermostats was so high that a small change in the dial would change the setpoint by several degrees, and the midpoint of the analog adjust in another thermostat corresponded to an increase in setpoint by $3\,^{\circ}$F. Such thermostat calibrations can lead to unintended temperature settings and cause both thermal discomfort and wastage of energy. We adjusted the thermostats for 8 of the zones to reduce the reheat required, which resulted in 50.7% savings in heating power. However, since our adjustments were not fine enough, it resulted in an increase in airflow rate which led to no savings in the total power.

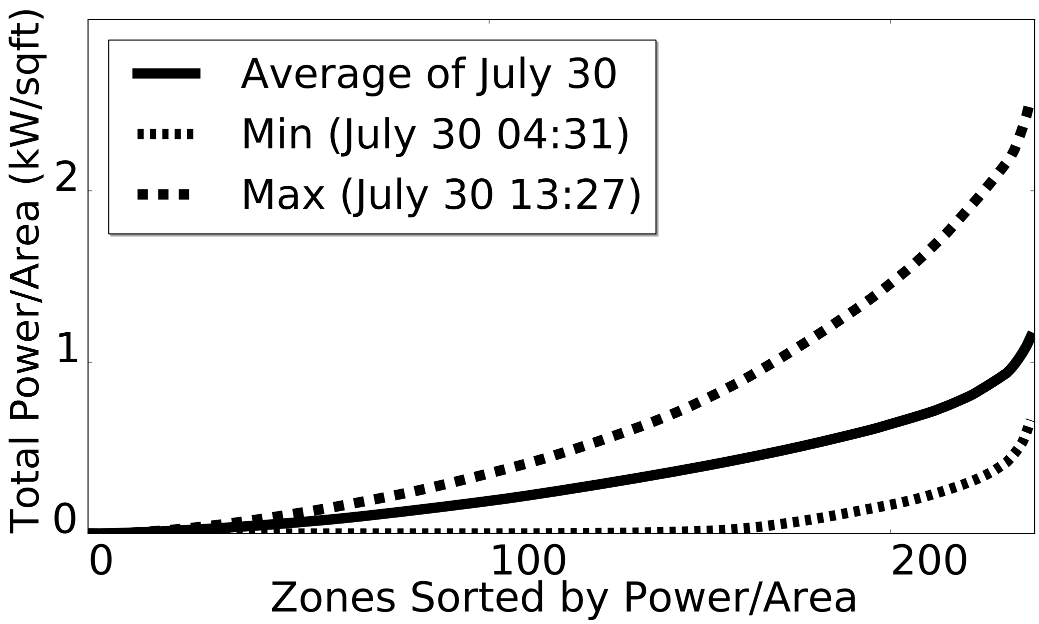

Figure 4.7. Distribution of HVAC Zones by Total Power per Unit Area

We also find that the rooms which have additional cooling demands affect the power consumption of the nearby zones. For example, one of the office rooms was repurposed to host computing equipment and the HVAC was requested to be always in "Occupied" mode and the thermostat was adjusted to its minimum to satisfy the cooling needs. As a result, significant amount of cold air was pumped to the zone to maintain the requested temperature. As the same air duct is shared by multiple zones, and due to heat transfer by conduction and convection across zones, the nearby zones were overcooled, requiring heavy use of hot water to maintain comfortable temperature. Facilities personnel informed us that such zones are known to cause inefficiencies and are installed with a special cooling unit to satisfy the additional cooling demands, but the zones are difficult to locate unless the occupant directly contacts them.

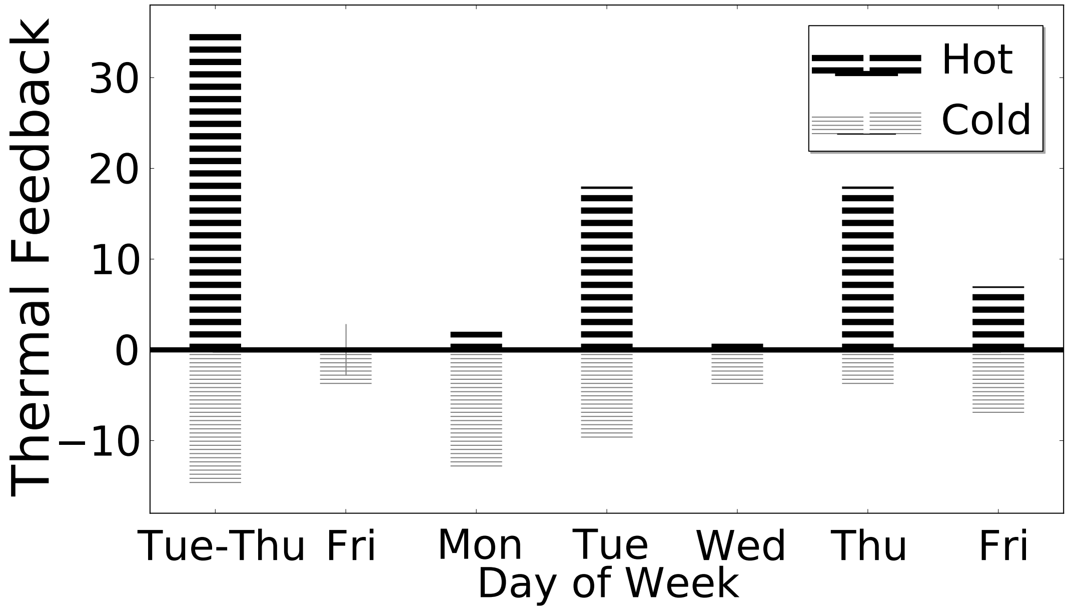

Figure 4.8. Distribution of the thermal feedback received for 10 days of deployment

4.3.2 Occupant Feedback

We deployed WebUI of ZonePAC for 10 weekdays at CSE, and report results obtained from 65 registered users across 51 zones. Participation was voluntary, and we did not provide any incentives to the occupants apart from providing feedback and control of HVAC system. We provided only the measurements from HVAC sensors on the first four days of deployment, then added the provision to change settings of the temperature setpoint and HVAC status. After two days of allowing control of settings, we added energy savings suggestions to the WebUI as explained in Section 4.2. We received over 140 feedback inputs on thermal comfort, and users changed their HVAC settings over 130 times during the course of the control period.

From the distribution of average zone temperature shown in Figure 4.11, we observe that most of the zones fall in the comfortable range of 70°F to 75°F. The zones which show large deviations from the ideal temperature are either anomalous or

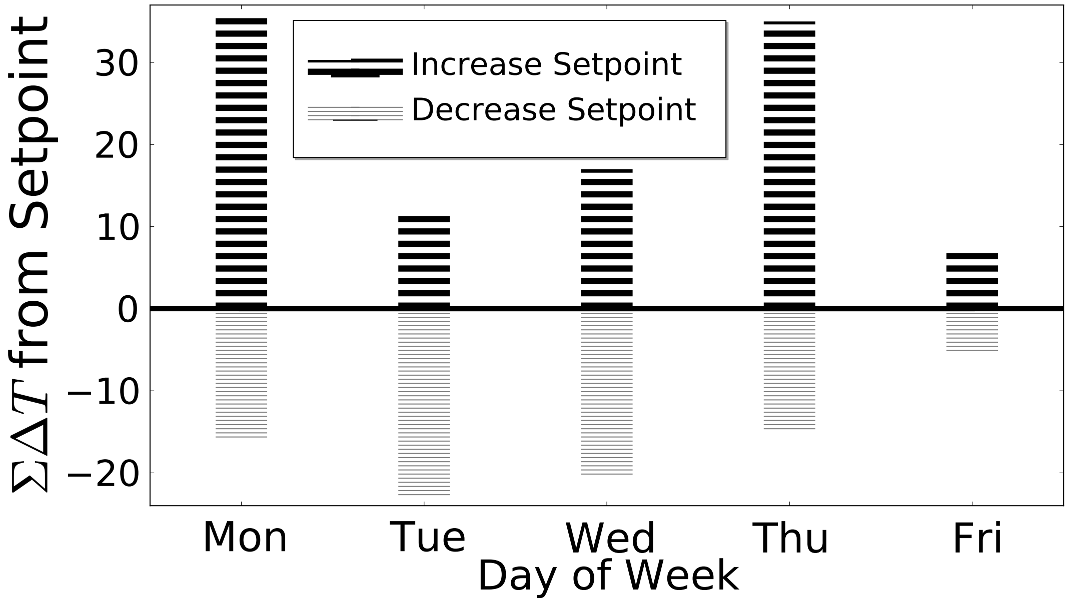

Figure 4.9. Changes in setpoints selected by the users for one week of control

unoccupied, such as the server room(60 °F), an unused office space(82 °F)and a zone with a damaged damper(78 °F). The thermal comfort feedback we received from WebUI confirmed that most users were comfortable, as 60% of the feedback inputs indicated acceptable comfort levels as per ASHRAE Standard 55 [3], i.e., Slightly Cool, Neutral or Slightly Warm.

Before the control of HVAC settings were enabled, most of the comfort complaints were received when the HVAC system was either running in “Standby” or “Unoccupied” mode. This indicates that occupants were not aware that they could change the status by pushing the button on the thermostat. After the control was enabled, the majority of the complaints were from zones which failed to meet the setpoint despite the HVAC settings being correct. We found a number of zones in which occupants felt colder or warmer than the measured temperature, and a few zones which were slow to respond to the changes in zone temperature. The former problem indicates that in multi-office

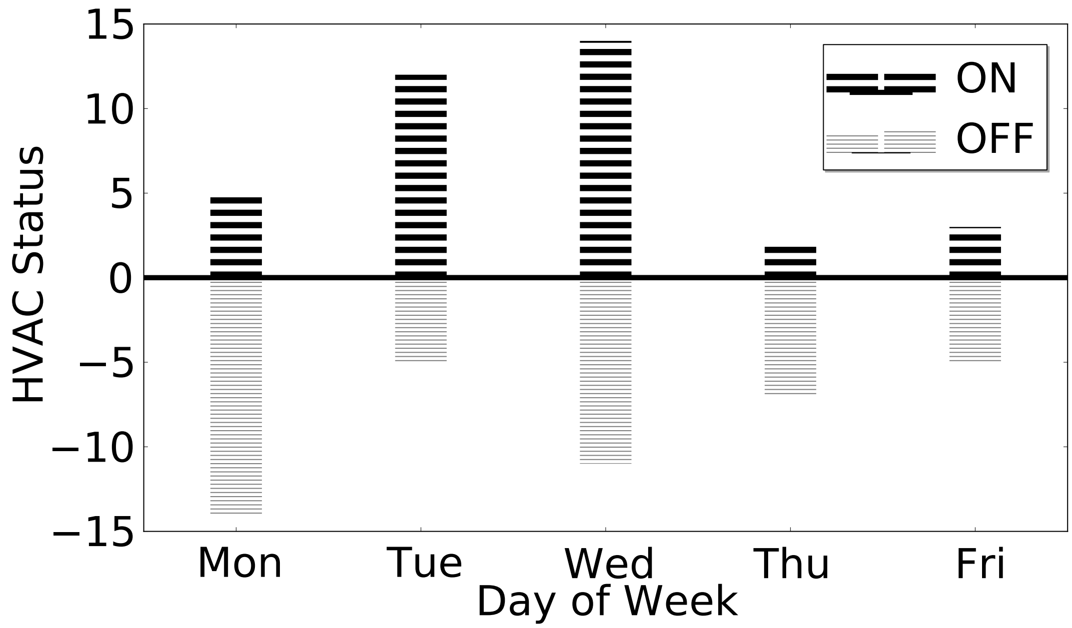

Figure 4.10. HVAC On/Off commands sent by the users for the control week

HVAC zones, a single temperature sensor does not represent the thermal environment of all the rooms in the zone, and a more granular temperature measurement is required for providing better thermal comfort to the occupants. The latter problem could be fixed by tuning the control system of the VAV box to respond faster to the changes in measured temperature.

Figures 4.9 and 4.10 show the distribution of control inputs from the users of WebUI across a week. With the flexibility to change HVAC status, many users put the zone to “Standby” mode if they were not currently in their office. The majority of the ON commands were all received during the period when HVAC would have been normally OFF (6pm - 6am). Some of the users also tried to duty cycle the HVAC system between ON and OFF to save energy. The changes in temperature setpoints do not follow any clear trends, as users changed the setpoint to whatever they felt comfortable with. Most users were content with their change in temperature settings in their first attempt, and only a few users would change their setpoint more than once a day.

Figure 4.12 shows the energy consumption of for six days of the deployment of ZonePAC for the 51 zones involved in the user study. We display the energy values only from 6am - 6pm as there was a bug in the WebUI which kept some of the zones in "Occupied" mode throughout the night on Monday, and there was an exception set to the regular schedule which set the HVAC zones to "Occupied" mode on Friday night. Apart from these exceptions, the trends in energy consumption follow similar trends shown in Figure 4.12. On an average, we measure 5% energy savings after providing control over HVAC settings to the users. The energy consumption on Wednesday is unusually low due to a Demand Response(DR) event from the campus managers which put all the zones in the building to "Unoccupied" mode from 2pm to 4pm. We do not include Wednesday in our savings estimate. The difference between energy consumption on the days with and without energy saving suggestions were negligible.

The low energy savings obtained were expected as occupants are not responsible for the power bills in their offices and are not completely in control of the HVAC settings. However, absolute energy savings do not necessarily capture the motivation of the users to save energy. For example, if a zone is already being conditioned with the minimum amount of airflow possible, changes in setpoint can only increase the energy consumption of the zone. One of the users indicated in her feedback that she would prefer to be slightly cold to prevent reheat of the system and waste energy. However, significant energy savings could be obtained if the occupants were given more options that could save energy. For instance, we plan to allow users to set their schedule on the WebUI, and the HVAC system would be conditioned according to user specific schedule rather than a global schedule.

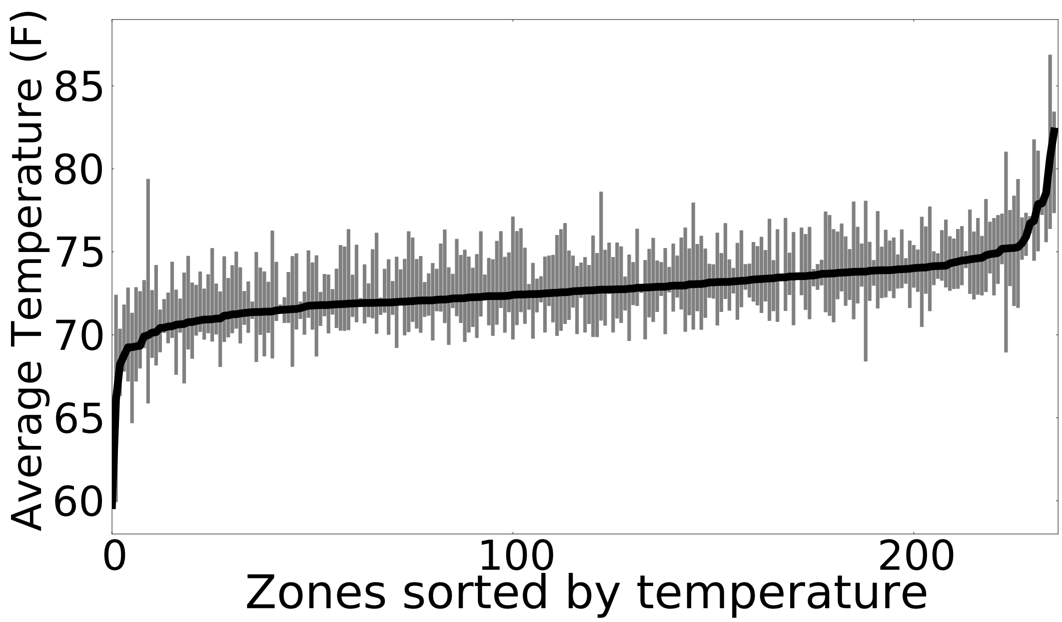

Figure 4.11. Distribution of average zone temperature with error bar for all the zones in the building

4.4 Discussion

We have built ZonePAC for a modern building with VAV type HVAC system, and provide feedback to the occupants using a webapp. There are other types of HVAC systems used in commercial buildings which do not use a centralized plant and use humidity control equipments. From the general principles presented in this paper, it is possible to estimate zone power consumption in different types of HVAC systems using the measurement from installed sensors and modeling the heat transfer process.

Occupancy based HVAC control has been proposed for significant improvement in energy efficiency of buildings [64, 181]. ZonePAC provides insight into a variety of situations in which occupancy information would be useful for saving energy(Section 4.3.1). With power consumption information at the zone level, researchers would be able to design more optimized solutions that would exploit the inefficiencies in current HVAC

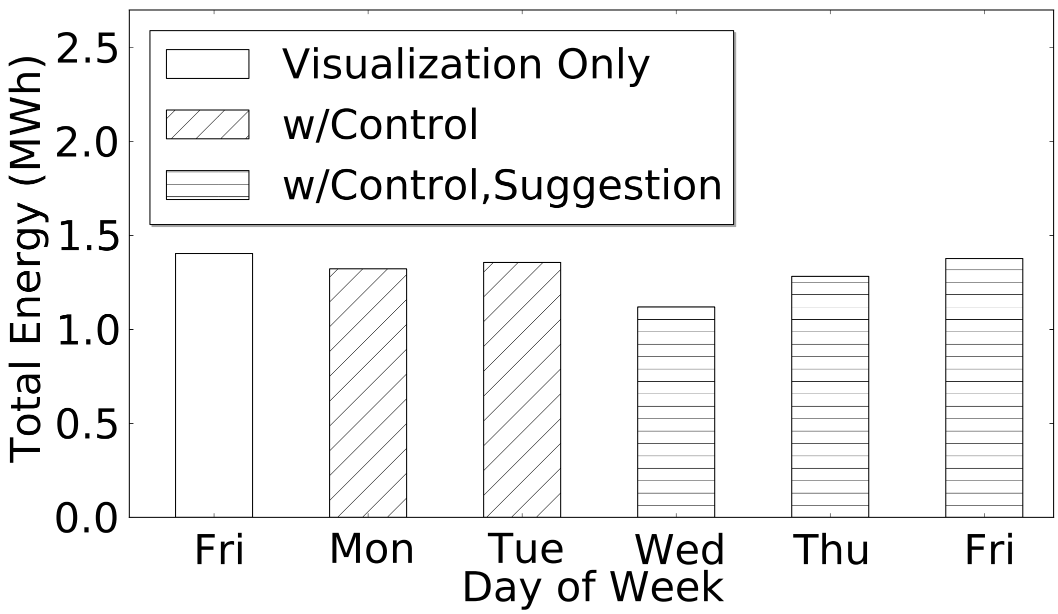

Figure 4.12. Average energy consumption of zones which were part of the feedback experiment

systems.

Feedback from occupants using ZonePAC showed that they care about their energy footprint on the building(Section 4.3.2). Although our WebUI provides the information about HVAC and zone power consumption in a clear manner, it does not adopt sustainable HCI concepts such as use of social network [122] or providing comparison against other zones [143]. We hope ZonePAC acts as a stepping stone to develop better feedback interfaces so that occupants are incentivized to save energy.

Several parameters used for estimating HVAC zone power required careful study of building plans. Although the information from sensors installed in the building were readily available through BACnet, the details about the type of VAV box, the size of the air ducts and water pipes are not provided in a manner that could be easily used for developing applications such as ZonePAC. Automated methods to scan the existing building plans and extraction of relevant information to form a building model would significantly accelerate development of next generation smart building applications.

4.5 Related Work

HVAC power estimation is well understood, and detailed energy analysis can be done using established simulation engines such as EnergyPlus [52] and DOE-2 [85]. Building models are built using the simulation tools, and the HVAC system is tuned based on the results of the analysis. The methodology is followed for both design of new systems [34], as well as existing buildings [47].

Continuous commissioning [126] and automated Fault Diagnosis and Detection(FDD) [99] have been proposed for monitoring of HVAC systems using sensors and BMS. Mills et al. [126] report 16% median energy savings in existing buildings due to commissioning, and the savings were accrued due to faults corrected in all parts of HVAC system [129]. FDD methods have also advanced over the years from practical decision based rules [154], system models [115] to data driven approach [61]. However, the commissioning and energy information systems developed are designed for domain experts, and no feedback is provided to the building occupants. Moreover, energy wastage due to behavorial faults such as anomalous thermostat settings remain unchecked. ZonePAC provides visibility into energy consumption of each zone, and the opportunity to detect behavioral faults using modern FDD methods. By providing feedback directly to occupants, ZonePAC also provides the opportunity for the behavorial faults to be self-corrected.

Prior work has shown that energy feedback can be effective in motivating users to save energy [24, 54]. In a energy conservation study, Peterson et al. show that motivated occupants saved 20% more energy when given feedback on energy consumption in a college dormitory [143]. Recognizing the importance of feedback, plug meters have been developed to provide feedback on appliance power consumption [94]. However, to the best of our knowledge, ZonePAC is the first attempt to provide feedback on HVAC energy consumption to building occupants.

Prior work has given web based feedback on HVAC system to the building occupants. Krioukov et al. [108] build a personalized control system, allowing occupants to view the current status of the system and change settings. Erickson et al. [63] and Jazizadeh et al. [91, 93] gather thermal comfort feedback from occupants, and change the HVAC settings to match their thermal needs. Unlike ZonePAC, none of the systems provide energy feedback to the occupants. Erickson et al. [63] do estimate zone energy consumption using heat transfer equation, but do not validate its accuracy and do not account for electrical power consumption. ZonePAC provides occupants with similar web based HVAC information and includes the estimated zone power consumption.

4.6 Summary

We have built ZonePAC, a real-time HVAC zone power estimation system, built on top of RESTful web service. We present the trends in zone energy consumption, and provide insights into improving the energy efficiency of HVAC system. We find that the usage characteristics of a zone such as aberrant thermostat settings and presence of cooling demanding equipment can lead to significant wastage of energy. Further, we designed and deployed an interactive webapp which provides HVAC sensor information, zone power consumption and control of local HVAC settings to the occupants of the building. We present the data collected from the feedback study over a period of 10 days, and show that HVAC energy feedback to the occupants in commercial buildings could be used to motivate them to save energy.

Chapter 4, in part, is a reprint of the material as it appears in Proceedings of ACM Workshop on Embedded Systems For Energy-Efficient Buildings (BuildSys 13), 2013 by authors Bharathan Balaji, Hidetoshi Teraoka, Rajesh Gupta and Yuvraj Agarwal with the

title “ZonePAC: Zonal Power Estimation and Control via HVAC Metering and Occupant Feedback”. The dissertation author is the primary investigator and author of this paper.

Chapter 5

Software Augmented Thermostats

In this Chapter, we examine the user experience of using the web application we developed for ZonePAC. The web application, called Genie, not only provided energy feedback, but allowed occupants to have better control over their local settings and send feedback to building manager. Across 21 months, the popularity of this web application grew in our building, and here we provide a detailed analysis of how use of Genie compares with that of traditional thermostat already installed in the building. Genie is another example of an application that exploits existing infrastructure to provide a better user experience. As a result of providing such an interface, occupants are not only more comfortable and aware of their energy impact, but also they do not waste energy by blocking thermostats or using space heaters. They also report on faults that causes discomfort. Thus, software solutions can help integrate the stakeholders with the goals of the system.

Office buildings' occupants interact with the HVAC system (Heating, Ventilation and Air Conditioning) using thermostats which provide information such as current room temperature and whether HVAC is operating, as well as enable minor adjustments to the temperature settings. Since the ability to maintain control over their thermal environment has been shown to have a major effect on occupant satisfaction [59, 137], it is critical that these devices are accurate, effective and usable by occupants. In addition, thermostats are a key component of HVAC operation as they complete the feedback loop in the control system and provide insights into several types of HVAC faults.

Most buildings typically use a variant of the ubiquitous *physical thermostat*, under the assumption that they are intuitive to use, without any occupant training. However, a recent survey of 215 buildings across US, Canada and Finland showed that 89% of the buildings do not meet thermal comfort standards [89]. More importantly, in the survey three of the top five reasons linked to occupant dissatisfaction were due to thermostats, specifically (a) thermostats are inaccessible, (b) thermostats are controlled by other people, and (c) HVAC systems do not respond quickly enough to changes on the thermostat. Meier et al. [124] studied the various thermostat designs available today and confirmed how a poor user interface (UI) and occupants' misconceptions have a significant impact on comfort and HVAC energy consumption.

Software thermostats provide an attractive alternative to physical thermostats [30, 63, 92]. They provide occupants with an interface to the HVAC system via a web service or a native application, allowing them to have personalized settings that maximize comfort. Erickson et al [63] showed that use of a native application feedback system led to an improvement in user satisfaction from 25% to 100% in a university building. Furthermore, unlike physical thermostats, software thermostats are incrementally deployable within existing HVAC systems, and are continuously upgradeable with new features or updates to control policies.

In order to investigate the usage of software thermostats and their impact on comfort and energy consumption, we designed and deployed *Genie*, a software-augmented thermostat, directly integrated with our building's HVAC system. Genie displays all essential information conveyed by traditional thermostats in a web application. Since software interfaces can be made richer than physical thermostats, Genie supports additional features such as (i) the ability for occupants to send thermal feedback to building managers, (ii) the display of current weather conditions, (iii) an expanded level of control of the local temperature to $\pm 3°F$, and (iv) the ability to turn On/Off HVAC as needed. Additionally, Genie estimates the energy use by each thermal zone using heat transfer equations [30] and display the results to the occupants of that space as a way to measure their energy impact.

To study real world usage of Genie, we deployed it in the CSE building. Genie has been in use by 220 users over the period of 21 months and in this paper we present a detailed analysis of its usage. We further augment our analysis with survey and interviews conducted at the end of our study to assess the usefulness and usability of Genie to the building occupants. As far as we are aware, this is the first longitudinal study of physical and software office thermostats at a large scale.

Our data show several interesting findings that can serve as key design recommendations for implementation and deployment of software thermostats. We observe that the majority of thermostats are seldom used and find that some thermostats change temperature settings erroneously without user input, leading to significant discomfort and equipment damage. Additionally, our results confirm observations made in prior work that occupants have misconceptions about thermostat operation, and resort to improvisations when they are uncomfortable. Our data also shows that occupants are more comfortable with additional status information and added control for the HVAC system, and that electronic occupants' feedback about their comfort provided immediate insights into HVAC's usage characteristics and faults.

All in all, the study we present here indicates how providing wider thermal control to users does not lead to system abuse and the effect on energy consumption is minimal - while improving comfort and energy awareness.

5.1 Background and Related Work

Maintaining occupant thermal comfort is essential for a satisfactory [71] and productive [157] office environment, and studies show that effective HVAC control by occupants themselves is key [59, 137, 171]. Hence, thermostats and thermal comfort have been studied extensively [59, 98, 141, 171, 172]. The usability of residential thermostats has been explored in depth [141], where thermostats have evolved from simple mechanical devices to digital programmable thermostats. The latest devices even include network connectivity, learning, energy feedback and updated UIs for occupant interaction[1]. On the other hand, the long-term usage of thermostats in office buildings has not been studied as much.

The thermal comfort model followed in most buildings in the US is specified by ASHRAE Standard 55 [163], itself based on Fanger's Predicted Mean Vote (PMV) model [66]. Fanger's PMV model considers various parameters such as air temperature, air velocity, humidity, clothing insulation and metabolism of the occupant to predict occupant comfort. The PMV expresses comfort with a 7-point scale, ranging from Hot(+3) to Cold(-3), and occupants are considered comfortable if the PMV is between +1 and -1. Using this model, engineers design systems to maintain a range of temperature that satisfies at least 80% of the occupants, and provide local control options for minor changes to the temperature setting.

Several studies have shown that occupants are not comfortable in office spaces [39, 89, 96, 98, 136]. A survey by Huizenga et al. [89] shows that 89% of buildings do not meet comfort standards and lists (a) hot/cold regions, (b) thermostat inaccessibility and (c) thermostats controlled by other people, as primary reasons for discomfort. Contextual interviews by Karjalainen et al. [98] found that users are unaware that thermostat exists,

[1]Nest: https://nest.com/, Ecobee: https://www.ecobee.com/

thermostats are inaccessible, they lack informative feedback, users think they are not allowed to control the thermostat, thermostat's dial is stiff or broken, and – most commonly – users did not know how much the thermostat dial should be turned to get desired room temperature. In a follow up work, Karjalainen et al. [97] provide design guidelines based on user studies for office thermostats emphasizing clarity of information, adequate control, acceptable default settings, informative help and aesthetics. However, these guidelines were not tested in practice.

