Proceedings of The National Conference
On Undergraduate Research (NCUR) 2012
Weber State University, Ogden, Utah
March 29 – 31, 2012

"Automated Energy Management Integrating Green Technologies"

Alexander Farmer
Blake Rulison
Information Technology
United States Military Academy
West Point, New York 10997 USA

Faculty Advisors: Dr. Aaron St. Leger and Colonel Edward Sobiesk

Abstract

Across the world, utility companies and consumers are seeking more effective methods of responding to power demands and integrating renewable energy sources. As emphasis on renewable energy increases, green power sources are becoming more readily available as supplements to standard utility power distribution. This paper presents the research and development of a load management system that autonomously controls and dynamically manages power allocation based on inputs from the following three sources: the user, the utility company, and the environment. More specifically, a computer program manages the interaction of renewable energy generation, energy consumption and energy provided from the grid. This system also allows for utility controlled demand response in urgent situations constrained to user preferences. The load management system presented here is implemented as a client server architecture using a Ruby on Rails web interface that coordinates with other hardware and software components to control power allotment based on pre-defined preferences and historical data that has been logged in a database. The automated control system allows the user to view the current state of activity in a web display and to dynamically update settings from remote locations and diverse computing platforms.

Keywords: Energy Management, Smart Grid, Green Technology, Renewable Energy

1. Introduction

In today's environment, electrical power comes from many different sources. The power that runs everything from household electrical appliances to the lights in large corporate offices comes from the power grid of electricity providers using generators that primarily run on nonrenewable resources. There are several limitations that this structure and these electricity providers, or utility companies, have to deal with on a daily basis. These limitations include an inability of companies to accommodate large demand requirements that overload the grid.[1] This is a significant issue faced by utility companies today. As recent legislation and popular trends in information technology add more complications and possibilities to this domain, pressure increases to find improved methods for resource management.

1.1 Demand Response

Demand response is the change in electric usage by consumers from their normal consumption patterns in response to changes in the price of electricity over time, or to incentive payments designed to induce lower electricity use at times of high wholesale market prices or when system reliability is jeopardized.[2] Utility companies are beginning to

establish mechanisms which encourage users to consume less power during the day, or during specified peak load hours. These peak hours require the greatest support from generators and put the utility companies under significant strain. Con Edison is a utility in the northeast that has tackled this problem by establishing a new pricing model that makes consuming power during peak times more expensive than during off peak hours.[3] Utility companies work with customers to curtail peak load requirements, responding to the load demanded by their consumers as a whole.

Individual homes or small businesses are the most responsive to this pricing structure as they are most able to modify their electricity usage during the normal business hours to increase profitability. However, the amount of money saved depends on the effort the consumer expends to lower their electricity requirements during peak pricing. Ranging from ensuring lights are turned off to only doing laundry during off peak hours, load restraint is all dependent on the consumers' ability to manually follow this structure. These consumers would be able to save more money if they can monitor their system and receive feedback about their power consumption in some manner.[4] Currently there is a lack of household level suppliers for this kind of visualized system and many users must pay their electric bill without any indication of how particular charges are tracked.

1.2 Going Green

Along with user driven support for applications that increase visibility and control over a house's electrical systems there has also been a significant rise in the use of home installed renewable energy sources.[5] Solar panels, which are also called photovoltaic panels (PV panels), are the primary renewable energy source currently used.[6] PV panels are a one-time purchase and once installed essentially provide cost-free energy to supplement the house they are connected to when the panels are in direct sunlight. Another inherent bonus renewable resources provide is that if more solar energy is being produced by a house than is being consumed, the extra solar energy will be automatically sold back to the power grid. Pay periods at companies like Con Edison are settled after an entire year and this type of sell-back option is facilitated because only the net consumption, or net metering, is charged. However, utility companies will not pay consumers full price for net excess power generation at the end of a full payment period.

One of the inherent qualities of these green energy resources is that the instantaneous power output is often unpredictable. When a large cloud passes between the sun and your solar panels the amount of power produced by those panels significantly decreases. Imagine a city of houses that each have a suite of solar panels which supplement their grid power consumption. A wall of clouds passing over for several hours would normally cause little reason for concern but this situation causes incredible strain and possibly even leads to power outages with such a large deprivation of PV power. This would cause several extra generators to begin to power up as the load is shifted by the power company to other utility power grids to respond to the sudden spike in load requirement. Management of load can help smooth the demand curve, allowing utility companies more time to react to changes.

