

# UC Berkeley

## Planning & Evaluation

### Title

Accelerated Preparation of Learn HVAC, a Cyber-Based Education Tool for Rapid Widespread Dissemination within 2-year and 4-year Educational Institutions and the Building Industry

### Permalink

<https://escholarship.org/uc/item/8b58b02c>

### Authors

Auslander, David M.  
Beckmeier, Nigel  
Rodriguez, Romina

### Publication Date

2010-10-15

# **Accelerated Preparation of Learn HVAC, a Cyber-Based Education Tool for Rapid Widespread Dissemination within 2-year and 4-year Educational Institutions and the Building Industry**

**University of California, Berkeley Participation in Learn-HVAC Project at Lawrence Berkeley National Lab (LBNL) - POB239-B58**

**David M. Auslander, Professor, Mechanical Engineering, U.C. Berkeley  
Nigel Beckmeier, Graduate Student, Mechanical Engineering, U.C. Berkeley  
Romina Rodriguez, Undergraduate Student, Mechanical Engineering, U.C. Berkeley**

**Final Report**

**October 15, 2010**

## **1 Introduction**

The Learn-HVAC project is an effort to use a large-scale building simulation as the base for an educational tool to teach a wide variety of people how heating, ventilating, and air conditioning systems work. These systems account for a substantial fraction of energy usage in this country and by many accounts are often not operated at efficiency levels that the systems are capable of (which varies tremendously depending on their age and design). HVAC systems are extremely complex internally so inefficient system operation can result from lack of knowledge or from malfunctions in system components that are not recognized.

Learn-HVAC aims specifically at the HVAC complexity issue in its educational approach. Because buildings themselves are not suitable educational tools, a large-scale computer simulation is used to give people the experiential experience needed to develop an in-depth understanding of HVAC system operation. The thesis of this effort is that using such a simulation as part of an HVAC curriculum can enable a level of understanding that is not possible with conventional curricula.

The previous LBNL effort focused on community college programs. The UC Berkeley participation in this phase of the project is primarily to extend the community college focus to the four-year undergraduate engineering program and to beginning graduate study.

The tasks associated with this work are:

Task 1. Refine Learn HVAC for Widespread Use

Task 2. Adapt Learn HVAC for Additional Learner Types

Task 3. Develop Conceptual Framework for More Comprehensive Cyber-Based Educational Software Tools

Task 4. Reporting Tasks

Our work on this project was driven largely by the software development efforts at LBNL. By the time the project actually started it had been decided to drop the Spark version of the Learn-HVAC software and rewrite it in Modelica. This effort was underway throughout the time of this project. Thus, there was no reason to work at all with the Spark software and the new Modelica software was not ready for any work that needed involvement in its internal structure. As a result, we focused our work on the curricular aspects of Learn-HVAC and on assisting with debugging of the new Modelica version. Because of the state of basic software development we were not able to hire a student for task #3 or make any progress on it.

Task #4, Reporting, is represented by this final report which covers all work done on the project.

## **2 Modelica: Acausal Representation of Dynamic Systems**

### **2.1 Dynamic Systems**

A physical system that is capable of storing energy is a “dynamic system.” A building under HVAC control certainly stores thermal energy, and also stores energy in the fluids used to transport thermal energy throughout the building. What being “dynamic” means is that the behavior of the building depends on the history of its inputs. Mathematically, such systems are represented by differential equations.

In developing simulations of such systems it is simplest if the energy storages are all independent. The classical spring-mass system from freshman physics is in this category – it stores potential and kinetic energy. The real world, however, is rarely so cooperative. In most interesting cases the energy storages are not independent. An example of that in HVAC systems is a tee in a duct system. At the tee the flows must all sum to zero (given a consistent sign convention for flow direction). In the mechanical world, a similar situation exists in a simple model of a lever (or teeter-totter), modeled as a massless, rigid rod, with a fulcrum in the middle somewhere, and point masses at each end. It appears to have two masses, and thus energy storage for both of them, but, in reality, the two masses are coupled by the rigid rod and are not independent.

Standard representation of dynamic systems with independent energy storages is called “causal” representation because inputs and outputs are clearly delineated. Once the complication of algebraic constraints are added, as with duct tees, levers, etc., unless the combined differential-algebraic equation (DAE) can be decoupled analytically the representation no longer clearly identifies inputs and outputs and is called an “acausal” representation.

### **2.2 Modelica for HVAC Systems**

Modelica is a language specification for representation of dynamic systems with constraints (see <http://www.modelica.org>). It allows the representation to remain in acausal form. It is thus well suited for modeling the HVAC-building model for Learn-HVAC. Modelica itself, however, is only a language specification not solution software. There are a number of solutions available build on the Modelica language. The Learn-HVAC project is currently using Dymola, a commercial implementation of Modelica solution software.

### **2.3 Open Source Modelica Solutions**

Part of Task #2 is to explore open source (or otherwise free) versions of Modelica solution software. The major open source version is Open Modelica (<http://www.openmodelica.org/>). Unfortunately, evalua-

tion of Open Modelica was found to be inadequate to the task of running the Learn-HVAC model. Another open version, JModelica (<http://www.jmodelica.org/>) is now available but has not been evaluated.

This is an important issue because the curricular needs for the university level use of Learn-HVAC will require, in some instances, access to the actual source code of the model. The Dymola version currently being used, as noted, is a commercial implementation and is not well suited for four-year engineering academic use (it does provide for unlimited use of already compiled models which works well for environments such as community colleges that do not need to drill down into the source code). In particular:

It is very expensive (even at “academic” pricing).

Its use is very restrictive. It is designed for the kind of server-based usage common in industry. Used in the more open a fluid academic environment, for example, a single license cannot be used by an individual on both a desktop and a portable computer.

The student version, while relatively low cost compared to the “full” version (\$25/student) is too restrictive to be able to solve the Learn-HVAC model.

Thus, finding an alternative to Dymola, or convincing the Dymola organization to issue a version that is more suitable to the academic environment, is critical to meeting the needs for different kinds of experiences typical in a four-year engineering curriculum. The open source versions are improving all the time and it is possible one of them will meet our needs in the future.

### **3 Engineering Curricular Recommendations**

#### **3.1 A Different Kind of Educational Tool**

This section introduces research targeting the inclusion of the Learn HVAC software into the curriculum at UC Berkeley. A number of potential courses were selected based on subject relevance and professor interviews, including: E10, ME40, ME132, ME142/ME145 and ARCH140 (‘E’ denotes engineering courses, ‘ME’ denotes mechanical engineering courses and ‘ARCH’ denotes architecture courses). Lastly, a new elective course on HVAC systems based completely around the Learn HVAC software will be described as an alternative to integrating it into an existing course.