Several variations of software thermostats have been proposed to improve the interaction between occupants and the HVAC system. Murakami et al. [130] introduced a desktop voting system that determines the temperature of the entire floor based on occupants' feedback. Occupants provide feedback whether they want temperature to be warmer or colder, and communicate comfort level on the standard 7-point scale. Although the system showed a promising 20% energy savings, it was only deployed for a few days. Jazizadeh et al. [92] developed a smartphone application that lets occupants provide feedback on required temperature, airflow and lighting level. Their input is mapped to a learning model to determine the HVAC settings. However, they do not deploy their system for real use. Thermovote [63] seeks to overcome the limitations of the PMV model by using a software interface to gather occupants' comfort levels in the standard 7-point scale. The occupant feedback was used to estimate a corrected PMV and the temperature settings of the office are adjusted automatically. User satisfaction rose from 25% to 100% with this strategy over a period of 5 months, with a decrease of 10% in energy consumption. However, the occupants were prompted every 10 minutes for their comfort feedback and were not provided any other feedback on the current status of HVAC. Comfy[2] provides a web interface to office occupants to collect their comfort feedback. The occupants are given a choice between "Warm" and "Cold", and

[2]https://gocomfy.com/

their feedback is used to adjust the temperature setting for the room. These temperature settings are gradually relaxed over time until there is another occupant input from the web interface. Occupants are provided no other information than the simplified “Warm” and “Cool” buttons. Comfy’s case study reports engagement of 77% of the users across 6 months and an energy reduction of 22% due to the relaxed setting employed when there is no input from occupants.

These prior work show the promise of software thermostats to overcome limitations of physical thermostat controls. However, these systems also force users to engage with the system while providing no information on the current HVAC status. It is also unclear how the existing thermostat works with these software systems and what happens when users do not have access to a computer or when there is a software failure. No user study has been conducted to investigate these aspects. Furthermore, the onus of maintenance of these systems is on the building manager, and prior studies indicate that building managers are already overwhelmed with HVAC management issues [128, 166].

We propose an alternative design approach where occupants are provided with essential information such as current room temperature and setpoints, allowing them to take control of their environment and send feedback based on the information provided. Balaji et al. [30] designed a web application that shows the HVAC system status, allows occupants to control their settings and send comfort feedback. This work focused on providing accurate per-zone energy feedback and on quantifying the effect on energy consumption when using a software thermostat prototype across five days. We use a similar design strategy, but study the effect of usage across 21 months. To the best of our knowledge, none of the prior work has studied the actual use of physical or software thermostats in a longitudnal study at a large scale. We compare the usage of hardware thermostats with Genie, studying their use in isolation and when combined. We also show how users’ feedback can be valuable in fault identification, and how information

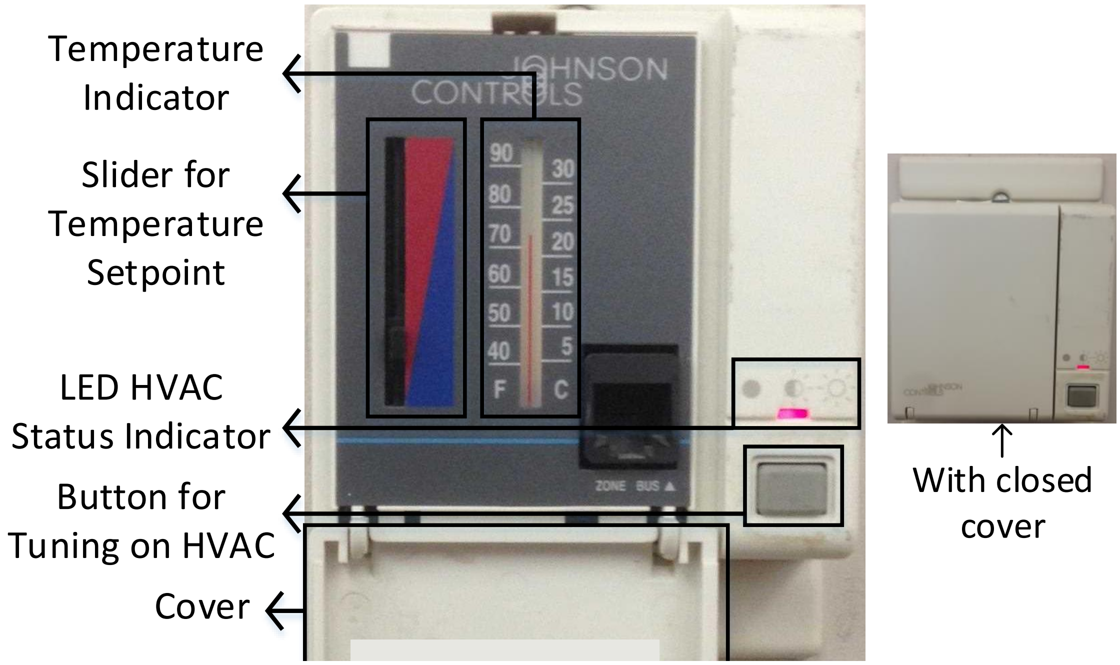

Figure 5.1. Thermostat used in the CSE building. Slider adjusts temperature setpoint by $\pm 1^o F$. HVAC power button turns On HVAC for 2 hours on nights/weekends.

about energy usage improves overall awareness.

5.2 CSE Building Thermostat

CSE building consists of 236 *thermal zones* and each thermal zone typically consists of a large room such as a conference room or multiple small offices. In both cases HVAC is managed by a single thermostat. Figure 5.1 shows the annotated picture of the thermostat in use in our building.

From Figure 5.1 we can see that when the thermostat cover is closed, its functionality is somewhat unclear to occupants. Once open, the thermostat consists of an analog thermometer and a slider to adjust the temperature setpoint by $\pm 1°$F. However, since there is no quantitative feedback on the effect of adjusting the slider, occupants are often unsure about its effect. In reality, the change in temperature due to the slider position is often non-linear and differs betweens zones depending on the degree of flexibility provided by the building manager in response to comfort complaints. Thus, occupants experience is inconsistent across different thermostats.

The LED on the panel indicates system status for that zone – when the LED is On (pink) the HVAC is in *Occupied* mode, when blinking it is in *Stand-by* mode and if the LED is Off, the HVAC is in *Unoccupied* mode (See Chapter 2 for details). If the occupants are in the building during off hours, they are expected to push the grey button to put the system into the Occupied mode for 2 hours. From Fig. 5.1, we can see that these features are not apparent without prior knowledge.

As there is only one thermostat installed per thermal zone even if the zone encompasses multiple offices, spaces without thermostats, i.e. Room 2 in Fig. 2.5, cannot provide direct feedback to the HVAC system. Hence, if an occupant in Room 2 is present during night/weekends, they cannot engage the HVAC system by pressing the thermostat power button in Room 1. Further, if Room 1 has high cooling demands, due for example to usage of heat dissipating equipment such as computers or copiers, Room 2 will be excessively cooled.

5.3 Genie Design and Implementation

We designed Genie to mitigate many of the problems associated with the use of thermostats outlined earlier, and satisfy several design goals. First, we want thermostats to be more accessible and intuitive to use with occupants getting more control of their environment. Second, occupants should be able to send feedback to the building manager when needed. Third, we wanted energy conscious occupants to be able to get immediate feedback on the impact of their settings on the HVAC energy usage. Finally, for the particularly curious occupants we wanted to provide detailed data for the different sensors in the HVAC system.

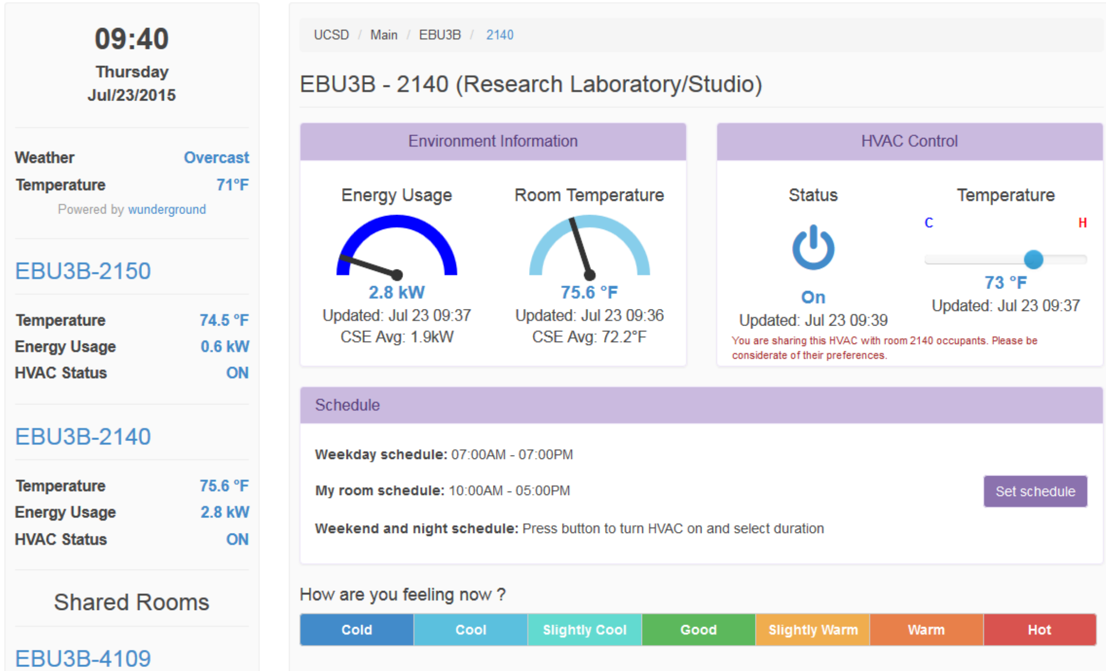

Figure 5.2. Screenshot of the Genie user interface. Users are given access to the rooms they have physical access to. They can change the temperature setpoint by $\pm 3°F$, choose to turn HVAC On/Off and set their own schedule.

5.3.1 User Interface Design

While designing the UI of Genie (Fig. 5.2), we emphasized transparent access to the HVAC data and functionality such as the current zone temperature, the temperature setpoint, HVAC system status, the estimated power consumed by the zone, as well as temperature control. The most pertinent information such as the room temperature as measured by the thermostat, and the energy consumption as estimated by ZonePAC are displayed prominently. We estimate zonal power consumption using available sensor data and heat transfer equations as described in Chapter 4. Users can provide feedback on their thermal comfort on a scale of -3 to +3, compliant with ASHRAE Standard 55 [3]. The Genie UI also shows a comparison of the current zone's temperature and power usage, with the average measurements of the overall building. Finally, we show the "Last update time" depicting the most recent change to the temperature, as a measure of the responsiveness of the system to changes made by occupants.

There are two types of control provided to the users - change in temperature setpoint, and change in HVAC occupancy status. For each of the zones, a common temperature setpoint is set by the BMS. The setpoint is typically set to 72°F, and is modified if the occupants of the zone register a comfort complaint with the building manager. Genie's web-based UI allows users to modify the temperature setpoint of their zone by ±3°F. Wyon et al. [178] show that this range is sufficient to meet the requirements of all the occupants in the building. The setpoints are allowed to be changed once every 10 minutes per zone. To mitigate issues caused by multiple rooms sharing a single zone-level thermostat, we list the rooms belonging to the particular thermal zone in the UI while nudging occupants to be considerate with colleagues in the same zone. If a conflict of temperature preferences occurs, we suggest that occupants resolve this offline as the offices in the same zone are usually co-located.

As explained in Chapter 2, there are three types of occupancy modes supported by the HVAC system in CSE: "Occupied", "Standby" and "Unoccupied". When a user turns OFF the HVAC using WebUI, the occupancy mode is changed to "Standby" during weekday (6am - 10pm), and is changed to "Unoccupied" on nights and weekends. We chose to use "Standby" mode during weekdays as the zone status is likely to be changed if occupants come in to the zone again, and the shallow setback temperature of "Standby" will reduce the thermal discomfort caused to the occupants. Users need to manually turn On the HVAC on weekends and set the number of hours they expect to be in their office through the UI, which puts that zone to the Occupied mode for the entire duration. The change in HVAC status is also restricted to once every 10 mins per zone.

Users can set their own schedule, and the union of all the user schedules in a thermal zone is computed to be the zone schedule (default schedule is set to 7am - 7pm based on our experience).

As shown in Fig.5.2, users can select different rooms using the navigation bar. They can request access to the rooms they have physical access to, which is manually verified before being approved. Genie only takes control of thermal zones whose occupants have registered, while the rest of the zones are managed by the traditional system. Note that the physical thermostat remains operational in zones with Genie controlling them, allowing users to manipulate temperature using either system. Public spaces such as kitchenettes, lobbies, and classrooms can in theory be accessed by any building occupant, which could lead to conflicts and abuse if anyone can exercise control. Hence, we initially restricted Genie access to only the personal offices in the building and then extended read-only access to public spaces a year later. In that way users could send feedback for public spaces to the building manager, who could decide to take action.

In addition to real-time monitoring, control and feedback features, Genie also provides information to users who want to learn more or diagnose faults when they occur. Each of the sensor measurements – airflow, temperature band, status of damper, etc. – can be clicked to get historical values in the “Show more details” section. The navigation bar also provides weather information which has been shown to be useful [124]. The About page illustrates the HVAC system functionality with detailed diagrams similar to the one in Fig. 2.5 (See Chapter 2).

The implementation details of Genie has been presented in Chapter 4.

5.4 Genie Deployment

We announced Genie to all the occupants of our testbed building on October 15, 2013 over email. After the initial announcement, we created an internal mailing list for registered users. Three additional emails were sent to occupants to announce new features over the 21 month period. Users were not prompted in any other way to use this service. As of June 2015, there are 220 registered users with a large number of these users being familiar with technology since they are student, staff and faculty in Computer Science.

In addition to collecting logs and sensor data, we deployed a user survey and conducted interviews with occupants at the end of our study to understand their perspective on Genie's use. Our questions focused on knowledge of thermostats, comfort, features that were useful, effect of energy feedback and improvements that can be made to the system.

In the remainder of this paper we present our mixed-methods analysis based on sensor data and log files collected by Genie from October 2013 to June 2015, combined with qualitative data from 32 survey respondents and 9 contextual interviews. We anonymized data about users and the individual rooms to protect users' privacy as per our university's human research protection office's guidelines and our IRB approved study.

5.5 Longitudinal Study

In our longitudinal analysis of thermostats' and Genie's use we focus on offices with individual occupants, and ignore common spaces such as conference rooms and kitchens. Individual offices make up 152 of the thermal zones in our building, of which 82 zones are controlled by Genie and the physical thermostat while the rest (70) are controlled by physical thermostat alone.

In order to compare usage and investigate emerging patterns we start by focusing our analysis on two main features provided by both the physical thermostat and Genie: (1) change of temperature setpoint and (2) HVAC actuation during nights (7pm - 7am) and weekends. Figure 5.3 shows an overview of the usage of Genie and the thermostats across all office zones. In general, thermostats are used much more than Genie, with thermostat usage constituting 73% of all activity. However, in general, only a few zones show high activity, with 81% of zones showing <5 interactions with the system per

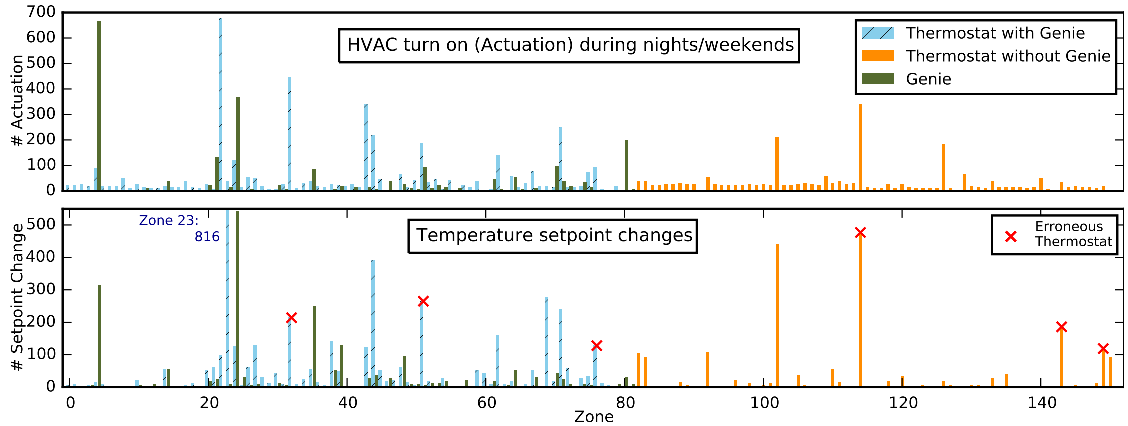

Figure 5.3. Comparison of temperature setpoint changes and actuation during nights/weekends made using Genie and physical thermostats across 152 office thermal zones. Note that both the thermostat and Genie are used for the first 82 zones.

month. To better understand how this overall usage is reflected in the two different interfaces we further analyze users' behavior by breaking it down in Physical Thermostat and Genie usage.

5.5.1 Physical Thermostat

Given the proliferation of thermostats in modern homes and buildings, it is not surprising that occupants used their thermostats at least a few times over our 21 month study. In fact, 74% of our survey and interview participants knew about the use of the physical thermostat's slider to adjust temperature, and 36% about the actuation button for nights/weekends.

Erroneous Thermostats

Upon manual inspection of thermostat setpoint changes we observed that some of these changes were erroneously attributed to user interactions. Figure 5.4 shows an example of frequent thermostat setpoint deviation in the middle of the night. Another thermostat showed an impossible change of $+12^oF$. These setpoint changes not only cause discomfort but also lead to energy wastage and equipment damage. We mark these thermostats as erroneous and do not consider them for further analysis. We consider a setpoint change only when it exceeds one-tenth the maximum range, i.e., for a thermostat slider with a range $\pm 1^o F$, we consider a change of $\geq 0.2^o F$.

Thermostats with High Activity

Some of the occupants are familiar with the thermostat, as one of our interviewee who works regularly on weekends surmises: *"I only interact with it on weekends, because I figure that's when the temperature control is shut down centrally. [...] at some point if I'm sitting still in the office for a long time and the detectors don't detect any motion I think it turns off automatically and it starts getting warmer. I have to occasionally turn it on again."* In reality, the HVAC is not connected to the sensors and turns Off after two hours independent of any motion, but the occupant knew to push the button repeatedly to keep HVAC working. We also found that occupants who work on weekends figured out how to use thermostats over time. As another interviewee explains: *"I didn't even know you could push the button to turn on the AC at that time. So I would remember like... when I would come in on the weekends it would be hot and I wouldn't know what to do about it. [...] it wasn't until later when someone showed me how to use the thermostat and where it was even."*

Upon manual inspection of data from zones which have high usage, we noticed that occupants in these zones have a habit of using the thermostat as soon as they enter the office in the morning or when it starts getting hot later in the afternoon. We saw an interesting correlation across users of thermostats with high activity in our data: in all of the cases the temperature setpoint range was widened to be $> \pm 1^o F$, and the average range was $\pm 7.3^o F$. Zone 23 with an abnormally high setpoint changes was a special case. Two of the occupants in a shared office had conflicting temperature requirements, and they changed the temperature settings several times in a day.

Temperature Control and Discomfort

Our interviews revealed that occupants have many misconceptions over how to use the thermostats and how it affected their office temperature. Many participants assumed the thermostat did not work, as an interviewee states: *"I never thought it ever did anything. On the days it was too cold it stayed too cold."* One of the occupants expressed frustration over the thermostat: *"we didn't realize you had to actually push the button. I mean we were just pushing everything..."*, and as a result improvised their own solution: *"Because it just blows down on me so forcefully that I actually went on top of my desk and I taped a manila folder to my ceiling."* Use of space heaters (even in summer) is also a common solution used by occupants to combat overcooling by HVAC. Such improvisations not only cause excessive energy waste, but also leads to equipment damage. Occupants who did not have a thermostat in their offices often did not realize they had control over the temperature. As another interviewee states: *"I was freezing to death. You can shut the door if that helps. I was freezing to death and I didn't know where the thermostat was to make at least my area...at least comfortable for me..."*. Our surveys corroborate these findings reporting an average comfort level of 2.9 out of 5 with the use of thermostats.

5.5.2 Genie

After looking at our log files we discovered that the overall usage of Genie seemed to be much lower than the thermostats (see Fig. 5.3). However, after carefully considering the possible reasons behind this potentially disappointing result, we recognized that Genie allows for a wider temperature control than thermostats, which may result in reduced number of changes as occupants are comfortable with that temperature. Furthermore, the physical thermostat turns On the HVAC only for 2 hours at a time, while Genie expands that to up to 14 hours. Thus, it is possible that Genie's absolute actuations count does not correspond to effective usage of the interface. Moreover, our survey indicated that comfort level after using Genie increased to 4.2 out of 5 vs 2.9 using thermostats, with the difference being statistically significant ($F_{1,33} = 29.42, p =< 0.0005$). To investigate how users consistently used temperature control across 21 months and why they reported such an increased comfort level, we further analyzed Genie's logs.

Engagement over time

Although Genie logs were only available for 122 of 220 users and for 13 out of the 21 months of deployment (logs are not available for the initial two months and for six additional months as indicated in Fig. 5.5) we were still able to get a detailed view of Genie's usage characteristics. Based on this analysis we were able to categorize Genie's users into four distinct types:

- **One-time**: Users visit the page a few times after registration and do not visit again.
- **Short-term**: Users actively use Genie for ≤ 2 months.
- **Sporadic**: Users whose regular use of Genie is spread across more than 2 months, although interspersed with gaps in their usage for several months.
- **Consistent**: Users who used Genie consistently for more than 6 months.

Figure 5.5 shows usage data from logs for three example users from each category and Table 5.1 summarizes the results across all users. From our analysis we conclude that a significant portion (45.1%) of users were actively engaged in using Genie for more than two months after their registration.

Table 5.1. Percentage of Genie users per category: one-time, short-term (<2 months), sporadic (gaps in usage) and consistent (> 6 months).

User Types	One-time	Short-term	Sporadic	Consistent
% Users	24.6%	30.3%	23.8%	21.3%

We investigated further to find out the specifics of when and why people wanted to use Genie through our surveys and interviews. Our data revealed that Genie was especially useful when users did not have a thermostat in their office. As one interviewee explains: *"I didn't actually use the older thermostat because I don't have a thermostat in this room. ... for me Genie is great because I have personalized access to my room."* Users also liked the precision of control made available by Genie, as one survey respondent comments: *"Digital control of the temperature is very, very useful. Moving the slider [on the thermostat] still leaves a lot of uncertainty as to what exactly will happen, and the temperature setting helps."* One of the survey respondents commented on how temperature control affected his productivity: *"Genie is awesome and has made a real difference in my ability to work in my office. I get migraines that are correlated with higher temperatures, and Genie allows me to set the office temperature to 67, which greatly reduces occurrence."*.

For the *consistent* users we found that Genie is often actively used because offices are uncomfortable on a regular basis. As one user says: *"I generally think its fine ... only in the late afternoon I have to make it cooler"*. On the other hand *sporadic* users use Genie occasionally because offices are already quite comfortable:, as reported by one of the interviewees: *"I mean, I haven't used it a lot. I just...uhm...will change the temperature if it's like too hot or too cold. And on the weekends if I'm working here I'll turn it on because the AC doesn't turn on automatically."*. *Short-term* users often indicated how the initial interest was high and then it vanished with time: *"I used it frequently at some point as in usually over the weekend, I would tweak the temperature* *through the web interface. Then nowadays I don't come in as often in the weekends. So if I do come, I might set up the thermostat manually coming in the room. Then usually I don't have to deal with it until I leave...so yeah, I may not have been used the web interface for a while now."*. Finally, *one-time* users typically forget the URL, or the password for their account, and do not visit the web page after their initial registration. As one user indicated: *"It looks pretty friendly. It's more of a matter of out-of-sight...out-of-mind."*

Dual Thermostat Usage

Many of our survey respondents revealed they used the physical thermostat despite having a Genie account. One of the main reasons echoed by several users was that the thermostat was sometimes easier to access compared to opening the computer and controlling the temperature via the web app. As one user says: *"I don't have to pull up the web interface. It's just a dedicated slider on the wall, which is pretty easy for occasional tweaks."* Another reason for using the physical thermostats was that many occupants were confused about the relationship between Genie and the thermostat on the wall. As one survey respondent explains: *"I don't quite understand how the physical thermostat and Genie interact and so I often adjust both."* Both Genie and the thermostat were functional, but Genie does not directly reflect the changes made through the thermostat slider. Having access to both controls confused some users; we realized that this is a design flaw and we are planning to address that in our future work, with the Genie interface directly reflecting the physical thermostat changes.

Thermal Feedback from Occupants

Genie introduced the ability to send feedback on how comfortable occupants are in their offices. Some users were unclear on the utility of the feedback, and whether it affected their HVAC settings. Users therefore initially sent feedback to express their comfort level or justify their control actions. As one of the feedbacks said: *"Felt cool* *for the past 1-2 wks. Just tried changing the room temp from 73 to 75 hoping we feel a difference!"* Other users would ask questions about the interface: *"AC seems to be off during weekend. Can I/anyone turn it on?"* Many users initially sent *"Good"* feedback, which we interpreted as being satisfied with the HVAC system. However, the majority of feedback messages we received were linked to users being uncomfortable despite changing their temperature settings, or complaining about Genie or the HVAC system not working correctly.

Occupants' feedback also served an additional means. As facilities managers do not have time to inspect the problems in every room in a building, faults that occur at the office zone level are often ignored and remain undiscovered unless an occupant sends a complaint [166]. The feedbacks from Genie proved to be a valuable resource to identify such faults and correct them to improve occupant comfort. Figure 5.6 shows the distribution of comfort feedbacks sent by the users along with the mean values of their comfort level in the standard 7 point scale. As can be seen from the graph, most users only send a few feedback messages. These messages usually correspond to extreme discomfort levels. The textual feedback sometimes elaborates on the issue. For instance, one user comments: *"I am wearing a sweater but I am cold in the office. Walking in the corridor, I am much colder. My hands are really cold."* Sometimes the users will directly send a symptom of a fault, for example: *"It's 64 in here now, though the setting is the max allowed at 73."* During our deployment these messages allowed the building manager to uncover many unknown or unreported system faults. Examples include sensors which stopped reporting information, thermostats which were blocked by computers, dampers getting stuck, Genie not reporting data, etc.

Energy Feedback to Occupants

Genie provides the estimated energy consumed within the thermal zone to the users and a normalized average energy consumption for the building to allow users to compare their energy usage with other zones in the building. While prior work did show the effect of energy feedback on occupant's behavior (5% reduction), the results were preliminary with a small set of users and over 5 days [30]. As part of our study we analyzed the effect of energy feedback across 21 months. Figure 5.7 shows the effect on zone energy consumption due to a temperature setting change by the user. We compare the energy consumption two hours before and after a change made by the user to infer if the user made an energy conscious decision. The data shows that the energy consumption could equally decrease or increase, and there is no bias towards energy conserving settings. As we show later, Genie zones show a 3% decrease in energy consumption on weekdays and 31% increase in weekends compared to physical thermostat zones.

In addition to the effects we registered in our system, we investigated the personal occupant's perception in terms of added energy consciousness. Our survey revealed that users were divided on whether they were more energy conscious after using Genie, with a mean score of 2.8/5. Many users commented that their comfort was a clear priority over the energy consumption. As one interviewee states: *"If I'm hot dude...I'm going to turn it on. I mean uh...I got work to do. You know...if I got to use a little bit of wattage I don't care."* Some users agree that it is good to be aware of the energy consumption, but it does not change their behavior in any way. As one user comments: *"I do care, but admittedly would do whatever I needed to be comfortable without regard to energy consumption."* A subset of users, however, expressed a desire to better understand their energy footprint, and wanted more indication in the interface on how they could act upon decreasing it. One user states: *"I think it would be helpful even to see what your peers...what their* *energy consumption is. Just to kind of see if I'm conserving a lot more, or...wow...I'm way over the top. Maybe I need to start being more conscientious about things."*

38% of the users responded that they were more energy conscious with the feedback Genie gave them. Therefore, although many users do not care, energy consumption's feedback does have an overall impact in behavior on an important subset of our user base.