1.3 Future Considerations

Utility companies are also looking for ways to leverage the power of a more intelligent grid, or smart grid. The smart grid is a general term for the introduction of more sophisticated control mechanisms that exist within the grid itself and the appliances that are attached. One example of this is an automated thermostat that can sense when it should turn on based not only on temperature but also adjusts according to the price of electricity.[7] Applications also now exist that allow customers and home owners to remotely manipulate their lights or garage doors using only their smart phones. These technology applications have driven consumer demand for integrated control over other systems in their homes.

Utility companies are recognizing the increasing importance of being able handle this type of situation. From the perspective of the utility company, there is little that can be done except to develop and build the infrastructure that would be able to compensate. This infrastructure is expensive and will require several years of development and installation. Indeed, in "Smart Integration," Ali Vojdani argues that the power grid is in dire need of a flexible and dynamic architecture.[4] Requirements include time-variable pricing, peak load curtailment through demand-side management, distributed generation and smart metering.

All of these strategies involve adding flexibility to a once rigid system. However, Katherine Hamilton and Neel Gulhar point out in their article "Taking Demand Response to the Next Level," the greenest method for controlling

power demand would be at the "demand-side" or at the level of the consumer through coordinating or incentivizing the consumers load requirements in order to lessen requirements for fossil fuels during that time.[8] In a nutshell, accomplishing this is a significant goal of the work presented in this paper.

1.4 Research Goal

This work designs and builds an effective system that employs user and power utility defined control attributes to autonomously manage power consumption of a home that draws on both grid and renewable power sources. The key component of this challenge is "effectiveness" as systems could be built that meet all other requirements but remain unusable to the consumer or utility, or not take advantage of pricing structures and renewable resources available.

1.5 Scope

Our proposed solution includes two operating modes. The first operating mode is a combination of demand response and integration of PV into the grid with the goal to minimize energy cost. The second mode allows utility controlled demand response constrained to user preferences. The first problem will be solved by regulating the amount of power used by an individual consumer through a software interface based on the pricing of grid energy, the current available amount of PV power, and user preferences. The second is similar but will be modified so that the utility company has access only to those devices the consumers have volunteered. The primary focus are the consumers, concerned with power availability and cost, and the utility company, concerned with efficient load management over the grid. Figure 1 depicts the position of the Load Management System.

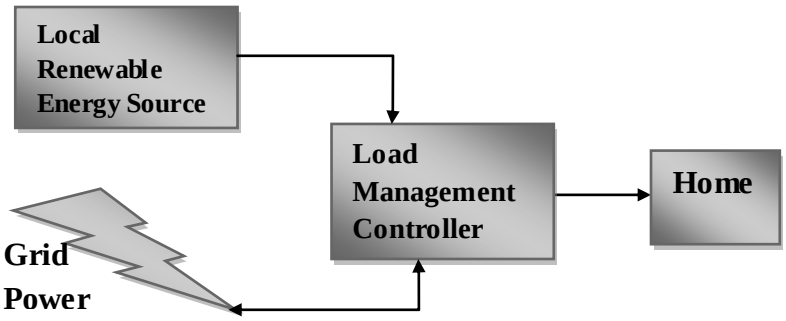


Figure 1: System Perspective

2. System Overview

The Intelligent Load Management System, or ILMS, involves a combination of software and hardware that provides an automated method for intelligently managing the power usage of a home. The user will interface with the system through a simple web application that stores information on and reads information from a local database. Additionally, management scripts will evaluate power consumption, green power generation, user inputs, and time of day every 30 seconds to make decisions in real time that determine the operation of the system.

2.1 Software Requirements

2.1.1 Interface with a microcontroller to provide real time data and control demand response
2.1.2 Communicate with the user through a GUI that also depicts the current state of the system
2.1.3 Run configurable consumer-based algorithms using present state, utility and user-set parameters
2.1.4 Construct and manage a database for information storage

2.2 Previous Works and Versions

This is the third year that this project has been a senior capstone design project for West Point. The only aspects of the previous project iterations used in the existing system are the hardware for simulating a home and the software running on the microcontroller associated with this hardware. All other software involved in this system has been designed and developed over the past year by the authors of this work.

2.3 Assumptions

It is assumed that the user will have a computer with connectivity to the ILMS. The current system uses an internally hosted web server. We are also assuming that the ILMS control hardware would be configured properly in relation to the power loads that are being controlled and has an uninterrupted serial port connection to the host server.

We are assuming the pricing structure to be what is shown in Table 1, showing times that are to be considered peak and off-peak. This structure is based on the Con Edison pricing scheme and acts as a general replacement for the wide variety of structures that utility companies are currently employing.