Use of a simulation tool is a departure from the traditional means of teaching most engineering courses. The simulation can be used in many ways which is why it is a particularly good tool to expand interest for students with different learning styles. The simulation can be used as-is, much as it is in the community college environment to observe and control the behavior of a building, explore possible faults, do energy audits, etc. This hands-on learning supplements the basic engineering science subject matter and puts it into an application context. When looking in more depth into specific subject areas, examination and verification of the source code for various components shows how the engineering science material can be integrated into a large-scale model, how it interacts with surrounding components, etc. At a still deeper level, models can be created for components not yet in the simulation, or models that already exist can be improved.

The material below shows all these kinds of uses of Learn-HVAC in the context of various existing courses and one new course.

## **3.2 E10 - Engineering Design and Analysis**

### **3.2.1 Course Overview**

E10 starts with 3 weeks of introduction followed by two six-week modules that include both lecture and laboratory time. This is an introductory course intended for first year students. The primary goal of these modules is to enhance design and analysis skills, both fundamentals in the engineering profession. Secondary goals include successful teamwork, effective communication and sparking an interest in selected fields early on in a student's collegiate career. Overall, the course is intended to give students early on a better glimpse into the nature of the engineering world.

### **3.2.2 Interview with Professor George Johnson**

This is a write-up for an interview conducted with Professor George Johnson concerning the inclusion of the *Learn HVAC* software in the E10 course. Professor Johnson is teaching the second module of E10 this semester (Spring 2010) on wind turbines. The first segment involves teaching the rudimentary knowledge needed to understand how wind turbines work, simple engineering formulas (e.g. Bernoulli's equation for fluid flow) and general approaches to design and analysis. The students are then taught how to design turbines with various physical attributes, and then they physically get to make a model in the rapid-prototyping machine. Finally, the students learn how to test their turbines in a wind tunnel and how to make meaning of their results. During the interview, Professor Johnson was introduced to the basics of the *Learn HVAC* software (version "lh\_build\_98\_3") and was allowed to briefly play around with the simulation features.

Johnson brought up a number of recommendations including adding: (1) a cost factor, (2) load meters to both the system and individual appliances, the (3) option to model your own system and (4) the ability to simulate faster than real-time. Johnson stressed the importance of being able to find the cost, or dollar value, associated with the HVAC parameters. For example, he wanted the software to be able to yield the cost difference in heating a building to 80°F versus 75°F. He would also like to be able to add various heat sources to a room (let's say a refrigerator) and see how that affects the system, namely in cost. Further, adding the option to upgrade various appliances and forcing the user to decide whether the energy saved would outweigh the initial cost and how long it would take to break even. Load meters would give a better sense of which appliances consume the most energy and on a grander scale what the energy bill looks like for a large-scale building. Johnson also liked the idea that a user could model his own house or apartment in order to make more educated choice with respect to load management. This recommendation seems to be more closely related to what he needed for inclusion in E10. Lastly, he would like to be able to run the simulation and more importantly obtain results in a faster than real-time mode. He could imagine asking the students to find the total energy consumption on a building for a given day, but admitted that it would be unreasonable for the student to have to run the model for 24 hours to obtain such a result. Overall, Professor Johnson was enthused by the software but hesitant to include it in a module for E10 as of now.

### **3.2.3 Proposal**

For inclusion in course like E10, we would propose using the *Learn HVAC* software in tandem with an actual building on campus. I believe that the new CITRIS building on north side is a perfect candidate, and we would recommend that the software tries to model this particular "scenario" [Figure 1]. The CITRIS building was outfitted with an ultra modern HVAC system that is run from a Siemens building operating system that is state-of-the-art. The unit would start by having the building operators to give a

talk on how their Siemens system works, and then take the students on a tour showing all of the system's major components. The students could then get a feel for what the various components actually looked like, relative to size and shape, and the general layout of the system. Students would also be able to take basic measurements (like vent cross-sectional area, air flow velocity, etc.) that they could then further analyze later. The Learn HVAC software would be used as a tool of comparison to the measurements that were taken in order to give meaning to seemingly random values. Ideally, the "CITRIS building scenario" could be loaded up and the student would input values such as the inside/outside temperatures and room thermostat readings. Then, run the system and output some sort of graph to verify their results. This graph of simulation data should closely resemble the empirical data gained from physical experimentation, given some margin for error. Although Johnson admitted that the software in its current state might not be a good fit for E10, I believe that a module based on the Learn HVAC software and the CITRIS building twosome would integrate nicely into the course.



Figure 1. The new CITRIS building the north-side UC Berkeley campus.

### 3.3 ME40 – Thermodynamics

#### 3.3.1 Course Overview

The class is an introductory Thermodynamics course intended for Mechanical Engineering majors in their second year. Students are introduced to fundamentals of energy storage, thermo-physical properties of liquids and gases and basic principles of thermodynamics. These fundamentals are applied to different engineering areas through assignments focusing on topics like energy conversion and air conditioning to strengthen students' knowledge of thermodynamics. The class usually consists of three major quizzes and a final at the end of the course.

#### 3.3.2 Interview with Professor Fernandez-Pello

During the interview, Professor Fernandez-Pello was shown a simulation and viewed the system's parameters that can be altered by the user. He was also shown all the components of the HVAC system, as seen in the Learn HVAC software, such as the cooling towers and VAV box. Besides running real-time

simulations, the Professor also learned about the “faults” that can be implemented into the program and how students can work in groups to figure out how to correct these faults.

Although the Professor thought the software was useful and that students could learn a lot from it, he quickly pointed out a problem with the HVAC system configuration shown in the simulation. When running a simulation, the user can observe in detail, the different system components such as the cooling towers or the mixing box by switching views using the pull-down menu on the main screen. Although the “system” view shows all the components in the building’s air ducts, and the “plant” view shows how all the components in the building are interconnected, Professor Fernandez-Pello believed that this would not suffice in helping undergraduates in an introductory thermodynamics class understand how all the components worked together. He believed that there had to be diagrams in the simulation that would help students jump from the thermodynamic theory to the actual engineering application of the HVAC system in a building.

### **3.3.3 Proposal**

While the HVAC system in a building may be too complex for students in an introductory thermodynamics class to fully understand, by making some changes as suggested by Professor Fernandez-Pello, the program can adapt for less experienced-users. These changes consist of implementing three different stages to the Learn HVAC software that will help students build their way up to fully understanding an HVAC system in a building.