Genie's Limitations

Despite the overall positive feedback from our users, Genie introduced its own set of problems and exposed some limitations. A common issue among many users was that the HVAC control was limited to once every 10 minutes. This was our design decision to protect the HVAC equipment from excessive usage. As a consequence of this conservative setting Genie was unresponsive to some specific user's behaviors and intended interactions with the system. For instance, when users made a minor mistake with the temperature setting, or accidentally pressed a button, the system would not let them change the settings for the next 10 minutes. As one user explains: *"I was trying to adjust it and I moved it down and I slipped...and so I let go of the mouse and it only moved a half degree. Then it was like you can do this again in 10 minutes..."* Another major issue occurred when Genie was temporarily unavailable due to system updates or maintenance. We have had only a few instances which led to unavailability over some weekends, and at that time users had to revert to using thermostats. One user sent us a message when Genie was down: *"For some reason the A/C wasn't running ... I don't have a thermostat in my office (it's in another office next to mine that I don't have access to), so genie was my only hope"*. Hence, when Genie fails, an alternative such as manual thermostat override should be available. This is important in case occupants cannot access a networked device or in case of a software failure. Thus, the system needs to be carefully designed to address these scenarios.

Additional Features

Genie provides several other features, most of which remain unused. Most users do not set their personal schedule if the default schedule is enough to make them comfortable. The history of each sensor can be obtained by clicking on the measurement (e.g. 72°F) in the UI. Although many users indicated history was useful, they did not realize this feature was available. We provide detailed sensor data and details about what each sensor means, but this is almost never used.

We did not provide users access to shared spaces such as conference rooms and lobbies due to conflicts that may occur between requests. To extend Genie functionality we synchronized the online conference room calendar with the Genie schedule so that users have control over the HVAC settings for the duration of the meeting. The HVAC is turned down during non-meeting times to save energy. Although many users liked this feature when we announced it, most users either forgot about it or did not eventually use it.

5.6 Impact On the HVAC System

As Genie provides more flexibility for occupants to control their temperature and turn HVAC On/Off, one of the risks from a building manager perspective is that Genie could lead to an increase in overall energy consumption or deviation of operation from the HVAC managament's original design and intended purpose. To investigate the impact of this added flexibility, we compared the overall energy consumption and the extent of control exercised using Genie versus the physical thermostats.

Energy Consumption

We first focus our attention on how Genie impacted energy consumption. Figure 5.8 shows a comparison of normalized energy consumption for weekdays and weekends separately. The weekday graph indicates that the energy consumption of Genie zones is comparable to the thermostats, and overall Genie zones save 3.5% energy, The difference is statistically insignificant ($F_{1,70} = 0.001, p = ns$), but we can still confidently say that Genie's usage is not linked too more energy consumption during the week. On the weekends, Genie zones consume more energy on average, and this points to the fact that users utilize Genie regularly to actuate the HVAC on weekends. Hence, this excess in energy consumption is justified as it serves to keep the occupants comfortable. Genie zones consume 31.6% more energy than zones with thermostats during the weekends but the difference is statistically insignificant ($F_{1,70} = 2.59, p = 0.11$). Comparing the overall energy consumption considering both weekends and weekdays, Genie zones consume 3.4% more than thermostat zones, but it is again statistically insignificant ($F_{1,70} = 0.092, p = ns$). Therefore, long term use of Genie has not had a significant effect on HVAC energy use.

Temperature Swing

As the temperature setting can be changed up to 6°F in Genie, users may tend to change the temperature settings to its extremes which may lead to excessive energy consumption or large swings in airflow. We compared the deviation in temperature settings across different zones over 21 months. Surprisingly, some physical thermostats show more deviation than Genie, with up to 6°F standard deviation. This can be attributed to those physical thermostats whose range have been increased by the building manager in response to comfort complaints. The occupants do not know by how much they are changing the temperature as there is no indication in the thermostat. There are a total of 63 out of 152 thermostats whose range is larger than the designed ±1°F, and the building manager does not keep a track of these thermostat changes. On the other hand, despite having the freedom to change the temperature by 6°F in Genie, surprisingly most extreme changes in Genie are around the 4°F mark. The standard deviation for the change is ±2.0°F, compared to ±3.5°F in thermostats, and this difference is statistically significant ($F_{1,>>100} = 95, p < 0.0005$). All in all, we can see here that providing users with clear information and more control results in better overall behavior than providing a slider without information on the thermostat.

5.7 Lessons Learned and Design Guidelines

Our combined analysis of thermostat's and Genie's usage data with user interviews and surveys revealed that the thermostats in our building fail to provide clear status and feedback information to occupants. In addition, some occupants do not know where thermostats are located, or do not have access to them. These findings confirm the outcomes of prior studies [96, 98]. We showed here how software-augmented thermostats can alleviate these issues as well as provide additional features such as getting feedback from occupants. Systems like Genie are especially attractive for existing buildings, where retrofitting can cost from $500-$2,500 for each thermostat [53]. Based on our experience with the design and development of Genie and our longitudinal study, we discuss below six specific design guidelines that we believe will guide and inform the future design and development of software-augmented thermostats.

Relationship to Physical Thermostats

Software thermostat should not aim to replace the physical thermostat. Thermostats have been around since 1572 [173], and many occupants are familiar with its basic functions. We claim that physical thermostats can still provide basic functions and occupants should be able to use them when they do not have access to a networked device or when there is a software failure. However, it is important that both the physical and software thermostats show a similar interface, and are synchronized with each other's updates, so that users do not get confused with the relationship between them.

Clarity of Information

Users value the precision of information available in a software graphical interface, since it allows them to better comprehend what the HVAC system is trying to accomplish. Thus, although a simplified interface is necessary, it should not leave out essential information such as if the HVAC is working currently, and what temperature settings are in use. Our data shows that users visit the software interface only when they feel uncomfortable, and that accurate information allows them to infer the current status quickly.

Provide Adequate Control

Users expressed immense satisfaction in having the ability to control their local office temperature, which confirms findings from prior studies [59, 137]. Showing users how much control is available to them and how it affects the HVAC operation allows users to make intelligent decisions. Our data shows that users are careful with their control decision and the impact on HVAC operation and energy consumption is minimal.

Comfort Complaints and Feedback to Managers

Comfort feedback not only provides building managers information on the level of comfort of occupants, but also helps in identifying hard to detect faults such as thermostat blockage. Fault detection algorithms and control strategies can use this information to crowdsource comfort information and further tune the HVAC system as per user requirements.

Actionable Information on Energy Usage

Many users like energy consumption feedback, and a number of them even indicated active interest in using the information to save energy. Prior studies have shown that providing actionable energy reduction information can be effective in residential settings [150]. Users need similar information in offices, as one interviewee requested: *"... if by changing this 1 degree I would save this percent of energy, I would do it."*

Prediction and Additional Features

In a software interface, users expect fast reaction times to inputs. Thus, the system needs to hide HVAC latency and show the effective change that will occur later. Another strategy is to provide users with predictions of HVAC behavior due to a change in setting, which has shown to be effective in homes [153]. Further, features such as historical data should be intuitive to discover for users to actually use them.

5.7.1 Limitations

We note that our study of physical thermostats and Genie usage has been conducted in a university building located in a temperate climate zone in the US. The analog thermostat we studied is from Johnson Controls, a popular vendor who install HVAC systems across 125 countries. Although the thermostat model we consider is installed across most buildings in our university campus, it predates the latest digital model provided by the vendor. Therefore, more research is needed to verify our findings across different cultures, climate zones and types of thermostats. Finally, our occupants are all from a Computer Science building, and more research is required to generalize our findings to other population pools.

5.8 Summary

We designed a software augmented thermostat, called Genie, that provides pertinent HVAC status information to the occupants and enables adequate control over their local temperature. We introduced additional features such as comfort feedback from occupants as well as energy consumption information to increase occupant awareness. To evaluate Genie, we deployed it in a five floor university building, and studied its usage as compared to the physical thermostat alone over 21 months.

We show that occupants have misconceptions about thermostat usage, and some of them did not know where thermostats were located. Genie users were satisfied with the clarity of information and level of control available, and 45% of users showed longer term engagement with the system. In addition, comfort feedback from users provided insights into non-obvious HVAC faults. The energy feedback provided by Genie increased user awareness with a subset of the user base motivated to change their behavior. Based on our usage analysis and design experience, we outlined key design guidelines for software augmented thermostats.

All in all, we believe that the insights presented in this study will benefit researchers and designers who want to further investigate temperature control in office buildings and develop user facing smart building applications

Chapter 5, in part, has been submitted for publication of the material as it may appear in SIGCHI Conference on Human Factors in Computing Systems (CHI '15), 2015 by authors by Bharathan Balaji, Jason Koh, Nadir Weibel, Yuvraj Agarwal with the title "Genie: A Longitudinal Study Comparing Physical and Software-augmented Thermostats in Office Buildings". The dissertation author was the primary investigator and author of this paper.

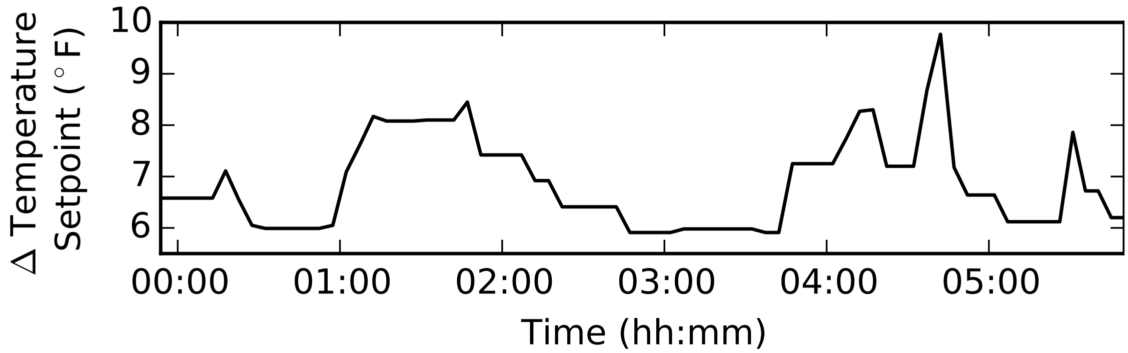

Figure 5.4. An example of erroneous thermostat behaviour where changes occur frequently in the middle of the night. These changes are frequent in identified erroneous thermostats.

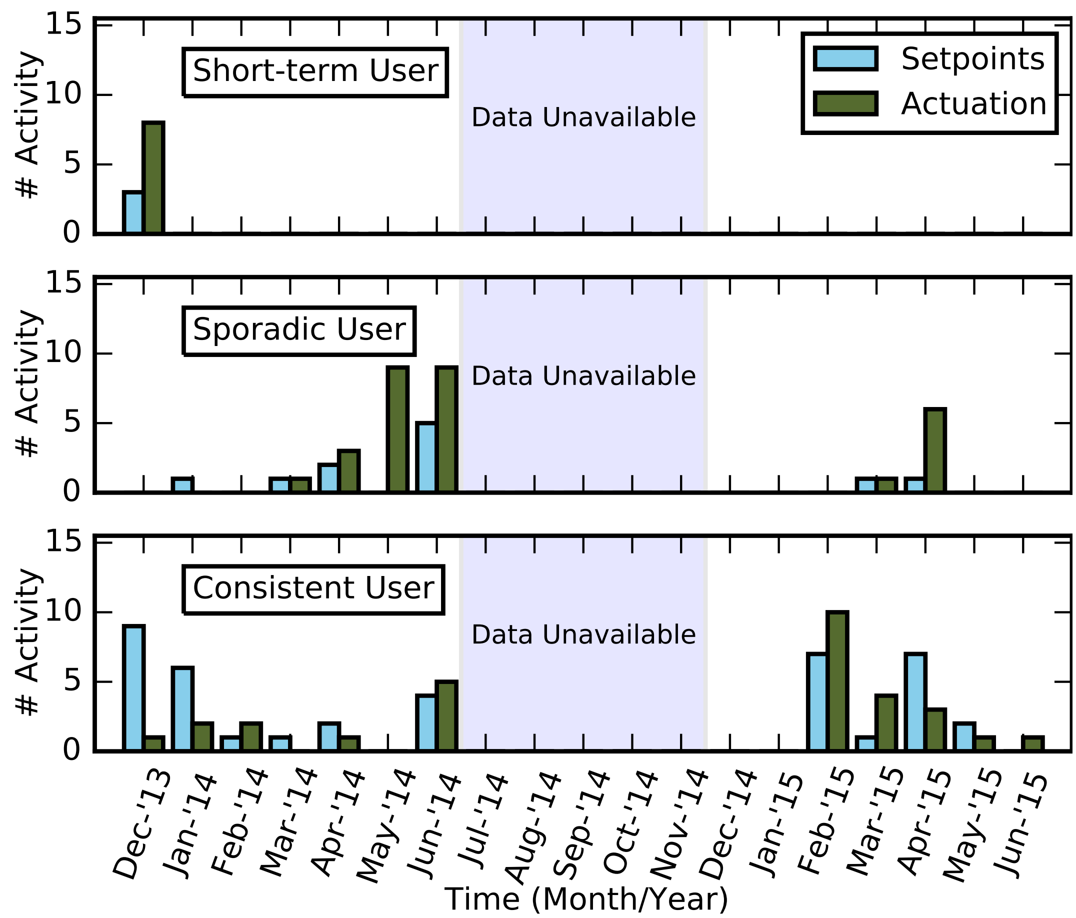

Figure 5.5. Genie activity comparison for a representative user from each category.

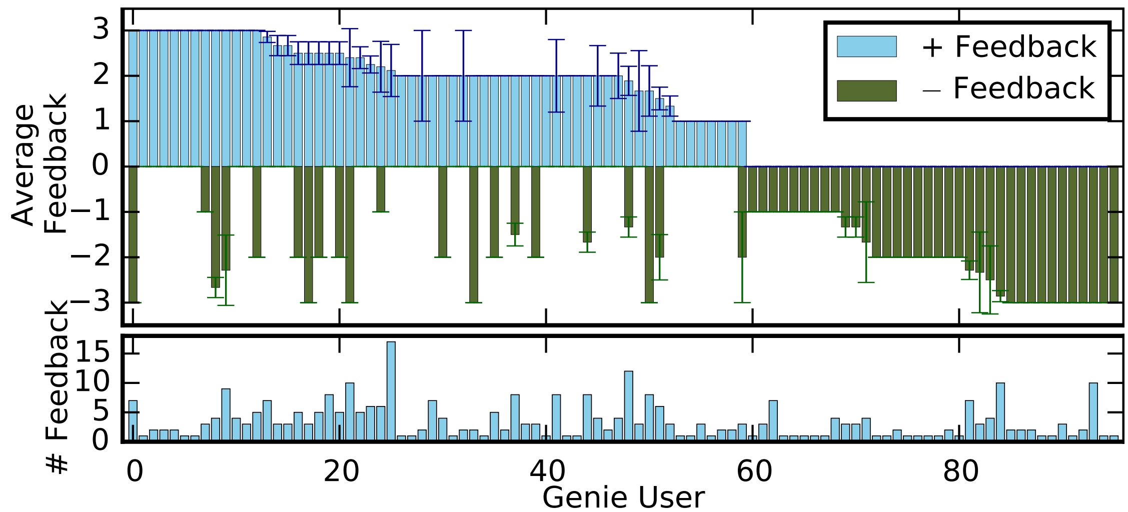

Figure 5.6. Distribution of feedback given using Genie across all users on standard PMV 7-point scale. Feedbacks help identify extreme conditions in the offices and insights into HVAC faults.

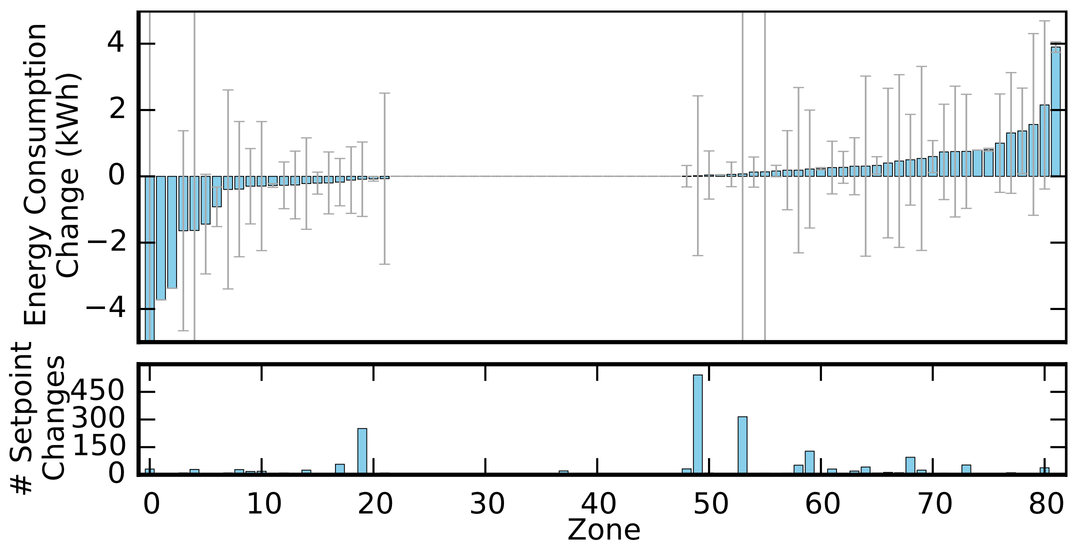

Figure 5.7. Average energy consumption difference 2 hours before and after a change in setpoint by a Genie user across all zones.

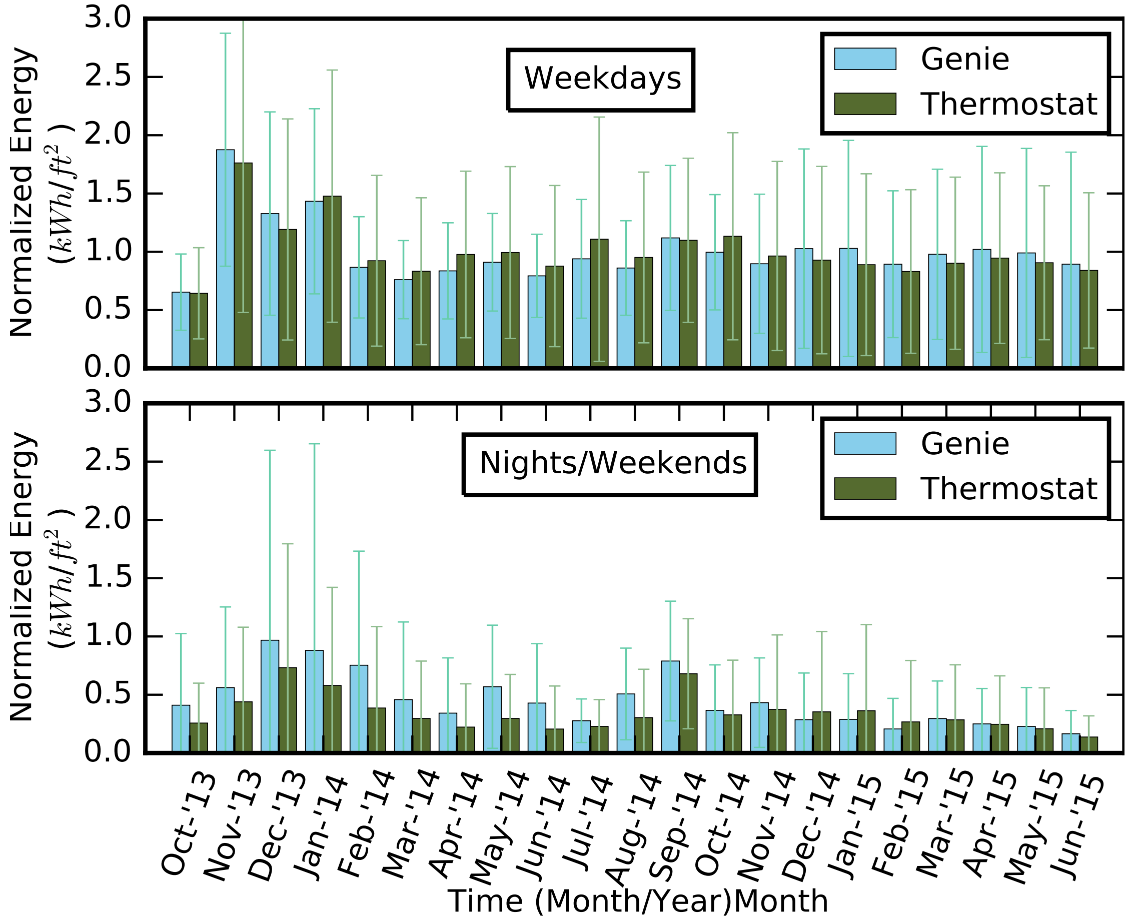

Figure 5.8. Comparison of Genie and Thermostat zone energy consumption across 21 months. The energy consumption has been normalized by area to account for varying room sizes. Other confounding factors such as presence of windows is assumed to be randomly distributed.

Chapter 6

Building Management Software

In this Chapter, we shift the focus from occupants of the building to the building manager and maintenance personnel. Building managers are essential to the success of energy management measures in a building. They can monitor building energy, attend to occupant needs and fix problems that lead to energy wastage. Software tools and interfaces available to them can enable them to be efficient and ease maintenance hurdles. Here, we present an analysis of the management systems used in buildings today and identify key pieces missing that leads to inefficiencies. We also present our fault management system that addresses these problems, and tease apart the challenges in implementing such software systems on a real building. This work builds on top of existing building management solutions, and strengthens my thesis that with the help of well designed software applications, large energy savings can be obtained in modern buildings.

Building Management Systems (BMS) are used for management of Heating, Ventilation and Air Conditioning (HVAC), lighting, security, irrigation, etc. Figure 6.1 outlines BMS's main architecture. They consist of complex set of equipment and control programs used by a few key operators such as building managers, maintenance personnel or service contractors. As large equipment can be controlled using BMS software, even small actions can affect the comfort and energy efficiency of the entire building.

We studied the usage of BMS across five institutions in the US and outline here the challenges of everyday use of these systems. Our analysis shows that these challenges are often due to incorrect design and development of the current systems, and we suggest design changes to help overcome them. Using contextual inquiry [35], we interviewed participants with diverse designations who interact with BMS software regularly. We focus on HVAC management using BMS as it is the only system installed in most buildings, and the software has advanced to address the challenges that emerge in a complex system. Our contribution from our building manager user study is therefore two-fold: first we contextually document key challenges of BMS use across operator roles, BMS software, type of institution, and geographic location; second, we distill important insights and design directions that can be incorporated in the development of the next generation interfaces.

To illustrate challenges in practical deployment of BMSes that overcome the limitations brought out by our user study, we have designed BuildingSherlock (BDSherlock), an extensible, web service based management framework for fault management in building HVAC systems. The goals of BDSherlock are to solve the current challenges in HVAC fault management. We have designed BDSherlock with two key principles in mind - (1) provide extensive HVAC information and allow fault reporting through

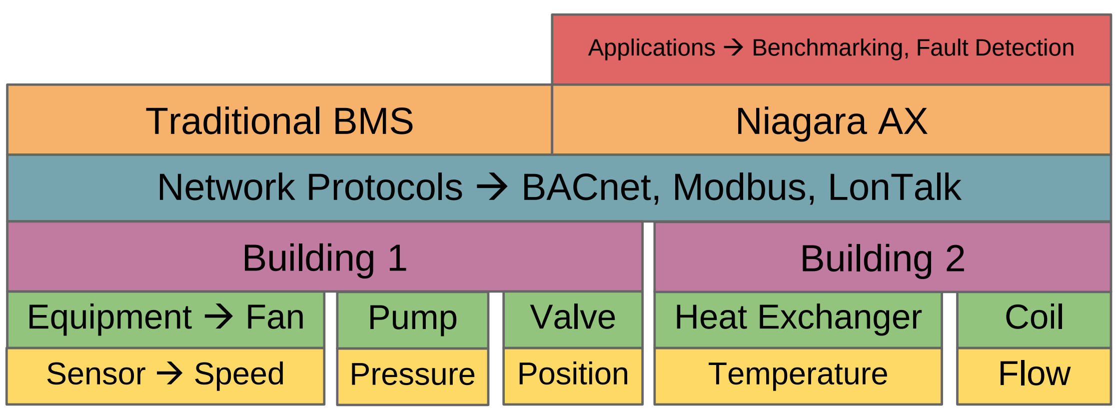

Figure 6.1. Architecture of a typical Building Management System

an open API so that sophisticated algorithms can be implemented and re-used across different buildings, and (2) provide relevant information to the facilities personnel to analyze these faults and so that they can act on them quickly. Our framework encourages collaboration between the stakeholders – facility managers, equipment vendors, service providers and building occupants.

As part of BDSherlock, we have implemented a fault dashboard for facility managers where building information from different sources is organized in a single interface that provides a prioritized list of faults and tools for analyzing these faults. We have integrated useful contextual information such as estimation of energy wastage due to a fault, and direct feedback from occupants of the building to aid building managers in rapid exploration and fault remediation.

We deployed BDSherlock in the CSE building and implemented algorithms to detect faults at the VAV units, and present our summary of findings here. Some of the faults we found were unique, and have not been found in commissioning studies in other buildings on our campus. We captured a total of 88 faults, with estimated savings of 410 MWh/yr.

6.1 Background and Related Work

Several stakeholders are involved in the management of HVAC systems. BMS operators can be grouped into three categories: people who ensure (a) day to day operation, (b) comfort of the occupants, and (c) energy efficiency. Facility managers and operators are common to all buildings in an institution. Building managers work at the building level, and help maintain all systems in the building. Issues which require technical expertise are forwarded to the facilities management. In addition, service contractors who specialize in certain services conduct repairs, upgrade software, etc. Finally, commissioning consultants are hired short term to ensure that building systems are functioning correctly and recommend changes to equipment or control programs. We use the term "*operator*" to represent a generic user of BMS.

Despite the need to employ different professionals for managing buildings, institutions cut costs by having less staff members, relying primarily on the BMS to assist with monitoring and automation of HVAC systems. BMS are routinely installed in newly constructed buildings, and older buildings are upgraded to BMS-enabled equipment. BMS provide services such as monitoring of sensor data across the system, programming of control sequences for proper equipment operation, reporting faults detected through sensor data, graphical visualization, and providing access control across users.

Traditionally, BMS are provided by HVAC equipment vendors such as Johnson Controls, Siemens, and Automated Logic, as end to end customer solutions. Although proprietary solutions typically do not communicate to systems from other vendors, communication protocols such as BACnet and Lontalk were introduced to increase interoperability across vendors. Nevertheless, compatibility remains a challenge as each vendor uses their own extensions of common protocols.

To overcome a number of those interoperability challenges, vendor-agnostic BMS platforms such as Niagara AX[1] and OpenBMCS[2] have been developed that provide interoperability across different vendors as well as support development of third party applications. Standards such as oBIX[3] and Haystack[4] are being developed to semantically represent building information and access data through REST APIs.

Despite the efforts so far to improve BMS, the work of operators remains challenging, and is becoming increasingly complex. With hundreds of buildings each with thousands of sensors, the design of effective, efficient and satisfactory BMS is key to

[1]http://www.niagaraax.com/
[2]http://www.openbmcs.com/
[3]http://www.obix.org/
[4]http://project-haystack.org/

overcome challenges and avoid operator overload. We report on a variety of issues that operators face with today's systems and outline directions for next generation BMSes.

Few studies have focused on user requirements for addressing building management issues. Lehrer et al. [113] study building experts and occupants for better visualization of information. They focus on the information and standards in BMSes that would help building operators monitor building performance. Khire et al. [102] develop a fault management framework based on user centered design, and they address fallacies in modern BMS with an architecture similar to BDSherlock. However, they do not provide the results of their user study, nor do they present the details of their deployment. Our interview results are in agreement with these studies showing building operators are indeed overwhelmed with the number of issues they can handle, and modern BMSes could be improved significantly to reduce manual labor, provide more information to benchmark building performance and fix faults. Unlike prior studies, with BDSherlock we focus on the system level changes to both the BMS and the fault management framework. In addition, we have implemented a prototype for a real building to demonstrate our framework's efficacy and share experiences gained with our deployment.

Several commercial fault management frameworks are available, and we examine SkySpark as a typical example [162]. SkySpark supports integration across data sources with compatibility with various standards - gbXML, oBIX, and external information such as utility data. SkySpark supports fault analysis with their custom designed Axon language, that allows users to write sophisticated rules using available library functions and can detect faults by executing these rules in their engine. Although this broadens the type of faults that can be detected considerably, the framework restricts the complexity of algorithms as they need to be executed within the Spark framework. Data analysts cannot use popular programming tools - Matlab, R, Python, along with their vast collection of available libraries. The APIs exposed are restricted for visualization, and does not

Table 6.1. Innovations and improvements introduced by Niagara AX with respect to the building operators needs outlined in the interviews.