Table 1: Peak / Off-Peak Pricing Hours

Times	Monday - Friday	Saturday - Sunday
Peak	10:00AM – 10:00PM	None
Off-Peak	10:00PM – 10:00AM	All

3. Design

The Intelligent Load Management System involves a combination of software and hardware that provides an automated way of intelligently managing the power usage of a home. The system provides a proof of concept for this solution in the form of a control architecture that focuses on usability, flexibility and automation from the user's perspective. At the highest level our system acts as a load management controller that can intelligently react to changes in renewable and grid power availability, environment variables, power utility and user preferences, as well as act as a management and power analysis system for the user's home or area of control. Specifically this is accomplished through a web interface that controls the availability of power toward specific loads and analysis of both grid and renewable power resources, while maintaining an extensive backend architecture that provides comprehensive algorithmic automation options and real time display of current environment, system, utility and user variables.

3.1 Host Server

The server that hosts the key functionalities of this project is a Ruby on Rails model view controller architecture. Additional components that were integrated into this architecture, shown in Figure 2, include the analysis and dashboard view pages, daemonized process scripts and the serial interface that connects via serial USB cable to the microcontroller. Each of these components that have been added to the Rails architecture will be explained in more detail.

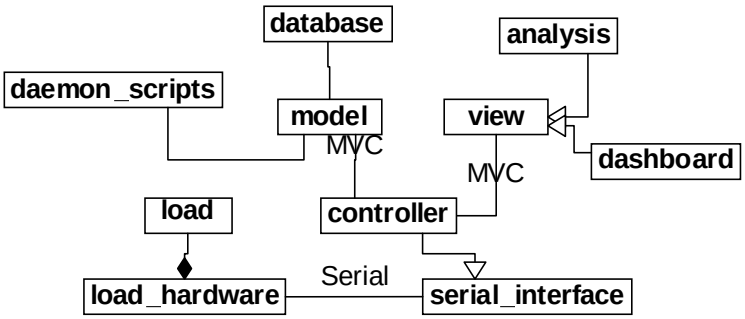


Figure 2: Server Architecture

3.2 Serial Interface

Essential to the functionality of this system, the serial interface provides the methodology to connect commands from the core Rails controllers via the microcontroller to affect the loads that are connected to the hardware which

simulates a home. Building on the open source Ruby gem SerialPort[9] which provides the basics for establishing serial connections, this interface is written to both read power consumption information from the microcontroller and send commands that will disconnect specified load links from the power grid.

3.3 View Pages

To accomplish our graphic user interface software requirement we designed two additional view pages that provide centralized locations for control and system analysis for the user. In Figure 3, the first view page is termed the dashboard and on the right presents information on the settings specified by the power utility, and the user's current settings. These user settings are selectable and can be changed from the dashboard via dropdown menus. On the left column of the dashboard is a table displaying the devices configured for the system and whether they are being supplied with power. The status of the outlets is checked every three seconds using an Ajax (asynchronous javascript) query that will reload this small portion of the page if needed to present the user with up-to-date information about their system, without needing continuous full page refreshs.

The second view page in Figure 3 is the Analysis page. This specially designed page contains a graph of the most recent power consumption readings from the microcontroller as well as the most recent readings from the green power simulation. Additionally an area graph overlays to display the summation of difference between these two values to present the user with a clear indication of the current usage of grid power. Again using an Ajax query, new information is appended to the graph to display to the user the current power readings. If the graph contains more than twenty entries it will scroll to the left and continue to display the most recent readings. This graph is built using the HighCharts[10] javascript engine where we have pooled asynchronous updating with multiple series of data.

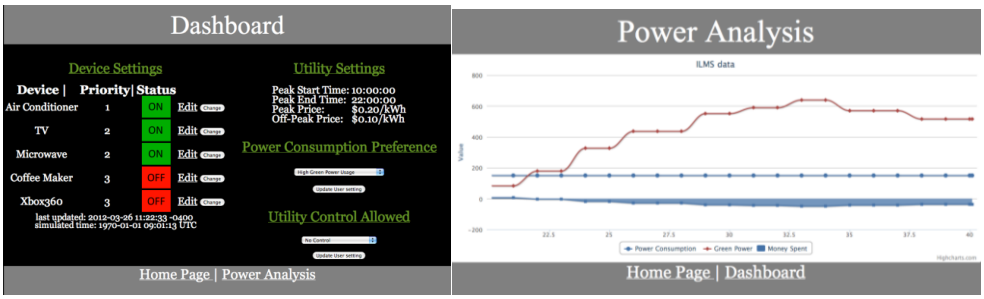


Figure 3: Dasboard and Analysis Pages

3.4 Daemon Process Scripts

There are two persistent tasks that must be continuously running in order to achieve our software requirements of power logging and autonomous management. Without the power logging and autonomous load processing that will continue to run while the user is disengaged from the website, this web application would be unable to provide functionality to users during those times. Our background processes use the Daemon Ruby gem[11] to aid in the generation of control scripts and make the Ruby on Rails environment variables available to the daemon processes.