#### 1. Theory behind Thermodynamics

The Learn HVAC software should have diagrams such as the one shown in Figure 2 of the refrigeration cycle that helps students tie in the theory they learn in class to the real-world applications shown in the program. Before even being able to run simulations on the software, they should be able to view air-cycle diagrams to better understand what thermodynamic concepts are being implemented when designing an HVAC system. The most important aspect of the ME 40 class is for students to walk away with basic thermodynamic principles, and this can be achieved by students reviewing basic concepts before trying to use them to solve problems.

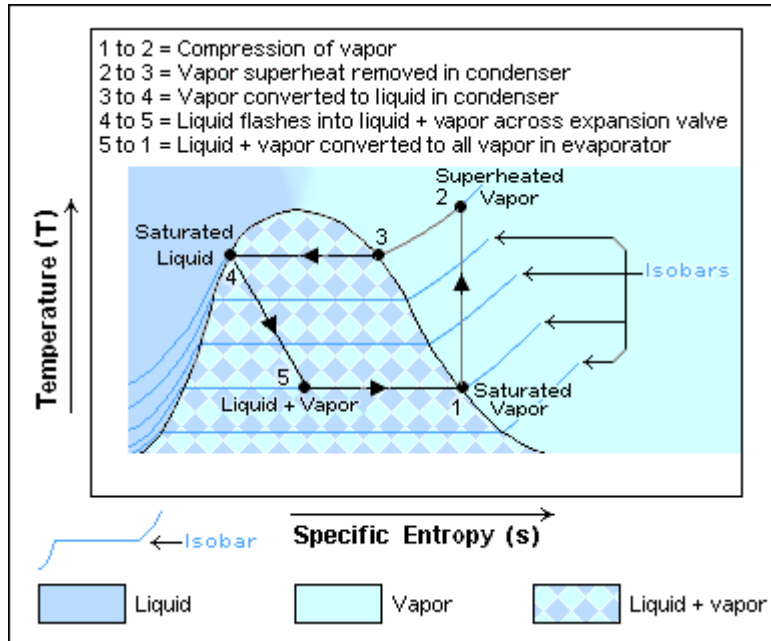


Figure 2. Temperature versus entropy for the refrigeration cycle

## 2. Concept Implementation

Once the student has been able to review the theory, the next step is to use their newly found concepts to solve a problem. For example, the question they would be asked in class would be to use their knowledge of the refrigeration cycle to make a system that would take in hot air and cool it before returning it to its environment. The schematic of the refrigeration cycle, such as the one shown in Figure 3, should be implemented into the HVAC software so that students have a step by step picture of how a refrigeration system can be made.

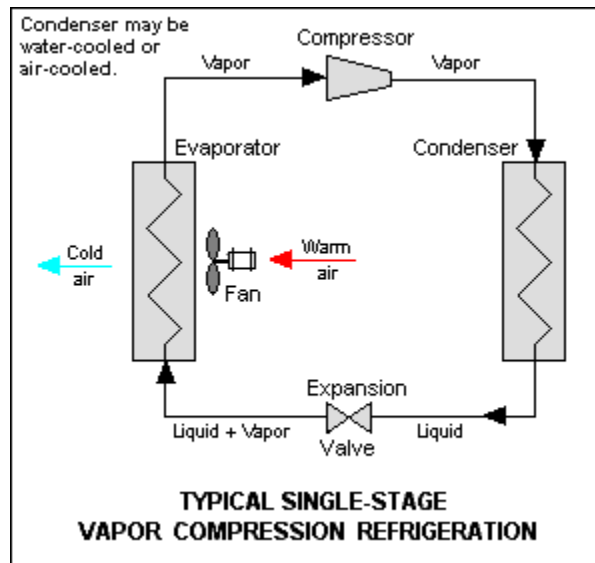
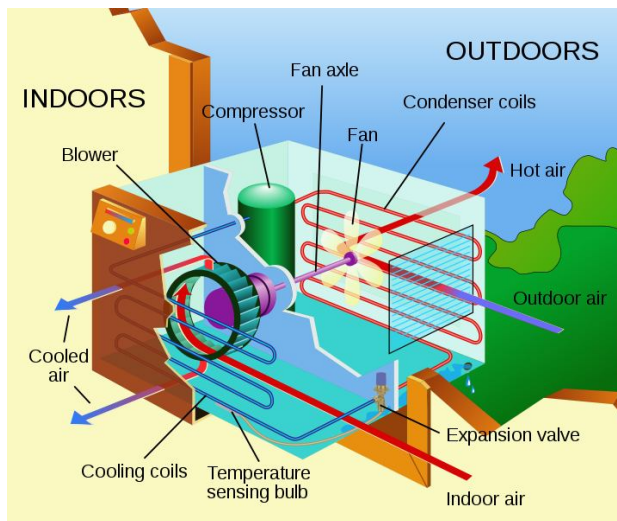


Figure 3. Schematic of refrigeration cycle implementation



### 3. Engineering Examples

Once the student has been able to learn the concepts and understands how to implement these concepts to solve engineering problems, the final stage in the Learn HVAC software should be real-world examples. Since the HVAC system of a building is too complex, it has been suggested that students start with running simulations on a typical window air-conditioning unit such as the one shown in Figure 4. By starting with a simpler system, it will be easier for students to understand how to use thermodynamic principles to analyze a system and how to analyze more complex systems.



**Figure 4. Real engineering example of window air-conditioning unit.**

The final step would be for students to tackle the intricate HVAC building example and run simulations, analyzing for example how to reduce a building's energy usage by looking at the mass flow rate of the air governed by the fan.

If the Learn HVAC would implement these changes, then the class ME 40 would use the software throughout the entire course by consistently referring to it for extra examples and for assignments. The final project for the class would consist of analyzing and understanding a simulation for the building HVAC system.

## **3.4 ME132 – Dynamic Systems and Feedback**

### **3.4.1 Course Overview**

This course focuses on the physical understanding of dynamics and feedback. Students are taught linear feedback control of dynamic systems and how to use mathematical tools for analysis and design of control systems. Modeling using differential equations is heavily enforced and solutions to linear, time-invariant differential equations are taught.

### **3.4.2 Interview with Professor Francesco Borrelli**

During the interview Professor Borrelli was shown how to use the Learn HVAC software. He was shown how to input parameters and how the simulation changes as these inputs are updated. Besides running

simulations and looking at the various model zones, he was also informed about the faults feature which professors are allowed to control.

Although Professor Borrelli was impressed with the program and thought it would be a good fit in the controls courses he teaches, he thought there was room for improvement. Specifically, he thought there should be more information on the description of each input parameter since most mechanical engineering students wouldn't be very familiar with HVAC systems in a building. Also, he stressed that this would be a good tool for students to learn the physical meaning behind control theory, but the program could not be focus of the class since various control systems have to be taught as a prerequisite.