Contextual and Historical Data	Third party products for search and visualization, including historical sensor data, can be added on top of Niagara AX. Operators can have personalized dashboards that show relevant indicators such as jumps in energy consumption.
Naming Convention	Introduced component object model for naming building entities – sensors, sensor metadata, actuators, and control sequences. Supports rising standards such as oBIX and Haystack.
Fault Reporting	SkySpark is a popular third-party tool for analyzing HVAC sensor data for fault detection and diagnosis that can be installed on top of Niagara AX. It supports open standards, and provides relevant information on each fault to the user.
Data Analysis	The platform supports storing of sensor data and auditing of user actions to understand the historical performance of the HVAC. Historical data also allows users to easily benchmark performance with respect to their past data.
Vendor Lock-in	JACE box is an intermediary between vendor equipment and the Niagara AX. Boxes contain drivers to port vendor specific protocols to proprietary protocols. Data is exposed to third party applications using BAJA open standard.
Search and Reporting	Third party applications developed in Java or as a web service enables data querying using SQL-like language. Reports can be built to periodically summarize performance, usage and energy characteristics.

Table 6.2. Portfolio of participants in our user study, the BMS platform they used and the number of buildings managed by the institution

Institution/Company	Participants	BMS	Buildings
University of California, San Diego	Energy manager (P1), HVAC technician (P2), Building Manager (P3)	Johnson Controls Metasys	100+
University of San Diego	HVAC technician (P4)	Siemens Apogee	50+
Carnegie Mellon University	Asset Preservation Manager (P5)	Automated Control Logic, Automatrix, Siemens, Johnson Controls	100+
University of California, Berkeley	Two building managers (P6, P7)	Automated Control Logic, Siemens, Barrington	100+
San Diego County	HVAC maintenance operator (P8)	Tridium Niagara AX	520+
Johnson Controls Inc.	BMS technician (P9)	Johnson Controls Metasys	-
Enernoc Inc.	Energy efficiency consultant (P10)	Various BMS, Enernoc Insight	-

include fault management. Our BDSherlock design builds on top these automation solutions, and mitigates their limitations by advocating an open framework and flexibility in implementation.

6.2 Understanding Building Operators

To understand the current experiences of building operators, we studied the use of BMS by ten building operators across five institutions managing more than

870 buildings (Table 6.2). We followed a hybrid semi-structured [120] and contextual learning [35] model that elicited direct feedback from the users and engaged them in detailed description of their experiences. We conducted all but two interviews on-site, at the participants office or in a nearby conference room, with the remaining two conducted remotely. We took detailed notes during all interviews, and with participants consent recorded audio for 7 out of 10 interviews, and for five of them we collected videos of the operator's interaction with the system. We transcribed the recorded interviews for in depth analysis.

Data collected was analyzed by two researchers who worked in the area of smart buildings for five years and an expert in human-computer interaction and user interfaces. We exploited elements from grounded theory [164] to perform a thematic analysis and we grouped emerging elements into seven key challenges that building operators currently face.

6.2.1 Challenges in Building Management

Regular maintenance of buildings include addressing comfort complaints, resolving BMS alarms indicating faults in building system, performing periodic tasks such as replacement of dirty filters or installing/upgrading of equipment or software.

All of our participants felt they were understaffed and underfunded to handle the number of issues they need to address as summarized by P10 (consultant): "*Almost everywhere we go they don't have enough maintenance staff to do things right most of the time. So that's very common that they do the quick fix rather than the right fix*". Operators admitted that they were aware that many of their buildings are operating inefficiently, but did not know which ones were inefficient and how inefficient they were. Their main strategies to overcome this issue were based on their past experiences, and on the age of equipment. They also relied on commissioning – i.e. manual checks of specific buildings

– to identify major inefficiencies. Although sometimes effective, these strategies are not scalable, do not transfer well to other operators, are not documented, and are not sustainable for a large campus.

To better understand the underlying reasons behind these challenges we now detail seven key problems that we identified across our interviews:

Simplistic Fault Reporting: The way faults are reported to the operators are simplistic and create an alert every time a sensor value goes beyond a pre-specified threshold. The underlying cause of the alert is not easily identifiable: it could be related to a sensor drift, an error in configuration, a damage in equipment or a combination of factors. Alerts which are related to each other are not grouped together, causing a deluge of alerts for the same fault. Therefore, faults often accumulate and some of them remain unresolved. For instance, one of the operator showed us >100,000 alerts that accumulated in her system that she would never be able to catch up with. Also, although energy efficiency is increasingly important, sensors installed only target critical faults to reduce costs.

Missing Contextual and Historical Data: Analyzing the underlying cause of a fault is vital to locate the problem and fix it. To this extent, various levels of data needs to be available to operators to analyze the status of the system. While historical sensor data was provided in all BMS, in one of the universities data was only stored for 3 days, and in another the trending had to be started manually, which at times happened only after the discovery of a fault. P1 expresses his frustration: "*one of its [data trending] biggest limitations is that I'm always being told that 'Don't ask us to map so many points' or 'Don't ask us to set too many trends' because it'll overload the system.*" Furthermore, relevant information is distributed across a variety of sources. Contextual information such as equipment location, connection to other units, model number, etc. are not available. In one institution, the power meter data is accessed separately from the

HVAC sensor data, and the relationship between different equipment is only present in architectural drawings. This missing information results in the operator visiting the site in person to diagnose the fault, which increases the time to fix it considerably.

Inconsistent Naming Convention: Even within the same institution and BMS, we witnessed lack of standard naming conventions across different buildings. Names are manually labeled by different operators (and, even renamed over the years), and therefore, do not follow a consistent naming convention. As P10 explains: "*I don't know if they do anything to make their point names consistent [...] sometimes they'll just leave them as AV1. And that's not very helpful to anybody.*"

No Integrated Data Analysis Tools: BMS only provided raw sensor data, and did not support easy addition of computed information. Thus, participants reported having to perform many calculations by hand to analyze data. As P1 explains: "*I was looking at that specific room [...] to see things like 'Ok does the total supply flow match up to the total exhaust flow?'I was doing summations in my head of these numbers, like ... is this making sense?*". BMS do not provide common metrics to be used with analysis such as benchmark against other buildings or calculate efficiency of operation.

Vendor Lock-in: Traditional BMS lock-in the facilities with their equipment, so operators can only use vendor-provided hardware and software. Even when open protocols such as BACnet are adopted, vendor extensions of the protocol do not match with other vendors. These vendors also provide versions of Niagara AX platform which are incompatible with other vendors. As P8, who uses a Niagara system complains: "*I wish that there was a way that I can put a third party item onto it so I don't have to upgrade the whole system. But that's not available. It has to come as a part of what they sell.*"

Forgotten Overrides: When fixing certain faults, it is common for operators to temporarily override current settings to conduct repairs. However, operators frequently forget to release their override leading to faults. BMS support storing past operator operations, but the number of entries is limited. No option is provided to integrate notes while overriding settings to enable later analysis. As P10 recalls from experience: "*... extremely common would be operators leaving something in override. So either switching the handoff auto switch on a VFD by hand and just leaving something flat out. Typically these are things that I'll do with the intention of having it be temporary but then you're too busy to come back and fix the root of the problem so it just stays for weeks or years.*"

Recognize Occupants Misuse: Occupants cause faults because they are unaware of how HVAC systems work. Space heaters are commonly used during winter, causing excessive energy wastage. Refrigerators or other appliances block thermostats or air vents, causing incorrect operation of HVAC. It is difficult to understand and recognize from the BMS when a faulty operation is due to a misuse: "*... a pet peeve of mine is when people set their air conditioner to like 69 degrees and the [occupants] have the 1500 watt electric heaters going on their desks at the same time.*" [P4].

6.3 Discussion

As outlined above, facility managers struggle with integration of different systems, lack of standardized data formats and are locked into vendors after the initial installation. The infrastructure for historical data collection is not robust, contextual information that is key to understanding the underlying situation is missing, and data analytics are simplistic, putting the onus on the operator to do calculations.

Addressing those issues through vendor-specific HVAC systems is hard, since they are monolithic and have not been designed with flexibility in mind. However, the vendor-agnostic solution provided by Niagara AX has the potential to overcome some of these limitations (Table 6.1). One of the institutions in our study had Niagara AX installed in eight (out of 520) buildings. The building manager reported that BMS is easier to use and helped with benchmarks and faults.

Despite the functionality introduced by Niagara AX, only one of the operators we interviewed makes use of it. This is due to several reasons. First, many of our interviewees were unaware of the benefits of vendor-agnostic platforms such as Niagara AX. Second, even though it helps in the long-term, installation costs in existing buildings are non-trivial: much of the costs account for manual translation of data from existing building(s) to Niagara platform. Finally, changes to another system result in at least a temporary drop in productivity and will introduce a variety of new and different interfaces that operators are not willing to embrace easily.

6.3.1 Designing Next Generation Interfaces

While the approach put forward by Niagara AX and the overall idea of vendor-agnostic BMS is useful, more radical changes are needed to exploit the inherent energy saving capability and improve occupant comfort in buildings. Based on the outcome of the interviews we identify a finite but carefully investigated list of design recommendations next.

Automation: Several parts of BMS remain expensive to create or program because of lack of automation and use of standard machine readable formats. Graphics for floor plans and equipment connections, for example, are hand drawn. Building architectural and mechanical plans are available in CAD drawings with standard formats like Green Building XML, and should be leveraged by BMS. Similar automation features can be developed for discovery of sensors installed, population of metadata such as location, and acquisition of equipment datasheets by providing them in machine readable form.

Data Analytics: With the amount of data available from sensors in the HVAC system, a wide variety of analytics and diagnostics can be performed. However, operators in our study stressed how the current system: "*is definitely suboptimal right now, I'm overwhelmed by the amount of data that's available and the lack of automation of it.*" [P1]. Several tools and techniques have been created by researchers [52, 99] to develop HVAC system models, detect faults and inefficiencies, and simulate different scenarios. However, these tools remain disconnected from BMS. We believe that integrating them into the BMS will provide useful insights to operators.

Contextual Information: When operators make changes to the HVAC system (e.g. temperature control), they do not get feedback on the effects on energy or comfort. Instead, they judge these effects based on the raw values of sensors provided, although effects are only visible after several minutes to hours. Operators need immediate feedback using form and metrics that is relevant (e.g. type of room or energy consumed). It will help them be more efficient and reduce mistakes. P1 commented on this point stating how "*If operators were getting immediate feedback in terms of energy waste ... as soon as you put that override in it could be like 'This is going to cost the university $40,000 this year, are you sure?'*"

Communication Among Stakeholders: Communication among management personnel is done through phone calls, emails and work orders, all of them kept separate from BMS. Novel BMS need to support contextual annotations to avoid misunderstanding. For example, the energy manager can infer that an override is in place due to repairs being conducted. BMS also need to involve occupants as their actions directly impact the system. An operator, for instance, can see that HVAC is active late at night due to a special occassion.

User Support: BMS usability would greatly increase by adopting standard user involvement practices. Instead of requiring specialized training, operators can be provided with wiki pages and discussion boards to encourage learning and adoption of best practices. BMS needs to support varying requirement of different roles in management, and provide appropriate levels of abstraction, permissions and data analytics to assist them in their daily work.

6.4 BDSherlock: Design and Implementation

To further concretize the lessons learned from our building management study, we design a fault management framework as a third party application on top of traditional BMS. We focus on fault management as it is one of the primary services provided by the BMS and improvements can lead to large savings in energy and reduce operator effort. To reduce skilled labor, we focus on providing adequate information to the operators so that they can determine the status of the system at a glance, prioritize their actions to maximize returns and understand the context of a fault to diagnose it quickly. We rely on algorithms to provide comprehensive coverage of faults that occur in the system, so that operators spend minimum amount of time on manual inspections. In addition, we seek to remove the restrictions in the current BMSes so that data from diverse sources can be integrated into the system, and the third party fault management applications can be developed. We have designed BDSherlock to satisfy these objectives by emphasizing information management, flexibility of implementation and scalability for large deployments. We advocate that fault management be a priority from the design phase of building systems, standardized data formats be used to encourage sharing of information as well as re-use of software applications, automated analysis be deployed to detect faults comprehensively.

6.4.1 Software Architecture

BDSherlock employs web services as the backbone of the framework as they are easy to use, scalable, and flexible to implement different types of services and policies [25]. Our framework is a composition of web services, each serving different

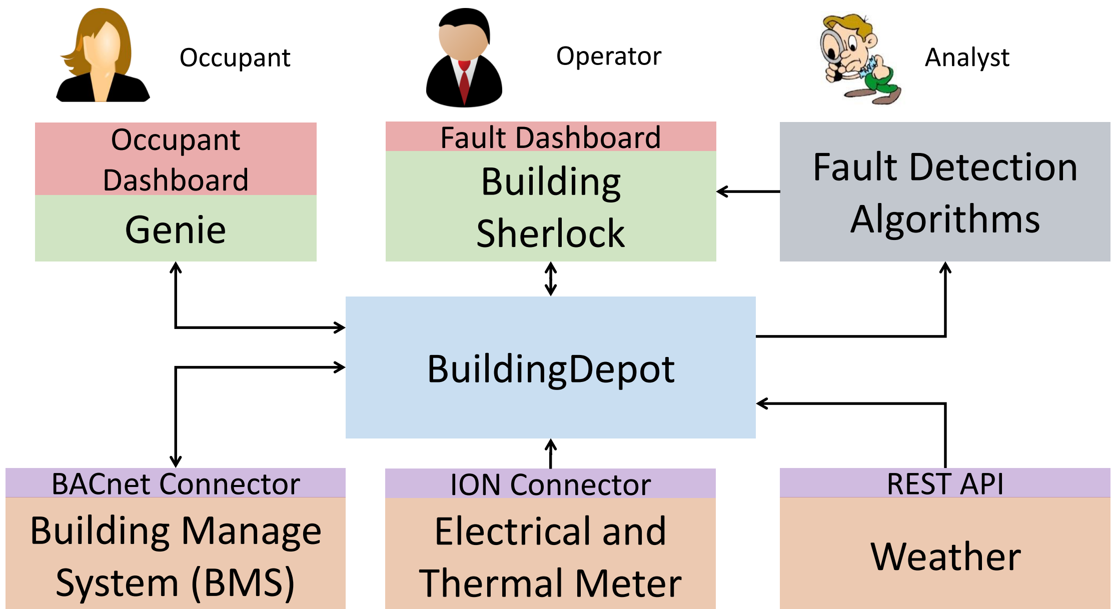

Figure 6.2. System Architecture of BuildingSherlock

functionality of HVAC management. Figure 6.2 depicts our software architecture.

Our framework builds on top of our information management web service, BuildingDepot (BDDepot) [22]. Currently our naming convention is the standard imposed by our university, and it can be easily extended to support standards such as Haystack [1]. Unlike traditional information management systems, BDDepot also supports actuation of control systems. This capability allows us to support applications such as automated functionality testing and fault correction.

BDSherlock is composed of three additional services built on top of BDDepot. The BDSherlock core service is used to register algorithms, report faults, and for user facing applications like the fault dashboard. The fault detection service uses sensor information from BDDepot, and reports faults to the BDSherlock service. Finally, the occupant service, called Genie, provides a web interface to occupants, and reports their feedback to BDSherlock.

6.4.2 Fault Detection and Reporting

BDSherlock relies on third party algorithms to detect faults in the HVAC system and report them using RESTful APIs. Developers register their algorithm with BDSherlock service, providing information such as the type of sensors they would use, the type of faults they detect, and parameters of faults they will report. By separating the detection algorithms and fault reporting, we do not impose any restriction on the type of analysis used by algorithms. Developers are free to employ tools of their choice, and implement sophisticated algorithms [99]. Developers use information from BDDepot for analysis, and we emphasize revealing as much information as possible to enable valuable insights on system status. We provide long term historical sensor data, and contextual information such as sensor location, equipment type, room usage model, etc. As we use a common naming convention for metadata, algorithms can be reused across buildings, which reduces cost of deployment.

Algorithms report detected faults to the BDSherlock service. We introduce a library of *fault types* to enforce standard naming convention when reporting faults. Each fault type is associated with specific faults associated with parts of equipment or sensor. Examples include ‘damper stuck’, ‘temperature high’, and ‘valve leak’. To help facilities personnel prioritize and analyze faults, we encourage algorithms to report additional metrics about a fault such as confidence of detection, energy savings, impact on comfort, duration of the fault, expected return on investment etc. In this paper, we focus on confidence of detection and energy savings.

With use of available information and sophisticated algorithms, we can detect many faults in HVAC. However, manual analysis is still needed for diagnosis of a fault, and some faults may never be detected as it occurs only in certain modes of operation. Automated functionality testing [84] can identify such faults and help with fault diagnosis.

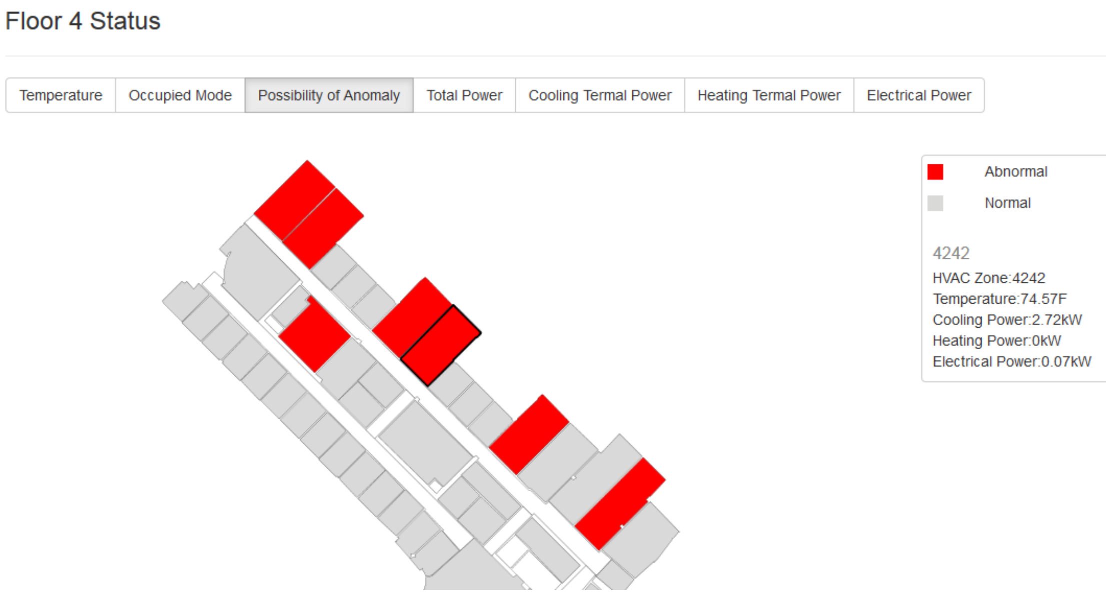

Figure 6.3. Floorplan view in BDSherlock Faults Dashboard.

For a large campus such as UCSD, there are many faults which are neglected in lieu of more critical faults. In the meantime, automated fault correction [67] can temporarily fix faults or run the system in a degraded mode using actuation. BDSherlock architecture supports both these applications with the use of BDDepot access control and BACnet priority table for actuation of HVAC system. In this work, we focus on detection, and leave study of actuation algorithms for future work.

6.4.3 Occupant Feedback

Occupants can provide valuable information about faults that occur in HVAC. With the help of our Genie web service [30], we enable active occupant participation in keeping their environment comfortable, and report faults. Details of Genie design and features has been provided in Chapter 5.

6.4.4 User Interface Design

The faults dashboard is designed to make it easy for building managers to check on faults. The front page of the UI shows an overview of the building, providing building power consumption information, and faults found in each part of the system - cooling system, heating system, or terminal units in each thermal zone.

The user can navigate the system hierarchically – from the building level, to each thermal zone and individual rooms. The UI shows a visualization of the HVAC control system as a symbol diagram with representation of fans, pumps, cooling coils, sensors and their connections. The page shows live data being collected from BDDepot, and is refreshed every minute. Each sensor value can be clicked to get historical data. For each floor, a floorplan view of the building is provided, with color graphs for different parameters - temperature, energy consumption, faults reported. Figure 6.3 shows the floorplan for one of the floors in our building testbed. The rooms in the floor are highlighted when moused over, and the important sensor values gets displayed in a sidebar. A click on this room leads to a symbol diagram of the corresponding VAV unit with relevant sensor information. These symbol diagrams are similar to Figures 2.4 and 2.5.

A faults tab shows the list of faults reported. Faults can be sorted or searched based on system, subsystem, type of fault, confidence, or time of reporting. Clicking on each fault provides the details of the fault. Figure 6.4 shows a snapshot of this page for one of the faults reported. Details include the user facing information provided by the algorithm - fault summary, confidence level, fault type, and it automatically plots one month's data from the relevant sensors. All the faults reported by algorithms are grouped together if they indicate the same type of fault, equipment and location. Thus, the user can analyze reports by multiple algorithms for a single fault and it also reduces

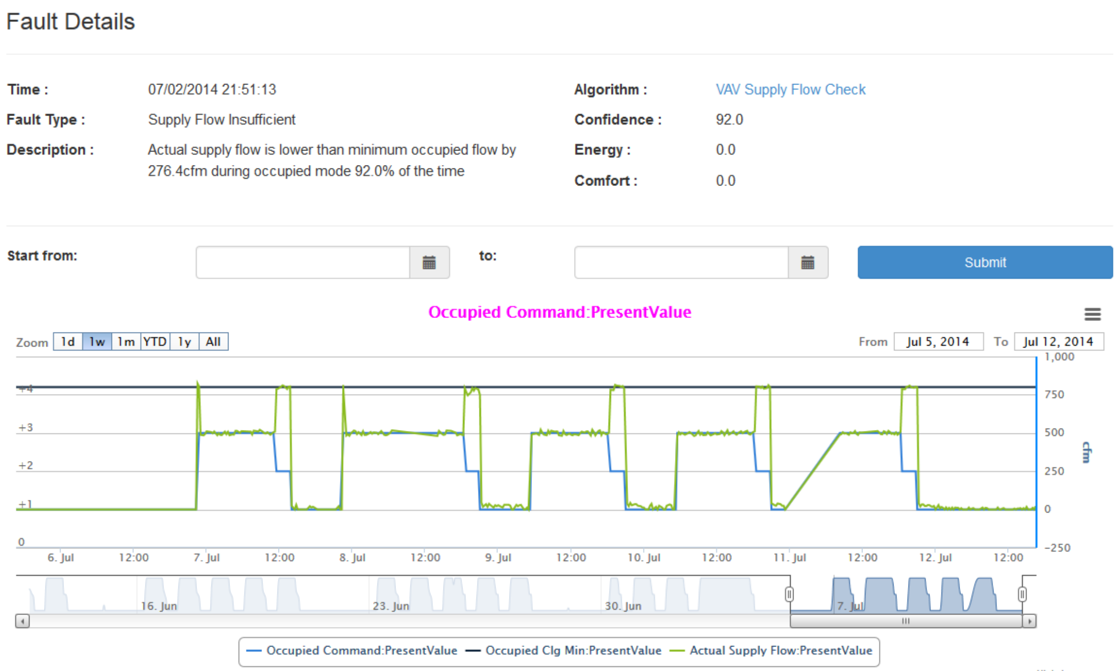

Figure 6.4. Fault report view in BDSherlock Faults Dashboard.

data deluge due to multiple reports.

6.5 Deployment

We deployed BDSherlock in the CSE HVAC system. Details of the HVAC system and our data collection methods have been provided in Chapter 2.

6.5.1 Manual Inspection

Before using the available data to find faults, we performed manual inspection to ensure correctness of sensor data. An experienced Johnson Controls contractor checked only the central unit of the HVAC as it time consuming to manually inspect each VAV box.

Some of the sensors as well as the hot water thermal meter were found to be miscalibrated, and were fixed. The contractor also noticed that Electronic to Pneumatic (EP) transducers for cooling coil valves were broken, which kept the valve in a fixed position leading to significant leakage. This is an example of a minor fault left undetected that costs <\$100 to fix but led to a massive energy penalty. The economizer damper was broken, letting 100% outside air into the air mixer irrespective of outdoor temperature. This leads to substantial wastage, and even though the large damper replacement cost >\$26,000, the expected return on investment is less than a year. The control loop tied to the economizer was also not working, and the contractor fixed this as well. The temperature of the supply air was fixed at 55°F as per design, and dependent on the cooling demand of the top three energy intensive zones in the building. The contractor changed it to be dynamic, with the setpoint determined by the average demand of the zones on each floor. This changed the average supply air temperature to 65°F. He made a similar change to the hot water supply temperature setpoint, reducing the average setpoint from 180°F to 120°F. Both of these changes led to significant savings, and made the HVAC power more proportional to the demand.

6.5.2 Algorithms

Many fault detection techniques have been proposed in the literature [99], and we focus on data driven algorithms and rules based detection. We focus on the terminal units (VAV boxes) in the system as they are typically ignored in commissioning processes.

Machine Learning

We implemented several popular unsupervised machine learning algorithms – subspace methods exemplified by Principle Component Analysis, correlation based methods based on the intuition behind Strip, Bind and Search [70], and developed our own algorithm, called Model, Cluster and Compare which compared thermal zones with similar characteristics to detect faults [131]. We did not implement supervised algorithms as it required sensor data with fault labels which is not available to us.

Our analysis revealed several interesting anomalies which are not normally tested for by the Facilities Management. For example, there were some thermal zones in which the temperature guard bands do not change with HVAC mode of operation (Section 2.2), and some zones which had unusually high airflow during unoccupied mode; both of these faults were due to programming errors. However, the machine learning algorithms could not identify all the instances of the anomalies in the building, and there were several false positives in the results. Therefore, we designed customized rules that would capture these anomalies accurately, and identified many faults in the HVAC after we implemented them. The details of our data driven investigation is presented in prior work [131].

Rules

Rule based methods are the most prevalent and well known way to detect faults. However, we aim to implement rules that go beyond checking for threshold violations by a single sensor as is common in BMSes. Rules can be powerful as they can be programmed to capture complex non-linear interaction in the control system, and precisely detect faults. Design of such rules which provide high accuracy require significant domain knowledge, usage characteristics of the building and thresholded based on historical performance. In practice, it is difficult to implement such specific rules, and generic rules given by certification bodies [49] are implemented instead.

We implement generic rules suggested in literature, as well as specific rules based on the anomalies obtained from our data driven analysis [131]. Our FM was only interested in faults that caused egregious wastage as they did not have time to fix minor faults. Hence, we use *conservative thresholds* to avoid false positives. For instance, when we checked for airflow leakage in terminal units, almost all the units leaked ¿10 cfm, but the FM was only interested in those which had ¿50 cfm leakage. We read data from BDDepot, and processed the data based on fault type, and report the fault if violation of

Table 6.3. Summary of HVAC faults detected

Rule	**System**	**# Instances**	**Estimated Energy Waste**
Supply Flow Excess	VAV	8	44.9 MWh/yr
Temperature Setpoint	VAV	27	167.6 MWh/yr
Insufficient Flow	VAV	10	-
Thermostat Adjust	VAV	33	-
Insufficient Cooling	VAV	8	-
High Temperature	VAV	1	-
Economizer Damper Stuck	AHU	1	197.8 MWh/yr
Total		88	410.3 MWh/yr

threshold occurred for a significant period of time. The number of violations is coded as the *confidence* of the fault, giving the frequently occurring fault a higher confidence. While reporting, the algorithm provides the location of the fault, and the sensor which is defective. It also provides a human readable summary giving a description of the fault, the amount of violation, and confidence of detection. An example summary is "Actual supply flow is lower than minimum occupied flow by 276.4 cfm during occupied mode 92.0% of the time". The complete list of rules implemented, along with the results obtained is presented in Section 6.6.

6.6 Evaluation

At the VAV controller level, we detect faults due to misconfiguration, insufficient or excessive airflow, cooling or heating when unoccupied, and any airflow leakage. We could not detect heating valve leakage due to lack of discharge air temperature sensors.

We were surprised to find a large number of configuration errors in the VAV controllers. In 27 zones, the temperature setpoints did not change with changes in occupancy mode. This fault kept the HVAC operational in these zones even during nights and weekends, an extravagant waste of energy. We found a similar configuration anomaly in 6 zones, where the airflow setpoint was set to be high in unoccupied mode even when the zone did not require cooling. A particularly wasteful example of a conference room is shown in Figure 6.5, where the airflow setpoint is high enough to cool the room to heating temperature setpoint, which then led to heating coil usage to increase the temperature of the supplied air. Thus, *simultaneous heating and cooling occurred in an unoccupied zone due to a configuration error*. The Johnson Controls contractor surmised that these errors were probably caused due to misunderstanding at the time of initial building commissioning. As these type of faults are not expected to be present, and not mentioned even in standardized rule sets [4, 49], they would have gone unrecognized even during a retro-commissioning process. Such faults are also not detected in any of the fault frameworks we have examined.