Using the serial interface, the power logging daemon process reads the present power consumption and current being passed through the hardware. This process reads at set three second intervals and writes the information it receives directly to a power log table in the database. The information gathered by this script is then available for the graphs, logs and other utilities that the server might need the information for.

A second daemon process ensures that loads are persistently managed and that changes in power availability and requirements are responded to if changes are required in the system. This daemon runs at set three second intervals and combines data from many of the different tables in the database including device status, user and utility settings as well as time calculations to be passed to the load management script for processing and decision logic.

3.5 Hardware Architecture

The hardware is configured to connect to the host server through a USB cable that allows serial connections with the Arduino Duemilanove microcontroller. The Arduino reads the total present power and current flowing through the system and transmits that information over the USB cable. The Arduino also controls demand response by discrete control of loads (ON or OFF) as directed by the host server and represented by indicator lights. Figure 4 shows the overall hardware used to simulate a home. Among other components, this hardware includes the Arduino microcontroller and five state switches.

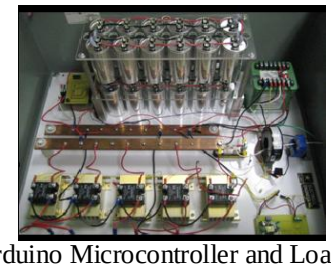


Figure 4: Arduino Microcontroller and Load Controllers

3.6 Load Management Script

As a key addition to the host server's supplemental design, the load management script could be considered the workhorse of the entire architecture. Given all of the accumulated user, utility, environment variables and parameters as well as recent power consumption history and state of the system, the load management script decides which loads need to be disconnected or reconnected to the power grid. This script is called by the load management daemon process which also passes all the necessary variables. Script functionalities include, among others, the ability for users to set preferences for dependency on available green power to supplement their load requirements, preferences for allowing the utility to make decisions for supplying certain devices power, and the ability to conduct a manual override that will force the management script to overlook the specified device for a given period of time.

3.7 Testing

The key functionality and structural additions made to the Ruby on Rails web application server were tested. During testing this project implemented some of the Rails autogenerated functional testing mechanisms to facilitate reproducible assessments, as well as unit tests that would ensure utility and model related functions retrieved variables and data correctly. Additionally, the system tested each available option in user and utility preferences to ensure the load management script was making the expected decisions in each mode. Testing was also conducted on the user override capabilities to ensure the load management process did not modify certain devices when otherwise it would have, in addition to the utility option to override certain devices if allowed.

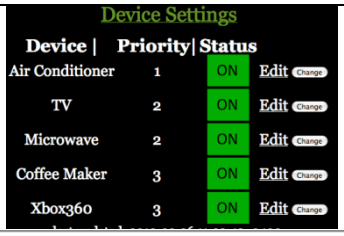
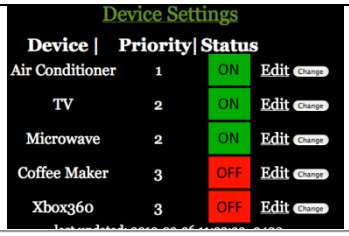
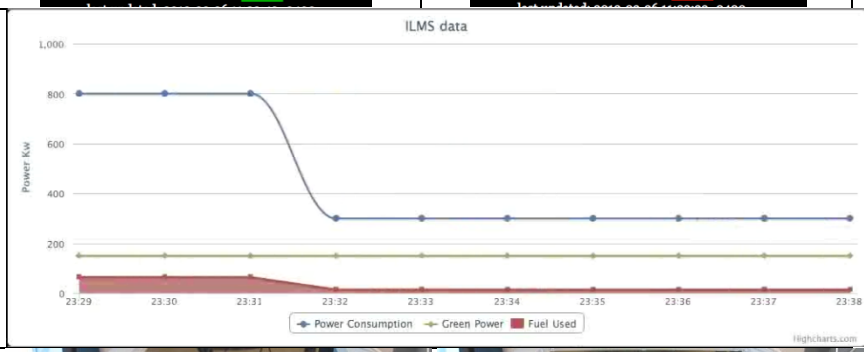
4. Applications and Discussion of Results

Each of the specified components in the design fit together to provide the functionality specified in the system requirements. Using a host server and the daemon processes, a user is able to run the system in an operational environment bringing in live power data from resistors and have the load management script continually process present variables to decide the current state of the attached load devices. As can be seen in Table 2, from the analysis page, it is possible to observe the current state of power in the system and observe how that changes in time. Simultaneously it is possible to observe the same results on a separate medium, the dashboard, as devices are disconnected or reconnected to the power grid. This updating happens autonomously though the Ajax and javascript present in both these view pages without requiring the user to reload the page. This adds convenience in addition to dependability as users can monitor the status of their systems with confidence that their current view is correct.