### 3.4.3 Proposal

While the Learn HVAC program could be useful in helping engineering students better understand control theory and its applications, the program needs to provide more information to the user as to what each component in the program physically stands for and how it's modeled. This proposal recommends: (1) implementing pop-up based education system on control theory, (2) increasing step-response diagrams of various control inputs and (3) adding a Learn HVAC software user manual.

#### 1. Control Theory

The Learn HVAC software has a lot of input parameters that allow the user to change the controller gain of a component in the HVAC system, for example the VAV damper control gain. However, the program does not specify what each of these gains means, for example in the heating coil system there is an input called: *Heating coil valve controller P gain* but it does not provide any more information. For someone who knows a lot about controls, this means that the *Proportional* controller gain constant can be changed to have a faster or slower response time from the damper. On the other hand, many of the students who will use this program will not know a lot about HVAC systems, so whenever they drag their cursor over the controller gains or the controller integrator time constants, a detailed description of each should pop up. Also the differential equations that govern these control systems should pop up so that students can understand what is physically going on. Mechanical engineering students who are first learning controls concepts may have a problem relating the control theory they learn in class to real world examples, so the Learn HVAC software should facilitate this process. This is why diagrams such as the one shown in Figure 5, which shows a common closed-loop feedback schematic, should be included in the diagram. Students will begin to understand what each component in a building's HVAC system refers to in control concepts, such as the thermostat being modeled as a sensor.

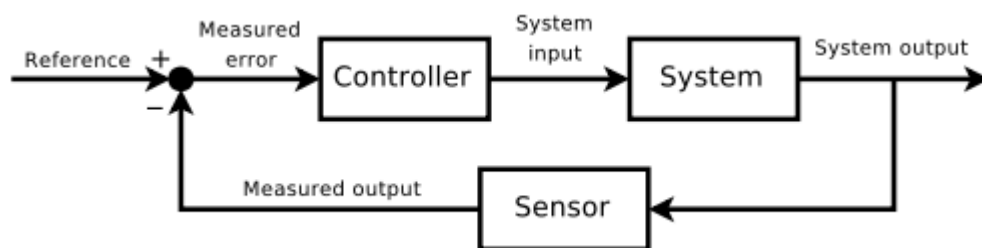


Figure 5. Typical controls closed-loop feedback block diagram.

## 2. Time Profile of Control Input

Besides learning the differential equations that govern control systems and how to read block diagram such as the one shown in Figure 5, students also need to understand how to control the outputs of a system. This is why it is important for the Learn HVAC software to include Time Profiles of Control Inputs, or the step-response of the signal input. For each control component in the HVAC program such as the VAV box, a step-response diagram should be outputted so that students can understand how input signals change over time due to the governance of the control system (Figure 6). As the user changes the controller gains of each component, a new plot will be generated and the user will better be able to understand what happens to the control input as each parameter is changed. For example, if the temperature of a system is an input, then over time it can be observed how long it takes for the temperature to converge to a specified temperature as the controller gains are varied or if the controller integrator time constant is altered. It was also suggested that the time response of the different inputs be able to be exported into Matlab or Excel since these are what students at Berkeley mostly use and they will better be able to analyze the data.

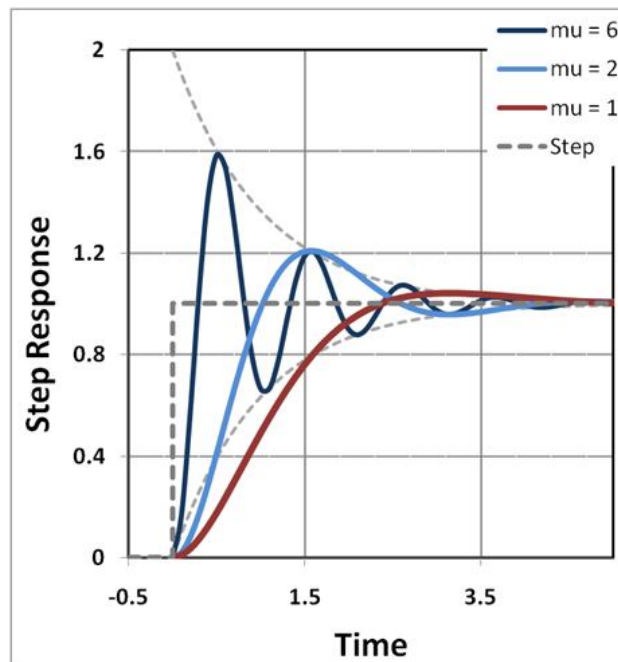


Figure 6. A common plot of the step-response of a signal over time.

## 3. Software Manual

Instructors are worried about how long it would take students to learn a new program since semesters are relatively short. Considering the Learn HVAC software would not be the entire focus of a controls class, and only a tool to provide students with more insight into control systems, a short manual that comes with the program is needed. This manual needs to have an explanation on how to run simulations and the types of analysis users can perform using the program. It should also include the theory behind temperature control because that is a primary function of the HVAC system

and provide example problems so that students can become familiarized with the program a lot quicker.

### **3.5 ME142/ME145 – Thermal Environmental Control/ Computer-Aided Thermal Design**

#### **3.5.1 Course Overviews**

Both of these are upper division Mechanical Engineering courses taught by Professor Van Carey on advanced topics in Thermodynamics.

*ME142 Thermal Environmental Control:* This course focuses on analyzing thermodynamic and transport phenomena associated with a wide variety of thermal control applications. Particularly, emphasis on the application of theoretical concepts to the design of advanced thermal control systems is stressed. Students are taught topics covering vapor compression heat pumps, gas compression cycles, cooling towers, thermoelectric cooling, heat exchangers, heat pipes and advanced insulation concepts. Also, thermal control of buildings, automobiles, spacecraft and electronics will also be discussed.

*ME145 Computer-Aided Thermal Design:* This course introduces mechanical engineering students to the use of modern computation tools to model thermal performance characteristics of components and systems. Particularly, the course focuses on the synthesis of thermoscience theory and advanced computational methods to better understand engineering design. The course heavily relies on the use of computer graphics tools to communicate design issues. By the end of the course, all students must complete two applications-related design projects using theoretical thermoscience concepts along with advanced computational methods developed in class.