Another uncommon fault we found was associated with thermostat adjust available to occupants. By default, the thermostat allows occupants to change their temperature settings by $\pm 1\,^{\circ}$F. If this change was found to be inadequate to keep a comfortable temperature within the zone, the maintenance operators increase their range upon a request from occupants. These changes in thermostat adjusts effectively shifted the temperature band maintained by HVAC in different occupancy modes (Section 2.2). Due to a flaw in the initial configuration of thermostats, these changes to thermostat adjusts remained in effect in unoccupied mode. For zones with large change in adjust, the temperature band shifted enough to require cooling or heating during unoccupied mode. We found 33 zones with adjusts greater than $\pm 3\,^{\circ}$F. FM found that it is hard to change the programming to disable these adjusts in unoccupied mode using the BMS, and we plan to leverage BDDepot to implement the fix ourselves. We skip the energy analysis for this fault, as it hard to estimate the energy consumed by these zones without such a shift in temperature band.

We found several faults that would have been detected with the standard set of rules. The temperature in one of the kitchens was found to be particularly high despite maximum cooling by the VAV box. Upon inspection, we found a water cooler was placed in front of the thermostat, which led to incorrect measurement. This is an example of a fault where the occupants were unaware of the implications of their actions on the HVAC system. We also found 10 zones to have insufficient airflow during the occupied mode, and our basement computer labs to require excessive amount of cooling. These faults occurred due to the recent change in static pressure settings during the manual inspection (Section 6.5.1). Although this setting leads to energy savings, insufficient airflow can be a health hazard, and will make occupants uncomfortable.

Although we used conservative thresholds to detect faults, there were a few false positives from these rules. Some zones such as mechanical rooms, kitchens, and restrooms were supposed to be ventilated even in unoccupied mode. Similarly, one of the rooms was under renovation, and the rules found the zone to have insufficient ventilation. As our rules did not take these usage characteristics initially, they led to false detection. This tells us the importance of contextual information to improve the accuracy of detection algorithms. We have removed these faults from Table 6.3.

We also confirmed the economizer damper fault from sensor readings. The FM has already fixed the configuration errors, and are in the process of replacing the economizer dampers. There are several types of fault that still remains to be examined, such as tuning of PID control loop, and analysis of equipment efficiency at the central units. We plan to address these in future work.

6.6.1 Energy Analysis

We leverage our prior work, called ZonePAC, to estimate the energy wastage due to faults [30]. ZonePAC used the information from design specifications, sensor data and applied heat transfer equation to estimate energy consumption of each zone in the

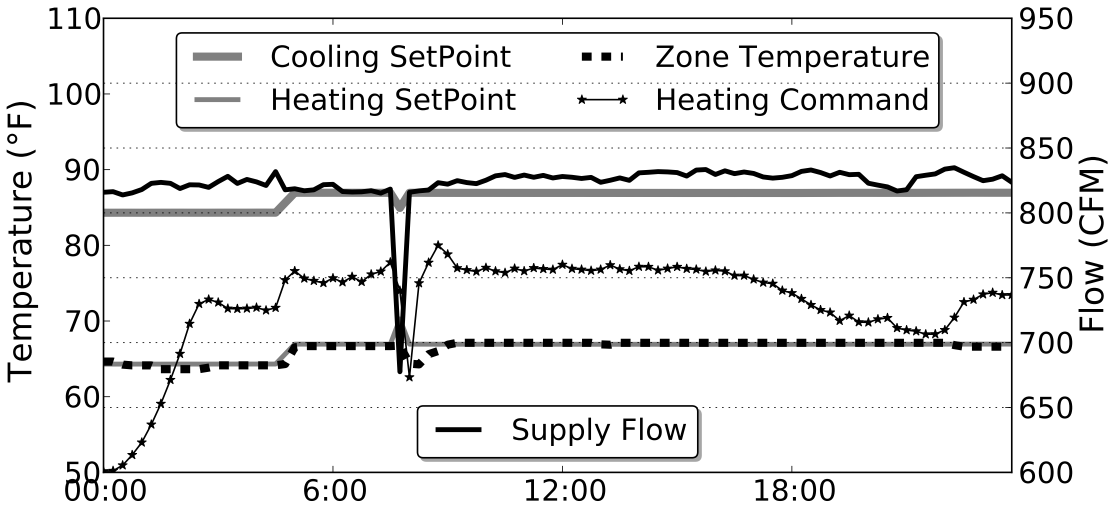

Figure 6.5. Supply flow excess in a conference room

building. We also apply heat transfer equation is to estimate wastage due to the broken economizer damper. To estimate wastage accurately, we need to compare the measured sensor data with the ideal values in the absence of faults. To estimate economizer energy wastage, we conservatively assume wastage occurs when the economizers are commanded to be at their minimum position. We compute the ratio of airflow mixture between outside air and mixed air based on design specifications, and use air temperature measurements to estimate mixed air temperature. For the VAV level faults, we accumulate the energy consumed during the periods when the zone requires no heating or cooling in unoccupied mode. The results of the analysis is summarized in Table 6.3. The total estimated energy wastage due to the faults discovered is 410.3 MWh/year, including the economizer damage which accounts for about half the wastage.

6.7 Summary

We collected data across five large institutions in the USA and engaged in discussions with ten different building operators on their experiences, frustrations and their informed requirements for novel BMS. After analyzing hours of interviews, we identified seven key recurrent problems that need to be addressed. With the help of our participants' suggestions we then distilled a list of overall directions to take for the design of the next generation BMS. We believe that these results have direct applicability and can be used to guide the development of novel interfaces, like the one that we are currently designing as part of our larger research on building management. We hope that the depth and breadth of our findings will support the much needed change in perspective and the bootstrapping of a new class of flexible, user-centered BMS.

We propose a list of requirements for fault management systems. Our suggestions include integration of information sources, long term data storage, standardized naming conventions, support for wide variety of fault detection algorithms, tools for analysis of faults, and reporting of contextual information with fault detection. We designed and implemented BuildingSherlock (BDSherlock), a web service based fault management framework which exposes RESTful APIs for reporting of faults by third party algorithms. We have deployed BDSherlock in the Computer Science building at UC San Diego, and successfully detected 88 faults in the HVAC system with estimated savings of 410 MWh/yr.

Chapter 6, in part is currently being prepared for submission for publication in the Journal of Energy and Buildings by authors Bharathan Balaji, Nadir Weibel, Rizhen Zhang and Yuvraj Agarwal with the title "Understanding Building Operators to Improve Building Management Software". The dissertation author was the primary investigator and author of this material.

Chapter 7

Normalizing Building Metadata

In prior Chapters, we have presented software solutions can save energy, improve thermal comfort and ease interaction with the building systems. However, to implement these applications on a large scale, we have to address the heterogeneity of systems across different types of buildings, from hospitals to shopping malls. Due to lack of a common sensor ontology in buildings, each vendor and institution have their own naming schema, and these are not strictly adhered to as they are meant to be human readable. In this Chapter, we present an algorithm that maps disparately named sensors to a common naming scheme using machine learning algorithms. This is a necessary step towards portability of software applications to multiple buildings.

Improvements in the design and manufacture of devices have led to the widespread availability of cheap sensors, actuators and data collection infrastructure. This, in turn, has led to increasing interest in "Smart Environments", which use these technologies to better understand user context and adapt to meet their requirements by controlling the physical environment around them. In pursuit of this vision, researchers have sought to create smart buildings that are responsive to occupants' needs and comfort while conserving energy and water resources.

Within commercial buildings, tasks involving indoor climate control and maintaining proper ventilation are typically performed using centralized Building Management

Systems (BMS), such as Metasys from Johnson Controls [51]. A BMS interfaces with a large number of sensors and actuators deployed within buildings during construction and commissioning such as thermostats, Variable Air Volume Boxes (VAVs), Air Handler Units (AHUs), Variable Frequency Drives (VFDs) and chillers. Collectively the sensors, actuators and the BMS form an integral part of the HVAC system. HVAC systems are relatively complex, typically interfacing with thousands of sensors and actuators, even in a moderately sized building (150,000 sq-ft). BMSs collect data from these sensors and provide vertically integrated tools to not only control the day to day operation of buildings, but also store and visualize data, analyze trends, and even detect faults [2, 51].

Vendors such as Johnson Controls, Siemens and Automated Logic provide proprietary tools to manage the complexity and provide different functions within buildings. These are often tied to expensive maintenance contracts and have not kept up with the state of the art in functionality, user interfaces and design. For instance, despite having fault management as a key function, facilities managers struggle to keep HVAC systems running efficiently and many faults remain unaddressed [127, 167]. Our own building managers report being notified of over 10,000 faults a day - most of which are ignored - thereby causing occupant discomfort, equipment deterioration and energy wastage [167]. This sensor management problem is compounded by lack of interoperability of methods to identify and manage sensors, and a general lack of tools to analyze large amount of sensor data generated [127, 112]. As an example, NIST estimates an annual loss of $15.8 billion in the US due to lack of building interoperability standards [76].

Recognizing this need for systems that enable ‘smarter’ buildings, several recent efforts have attempted to address the problems of interoperability, information integration, data storage and access control. These efforts primarily propose middleware services for buildings that gather information from disparate sources of information, including a multitude of sensor protocols, and make it available to application developers through standardized APIs [2, 22, 26, 58]. Based on our work with building infrastructure, we see that a key missing piece in all these efforts, however, is related to the assumption that the underlying sensor information is named consistently and accurately. Given the long lifetime of buildings relative to individual sensors or their networks, it is common to see sensor data information fall into disuse over time. This is exacerbated by the current practice of manually mapping sensor information for each building to a particular data model by the developers and building managers [167]. This manual mapping is expensive (requiring domain experts), time consuming, and does not generalize since it needs to be repeated for every vendor, equipment and building. The lack of standardized and automated naming has become an impediment to the creation and adoption of smart-building applications by developers that are portable across buildings and deployments.

The problem of automatically naming sensor metadata correctly and mapping the sensors and actuators to a uniform ontology is not easy. The challenges include the scale (thousands of endpoints in a moderately sized building), diverse lifetimes of buildings and BMSes - that are easily over 50 years in academic campuses - leading to heterogeneity in equipment types, and varying usage requirements. Researchers have identified this problem [36, 42, 155], and proposed solutions that still require significant manual effort. At the same time approaches based on using regular expressions (regex) and training examples [36, 149], do not generalize due to varying inconsistencies in sensor naming and do not leverage complementary sensor information such as its metadata and time series sensor data.

To address these challenges, we present Zodiac, a framework to analyze large numbers of sensors and actuators - including the time-series based data and the sensor metadata – and map them to a standard naming scheme with minimal human supervision. To show the efficacy of Zodiac, we applied it to four buildings on the UCSD campus comprising of over 20,000 end-points in total. To evaluate the accuracy of Zodiac we manually labeled the ground truth in terms of the sensor metadata for these sensors. We show that Zodiac classified sensor types in these buildings with an average accuracy of 98% accuracy using 28% fewer training examples, when compared to a regex based look up method, and only 15% more manual inputs than a hypothetical oracle algorithm.

The key contributions of our work are as follows.

- We show that the manual effort required to label sensors is significantly reduced through hierarchical clustering methods, without requiring customized regex that need building specific domain knowledge.
- We propose an active learning based approach that is effective in automatically identifying new sensor actuator types. Manual input is automatically requested to label these examples, and this labeling is expanded to improve coverage.
- We show that Zodiac is able to classify sensor types with high accuracy with only a small number of additional training examples than an oracle system with perfect knowledge. As compared to using regex, Zodiac uses significantly less examples and provides high sensor type classification accuracy, without requiring the significant manual effort of writing complex regexes.

We plan to release the sensor metadata to encourage researchers to develop systems that can automatically learn sensor relationships, pending permission from the university since some of the sensor data and metadata could be potentially privacy invasive.

7.1 Background

Modern HVAC systems consist of thousands of sensors and actuators that report information to a building control system for monitoring and maintenance. For example, a

Table 7.1. Sample points from HVAC system across three buildings on the UCSD campus. Metadata of points which have the same point type have inconsistencies, and points which are different point type can have similar metadata.

Vendor Given Name	BACnet Name	Description	Data Type	Unit	Point Type
BLDG1.N1STFLR.VAV-1NW.VAV-47.FLOW-SP	NAE-66/N2-1.VAV-47.FLOW-SP	Flow Setpoint	Analog Output	Cubic Feet per Minute	Supply Air Flow Setpoint
BLDG2.RM-2819.SUP-FLOW	NAE-14/N2 Trunk 2.VAV-35.SUP-FLOW	Supply Air Flow	Analog Input	Cubic Feet per Minute	Supply Air Flow
BLDG3 1stFl RM-111.SUPFLOW	NAE-10/N2-1.VMA101.SUPFLOW	Process Variable	Analog Input	Cubic Feet per Minute	Supply Air Flow
BLDG2.RM-1704.RM1705-T	NAE-14/N2 Trunk 2.VAV-36.RM1705-T	Room 1705 Temperature	Analog Input	Fahrenheit	Zone Temperature
BLDG3 1stFl RM-135.ZN-T	NAE-10/N2-2.VMA129.ZN-T	Zone Temperature conference rm	Analog Input	Fahrenheit	Zone Temperature
BLDG2.WBASEMENT.RM-B241.PHX-1.ZNT-SP	NAE-65/N2-2.PHX-1.ZNT-SP	Zone Temperature Setpoint	Analog Output	Fahrenheit	Common Setpoint

room thermostat informs the control system on how much cooling or heating is required, and helps an operator determine when the room is too hot or cold. In addition, there are configuration parameters that determine the operating point of the equipment such as cooling and heating temperature setpoints for each room. In our buildings, these serve as the higher and lower temperature bounds that the HVAC system tries to adhere to. Configuration parameters also include actuation commands such as switching ON a fan, or scheduling of equipment operation. We refer to the sensors and the associated configuration parameters in the HVAC system as *points*.

Points report data to their respective equipment *controllers*, which are embedded devices that operate the equipment control system, and react to changes in configuration parameters. Each of these controllers communicate with middle box servers, called Network Application Engines (NAEs) in Johnson Controls systems, that collect data from the controllers, and act as the interface between the HVAC system and BMS software. A subset of the NAEs in our university are connected to a dedicated network (VLAN), and they expose the points available via the BACnet protocol [41]. We have

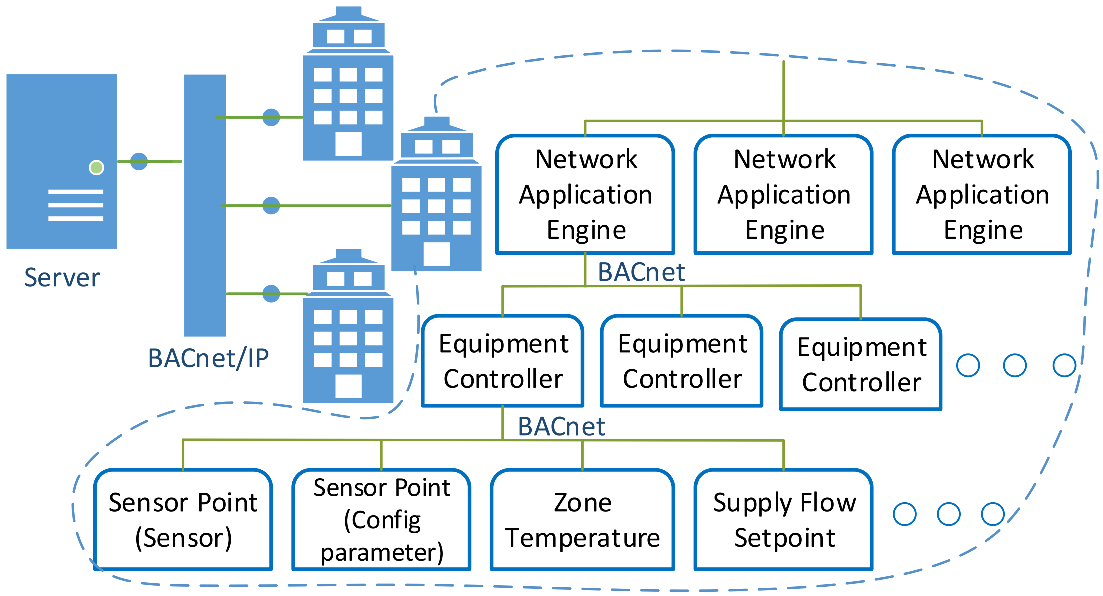

Figure 7.1. Figure shows the architecture of HVAC System points and the BACnet network layout leveraged for data collection

deployed our own BuildingDepot server [22] on this network to collect sensor data from 180,000 points across 55 buildings on the UCSD campus as of July 2015. (Figure 7.1). For this paper we focus on a subset of these buildings (four) which are of different sizes and usage modalities. In particular, for these four buildings, comprising of over 20,000 points, we had to manually label the ground truth for the sensor metadata, including their *type*. We use this labeled ground truth for both learning and testing of our automated labeling framework. The ground truth point types were based on a standardized naming template specified by UCSD contracts and our prior experience working with HVAC systems and consultation with the campus building managers. led ground truth for both learning and testing of our automated labeling framework. The ground truth point types were based on a standardized naming template specified by UCSD contracts and our prior experience working with HVAC systems and consultation with the campus building managers.

Each point in BACnet has associated metadata that describes the point and its properties. Some of these properties are specified by the BACnet standard, and others are defined by the vendor. Table 7.1 shows examples of six points along with a subset of their metadata. It is common for large campuses and vendors to follow a naming convention specific to the enterprise or campus [8, 17, 135]. For example, according to our university's naming standard, "vendor given name" uses a structured format to describe a point which when split by '.' gives the building name, the floor and room at which the sensor is located, the type of equipment it belongs to and the ID of the equipment, and finally, an abbreviation for the type of point. The "description" of the point gives the point type, and the "data type" gives both the type of data as well as whether it is an input or output point. As can be observed in Table 7.1, this naming convention is not strictly followed or enforced. The ordering of words or the punctuation may change, abbreviations and their description may change for the same point type, and as these names are entered manually per equipment, there are typographical errors and inconsistencies.

To standardize naming across buildings, we need to map the existing points into a standardized ontology [1]. We focus on accurately mapping the building points to standardized point types in this paper. Table 7.2 shows the number of point types for the four buildings we use for this paper. Building 1, has 3213 points that come from 154 distinct point types based on the ground truth labeling we do. Therefore, a perfect oracle algorithm that could label similar point types from a single example would still require at least 154 examples (provided by a domain expert) to label all the points in this building. Our goal is to design algorithms that can accurately label all points and require as few manual labels as possible, preferably close to unique point types in the building. Furthermore, the algorithm should be able to learn the patterns in one building, and use it for labeling points in other buildings - that is it should be able to *transfer* knowledge and labels.

As there is a naming convention based on which points are labeled on our campus and in other enterprises, regex are a natural fit for identification of sensor type [36]. As per the naming convention, the “description” of the point gives its point type, and the last part of the “vendor given name” is the abbreviation for the point type. Thus, we could maintain a mapping of description and point type abbreviations to their respective ground truth point types, and label points if their description is present in this map. As new descriptions are discovered, the domain expert is prompted to enter the point type. For Building 2, there are 922 unique descriptions mapping to 367 point types and 11910 total points. There are multiple descriptions that map to a single point type due to variations in the description, as the naming convention is not strictly enforced. The variations in descriptions can occur due to various reasons – spelling errors, additional information such as room number, or an alternate description that has the same meaning. For example, the point type “zone temperature” is also written as “zone temp”(shortening of word), “zone tempeartuer” (spell error), “room temperature” (alternate version), “zone1 temperature” or “zone temperature room 2102”(additional information). To reduce this variation, we remove special characters and numbers, and use uniform case. The number of unique descriptions for Building 2 reduce to 527. Some of the points in the dataset do not have any description, and we use point type abbreviation to label these points. These abbreviations can sometimes reveal the point type more accurately, as they do not necessarily vary due to changes in description. “ZN-T” is an example abbreviation of the point type “zone temperature”. However, these point types themselves vary, with use of punctuation, numbers or alternate versions of the abbreviation for the same description. The total number of unique labels with the combination of descriptions and sensor abbreviations for blank descriptions for Building 2 is 589. Thus, by some preprocessing the number of unique descriptions for 11910 points in Building 2 have been reduced from 922 to 589. It would take 589 manual inputs from experts to label Building 2, using regex to expand labeled examples. Table 7.1 summarizes the variation observed in descriptions and abbreviations observed across a few example point types. We design our regex to be highly accurate, and it is possible to reduce the number of manual inputs for a decrease in manual inputs. For example, regex for "zone temp" could include both descriptions "zone temperature" and "zone temp" to reduce one manual input, but may also include a false description "zone temperature setpoint". Thus, we rely on exact matches for these descriptions. Figure 7.2 shows the number of points that could be labeled by regex versus number of points manually labeled for Building 1.

Among possible errors in managing sensor data are sensor naming errors. Other errors occur when descriptions of certain point types are used interchangeably across different equipment. For Building 2, 58 points were mislabeled out of 11910 total points. Note that some points are hard to label even manually because of lack of metadata, and we mark these points as "unknown". In Building 2, 23 points were labeled as unknown.

We observe that using regex requires fairly involved domain expertise, in terms of the naming convention followed, yet can require large amounts of data. Further, regex fail to exploit additional metadata information such as unit or data type, and the actual time series of measurements, all of which can give additional clues to identify the type of point. We next describe our approach to automatically mapping each point to its sensor type using minimal manual labeling and no domain knowledge.

7.2 Identifying Point Type

To reduce the number of manual labels required, we *group* or *cluster* the features we have for each point. We use *hierarchical* clustering to improve the grouping of points with similar metadata. Starting with a small number of labeled points, preferably belonging to different clusters, we train a model that automatically labels other clusters, thus achieving point label expansion. When the model determines that a (new) cluster is

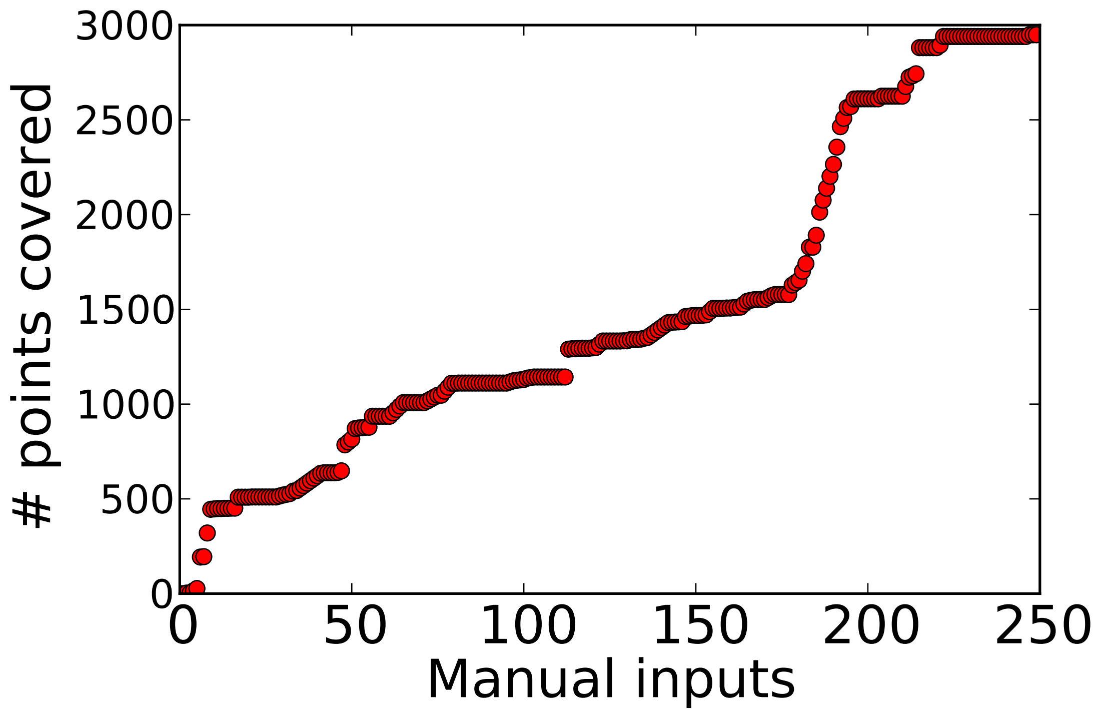

Figure 7.2. Number of points covered by using regex with respect to the number of sensors that were manually labeled for Building 1. Note that this graph does not account for domain expertise required to build the regex.

unrelated to any of the ones already labeled, manual labeling is requested from a domain expert for a member of this new cluster. We show that this process drastically reduces the manual effort required to assign types to points, with very few errors. Our machine learning algorithms rely on implementations from Scikit Learn [156].

7.2.1 Hierarchically Clustering Points

As discussed in Section 7.1, regex can group points which are of similar type thus reducing the manual effort in assigning point types. However writing regex requires domain knowledge to map point metadata to its type, and are dependent on the naming convention, building, and equipment providers. Clustering using point metadata offers two primary advantages over using regex for grouping. First, clustering uses the intrinsic

Table 7.2. Table lists the characteristics of our four testbed buildings. The number of examples required to learn point types by an oracle (c) and regex algorithm (d) are compared with those of Hierarchical clustering (e), and Active Learning (g,i,j).

Building Name	Total # Points	# Point Types	# Unique Descriptions	# Clusters	Accuracy %	Learning with Hierarchical # Manual	Learning with Hierarchical Accuracy	# Merged Clusters	Learning with Merged # Manual	Learning with Merged Accuracy
(a)	(b)	(c)	(d)	(e)	(f)	(g)	(h)	(i)	(j)	(k)
Bldg 1	3213	154	251	300	98.7%	245	99.3%	191	181	98.3%
Bldg 2	11910	367	589	1105	99.3%	548	94.5%	499	453	96.0%
Bldg 3	1913	156	228	215	97.1%	204	99.8%	174	169	98.8%
Bldg 4	4380	192	316	329	98.8%	299	100%	206	198	99.1%

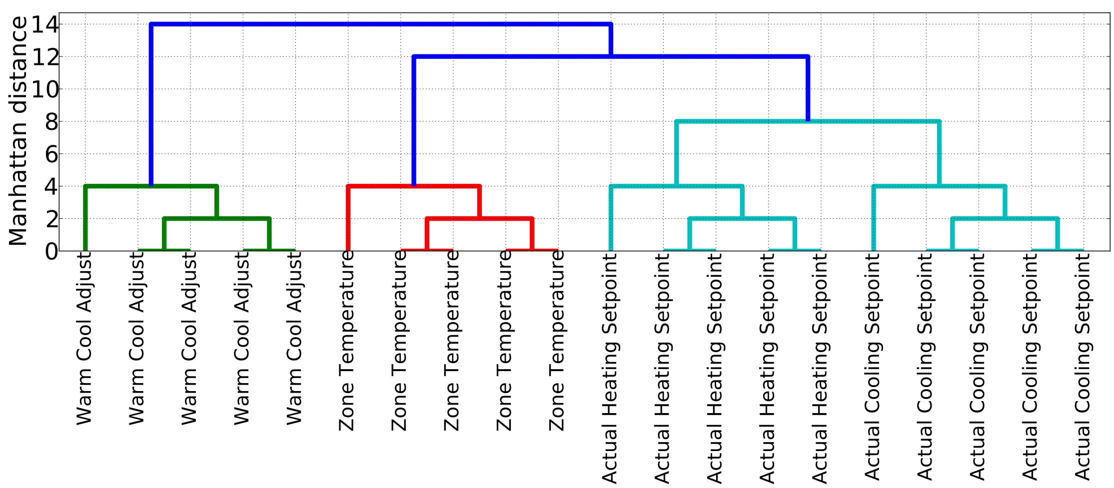

Figure 7.3. Example dendrogram of hierarchical clustering. Points whose metadata features are similar grouped together first and clusters which are closer to each other are consecutively grouped in the next stages.

similarities in sensor metadata rather than rely on a pre-specified pattern which may only be able to capture similarity in terms of few pre-defined descriptors. Thus, clustering can learn patterns using additional metadata such as units and data type, and can group together points which have minor variations in their metadata. As a result the grouping mechanism is more robust than an approach based on individual rules created using regex, and can generalize to a variety of naming conventions. The second advantage is that clustering based grouping of points is not dependent on domain expertise to extract useful information from the metadata. We use hierarchical clustering [95] to group points.