Table 2 shows the before and after states of the system from the hardware, dashboard, and analysis pages. The consumption preference in this example is set to turn off the lowest priority devices if the percentage of green power

available is not able to supplement more than sixty percent of the total load. In this example there is only 23% green power supplemented for the total load in the "before" capture shown. The power management scripts automatically turn off the two priority three devices based on these environment variables and settings, in this case the coffee maker and Xbox 360. This change is represented in the capture shown in both the hardware and automatically in the web interface. The analysis page also reflects this change in state of the system by illustrating the sudden decrease in power consumption that occurs as these devices are disconnected from the power grid.

Table 2: Demonstration of System

	Before	After
Dashboard Pages		
Analysis Page		
Hardware Simulating a Home		

Additionally, because this project is a web application, it is easily accessible from any network-connected media. During testing we used an Apple Ipad2[12] to supplement the web client running on the host server machine. This additional media provided the exact same functionalities as the web application running on the laptop. All clients could be viewing pages as well as interacting with the database and load hardware simultaneously.

5. Conclusions

This proof of concept prototype provides clear insight into the importance and potential usefulness of a commercial, user-based, control architecture that manages demand response and integration of renewable energy sources. Not only could it revolutionize the way that power is managed, it also streamlines the integration of renewable energy sources. This system would also bolster the usage of green energy and cut down on fossil fuel emissions. There is also an economic impact caused by the ILMS -- for the individual, they gain the ability to save money by managing energy more effectively during peak hours. Nationally, a lowered dependence on non-renewable energy sources would shift demand for nonrenewable resource imports.

6. Future Work

The biggest future task that would improve functionality would be the ability to measure individual loads within the simulated home as opposed to the measurement of overall net power consumption that is presently used. It is also important to have this system tested as a part of an actual home tied to both green energy and grid power. Another interesting potential for expansion in this project would be the introduction or significant use of smart appliances that are able to retain track of their task completion and time required to complete which would adjust the algorithm to accommodate an appliance's specific information. Future work in this area could also include an increase in algorithm functionality that allows for the prediction of possible future load requirements based on usage history.

7. References

[1] TheElectricCompanyReno.com. "An Electrician In Reno Explains Rolling Brownouts vs Rolling Blackouts". Accessed 26 March 2012. <http://electriccompanyreno.com/electrician-reno-explains-rolling-brownouts-vs-rolling-blackouts>

[2] FERC.gov. "Definition of Demand Response". Accessed 28 April 2012. <http://www.ferc.gov/industries/electric/indus-act/demand-response/dem-res-adv-metering.asp>

[3] ConEd.com. " Save with Time-Of-Use". Accessed 26 March 2012. <http://www.coned.com/customercentral/energyresvoluntary.asp>

[4] Vojdani, Ali. "Smart Integration", *IEEE Power & Energy Magazine,* Nov/Dec 2008

[5] UNEP.org. "Global Investments in Green Energy Up Nearly a Third to US$211 billion". Accessed 26 March 2012. <http://www.unep.org/newscentre/default.aspx?DocumentID=2647&ArticleID=8805>

[6] NewEnergyCities.org. "Forecast for Solar: Don't Like the Weather, Wait Five Minutes and It'll Change". Accessed 26 March 2012. <http://newenergycities.org/forecast-for-solar-don2019t-like-the-weather-wait-five-minutes-and-it2019ll-change>

[7] EnergyStar.gov. "Programmable Thermostats for Consumers". Accessed 26 March. <http://www.energystar.gov/index.cfm?fuseaction=find_a_product.showProductGroup&pgw_code=TH>

[8] Gulhar, Neel. Hamilon, Katherine. "Taking Demand Response to the Next Level". *IEEE power & EnergyMagazine*. May/June 2010

[9] RubyGems.org. "serialport 1.0.4". Accessed 26 March 2012. <http://rubygems.org/gems/serialport>

[10] Highcharts.com. "Highcharts JS". Accessed 26 March 2012. <http://www.highcharts.com/>

[11] RubyGems.org. "daemons 1.1.8". Accessed 26 March 2012. <http://rubygems.org/gems/daemons>

[12] Apple.com. "iPad". Accessed 26 March 2012. <http://www.apple.com/ipad/>