#### **3.5.2 Interview with Professor Van Carey**

Although Professor Carey was originally interviewed to discuss how the program could be implemented into an introductory thermodynamics class, after viewing the program he thought it would be a better fit in an upper division mechanical engineering course. He believed that the software would be too advanced for students who are trying to grasp introductory thermodynamic concepts. However, since he teaches two advanced thermal design courses, which are described above, he made suggestions on how the Learn HVAC software could be improved to be used in these courses.

Since ME 145 is a class that focuses on using computational tools to design thermal systems and ME 142 focuses on thermal control applications, Professor Carey has been presented with other software like Learn HVAC that could be used by the students in his class. However what makes him apprehensive about using various modeling software is twofold: (1) it will take a lot of time for students to become comfortable with the new software and (2) usually such programs don't allow the users to know the idealizations used to create the software. It is very important for students to know the limitations of the program before they are given an assignment intended to better understand the concepts behind thermal design.

#### **3.5.3 Proposal**

While the Learn HVAC software can be a powerful teaching tool for students to learn how to synthesize computational tools with thermoscience theory, it could greatly improve if more information was given on how to use the software to students who are learning how to model thermal systems.

1. Facilitate Student Learning Through Website Tutorials

Although there is currently a Learn HVAC website that allows both administrators and students to learn more about the program, there is not much information on the website. As can be seen below in Figure 7, there isn't even a tab for a tutorial for students to better understand how to use the program. Instructors have a lot of material to teach students throughout a short semester, so they worry about how long it would take to teach students a new program. For example, entire classes are structured around teaching students software like Solidworks or Matlab. However, if the Learn HVAC website included easy-to-follow tutorials, students would be able to learn how to use the program much faster. Specifically, the tutorial should provide step-by-step guidelines as to how to use the software to solve an example problem, which the students can refer back to repeatedly to speed up the learning curve. It was also suggested that the website include a theory section, which goes over thermodynamic concepts so that students can review the material at their own pace.



**Figure 7. Current Learn HVAC website still under development**

## 2. Greater Emphasis on How the system is Modeled

Before students are given assignments that require the use of modeling software, it is very important for them to understand the idealizations the program uses for its analysis. Most modeling programs, do not allow the user to view how the software was built, but since Learn HVAC will be open-source software, it was suggested that more detail be given to the user on the analysis used to model the depicted HVAC system. For example: How was each room in the building analyzed? Was each room used as a control-volume or was a nodal analysis used to compute the temperature variations throughout the building? These idealizations in conjunction with the limitations of the program need to be understood by students before they use the program in order to better understand the simulation results. It was suggested that parts of the code, used to create the Learn HVAC software, should be allowed to be viewed so that students get a more a profound understanding of how thermal system components are modeled.

### 3. Illustrate thermal component interactions

To improve the graphic user-interface, diagrams should be added so that when the user moves their cursor over a certain component of the HVAC system, a diagram of the how components interact pops up. For example, if the user moves their cursor over the heating coils when running a simulation, instead of just seeing the temperature of the air, a diagram depicting the direction of heat coming in or leaving the components such as the wall and coils should be shown.

## **3.6 ARCH140 – Energy and Environment**

### **3.6.1 Course Overview**

This course focuses on an introduction to issues of physical building performance, which includes building thermodynamics, day-lighting and solar control. The fundamentals of building science are emphasized, while keeping in mind the evolving nature of building technologies, energy efficiency, ecology, and responsible design. In the beginning of the course, students are taught thermal properties of materials, heat transfer through building assemblies, balance point temperature, solar geometry, and shading analysis. By the end of the course students are expected to apply these principles in a design project. The latter part of the course also provides a survey of broader building science topics including mechanical system design, microclimate, and current developments in energy-efficient design.

### **3.6.2 Interview with Professor Ed Arens**

During the interview, Professor Arens provided information as to how the Learn HVAC software would be helpful to architecture students. Learn HVAC would be a program that architecture students would use in a straightforward manner; meaning that they would only be interested in the output model and would not perform any sort of engineering analysis.

Although Professor Arens thought the Learn HVAC program would be a good tool for architecture students to better understand thermal aspects of building design, he did have some suggestions as to how to improve the program. He thought that the software should include solar loading on the building, especially how radiation through the windows would have an effect on the overall temperature of the building. Arens said that students in Arch 140 currently use another program (SHADE) for building thermal analysis, which deals with orientation, windows, lights, and insulation, in how they affect the total heat balance in a simplified building zone. He thought that these features should also be implemented into the Learn HVAC program. He suggested that there should be diagram representing the comfort zone, including a figure to show how increased levels of the air flow in the building changes comfort requirements. Lastly he thought that energy analysis would be extremely important to architects, and that the output energy might also be linked to a simple schedule of energy costs.

### **3.6.3 Proposal**

Architecture students need to have the ability to design their own buildings and test that design's particular energy efficiency. A tailored version of the Learn HVAC software could allow architecture students to pick from a number of different building designs, not just the typical one-room view. In addition, the student should be able to add heat sources into these building designs to test how they affect the system. These heat sources should include at minimum typical office equipment: computers, print-

ers, office machines, refrigerators, coffee-makers, and various types of lighting. Students could use this aspect of the program to gain a sense of how such heat flows compare to those of the system.

Adding the cost component to Learn HVAC will show the student economic value in their designs, and teach them about cost bracketing in real-world utility companies. Designing for comfort is an important subject in architecture, and the software needs to be structured to allow experimentation in this field. One common experiment would be to test various levels of room temperature in a given season (probably the extremes summer or winter) and the cost factor associated with it. The students will then be able to see whether decreasing the temperature by five degrees, let's say, to increase comfort makes sense from an economic point of view. This cost should be able to be broken down into its components, and more specifically should highlight the fact that energy costs more for different levels of consumption. Energy prices for these various levels of power should be constant in the software, and based upon typical utility prices. By the end of this course, students should be able to tell the difference between designs that are slightly more energy efficient but not nearly as cost efficient and those that truly make sense both economically and from an energy standpoint.

### **3.7 New Elective HVAC Course**

#### **3.7.1 Course Overview**

This would be an upper-division undergraduate engineering elective, most likely in the field of Mechanical Engineering. The focus of the course would be HVAC systems, specifically using advanced knowledge to increase design efficiency. The course would be split into two main sections: (1) focusing on the theory, components and systems overview of HVAC systems and (2) advancing knowledge of the object-oriented, equations based programming language, Modelica. The latter is a modern computer language that is highly proficient at modeling complex systems, specifically liquid/gas fluid flows, mechanical machines and temperature gradients. Prerequisites for the course would be: Thermodynamics, Heat Transfer and Controls. Students would also be encouraged to have a design class thrown into the mix.