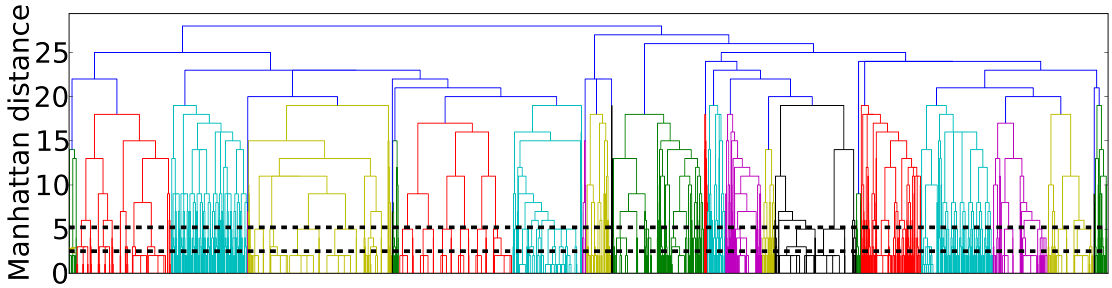

Figure 7.4. Dendrogram for 1000 points in a building. The two dashed lines indicate thresholds to obtain clusters of points. As the threshold increases the number of clusters decrease and the accuracy of clustering points of same type decreases.

The features we use for hierarchical clustering are created based on the "vendor given name", description, unit, and the type. The strings are tokenized into individual words and pre-processed to remove special characters and numbers and to convert to uniform case. A *bag of words* [183] representation is used for the feature set. Hierarchical agglomerative clustering computes the distance between a pair of points using their feature vectors, and merges those points which have the least distance between them. These clusters are then recursively merged again based on the linkage metric used. We use *complete* linkage, which combines clusters after examining each point within the cluster, and use *manhattan distance* as the distance metric. The results obtained are similar for other distance metrics such as euclidean distance and jaccard index. Manhattan distance is used for our results since it is easier to interpret in terms of the difference between point features.

Figure 7.3 shows an example dendrogram obtained by hierarchical clustering of 20 points for illustration. Since the metadata used for describing points of the same type are similar, the distance between their feature vectors is small, and they naturally get clustered together in the first few stages. As the clusters get bigger, points of different types also get merged eventually forming one big cluster. An appropriate threshold distance on the Manhattan distance needs to be identified that would prevent the merging of clusters with different point types.

Figure 7.4 shows the dendrogram obtained with hierarchical clustering of 1000 points from one of the buildings in our testbed. The horizontal lines represent choices for threshold for obtaining point clusters. As we increase the threshold, the number of points in a cluster increase, and points of different types may be clustered together. As we decrease the threshold the total number of clusters increases. As the number of clusters reflect the number of manual checks that may be required for labeling points, we would like to obtain as few clusters as possible. These trade offs can be quantified using network motif methods [81], but for this work we pick a threshold from the dendrogram based on an estimate of the number of manually labeled points. We conservatively pick a low threshold to minimize the number of errors in clustering, where errors correspond to points of different types clustering together. As the feature set we use is the same across buildings, this threshold remains the same for hierarchical clustering of points in other buildings as well.

We *conservatively* define a cluster to be erroneous if it contains points of more than one ground truth point type. So, a cluster and all the points within it are marked incorrect even if only one out of hundreds of points is included incorrectly. We applied hierarchical clustering on 11,900 points in Building 2, and obtained 1105 clusters. Only 18 of these clusters were erroneous, giving an accuracy of 99.3% with only 85 points mislabeled. An error in clustering occurs when the metadata used to describe points of different type are very similar. As an observed example, two point types “hot water pump status” and “chilled water pump status” were misclassified because their descriptions were blank, and only two letters in their entire metadata were different. It was observed that by examining the errors in clustering, it may be possible to identify errors in metadata as well. For example, point types “zone temperature” and “zone temperature setpoint” are input and output points respectively, but both of them were labeled as inputs, causing an error in clustering. As discussed earlier, the clustering was able to combine points which would have been difficult to identify as similar using regex rules. For example, points with descriptions “zone temperature”, “cold box temperature” and “freezer temperature” were clustered together as their other metadata are similar. Column (e) of Table 7.2 summarizes the clustering results of four buildings.

From Table 7.2 we see that Building 2 has 367 unique point types (column c), 922 unique descriptions, 589 unique descriptions (column d) after their case is normalized, numbers are removed and blank descriptions are mapped to abbreviations. Hierarchical clustering gives 1105 clusters (column e), which is more than domain based heuristics. However, the clusters obtained from hierarchical clustering capture the inherent variation in the naming structure, which is different from those obtained using domain knowledge. This is because hierarchical clustering looks for similarities across all the metadata of points while regex is based on domain knowledge that specific metadata such as point description is more indicative of the point type. As we show in Section 7.2.2, hierarchical clusters can be useful in learning data driven models. The low intra-cluster error generated by the hierarchical clustering can then be leveraged to efficiently label a cluster by manually assigning type of one point in each cluster and propagating the same to other points in the same cluster.

Furthermore, domain knowledge can be used to improve the clustering further. We combine two clusters (obtained using hierarchical clustering) when the description of each point across these clusters were identical. As a result of this, for Building 2, the number of unique clusters dropped to 499 (column i) compared to 589 unique descriptions. Thus, under the availability of the knowledge of which parts of the point metadata are most important, the number of lookups required can be reduced when compared to using regex. Figure 7.5 shows the comparison of the clusters obtained for Building 1. We next show how manual input can be further reduced by learning

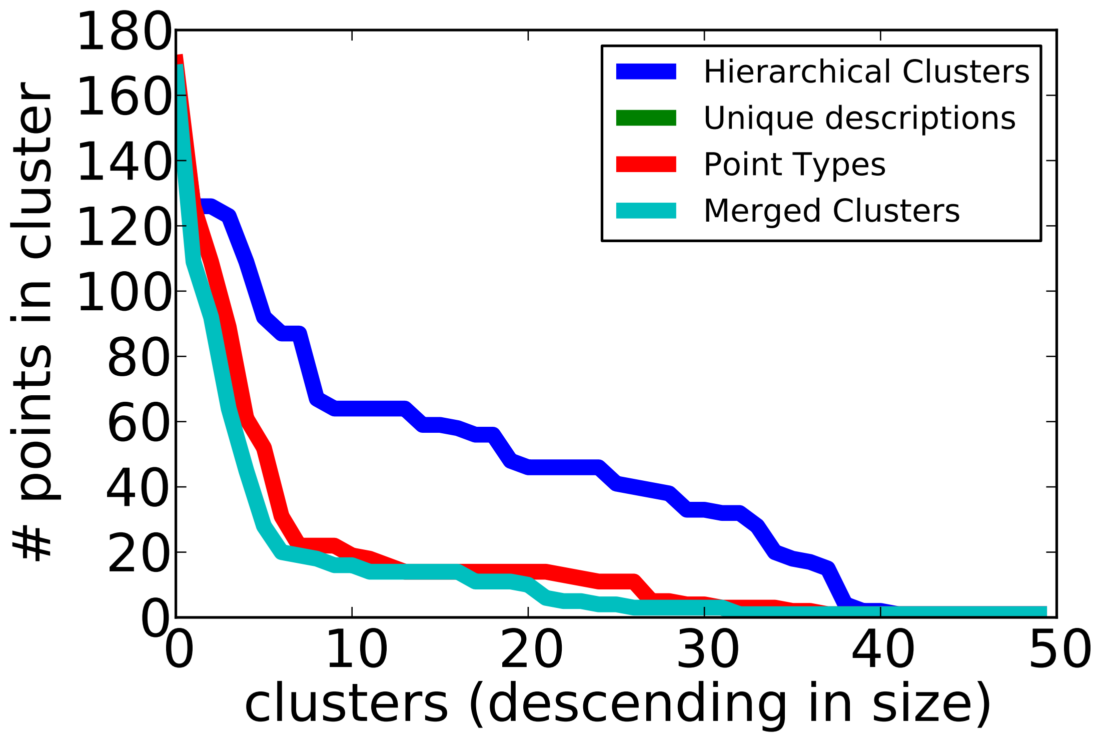

Figure 7.5. Histogram of points for Building 1 by unique descriptions, hierarchical clustering, merged clustering and ground truth point types. The number of clusters have been cutoff at 50 out of a maximum of 300

predictive models based on point metadata.

7.2.2 Learning Point Types

The regex based labeling of point types uses a look up table for different kinds of metadata, and maps it to its type. Hence, it relies on exact matches on point types, and each variation of metadata needs to be manually verified before it is added to the table. Although some of this variation in metadata is captured using hierarchical clustering, it does not comprehensively capture all the variations that occur. For example, suppose the point type “supply air flow” has two points with descriptions “supply air flow” and “supply flow feedback”. A look up table based match cannot automatically learn that “supply air flow feedback” is also a description of the same point type. Thus, there will be three separate lookups for “supply air flow”, “supply flow feedback” and “supply air flow feedback” using the regex method. If the metadata features used by hierarchical clustering is different for these three points, they would also be put into three separate clusters.

A data driven model can learn the relationship between metadata and ground truth point types, and it can give a prediction for metadata whose examples have not been observed before. Hence, a data driven model is capable of learning that “supply air flow”, “supply flow feedback” and “supply air flow feedback” belong to the same point type. Further, sensor timeseries data can be used to learn models (or rules, or regex) even when a pre-defined format is unknown for using regex. They can also incorporate additional features for learning the characteristics of a point. We present our timeseries data based features used for learning a model in Section 7.2.3

To validate our hypothesis that it is possible to learn an effective model that can use sensor metadata to predict the point type, we micro benchmark the performance of a Random Forest classifier [116] for Building 1. We use three fold cross-validation with the training set having at least one example of each point type. We observe that the Random Forest classifier can successfully identify point types with an average accuracy of 97.1%.

A key challenge for training a model that can predict point types based on their metadata is the availability of labeled training points. In order to train a mapping model with minimal manual input, we use *active learning*. To begin, we seed the learning with ten points which have been labeled by a domain expert. When we inspect the next point, we need to identify if this point is of the same types as the ones we have already have a label for, and if not, ask the domain expert if it is a new type.

There are two conflicting forces in learning such a model. There are point types which have metadata, i.e., features, which are very close to an existing point type, but it is of a different type. For example, “occupied temperature setpoint” and “unoccupied temperature setpoint”. For a poorly trained model, these will be misclassified. And then there are points whose features are very different, but actually are of the same type. For example, “airflow rate” and “supply flow feedback”. The model may mark one of these as a new type, and hence, it increases the number of manual inputs required.

To check whether an unseen point is of a new type, we can assign a probability of the point belonging to one of the existing classes (i.e., point types). We can build a generative model for each of the point types seen so far, and use this model to assign this probability. We experimented with Gaussian Mixture model [148], Multinomial Naive Bayes model [103] and one class SVM [121], and they did not work well with our dataset. A discriminative model on the other hand would give the probability of a point mapping to one of the existing classes. If the model assigns low probability to all the existing classes, then there is a good chance that this is of new point type. In line with this intuition, the Random Forest classifier [116] consistently gives low probability to a point of new type in our dataset. Hence, we use the Random Forest classifier to determine if we need to ask the domain expert for the correct label.

To label the points of a building, we first cluster the points using hierarchical clustering, and assume the points within a cluster are of the same type (Section 7.2.1). We ask for labels for 10 randomly chosen points from distinct clusters from the domain expert, propagate these labels to all the points in their respective clusters and build a Random Forest classifier based on these points. Next, we obtain the prediction probability for a new point. If its probability is high (¿0.9), we assume the prediction to be correct, and add the points in the corresponding cluster to the training set. If the probability is low (¡0.2), we ask the domain expert for the point type. We iteratively retrain the model using the labels learned, and add more points to the cluster. When there are no more points which satisfy the upper/lower probability thresholds, we decrease/increase the thresholds respectively, to learn more points.

Figure 7.6 shows the results for the random forest based active learning for Building 1. The number of manual inputs required for all points of Building 1 is 245, and the accuracy of labeling with the obtained random forest classifier is 99.3%. Thus, our random forest based active learner is able to learn the mapping of points in Building 1 with 6 fewer examples than regex methods (251 examples, Figure 7.2) without any prior knowledge about the structure of the naming convention. When we used the merged clusters obtained by combining unique descriptions and hierarchical clusters (Section 7.2.1), the number of manual examples required dropped to 181, with an accuracy of 98.3%. Columns (g, h, j, k) of Table 7.2 summarizes the results for active learning methods on four buildings. A limitation of our algorithm is that learning rate with manual inputs is linear as seen in Figure 7.6. However, a better algorithm could be devised that takes advantage of frequently occurring point types (Figure 7.10) to increase the learning rate.

Learning Across Multiple Buildings

With regex and look up tables, it is easy to use mapping from one building to learn the mapping of another. Some point types such as “zone temperature” and “supply air flow” are common across buildings, and once their mapping is learned, points of the same type in other buildings can be labeled. To test how much information can be learned across buildings using regex, we created a look up table using ground truth point types of Building 2 and used it to label points of Building 1. Figure 7.7 shows the learning curve obtained for Building 1. All the points in the building were learned using 176 manual inputs, a reduction of 75 labels compared to regex based learning without any prior knowledge. However, as the description of the two buildings do not follow the exact same terminology, some errors are introduced, and the accuracy drops to 99.6%.

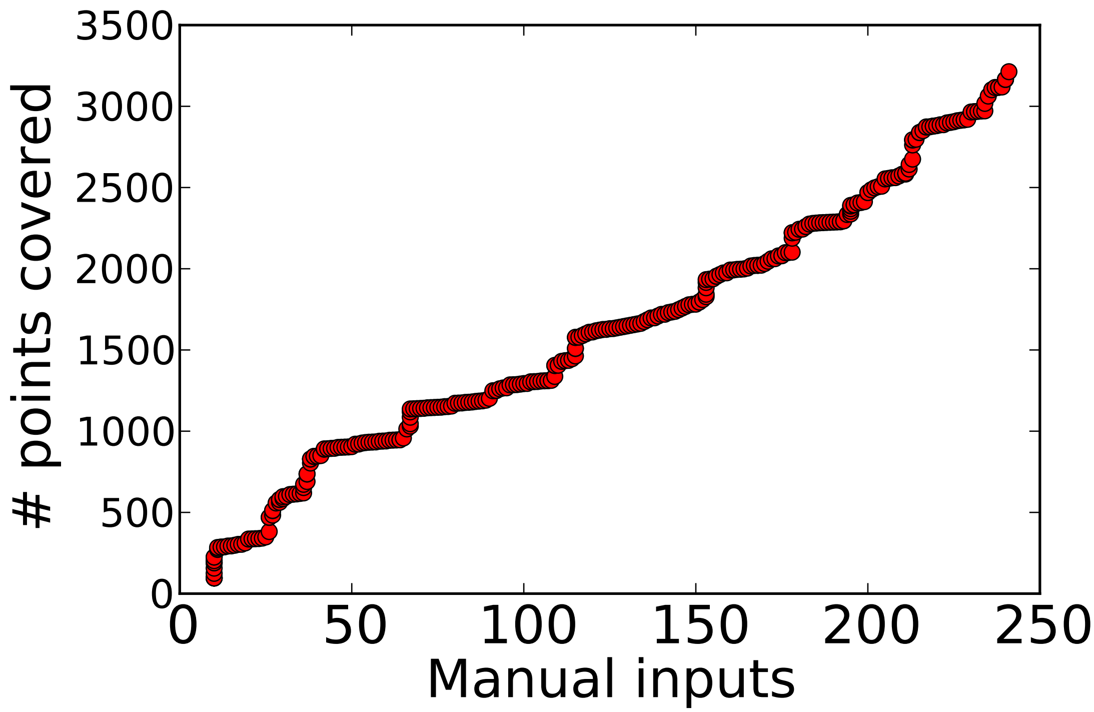

Figure 7.6. Learning curve of random forest based active learning algorithm. It required manual labeling of 245 points for labeling 3213 points in Building 1 with accuracy of 99.3%.

The active learning method used by Zodiac also learns the mapping between point metadata and its type, it should be able to label points of the same type even across different buildings as their metadata will be similar. To evaluate the *transfer learning* capability of Zodiac, we first built a random forest classifier using ground truth point types of Building 2, and used the iterative learning method to label points of Building 1. We again used 10 randomly chosen points as seed examples, and the feature vector for the classifier consists of bag of words of point metadata from both buildings. Figure 7.8 shows the learning curve obtained for mapping of Building 1 points. 173 manual inputs were used for learning, an improvement of 51 manual labels compared to learning without any prior experience. Thus, the active learning method is able to successfully learn from prior experience without domain knowledge, and is about as successful as the

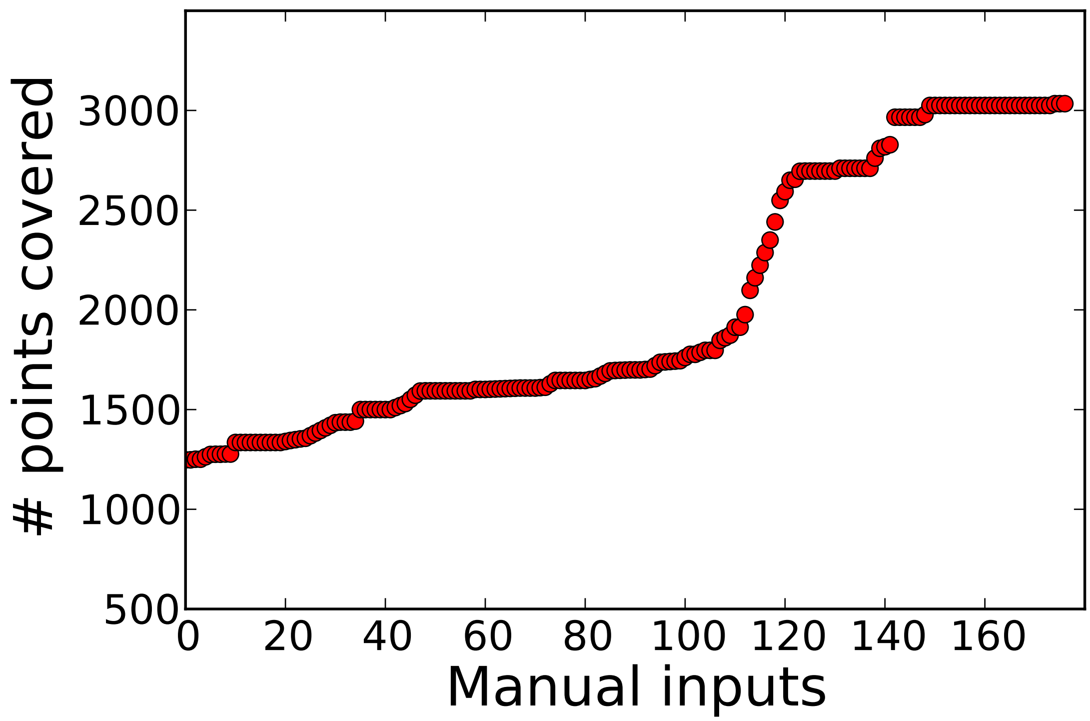

Figure 7.7. Learning curve of regex based naming across multiple buildings. It requires manual labels for 176 points in Building 1 (total 3213 points) with accuracy of 99.6%.

regex method. The accuracy of classification was 99.5%, and hence, the learning model is able to label points using information from another building. This is an initial result based on one example, and we are in the process of evaluating transfer learning across other buildings.

7.2.3 Using Time Series Data

To improve the accuracy of point labeling and to further reduce the manual input, we next try and leverage the time series data from points. This highlights an advantage of our learning based model, as it can incorporate any additional information that is available.

In general, time series can be divided into episodes, where each episode is a sample. For example, in many sensor applications, such as building HVAC, there is a

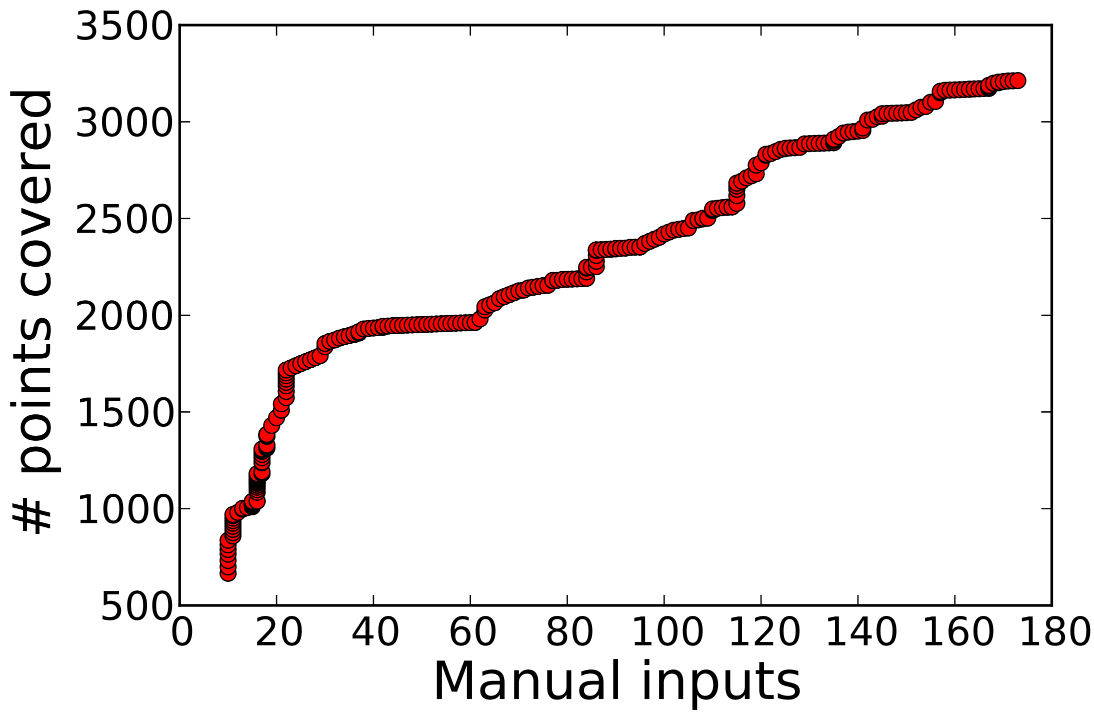

Figure 7.8. Learning curve of random forest based active learning across multiple buildings. It required manual labels for 173 points in Building 1 (total 3213 points) with accuracy of 99.5%.

natural diurnal variation leading to day length episodes. In order to leverage the time series data, the problem we are faced with is one of time series classification [77]. This is a problem that has been well studied in the data mining literature, see for example [101, 117] for representative techniques and applications. However, these methods often exclusively focus on coarse patterns or motifs that can be used to distinguish time series generated from very different processes. In our case, due to the fact that most points are associated with a common HVAC process - with common diurnal variation and dependence on external temperature - many time series will have similar patterns. Time series classification based on fine grained time series features arises in applications like speaker recognition [33] and signature based appliance classification [88]. In our experiments we combine features that capture many levels of the time series structure.

We use four classes of features, namely, scale based, pattern based, texture based and shape based features. *Scale based* features capture the range of values that the point readings can take. We use mean, max, min, upper and lower quartiles and range. For example, the mean and range of sensor measurements can tell us if a sensor measures supply air temperature or supply water temperature. *Pattern based* features capture the structure of repetitive sub-patterns in the time series. We use three Haar wavelet and three Fourier coefficients from the power spectral density of the signal as features. *Shape based* features capture the coarse structure of the time series, but are insensitive to fine structure. We use a piece-wise constant model of the time series, as shown in Figure 7.9, and use the location and magnitude of top two components as features. The error variance between the piece-wise constant model and the true signal is used as a texture feature. *Texture based* features capture the roughness, smoothness and other fine scale features of the time series. Texture based feature have a history in image processing applications, but have found limited application in time series classification. However, we find that the texture based features we use are particularly useful in distinguishing between points like supply flow - which is rough - and their corresponding set-points which are smooth. In addition to the error variance mentioned above, we use the variance of the difference and second difference between consecutive samples, max variation, number of up and down changes along with an edge entropy measure. The edge entropy measure is intended to capture the regularity of the time series across multiple episodes (a day in our case). For each episode, we capture the times at which large changes in value occur, and accumulate these as counts across episodes. We normalize these counts to sum to one, and compute the entropy of the resulting probability distribution. If this entropy is high, we can infer that the point either has limited structure within each episode or between episodes - a useful feature for point data classification.

As described, we select six features of each type, for a net total of 24 data

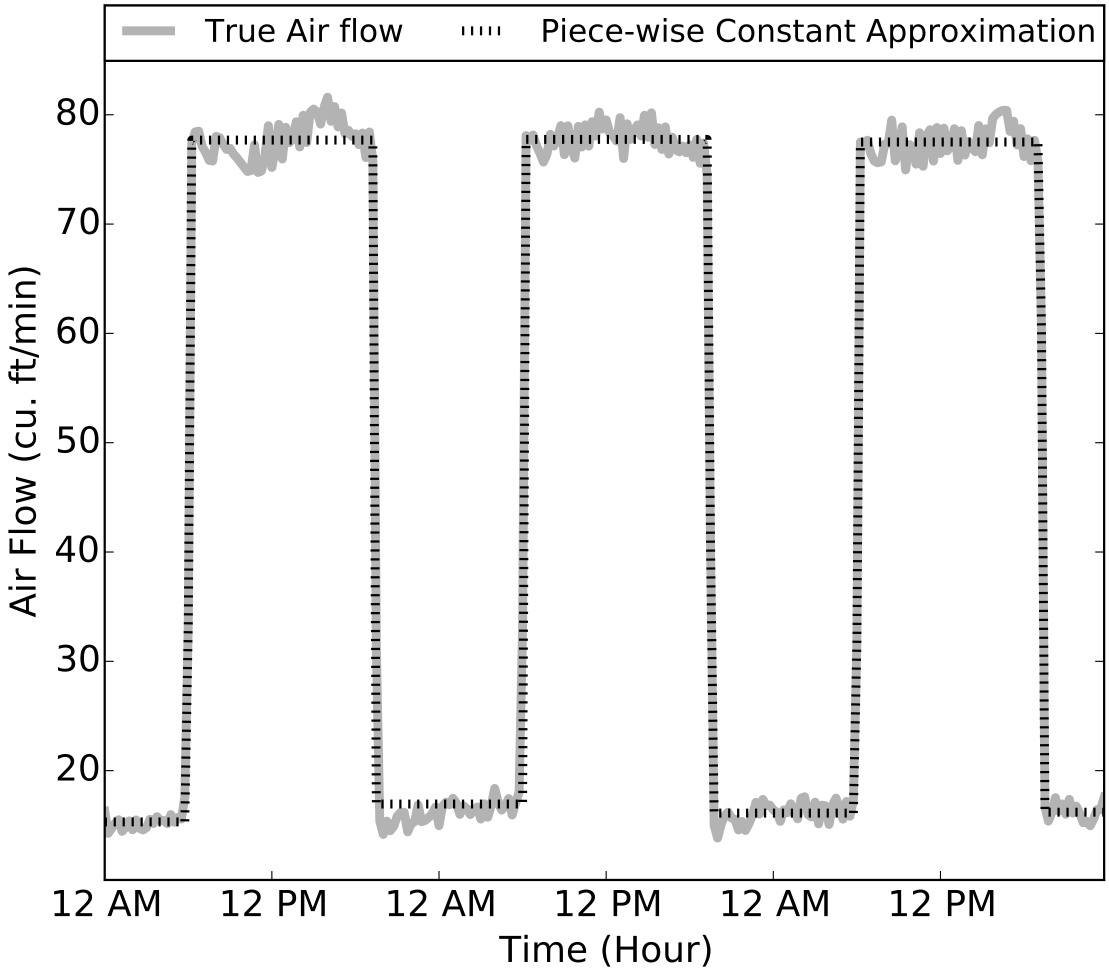

Figure 7.9. Piece-wise constant approximation of time series

dependent time series features. While the selection and design of more application specific features may be useful, we find that these features perform well in practice. To test this, we evaluate the *time series data only*, *metadata only* and *time series data+metadata* features on a separate building with 5857 points divided into 198 unique types. The distribution of point frequencies is shown in Figure 7.10.