#### **3.7.2 Proposal**

The first section of this course would be an introduction to the HVAC system and the basic theory that is necessary to understand the major pieces. The instructor would start by presenting each of the major components: the Mixing Box, Filters, Heating/Cooling Coils, Fan and VAV Box (including Reheat Coil). The Learn HVAC software will be used to illustrate how each of these components works together as a system. Remember that students will be able to use the software in class to zoom into each component to facilitate the learning process. Like in the E10 proposal, we believe this course should be taught in conjunction with a physical building, like Sutardja Dai Hall (CITRIS Building). Allowing students to experience what an actual HVAC system looks like, ie physical shapes and sizes, in the real world sticks with them longer than just learning out of a book. The course would refresh basic fundamentals like thermodynamic principles and heat transfer properties, and show how they are used in real-world applications. Also, new theory like psychrometry, the study of the properties of gas-vapor mixtures, and load calculation will be touched upon. Once the underlying HVAC knowledge has been laid, the course will shift to the modeling phase in Modelica.

The latter half of this course will center on the programming language Modelica and why and how it is used to model the Learn HVAC software. An air flow modeling problem like this involves a series of differential equations to simulate both temperature and energy throughout the system as well as a number of constraints defined by various components. MATLAB functions like ODE45 can be used to solve

these series of differential equations, but when you have various constraints as well these have to be dealt with manually. Modelica is ideal for this type of modeling because it is very efficient at dealing with both differential equations and constraints at the same time. This saves the programmer time and reduces the probability of error, which is the leading reason that this language is becoming increasingly more popular in a number of energy applications. This course will attempt to teach Modelica through a series of modeling exercises pertaining to smaller HVAC components. The final project would be to combine the modeled components from the exercises into a single simplified HVAC system, similar to that of the Learn HVAC software. The goal of this segment is to both understand the methodology used by developers to create Learn HVAC and to more generally apply the process to creating more complex systems.

This course should be intended for students who plan on or desire the knowledge to be able to design efficient HVAC systems of the future. By today's standards, most HVAC courses are taught to technicians who want the means to work on such applications out in the field. However, the University of California, Berkeley is not training technicians but rather future systems designers who will apply their advanced knowledge to improve upon the status quo. Thus, the instructor should intertwine various advanced HVAC topics into the curriculum at his discretion. These topics might include: equipment management, regulations (Title 24), energy management, refrigeration concepts, indoor air quality/health concerns, special cases in electronic and biotech manufacturing or LEEDs certification. The major advantages of creating a stand-alone HVAC course would be the greater emphasis on a somewhat neglected design field, but the disadvantages would be that less than 20% of the engineering student body might actually take the class because of its status as an elective.

## **4 Learn HVAC Model Recommendations**

### **4.1 User Operated Interface**

This is a recommendation to integrate a user-operated interface into the Learn HVAC simulation software. This proposal stems from an analysis of the Siemen's Insight® Advanced Workstation, the interface that is used to control the HVAC system in the CITRIS building on UC Berkeley campus. The Insight® Advanced Workstation offers a number of advantages from direct control over individual system components to the easy management of hundreds of sensors that maintain optimal efficiency. Navigating through the system is straightforward, and the operator can electronically monitor system components, even in individual rooms. Figure 8 depicts the third floor layout of the CITRIS building, and individual room temperatures can clearly be monitored and altered if necessary.



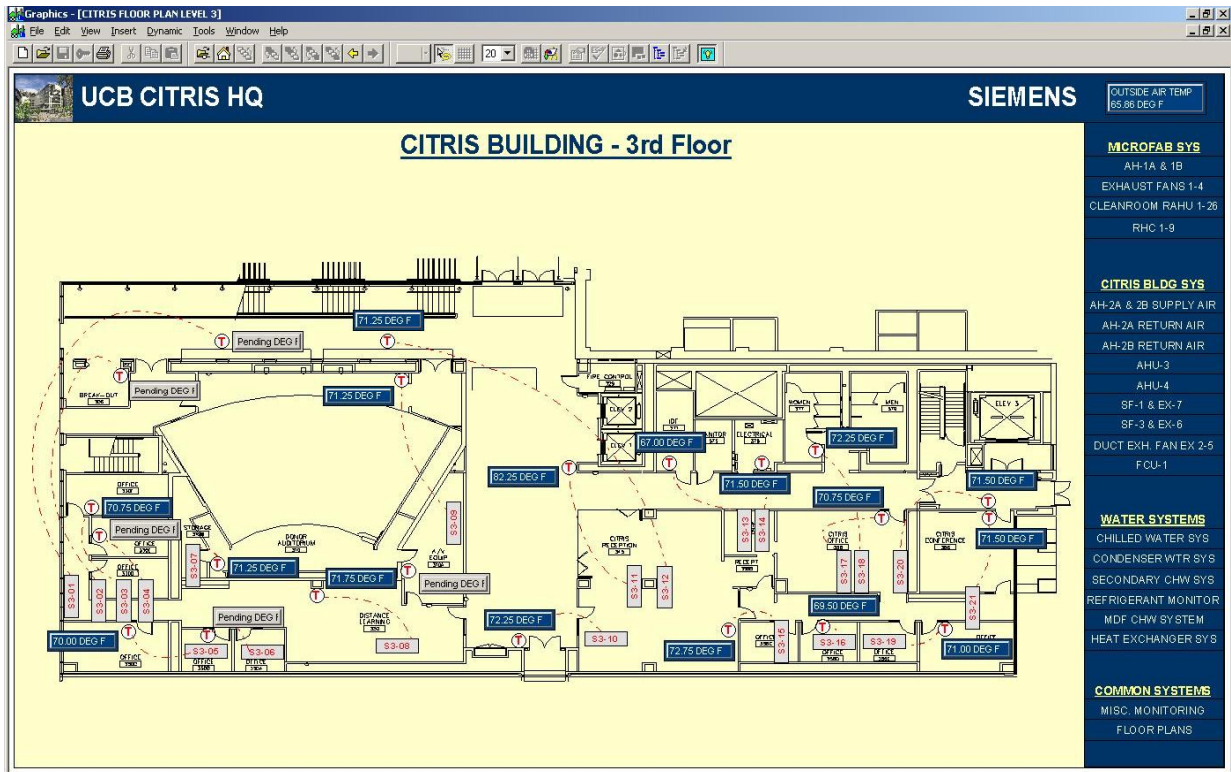


Figure 8. Third floor layout of the CITRIS building, including individual room temperatures.

The Siemens operator interface is useful for looking at sub-systems like the heat exchanger system, chilled water system, or the air handling units. Figure 9 illustrates a schematic of the heat exchanger system; notice that the green boxes ensure that those particular components are turned on and operating correctly.

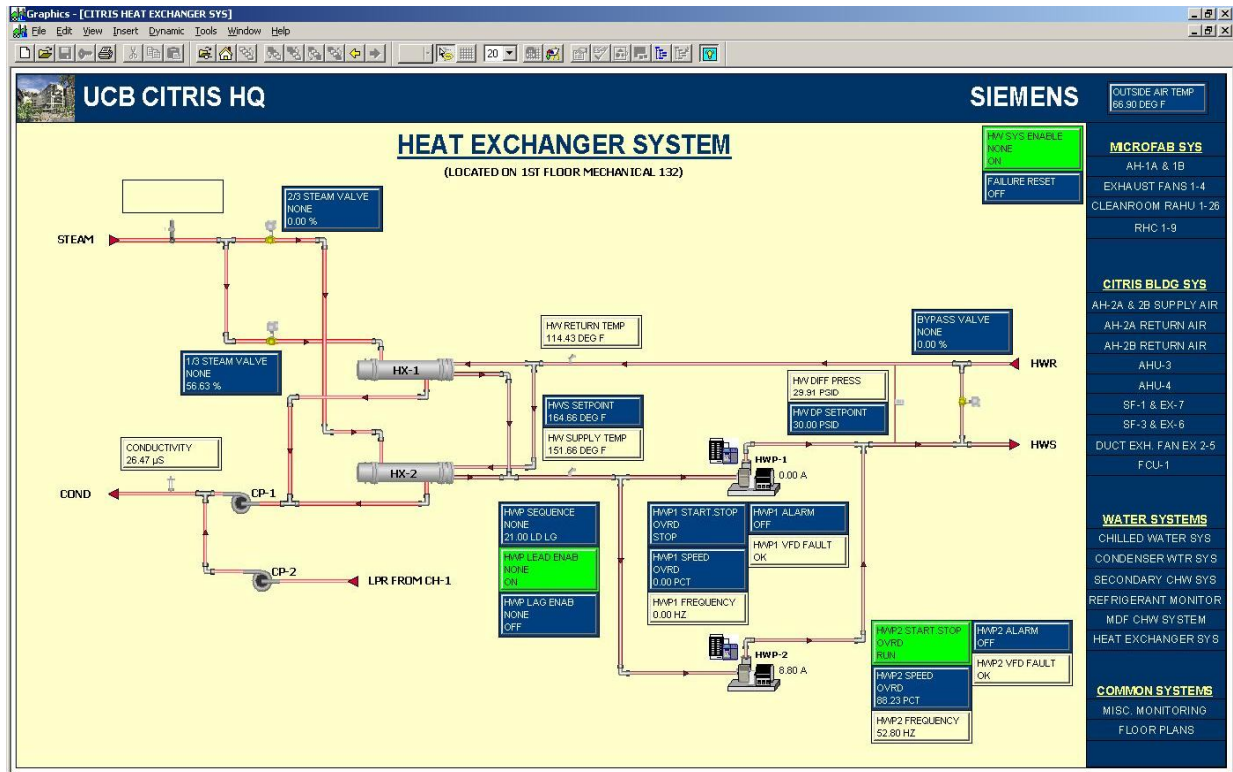


Figure 9. View of the Heat Exchanger System and all of its components in the CITRIS building.

In a building as large as CITRIS, routine maintenance occurs on a weekly basis. Building operators can quickly shut down the particular section that needs work without ever leaving the computer. This same system monitors all sensors in the building, provides real-time alerts based on malfunctions or inefficiencies, and prioritizes based on level of severity. Operators can also set measures to collect and analyze trend information to monitor performance over longer periods of time. In Figure 10, while viewing the sub-system “Chilled Water System,” an alert in the form of a yellow box has popped up on the screen. Alerts like these direct the attention of the building operator to particular system components that warrant a closer look. These alerts can range from minor system checks, like replacing a filter, to more serious matters such as faults or system failures. As it turns out, the alert in Figure 10 is only a minor signal indicating that this particular component needs to be reset. These schematics are also useful for dealing with components that are situated outside of the building, such as the Main Air Handling units located on the roof (Figure 11).

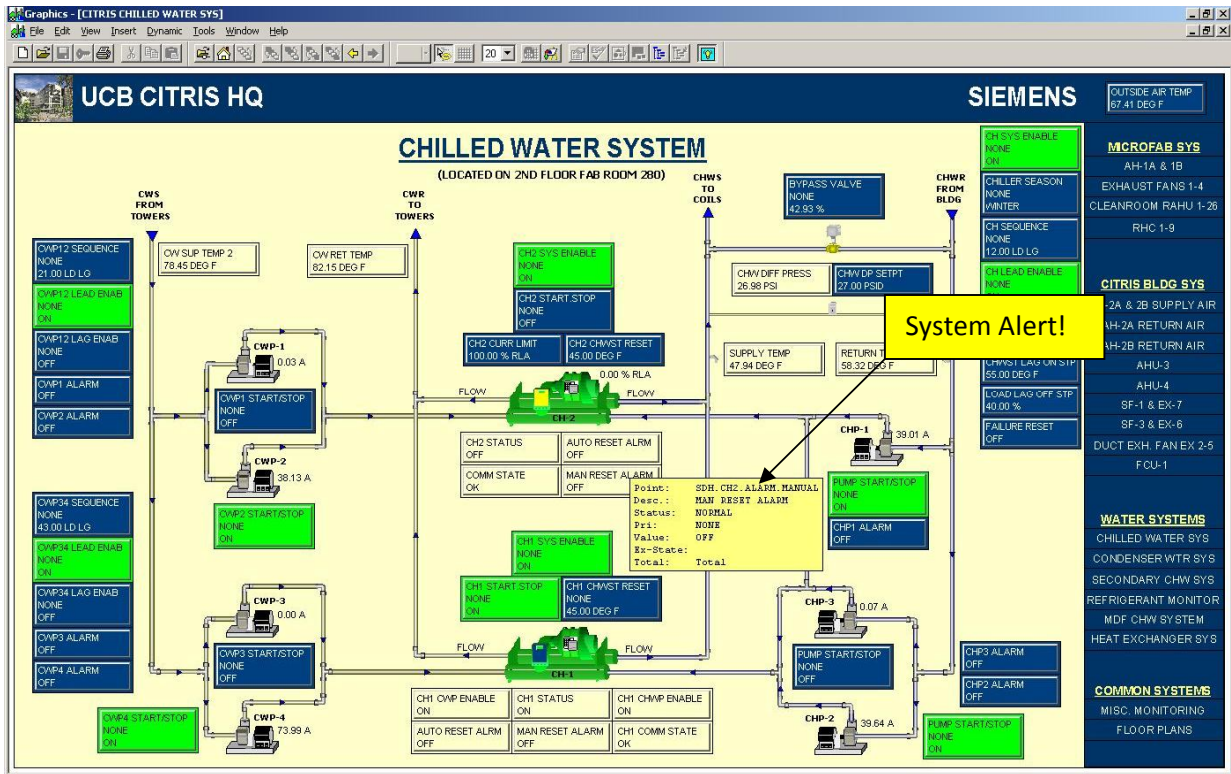


Figure 10. : View of the Chilled Water System. Note that an alert is warranting further attention.