The accuracy of the three methods, as a function of the number of labeled points of each type available is shown in Figure 7.11. We observe that the time series or data based features are not very effective on their own. This is because of two reasons. There are some point combinations 'supply air flow setpoint' and 'cooling minimum flow'

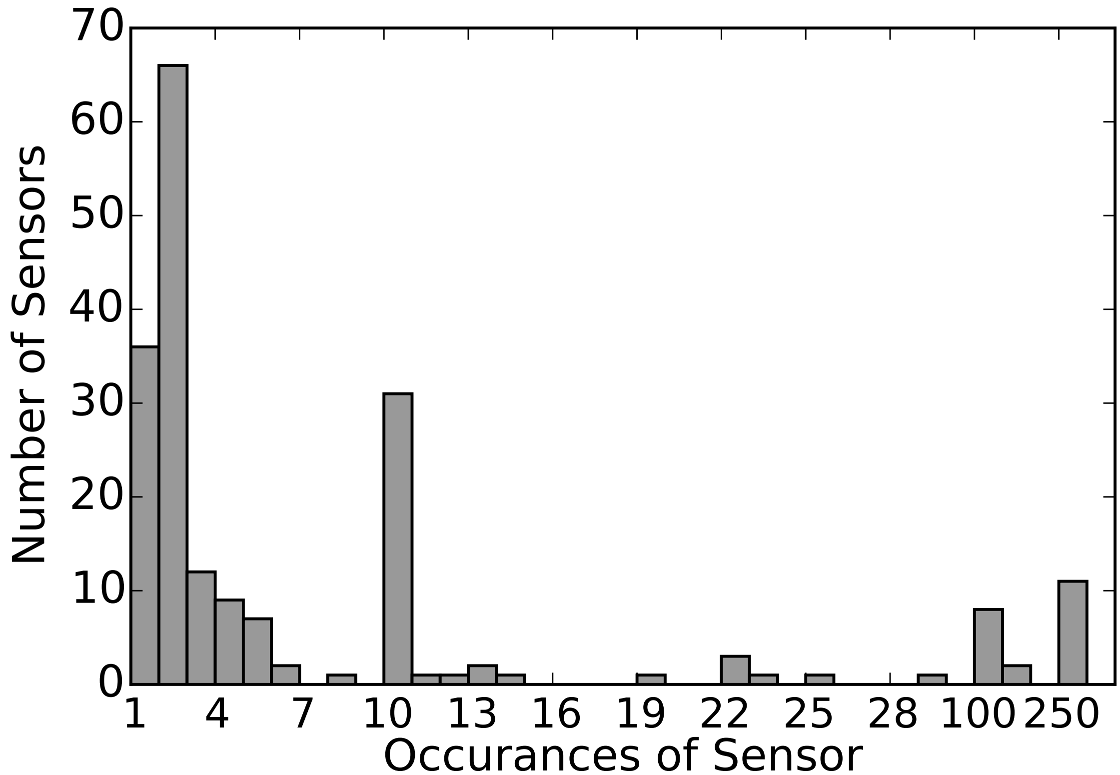

Figure 7.10. Relative occurrence counts of different point types in a building

which are essentially identical as time series in the way our buildings are configured. Secondly, there are many points - such as some set points and heating commands that never change (are always 0) hence are again impossible to differentiate using data alone. Finally, there are some points like ‘Heating Command’ and ‘Cooling Command’ that are very similar at coarse and fine time scales (taking values between 0 and 1, sharp changes at apparently arbitrary points) that are essentially impossible to distinguish using the features we use. We note that they can be distinguished using point inter-relationships (heating command will be high when zone temperature is too low), and does suggest a direction of future work. However, in Figure 7.11 we do see that using point data is able to provide significant boost to accuracy over using point metadata alone, particularly when only a few labeled examples are available - exactly the regime that is of interest to us. This demonstrates potential of using additional information such as point time series

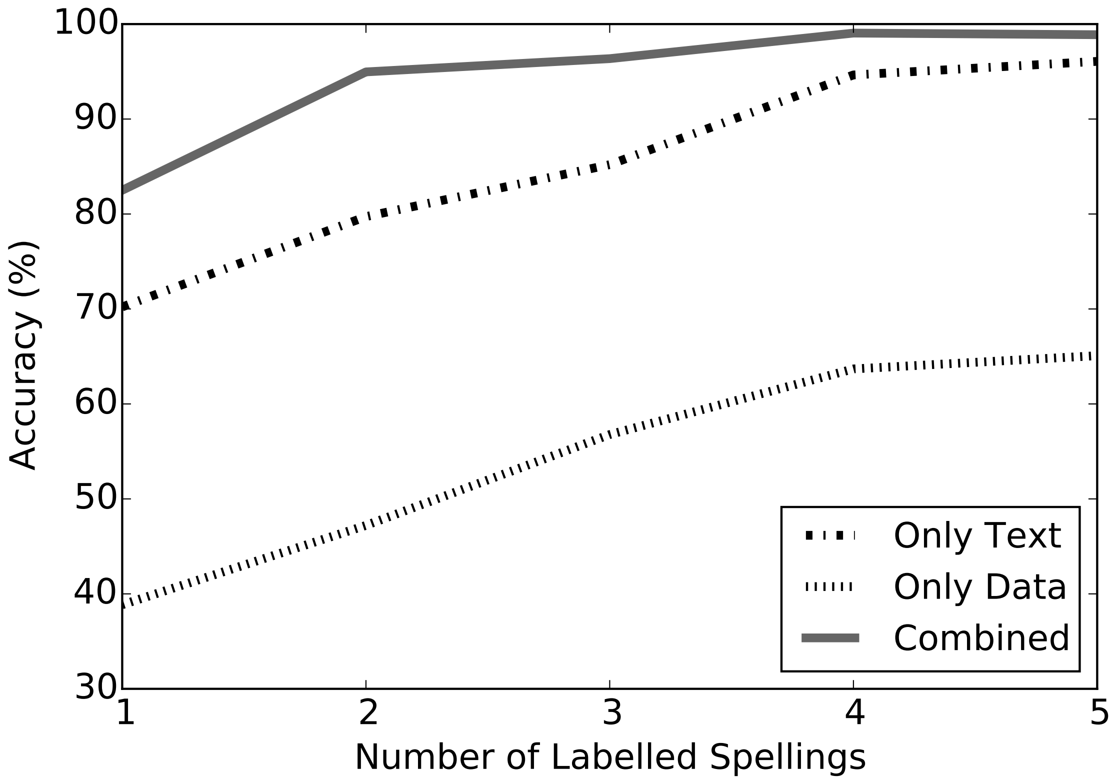

Figure 7.11. Comparison of text (i.e., point metadata) and (time series) data based learning methods. The x-axis represents the number of labels required for each point type. We observe significant improvement when both metadata and time series data features are used together.

data to improve the classification models.

We add the time series data features to the metadata features to test if they help our active learning algorithm. Figure 7.12 shows the learning curve of the active learning algorithm for Building 1 with 3213 points and 300 hierarchical clusters. The additional data features lead to a slight drop in accuracy (98% vs 99.3%) and an increase in manual examples required (261 vs 245). We observed similar results when we used merged clusters for Building 1. As the random forest based active learning algorithm is based on confidence of label classification, the result indicates that the added data features led to ambiguity and decreased the confidence threshold. Hence, additional manual inputs are solicited until lower thresholds can be included. It is possible that a better active learning

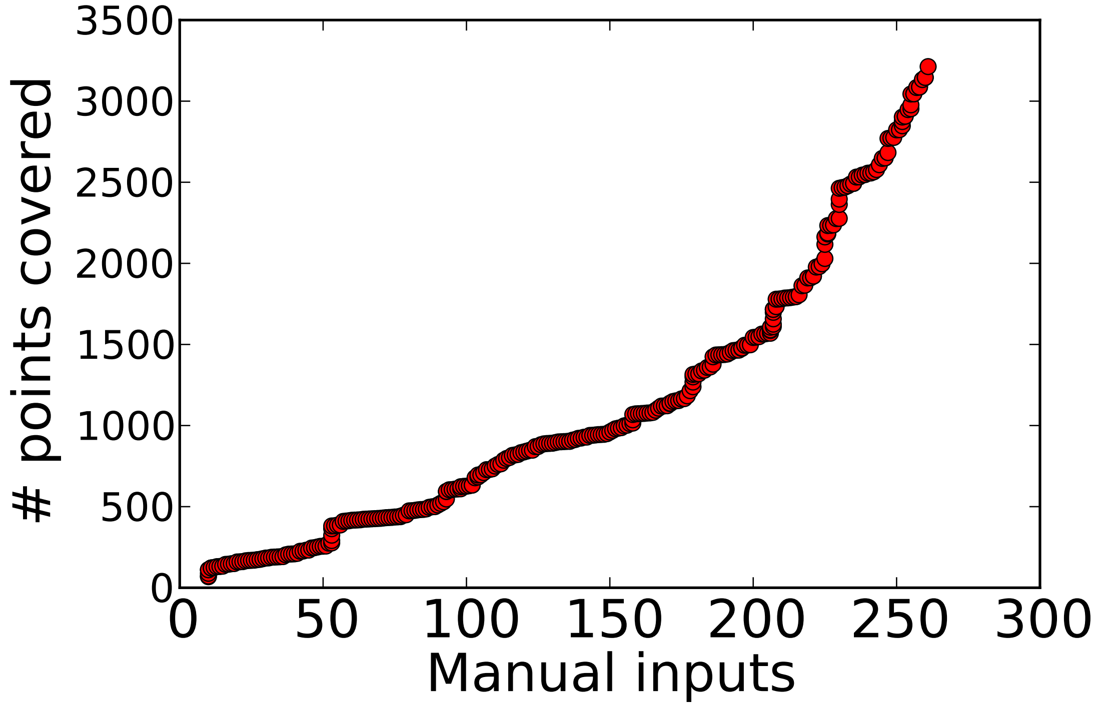

Figure 7.12. Learning curve of random forest based active learning algorithm with additional timeseries data features for Building 1 with use of hierarchical clusters. It required manual labeling of 261 points for labeling 3213 points with accuracy of 98%.

algorithm can incorporate data features with improved accuracy. Moreover in the absence of point metadata, the point time series data can be leveraged to reduce manual labeling.

7.3 Related Work

The organization of sensors using standardized ontologies has been recognized in literature as a key component for building useful, reusable applications [50, 152, 161]. OntoSensor [152] is a system that labels sensors using an ontology and describes them using a UML like language for representation and querying. Within the buildings domain, mapping of sensors, or points, to standardized ontologies is considered to be critical for information re-usability and development of apps that improve energy efficiency [42, 82, 111, 149]. Standards are being developed for naming of points in the

HVAC system [1, 111], and system architectures have been proposed that build upon a standardized information ontology to build applications that understand contextual information [82]. However, none of these works focus on mapping of existing points in a building to a standard ontology.

Schumann et al. [155] identify the mapping of existing points to a standard format to be a challenge, and propose artificial intelligence based methods such as hierarchical clustering for learning such a mapping automatically. However, they do not implement or evaluate their proposed method on real sensors. Reinisch et al. [149] propose a platform that facilitates the mapping of points to a standard ontology. They do not, however, learn the mapping, and still rely on manual inputs. Bhattacharya et al. [36] address the problem of organizing points to a standard template, and their work is closest to our work. They use a synthesis technique that constructs a metadata structure using transformation rules, and evaluate their technique on point metadata from several building in their university. We propose a different approach to the same problem, with use of learning based methods. The advantage of using a mixture of hierarchical clustering and active learning based methods is that we can use known information about the points to learn their model. For example, we used information such as unit, data type, and characteristics of data variation as features in our model. In contrast, the approach taken by Bhattacharya et al. [36] requires a human to recognize and formulate the patterns to identify sensor point types.

In machine learning terminology, the learning paradigm where both labeled and unlabeled points are available is called semi-supervised learning [44]. Active learning is a form of semi-supervised learning where the learning algorithm presents unlabeled points to an expert who returns labels for them [48]. It is expected that efficient querying algorithms will require fewer labeled samples. We consider pool based active learning algorithms, which exploit situations where a small set of labeled data and a large pool of unlabeled data are available [114]. Our active learning algorithms are uncertainty based - that is we query points we are least confident about. However, we modify these algorithms to be partly density based [133], i.e., we select which points to query based on cluster sizes. Based on the intuition that sensors in buildings and related large scale applications are spatio-temporally organized, we incorporate ideas from hierarchical active learning [55]. Finally, we note that the use of random forest ensembles with uncertainty based sampling is related to the idea of Query-by-Committee [159]. While many variants of active and interactive learning have been proposed in the literature [158], we find that the algorithms we have chosen work well in practice for our application. Investigating other alternatives remains an interesting direction for future work.

7.4 Limitations and Future Work

We have shown that it is possible to learn the naming patterns in HVAC systems, and classify points according to their types with minimal supervision. We have tried our methodology across four buildings with promising results. Our dataset, however, is limited to the UC San Diego's university campus, and most of the equipment is installed by one vendor. The point naming standards used across many different institutions are similar to ours [8, 17, 135], but it remains to be seen how our algorithm will generalize to a different set of equipment, vendors and facilities management.

Standardized point names are are important step towards portable smart building applications, however there is additional domain specific context that is not captured by uniform naming. For example, points need to be categorized according to the equipment they belong to, and the type of equipment needs to be identified for applications like fault detection and energy analysis. With our text metadata from BACnet, we used equipment specific features to identify equipment ID and equipment type. For buildings with well labeled points, hierarchical clustering successfully grouped points by equipment and clustered the equipment by their type. However, many buildings had point metadata that lacked equipment information or had poor equipment naming, and hierarchical clustering failed for these buildings because the metadata features were not adequate to cluster points by their equipment type. In future work, we will pursue methods that would map the points to their respective equipment, identify the equipment and learn the relationship between the points.

An example of such a problem is the mapping between VAV boxes and AHUs in a building HVAC system. Many buildings in our campus do not have this mapping information in the metadata, and facilities managers resort to manually maintained documents, or scrutinizing building architectural diagrams. In preliminary experiments, we attempted to find this mapping using data driven methods that identify the correlation between AHU behavior and the corresponding VAVs served by that AHU. However, the variation in temperature or airflow data was inadequate to capture this correlation. One promising approach is to use actuation of HVAC system according to a controlled sequence to learn such relationship between equipment across the building empirically.

Understanding such domain specific context and standardized representation of this context is key to developing portable applications that provide useful insights based on sensor information and provide value added services. To encourage research for development of methods that automatically learn relationship between points and map them to a standardized representation, we release the dataset consisting of metadata of 180,000 points across 55 buildings in the URL: *http://www.synergylabs.org/datasets/zodiac.html*.

7.5 Summary

Heterogeneity in sensor naming and metadata are an impediment to development of reusable applications in large scale sensor deployments. We illustrate the scale and challenges in mapping sensors in HVAC systems in our university buildings to a standardized naming schema. Regular expression based methods can map sensors to their respective types but tend to be too sensitive to minor variations in the sensor metadata and require substantial domain expertise. Our proposed framework, Zodiac, uses hierarchical clustering for grouping the sensors based on the inherent patterns in the sensor metadata. We applied hierarchical clustering on four buildings in our campus, and the clusters were grouped together based on sensor type with an average accuracy of 98% as compared to the manually labelled ground truth. Zodiac uses the clusters to train a random forest classifier using active learning. Our active learning algorithm labeled sensors across four buildings to their respective types with an average accuracy of 98% requiring 27% fewer ground truth labels than regular expression based methods.

Chapter 7, in part, is a reprint of the material as it appears in Proceedings of ACM Conference on Embedded Systems For Energy-Efficient Built Environments (BuildSys ’15), 2015 by authors Bharathan Balaji, Chetan Verma, Balakrishnan Narayanaswamy and Yuvraj Agarwal with the title “Zodiac: Organizing Large Deployment of Sensors to Create Reusable Applications for Buildings”. The dissertation author was the primary investigator and author of this paper.

Chapter 8

Future Work

Building software infrastructure systems such as BuildingDepot [22] have integrated information from different subsystems in buildings and made it available for developers using published APIs. Several similar systems have been developed recently. sMAP [57] provides an API for accessing building sensor timeseries data. Building Application Stack connects different components in the building and uses a domain specific language for ease software development [109]. BOSS [58] provides support for transactions and locking mechanisms to support actuation by software applications. NiagaraAX by Tridium [2] integrates with contemporary BMSes and supports third party applications. With the advent of these systems, it is now possible to access building information that was spread across disparate systems.

Many innovative software applications have been proposed that builds on top of these building management infrastructure [37]. However, several obstacles remain for adoption of these software solutions on a wide scale. Although building sensor data is available, they do not follow a standard naming scheme. There are multiple building naming standards such as IFC, Green Building XML and Haystack, but these do not adequately address the requirements of recent software applications [37]. Hence, there is a need for a standard building ontology that can map existing building information as well as support software development.

Several algorithms have been proposed recently that map sensors in existing building sensors to a standard format – Zodiac [31] from our research group, Building Adapter [87], Bhattacharya et al. [38] and Gao et al. [74]. These proposed solutions are a first step as they standardized sensors names, equipment type and location information. However, building sensor ontology need to encompass relationship between sensors and equipment to present a holistic view of the system, and developers need tools for discovering this ontology. Preliminary work has been proposed by Pritoni et al. [146] and Koh et al. [105] from our research group that use perturbations in the control system to discover building ontology information in existing buildings.

To enable rapid development and deployment of software applications, developers need to be supported with development and debugging tools. Modern smartphone application developers are provided with development environment and emulation tools, and that has led to a plethora of applications available for users. Similar tools need to be developed to tackle the complexity of buildings with thousands of sensors and actuators. Software applications developed recently deploy and test directly on buidings which is both expensive and time consuming. Apple iOS HomeKit [13] has taken initial steps towards this direction for home environments with support for networked devices such as lights and thermostats. Considerable research is necessary to expand such tools for commercial buildings.

Chapter 9

Conclusion

Buildings are an essential part of our society. We spend majority of our time inside buildings and we use them to protect many of our resources from the environment. Our needs have evolved over time, and buildings have changed to meet our diverse requirements. This evolution is evidenced by advances in building infrastructure in the past century. Modern buildings consist of various systems such as HVAC, security, water and fire safety. However, these systems are designed and operated as stand alone units. But, a building is an integrated entity serving the needs of its owners, and the individual systems can better serve the needs of the owner if they work together as a single entity.

This dissertation has focused on the energy efficiency of buildings as a primary requirement. The overall energy consumption of the building depends on the interplay of all the different systems deployed in buildings, and hence, connecting these different systems lead to improvements in building energy efficiency. Therefore, instead of focusing on improvements in a single system or in deployment of additional hardware infrastructure which increase management overhead, our thesis focuses on software infrastructure that integrates information from different systems, and exploits them for implementing energy saving solutions.

We have presented four software systems in this dissertation:

- An occupancy based control of HVAC system that infers occupancy using existing WiFi infrastructure in buildings.
- An energy apportionment system that estimates zone level energy consumption by applying heat transfer equation to available sensor data.
- A web application designed to improve occupant interaction with the HVAC system. It allows occupants to manage their local settings and send complaints to building manager.
- A fault management system that addresses the limitations of the current building management tools.

With real-world implementation and evaluation of these systems we have illustrated that software solutions can indeed lead to significant energy savings. In addition, they can improve occupant comfort and ease maintenance of buildings. We have also elucidated the challenges involved in deployment of software systems in modern building infrastructure. To truly enable rapid development and deployment of software systems, we need to create the infrastructure that assists software developers. For example, machine readable formats of building information will standardize access of heterogenous information and improve information flow. Similarly, emulation tools will help developers test ideas in a virtual environment before expensive real-world deployments are made.

A first step towards this direction is normalization of information across different buildings. We presented our active learning based methodology of mapping existing building metadata to a standard format, so that developers do not need to repeat this mapping for each building they encounter.

My vision is to incorporate software and communication across systems as a core component of modern buildings. Given a suitable software infrastructure, we can deploy software applications that are customized to meet the requirements of the occupants or the building owner. Such an infrastructure is already in place for software applications in smartphones and laptops, and their benefits can be seen in all aspects of our society. With proliferation of such cyber physical systems, not only will we able to make our buildings smarter, but by connecting buildings and other types of infrastructure together, we can create entire societies that can mitigate our current problems and evolve for our future needs.

Bibliography

[1] Project Haystack. http://project-haystack.org/.

[2] Tridium Niagara AX. http://www.niagaraax.com/.

[3] Standard 55-2004, Thermal Environmental Conditions for Human Occupancy, Atlanta: American Society of Heating, Refrigerating, and Air-conditioning Engineers(ASHRAE). *Inc., USA*, 2004.

[4] Advanced automated hvac fault detection and diagnosis commercialization program. *Energy Research and Development Division Final Project Report*, 2008.

[5] Buildings Energy Databook(DoE). *Energy Efficiency & Renewable Energy Department*, 2011.

[6] Aereco - Demand Controlled Ventilation. http://www.aereco.com/ventilation-systems/demand-controlled-ventilation, March 2013.

[7] BACnet Stack. http://bacnet.sourceforge.net/, March 2013.

[8] Building Automation System Design and Construction Standards. *Energy Operations Group, Utilities and Energy Management, University of Rochester*, 2013.

[9] Enmetric Systems. http://www.enmetric.com/, March 2013.

[10] FPL - Demand Controlled Ventilation. http://www.fpl.com/business/energy_saving/programs/interior/dcv.shtml, March 2013.

[11] Philips Hue. https://www.meethue.com/, March 2013.

[12] pyrad 2.0. https://pypi.python.org/pypi/pyrad, March 2013.

[13] Apple iOS HomeKit. http://www.apple.com/ios/homekit/, December 2015.

[14] Comfy by Building Robotics. https://gocomfy.com/, December 2015.

[15] Energy Flow Charts: Charting the Complex Relationships among Energy, Water and Carbon. https://flowcharts.llnl.gov/, December 2015.

[16] Residential Buildings Integration. http://energy.gov/eere/buildings/residential-buildings-integration, December 2015.

[17] Unified Facilities Guide Specifications. *Department of Defense Unified Facilities Criteria*, 2015.

[18] Y. Agarwal, B. Balaji, S. Dutta, R.K. Gupta, and T. Weng. Duty-Cycling Buildings Aggressively: The Next Frontier in HVAC Control. In *Proc. of IEEE IPSN*, 2011.

[19] Y. Agarwal, B. Balaji, R. Gupta, J. Lyles, M. Wei, and T. Weng. Occupancy-Driven Energy Management for Smart Building Automation. In *Proc. of the 2nd ACM Workshop on BuildSys*.

[20] Y. Agarwal, T. Weng, and R.K. Gupta. The Energy Dashboard: Improving the Visibility of Energy Consumption at a Campus-Wide Scale. In *Proc. of ACM BuildSys*, 2009.

[21] Yuvraj Agarwal, Ranveer Chandra, Alec Wolman, Paramvir Bahl, Kevin Chin, and Rajesh Gupta. Wireless Wakeups Revisited: Energy management for VoIP over WiFi smartphones. In *Proc. of ACM MobiSys*, 2007.

[22] Yuvraj Agarwal, Rajesh Gupta, Daisuke Komaki, and Thomas Weng. BuildingDepot: An Extensible and Distributed Architecture for Building Data Storage, Access and Sharing. In *Proc. of ACM BuildSys*, 2012.

[23] Mohamed H Albadi and EF El-Saadany. A summary of demand response in electricity markets. *Electric power systems research*, 78(11):1989–1996, 2008.

[24] Hunt Allcott. Social norms and energy conservation. *Journal of Public Economics*, 95(9):1082–1095, 2011.

[25] Gustavo Alonso, Fabio Casati, Harumi Kuno, and Vijay Machiraju. *Web services*. Springer, 2004.

[26] Pandarasamy Arjunan, Nipun Batra, Haksoo Choi, Amarjeet Singh, Pushpendra Singh, and Mani B Srivastava. SensorAct: A Privacy and Security Aware Federated Middleware for Building Management. In *Proc. of ACM BuildSys*, 2012.

[27] Anil Aswani, Neal Master, Jay Taneja, David Culler, and Claire Tomlin. Reducing Transient and Steady State Electricity Consumption in HVAC using Learning-Based Model-Predictive Control. *Proc. of IEEE*, 2012.

[28] Stefan Aust, R Venkatesha Prasad, and Ignas GMM Niemegeers. IEEE 802.11 ah: Advantages in standards and further challenges for sub 1 GHz Wi-Fi. In *Proc. of IEEE ICC*, 2012.

[29] Paramvir Bahl and Venkata N Padmanabhan. RADAR: An In-Building RF-Based User Location and Tracking System. In *Proc. of IEEE Infocom*, 2000.

[30] Bharathan Balaji, Hidetoshi Teraoka, Rajesh Gupta, and Yuvraj Agarwal. ZonePAC: Zonal power estimation and control via hvac metering and occupant feedback. In *Proceedings of the 5th ACM Workshop on Embedded Systems For Energy-Efficient Buildings*, pages 1–8. ACM, 2013.

[31] Bharathan Balaji, Chetan Verma, Balakrishnan Narayanaswamy, and Yuvraj Agarwal. Zodiac: Organizing Large Deployment of Sensors to Create Reusable Applications for Buildings. In *Proceedings of the 2nd ACM International Conference on Embedded Systems for Energy-Efficient Built Environments*, pages 13–22. ACM, 2015.

[32] Niranjan Balasubramanian, Aruna Balasubramanian, and Arun Venkataramani. Energy consumption in mobile phones: A measurement study and implications for network applications. In *Proc. of ACM SIGCOMM IMC*, 2009.

[33] Michele Basseville. Distance measures for signal processing and pattern recognition. *Signal processing*, 18(4):349–369, 1989.

[34] Willy Bernal, Madhur Behl, Truong X Nghiem, and Rahul Mangharam. MLE+: a tool for integrated design and deployment of energy efficient building controls. In *Proc. of the 4th ACM Workshop on BuildSys*. ACM, 2012.

[35] Hugh Beyer and Karen Holtzblatt. *Contextual design: Ddefining customer-centered systems*. Elsevier, 1997.

[36] Arka Bhattacharya, David E Culler, Jorge Ortiz, Dezhi Hong, and Kamin Whitehouse. Enabling portable building applications through automated metadata transformation. Technical report, Technical Report UCB/EECS-2014-159, EECS Department, University of California, Berkeley, 2014.

[37] Arka Bhattacharya, Joern Ploennigs, and David Culler. Short Paper: Analyzing Metadata Schemas for Buildings: The Good, the Bad, and the Ugly. In *Proceedings of the 2nd ACM International Conference on Embedded Systems for Energy-Efficient Built Environments*, pages 33–34. ACM, 2015.

[38] Arka A Bhattacharya, Dezhi Hong, David Culler, Jorge Ortiz, Kamin Whitehouse, and Eugene Wu. Automated metadata construction to support portable building applications. In *Proceedings of the 2nd ACM International Conference on Embedded Systems for Energy-Efficient Built Environments*, pages 3–12. ACM, 2015.

[39] Bill Bordass, Ken Bromley, and Adrian Leaman. User and occupant controls in office buildings. In *International conference on building design, technology and occupant well-being in temperate climates, Brussels, Belgium*, pages 12–5, 1993.

[40] Michael J Brandemuehl and James E Braun. The Impact of Demand-Controlled and Economizer Ventilation Strategies on Energy Use in Buildings. Technical report, Univ. of Colorado, Boulder, CO (US), 1999.

[41] Steven T Bushby. BACnet™: A standard communication infrastructure for intelligent buildings. *Automation in Construction*, 6(5):529–540, 1997.

[42] James F Butler and Robert Veelenturf. Point naming standards. *ASHRAE Journal*, (November issue):B16–B20, 2010.

[43] Aaron Carroll and Gernot Heiser. An analysis of power consumption in a smartphone. In *Proc. of USENIX ATC*, 2010.

[44] Olivier Chapelle, Bernhard Schlkopf, and Alexander Zien. *Semi-Supervised Learning*. The MIT Press, 1st edition, 2010.

[45] Krishna Chintalapudi, Anand Padmanabha Iyer, and Venkata N Padmanabhan. Indoor localization without the pain. In *Proc. of ACM MobiCom*, 2010.

[46] Jaewoo Chung, Matt Donahoe, Chris Schmandt, Ig-Jae Kim, Pedram Razavai, and Micaela Wiseman. Indoor location sensing using geo-magnetism. In *Proc. of ACM MobiSys*, 2011.

[47] David E Claridge. Using simulation models for building commissioning. In *International Conference on Enhanced Building Operation, Energy Systems Laboratory, Texas A&M University*, 2004.

[48] David Cohn, Les Atlas, and Richard Ladner. Improving generalization with active learning. *Machine learning*, 15(2):201–221, 1994.

[49] California Energy Commission. Nonresidential compliance manual, 2005.

[50] Michael Compton, Cory Henson, Laurent Lefort, Holger Neuhaus, and Amit P Sheth. A survey of the semantic specification of sensors. 2009.

[51] Johnson Controls. Metasys Facility Management System, 1999.

[52] Drury B Crawley, Linda K Lawrie, Frederick C Winkelmann, Walter F Buhl, Y Joe Huang, Curtis O Pedersen, Richard K Strand, Richard J Liesen, Daniel E Fisher, Michael J Witte, and Jason Glazer. Energyplus: creating a new-generation building energy simulation program. *Energy and buildings*, 33(4):319–331, 2001.

[53] Cypress Envirosystems. Retrofitting Existing Buildings for Demand Response and Energy Efficiency, 2015. Available online at: http://www.aashe.org/files/documents/webinars/Retrofitting_Existing_Buildings.PPT_.pdf.

[54] Sarah Darby. The effectiveness of feedback on energy consumption. *A Review for DEFRA of the Literature on Metering, Billing and direct Displays*, 486:2006, 2006.

[55] Sanjoy Dasgupta and Daniel Hsu. Hierarchical sampling for active learning. In *Proceedings of the 25th international conference on Machine learning*, pages 208–215. ACM, 2008.

[56] S. Dawson-Haggerty, S. Lanzisera, J. Taneja, R. Brown, and D. Culler. @ scale: Insights from a Large, Long-Lived Appliance Energy WSN. In *Proc. of IEEE IPSN*, 2012.

[57] Stephen Dawson-Haggerty, Xiaofan Jiang, Gilman Tolle, Jorge Ortiz, and David Culler. sMAP: A Simple Measurement and Actuation Profile for Physical Information. In *Proc. of ACM SenSys*, 2010.

[58] Stephen Dawson-Haggerty, Andrew Krioukov, Jay Taneja, Sagar Karandikar, Gabe Fierro, Nikita Kitaev, and David Culler. BOSS: Building Operating System Services. In *Proc. of USENIX NSDI*, 2013.

[59] RJ Dear, T Akimoto, EA Arens, G Brager, C Candido, KWD Cheong, B Li, N Nishihara, SC Sekhar, S Tanabe, J Toftum, H Zhang, and Y Zhu. Progress in thermal comfort research over the last twenty years. *Indoor air*, 23(6):442–461, 2013.

[60] Samuel DeBruin, Bradford Campbell, and Prabal Dutta. Monjolo: An energy-harvesting energy meter architecture. In *Proceedings of the 11th ACM Conference on Embedded Networked Sensor Systems*, page 18. ACM, 2013.

[61] Zhimin Du, Xinqiao Jin, and Yunyu Yang. Fault diagnosis for temperature, flow rate and pressure sensors in VAV systems using wavelet neural network. *Applied energy*, 86(9):1624–1631, 2009.

[62] SJ Emmerich, JW Mitchell, and WA Beckman. Demand-Controlled Ventilation in a Multi-zone Office Building. *Indoor and Built Environment*, 1994.

[63] Varick L Erickson and Alberto E Cerpa. Thermovote: Participatory sensing for efficient building HVAC conditioning. In *Proc. of the 4th ACM Workshop on BuildSys*. ACM, 2012.

[64] V.L. Erickson, S. Achleitner, and A.E. Cerpa. POEM: Power-Efficient Occupancy-Based Energy Management System. In *Proc. of IEEE IPSN*, 2013.

[65] V.L. Erickson, M.Á. Carreira-Perpiñán, and A.E. Cerpa. OBSERVE: Occupancy-Based System for Efficient Reduction of HVAC Energy. In *Proc. of IEEE IPSN*, 2011.

[66] Poul O Fanger. Thermal comfort: Analysis and applications in environmental engineering. *Thermal comfort. Analysis and applications in environmental engineering.*, 1970.

[67] N Fernandez, H Cho, MR Brambley, J Goddard, S Katipamula, and L Dinh. Self-Correcting HVAC Controls. 2009.

[68] William J Fisk and Anibal T De Almeida. Sensor-Based Demand-Controlled Ventilation: A Review. *Energy and buildings*, 1998.

[69] William J Fisk, D Faulkner, and DP Sullivan. A Pilot Study of the Accuracy of CO_2 Sensors in Commercial Buildings. *Lawrence Berkeley National Laboratory Paper LBNL E*, 2008.

[70] Romain Fontugne, Jorge Ortiz, Nicolas Tremblay, Pierre Borgnat, Patrick Flandrin, Kensuke Fukuda, David Culler, and Hiroshi Esaki. Strip, Bind, and Search: A method for identifying abnormal energy consumption in buildings. In *IPSN*. ACM, 2013.

[71] Marc Fountain, Gail Brager, and Richard de Dear. Expectations of indoor climate control. *Energy and Buildings*, 24(3):179–182, 1996.

[72] Jon Froehlich, Leah Findlater, and James Landay. The design of eco-feedback technology. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 1999–2008. ACM, 2010.

[73] Jon E Froehlich, Eric Larson, Tim Campbell, Conor Haggerty, James Fogarty, and Shwetak N Patel. HydroSense: Infrastructure-mediated single-point sensing of whole-home water activity. In *Proceedings of the 11th international conference on Ubiquitous computing*, pages 235–244. ACM, 2009.

[74] Jingkun Gao, Joern Ploennigs, and Mario Berges. A Data-driven Meta-data Inference Framework for Building Automation Systems. In *Proceedings of the 2nd ACM International Conference on Embedded Systems for Energy-Efficient Built Environments*, pages 23–32. ACM, 2015.

[75] Peter Xiang Gao and Srinivasan Keshav. SPOT: A smart personalized office thermal control system. In *Proceedings of the fourth international conference on future energy systems*, pages 237–246. ACM, 2013.

[76] NIST GCR. Cost analysis of inadequate interoperability in the US capital facilities industry. *National Institute of Standards and Technology (NIST)*, 2004.

[77] Pierre Geurts. Pattern extraction for time series classification. In *Principles of Data Mining and Knowledge Discovery*, pages 115–127. Springer, 2001.

[78] Sunil Kumar Ghai, Lakshmi V Thanayankizil, Deva P Seetharam, and Dipanjan Chakraborty. Occupancy Detection in Commercial Buildings using Opportunistic Context Sources. In *In IEEE Percom Workshops*, 2012.

[79] Siddharth Goyal, Herbert A Ingley, and Prabir Barooah. Occupancy-based zone-climate control for energy-efficient buildings: Complexity vs. Performance. *Applied Energy*, 2013.

[80] Jessica Granderson, Mary Ann Piette, and Girish Ghatikar. Building energy information systems: User case studies. *Energy Efficiency*, 4(1):17–30, 2011.

[81] Joshua A Grochow and Manolis Kellis. Network motif discovery using subgraph enumeration and symmetry-breaking. In *Research in Computational Molecular Biology*, pages 92–106. Springer, 2007.

[82] Jinsoo Han, Youn-Kwae Jeong, and Ilwoo Lee. Efficient building energy management system based on ontology, inference rules, and simulation. In *Proceedings of the 2011 International Conference on Intelligent Building and Management, Singapore*, volume 5, 2011.

[83] Tom Hargreaves, Michael Nye, and Jacquelin Burgess. Making energy visible: A qualitative field study of how householders interact with feedback from smart energy monitors. *Energy policy*, 38(10):6111–6119, 2010.

[84] Philip Haves and Moosung Kim. Model-Based Automated Functional Testing - Methodology and Application to Air-Handling Units. 2005.

[85] James J Hirsch. DOE 2.2 Building Energy Use and Cost Analysis Program. *See http://www. doe2. com*, 2003.

[86] T.W. Hnat, V. Srinivasan, J. Lu, T.I. Sookoor, R. Dawson, J. Stankovic, and K. Whitehouse. The Hitchhiker's Guide to Successful Residential Sensing Deployments. In *Proc. of the 9th ACM Conference on Embedded Networked Sensor Systems*, pages 232–245. ACM, 2011.

[87] Dezhi Hong, Hongning Wang, Jorge Ortiz, and Kamin Whitehouse. The Building Adapter: Towards Quickly Applying Building Analytics at Scale. In *Proceedings of the 2nd ACM International Conference on Embedded Systems for Energy-Efficient Built Environments*, pages 123–132. ACM, 2015.

[88] Shyh-Jier Huang, Cheng-Tao Hsieh, Lun-Chia Kuo, Chun-Wei Lin, Che-Wei Chang, and Shyang-An Fang. Classification of home appliance electricity consumption using power signature and harmonic features. In *Power Electronics and Drive Systems (PEDS), 2011 IEEE Ninth International Conference on*, pages 596–599. IEEE, 2011.

[89] C Huizenga, S Abbaszadeh, Leah Zagreus, and Edward A Arens. Air quality and thermal comfort in office buildings: Results of a large indoor environmental quality survey. *Center for the Built Environment*, 2006.

[90] Mark Hydeman. *Advanced Variable Air Volume: System Design Guide: Design Guidelines*. California Energy Commission, 2003.

[91] Farrokh Jazizadeh and Burcin Becerik-Gerber. Toward adaptive comfort management in office buildings using participatory sensing for end user driven control. In *Proc. of the 4th ACM Workshop on BuildSys*. ACM, 2012.

[92] Farrokh Jazizadeh, Ali Ghahramani, Burcin Becerik-Gerber, Tatiana Kichkaylo, and Michael Orosz. Human-building interaction framework for personalized thermal comfort-driven systems in office buildings. *Journal of Computing in Civil Engineering*, 2013.

[93] Farrokh Jazizadeh, Franco Moiso Marin, and Burcin Becerik-Gerber. A Thermal Preference Scale for Personalized Comfort Profile Identification via Participatory Sensing. *Building and Environment*, 2013.

[94] Xiaofan Jiang, Stephen Dawson-Haggerty, Prabal Dutta, and David Culler. Design and Implementation of a High-Fidelity AC Metering Network. In *IPSN 2009*, pages 253–264. IEEE, 2009.

[95] Stephen C Johnson. Hierarchical clustering schemes. *Psychometrika*, 32(3):241–254, 1967.

[96] Sami Karjalainen. Thermal comfort and use of thermostats in finnish homes and offices. *Building and Environment*, 44(6):1237–1245, 2009.

[97] Sami Karjalainen. Usability guidelines for room temperature controls. *Intelligent Buildings International*, 2(2):85–97, 2010.

[98] Sami Karjalainen and Olavi Koistinen. User problems with individual temperature control in offices. *Building and Environment*, 42(8):2880–2887, 2007.

[99] Srinivas Katipamula and Michael R Brambley. Review Article: Methods for Fault Detection, Diagnostics, and Prognostics for Building Systems-A Review, Part I. *HVAC&R Research*, 11(1):3–25, 2005.

[100] Srinivas Katipamula and Michael R Brambley. Review Article: Methods for Fault Detection, Diagnostics, and Prognostics for Building Systems-A Review, Part II. *HVAC&R Research*, 11(2):169–187, 2005.

[101] Eamonn J Keogh and Michael J Pazzani. An enhanced representation of time series which allows fast and accurate classification, clustering and relevance feedback. In *KDD*, volume 98, pages 239–243, 1998.

[102] Ritesh Khire, Francesco Leonardi, Paul Quimby, and Soumik Sarkar. A Novel Human Machine Interface for Advanced Building Controls and Diagnostics. 2014.

[103] Ashraf M Kibriya, Eibe Frank, Bernhard Pfahringer, and Geoffrey Holmes. Multinomial naive bayes for text categorization revisited. In *AI 2004: Advances in Artificial Intelligence*, pages 488–499. Springer, 2005.

[104] Neil E Klepeis, William C Nelson, Wayne R Ott, John P Robinson, Andy M Tsang, Paul Switzer, Joseph V Behar, Stephen C Hern, and William H Engelmann. The National Human Activity Pattern Survey (NHAPS): A resource for assessing exposure to environmental pollutants. *Journal of exposure analysis and environmental epidemiology*, 11(3):231–252, 2001.

[105] Jason Koh, Bharathan Balaji, Rajesh Gupta, and Yuvraj Agarwal. Controlling Actuation in Central HVAC Systems in Buildings. In *Proceedings of the 2nd ACM International Conference on Embedded Systems for Energy-Efficient Built Environments*, pages 105–106. ACM, 2015.

[106] J Zico Kolter and Matthew J Johnson. REDD: A public data set for energy disaggregation research. In *Workshop on Data Mining Applications in Sustainability (SIGKDD), San Diego, CA*, volume 25, pages 59–62. Citeseer, 2011.

[107] So Young Koo, Myoung Souk Yeo, and Kwang Woo Kim. Automated blind control to maximize the benefits of daylight in buildings. *Building and Environment*, 45(6):1508–1520, 2010.

[108] A. Krioukov, S. Dawson-Haggerty, L. Lee, O. Rehmane, and D. Culler. A Living Laboratory Study in Personalized Automated Lighting Controls. In *Proc. of ACM BuildSys*, 2011.

[109] Andrew Krioukov, Gabe Fierro, Nikita Kitaev, and David Culler. Building Application Stack (BAS). In *Proceedings of the Fourth ACM Workshop on Embedded Sensing Systems for Energy-Efficiency in Buildings*, pages 72–79. ACM, 2012.

[110] Patrick Lazik and Anthony Rowe. Indoor pseudo-ranging of mobile devices using ultrasonic chirps. In *Proc. of ACM SenSys*, 2012.

[111] Kwang Jun Lee, Omer Akin, Burcu Akinci, James Garrett, and Steven Bushby. Ontology development for low-energy building embedded commissioning. 2009.

[112] David Lehrer and Janani Vasudev. Visualizing Information to Improve Building Performance: A study of expert users. *Center for the Built Environment*, 2010.

[113] David Lehrer and Janani Vasudev. Visualizing Energy Information in Commercial Buildings: A Study of Tools, Expert Users, and Building Occupants. 2011.

[114] David D Lewis and William A Gale. A sequential algorithm for training text classifiers. In *Proceedings of the 17th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 3–12. Springer-Verlag New York, Inc., 1994.

[115] Jian Liang and Ruxu Du. Model-based fault detection and diagnosis of HVAC systems using support vector machine method. *International Journal of refrigeration*, 30(6):1104–1114, 2007.

[116] Andy Liaw and Matthew Wiener. Classification and regression by randomForest. *R news*, 2(3):18–22, 2002.

[117] Jessica Lin, Eamonn Keogh, Stefano Lonardi, and Bill Chiu. A symbolic representation of time series, with implications for streaming algorithms. In *Proceedings of the 8th ACM SIGMOD workshop on Research issues in data mining and knowledge discovery*, pages 2–11. ACM, 2003.

[118] Siu Hing Lo, Gjalt-Jorn Y Peters, and Gerjo Kok. Energy-Related Behaviors in Office Buildings: A Qualitative Study on Individual and Organisational Determinants. *Applied Psychology*, 61(2):227–249, 2012.

[119] David S Loughran and Jonathan Kulick. Demand-side management and energy efficiency in the United States. *The Energy Journal*, pages 19–43, 2004.

[120] K Louise Barriball and Alison While. Collecting Data using a semi-structured interview: A discussion paper. *Journal of advanced nursing*, 19(2):328–335, 1994.

[121] Larry M Manevitz and Malik Yousef. One-class SVMs for document classification. *the Journal of machine Learning research*, 2:139–154, 2002.

[122] Jennifer Mankoff, Deanna Matthews, Susan R Fussell, and Michael Johnson. Leveraging social networks to motivate individuals to reduce their ecological footprints. In *System Sciences, 2007. HICSS 2007. 40th Annual Hawaii International Conference on*, pages 87–87. IEEE, 2007.

[123] C. Martani, D. Lee, P. Robinson, R. Britter, and C. Ratti. ENERNET: Studying the Dynamic Relationship between Building Occupancy and Energy Consumption. *Energy and Buildings*, 2011.

[124] Alan Meier. Thermostat interface and usability: a survey. *Lawrence Berkeley National Laboratory*, 2011.

[125] Ryan Melfi, Ben Rosenblum, Bruce Nordman, and Ken Christensen. Measuring Building Occupancy using Existing Network Infrastructure. In *Proc. of IEEE IGCC*, 2011.

[126] Evan Mills. Building commissioning: A golden opportunity for reducing energy costs and greenhouse-gas emissions. Technical report, Ernest Orlando Lawrence Berkeley National Laboratory, Berkeley, CA (US), 2009.

[127] Evan Mills. Building commissioning: A golden opportunity for reducing energy costs and greenhouse-gas emissions. *Lawrence Berkeley National Laboratory*, 2010.

[128] Evan Mills. Building commissioning: A golden opportunity for reducing energy costs and greenhouse gas emissions in the United States. *Energy Efficiency*, 4(2):145–173, 2011.

[129] Evan Mills and Paul Mathew. Monitoring Based Commissioning: Benchmarking Analysis of 24 UC/CSU/IOU Projects. Technical report, Ernest Orlando Lawrence Berkeley National Laboratory, Berkeley, CA (US), 2009.

[130] Yoshifumi Murakami, Masaaki Terano, Kana Mizutani, Masayuki Harada, and Satoru Kuno. Field experiments on energy consumption and thermal comfort in the office environment controlled by occupants requirements from PC terminal. *Building and Environment*, 42(12):4022–4027, 2007.

[131] Balakrishnan Narayanaswamy, Bharathan Balaji, Rajesh Gupta, and Yuvraj Agarwal. Data driven investigation of faults in HVAC systems with Model, Cluster and Compare (MCC). In *Proceedings of the 1st ACM Conference on Embedded Systems for Energy-Efficient Buildings*, pages 50–59. ACM, 2014.

[132] Inc. Navigant Consulting. Energy savings potential of solid state lighting in general lighting applications. *US Department of Energy, Washington, DC*, 2014.

[133] Kamal Nigam and Andrew McCallum. Employing EM in pool-based active learning for text classification. In *Proceedings of ICML-98, 15th International Conference on Machine Learning*, 1998.

[134] Frauke Oldewurtel, Alessandra Parisio, Colin N Jones, Manfred Morari, Dimitrios Gyalistras, Markus Gwerder, Vanessa Stauch, Beat Lehmann, and Katharina Wirth. Energy Efficient Building Climate Control using Stochastic Model Predictive Control and Weather Predictions. In *Proc. of IEEE ACC*, 2010.

[135] Levi Olsen. Equipment Naming Convention. *University of Alberta, Facilities and Operations Design Guidelines*, 2015.

[136] Building Owners. Managers association (boma) international and uli the urban land institute. what office tenants want: 1999 boma/uli office tenant survey report. washington, dc, boma international and ulithe urban land institute. *Results of a survey of 1800 office building tenants across the US and Canada*, 1999.

[137] Monica Paciuk. *The role of personal control of the environment in thermal comfort and satisfaction at the workplace*. PhD thesis, University of Wisconsin-Milwaukee, 1989.

[138] Shwetak N Patel, Matthew S Reynolds, and Gregory D Abowd. Detecting human movement by differential air pressure sensing in HVAC system ductwork: An exploration in infrastructure mediated sensing. In *Pervasive Computing*. 2008.

[139] Shwetak N Patel, Thomas Robertson, Julie A Kientz, Matthew S Reynolds, and Gregory D Abowd. At the flick of a switch: Detecting and classifying unique electrical events on the residential power line. *Lecture Notes in Computer Science*, 4717:271–288, 2007.

[140] Shwetak N Patel, Khai N Truong, and Gregory D Abowd. Powerline positioning: A practical sub-room-level indoor location system for domestic use. In *In Proc. of UbiComp*. 2006.

[141] Therese Peffer, Marco Pritoni, Alan Meier, Cecilia Aragon, and Daniel Perry. How people use thermostats in homes: A review. *Building and Environment*, 46(12):2529–2541, 2011.

[142] Luis Pérez-Lombard, José Ortiz, and Christine Pout. A Review on Buildings Energy Consumption Information. *Energy and Buildings*, 40(3):394–398, 2008.

[143] John E Petersen, Vladislav Shunturov, Kathryn Janda, Gavin Platt, and Kate Weinberger. Dormitory residents reduce electricity consumption when exposed to real-time visual feedback and incentives. *International Journal of Sustainability in Higher Education*, 8(1):16–33, 2007.

[144] British Petroleum. BP Statistical Review of World Energy 2015, 2015.

[145] Mary Ann Piette. Open automated demand response communications specification (Version 1.0). *Lawrence Berkeley National Laboratory*, 2009.

[146] Marco Pritoni, Arka A Bhattacharya, David Culler, and Mark Modera. Short Paper: A Method for Discovering Functional Relationships Between Air Handling Units and Variable-Air-Volume Boxes From Sensor Data. In *Proceedings of the 2nd ACM International Conference on Embedded Systems for Energy-Efficient Built Environments*, pages 133–136. ACM, 2015.

[147] Ahmad Rahmati and Lin Zhong. Context-for-wireless: Context-sensitive energy-efficient wireless data transfer. In *Proc. of ACM MobiSys*, 2007.

[148] Carl Edward Rasmussen. The infinite Gaussian mixture model. In *NIPS*, volume 12, pages 554–560, 1999.

[149] Christian Reinisch, Wolfgang Granzer, Fritz Praus, and Wolfgang Kastner. Integration of heterogeneous building automation systems using ontologies. In *Industrial Electronics, 2008. IECON 2008. 34th Annual Conference of IEEE*, pages 2736–2741. IEEE, 2008.

[150] Simon Roberts, Helen Humphries, and Verity Hyldon. Consumer preferences for improving energy consumption feedback. *Report to Ofgem, Centre for Sustainable Energy*, 2(3):19, 2004.

[151] A. Rowe, M.E. Berges, G. Bhatia, E. Goldman, R. Rajkumar, J.H. Garrett, J.M.F. Moura, and L. Soibelman. Sensor Andrew: Large-Scale Campus-Wide Sensing and Actuation. *IBM Journal of Research and Development*, 2011.

[152] David J Russomanno, Cartik R Kothari, and Omoju A Thomas. Building a Sensor Ontology: A Practical Approach Leveraging ISO and OGC Models. In *IC-AI*, pages 637–643, 2005.

[153] J Sauer, DG Wastell, and C Schmeink. Designing for the home: A comparative study of support aids for central heating systems. *Applied Ergonomics*, 40(2):165–174, 2009.

[154] Jeffrey Schein, Steven T Bushby, Natascha S Castro, and John M House. A rule-based fault detection method for air handling units. *Energy and Buildings*, 38(12):1485–1492, 2006.

[155] Anika Schumann, Jer Hayes, Pascal Pompey, and Olivier Verscheure. Adaptable fault identification for smart buildings. In *Artificial Intelligence and Smarter Living*, 2011.

[156] scikit-learn Machine Learning in Python, 2015. http://scikit-learn.org/.

[157] O Seppanen, William J Fisk, and QH Lei. Room temperature and productivity in office work. *Lawrence Berkeley National Laboratory*, 2006.

[158] Burr Settles. Active learning literature survey. *University of Wisconsin, Madison*, 52(55-66):11, 2010.

[159] H Sebastian Seung, Manfred Opper, and Haim Sompolinsky. Query by committee. In *Proceedings of the fifth annual workshop on Computational learning theory*, pages 287–294. ACM, 1992.

[160] Atul Sharma, VV Tyagi, CR Chen, and D Buddhi. Review on thermal energy storage with phase change materials and applications. *Renewable and Sustainable energy reviews*, 13(2):318–345, 2009.

[161] Amit Sheth, Cory Henson, and Satya S Sahoo. Semantic sensor web. *Internet Computing, IEEE*, 12(4):78–83, 2008.

[162] SkyFoundry. SkySpark Analytic Rules: Combining a Comprehensive Library of Analytic Functions with Full Programmability. Feb 2014.

[163] ASHRAE Standard. Standard 55-2004. *Thermal environmental conditions for human occupancy*, 2004.

[164] Anselm Strauss and Juliet M Corbin. *Basics of qualitative research: Grounded theory procedures and techniques.* Sage Publications, Inc, 1990.

[165] Jay Taneja, Andrew Krioukov, Stephen Dawson-Haggerty, and David E Culler. Enabling Advanced Environmental Conditioning with a Building Application Stack. 2013.

[166] Hidetoshi Teraoka, Bharathan Balaji, Rizhen Zhang, Anthony Nwokafor, Balakrishnan Narayanaswamy, and Yuvraj Agarwal. BuildingSherlock: Fault Management Framework for HVAC Systems in Commercial Buildings. Technical report, Technical Report, CSE, UCSD, 2014.

[167] Hidetoshi Teraoka, Bharathan Balaji, Rizhen Zhang, Anthony Nwokafor, Balakrishnan Narayanaswamy, and Yuvraj Agarwal. BuildingSherlock: Fault Management Framework for HVAC Systems in Commercial Buildings. *Technical Report, CSE, UCSD*, 2014.

[168] L.V. Thanayankizil, S.K. Ghai, D. Chakraborty, and D.P. Seetharam. Softgreen: Towards Energy Management of Green Office Buildings with Soft Sensors. In *In Proc. of IEEE COMSNETS*, 2012.

[169] Tali Trigg, Paul Telleen, Rachael Boyd, Francois Cuenot, Davide DAmbrosio, Rebecca Gaghen, Jean-Francois Gagné, Annette Hardcastle, Didier Houssin, Amb. Richard Jones, Hiroyuki Kaneko, Melissa C Lott, Lizzy Spong, Kathleen Sullivan, Cecilia Tam, and Markus Wrake. Global EV outlook: Understanding the electric vehicle landscape to 2020. *International Energy Agency*, pages 1–40, 2013.

[170] Daniel Turner, Stefan Savage, and Alex C Snoeren. On the Empirical Performance of Self-Calibrating Wifi Location Systems. In *Prof. of IEEE LCN*, 2011.

[171] Joost Van Hoof. Forty years of Fangers model of thermal comfort: Comfort for all? *Indoor air*, 18(3):182–201, 2008.

[172] Joost Van Hoof, Mitja Mazej, and Jan LM Hensen. Thermal comfort: Research and practice. *Frontiers in Bioscience*, 15(2):765–788, 2010.

[173] Iain S Walker and Alan K Meier. Residential thermostats: Comfort controls in California homes. *Lawrence Berkeley National Laboratory*, 7, 2008.

[174] He Wang, Souvik Sen, Ahmed Elgohary, Moustafa Farid, Moustafa Youssef, and Romit Roy Choudhury. No need to war-drive: Unsupervised indoor localization. In *Proc. of ACM MobiSys*, 2012.

[175] Thomas Weng, Bharathan Balaji, Seemanta Dutta, Rajesh Gupta, and Yuvraj Agarwal. Managing plug-loads for demand response within buildings. In *Proc. of the 3rd ACM Workshop on BuildSys*. ACM, 2011.

[176] Thomas Weng, Anthony Nwokafor, and Yuvraj Agarwal. BuildingDepot 2.0: An Integrated Management System for Building Analysis and Control. In *BuildSys*. ACM, 2013.

[177] Ernst Worrell, Lenny Bernstein, Joyashree Roy, Lynn Price, and Jochen Harnisch. Industrial energy efficiency and climate change mitigation. *Energy Efficiency*, 2(2):109–123, 2009.

[178] David Wyon. Individual microclimate control: Required range, probable benefits and current feasibility. In *7th International Conference on Indoor Air Quality and Climate*, pages 1067–1072, 1996.

[179] Jin Yang, Hugues Rivard, and Radu Zmeureanu. On-line building energy prediction using adaptive artificial neural networks. *Energy and buildings*, 37(12):1250–1259, 2005.

[180] Moustafa Youssef and Ashok Agrawala. The Horus WLAN location determination system. In *Proc. of ACM MobiSys*, 2005.

[181] Jian Zhang, G Liu, RG Lutes, and Michael R Brambley. Energy Savings for Occupancy-Based Control (OBC) of Variable-Air-Volume (VAV) Systems. Technical report, Pacific Northwest National Laboratory (PNNL), Richland, WA (US), 2013.

[182] Lide Zhang, Birjodh Tiwana, Zhiyun Qian, Zhaoguang Wang, Robert P Dick, Zhuoqing Morley Mao, and Lei Yang. Accurate online power estimation and automatic battery behavior based power model generation for smartphones. In *Proc. of ACM CODES+ISSS*, 2010.

[183] Yin Zhang, Rong Jin, and Zhi-Hua Zhou. Understanding bag-of-words model: a statistical framework. *International Journal of Machine Learning and Cybernetics*, 1(1-4):43–52, 2010.