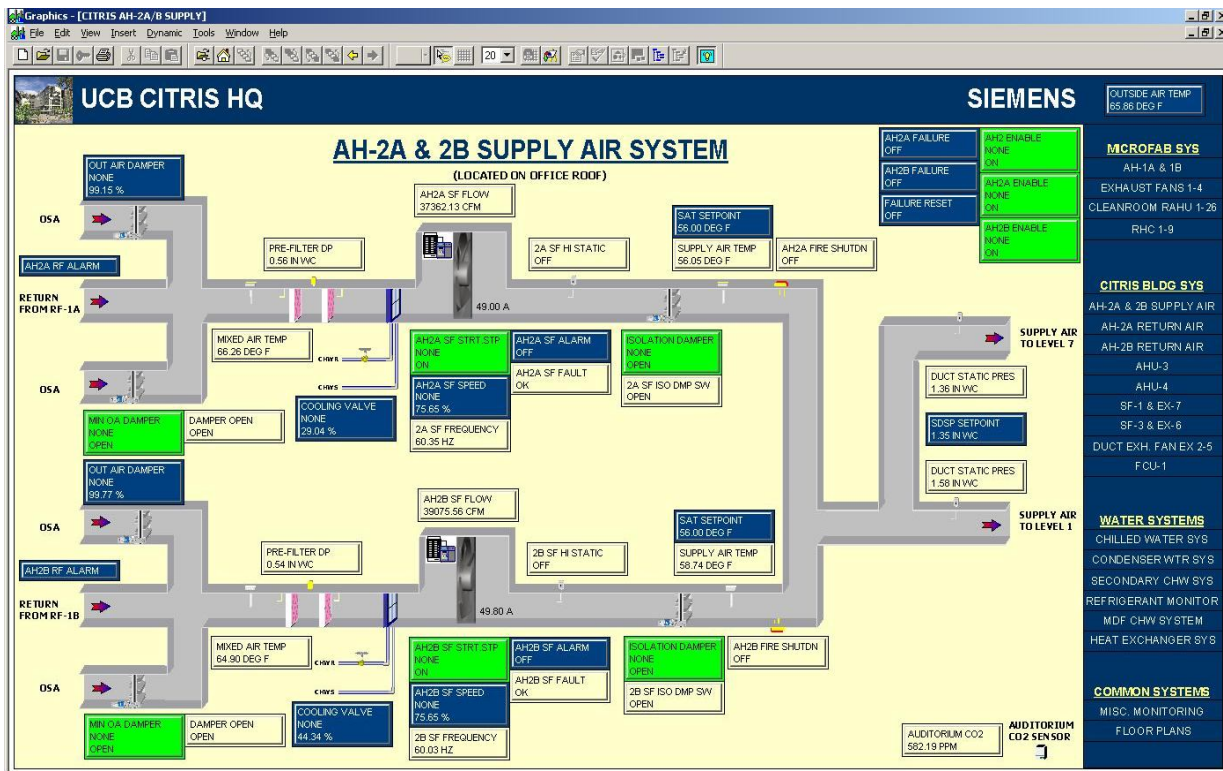


Figure 11. View of the Main Air Handlers located on the roof of the CITRIS building.

In general, the user-operated interface allows a building operator to efficiently manage large-scale building HVAC systems in an organized manner from the comfort of a computer screen. Siemen is on the cutting edge with respect to HVAC system controls and their particular user interface is, at the very least, on par with industry standards. Thus, introducing a user interface that is similar to that of Siemens' would prepare users for real-world industry applications.

## **5 Experience and Reflections with Learn HVAC**

Nigel Beckmeier, the graduate student working on this project, contributed his personal observations and thoughts on Learn-HVAC. As somebody recently through the mechanical engineering undergraduate curriculum, and in the midst of the Masters curriculum, he brings an important perspective to this project. These are his comments:

*My goal from the start of this semester has always been to find ways to integrate this software into the academic life here at UC Berkeley. First, I needed to understand the program from back to front, and thus I have spent the past semester experimenting with every major facet of the Learn HVAC program. Once I felt that I had a strong grasp on its capabilities, I began interviewing professors to obtain their thoughts for improvement. I selected teachers based on their expertise and ones that I thought might benefit from the software. I began each meet with a walk-through of the basic functions, and would then turn it over to them to see how they interacted with the interface. I first listened to their feedback, and then interjected my own thoughts on how I saw it helping them in the classroom. After each interview, I would collect my thoughts and write a proposal based on the research I had ascertained on how to change Learn HVAC to better fit the needs of a particular professor.*

*My overall impression of the software in its current state is that it would be an excellent tool to accompany a class on HVAC systems; however, it should be altered if it hopes to teach without a formal lecture as the base. The visuals to me facilitate the learning process and are key in illustrating how the system comes together as a whole. However, I needed outside research (library materials) to learn about the individual components themselves. My studies have taught me the basics on how a heat exchanger works, but I'll admit that I had to go back and reread the specifics to fully understand the cooling/heating coils. I really like the idea of the faults, but I can't really envision a university professor assigning a problem to figure out which part of the system is failing. I'm not sure whether this fault feature is something that research indicated as a market need, but I strongly support going out and interviewing the target market to find out what that need is exactly. It would save a lot of time and effort in the long run and stop new features from being added that the clients don't really want or need.*

*I approached the software from several points of views, including that of a student, teacher and designer. The student mindset came rather easily to me considering that I am a student and have had little prior experience with HVAC systems. I found myself wondering what to do with the program. I understand that the software is still being developed, but I felt that I had no direction in how best to approach the learning process. I think that the addition of a general walk-through is imperative to the product's success. Even something as simple as pop-up windows would give the students a world of direction, "This is the Cooling Coil – this is where a heat exchange occurs as the hot air comes into contact with metal coils that have chilled water running through them. Next is the ..." I wish that I had asked more of my classmates to test out the software because only with student feedback could we really understand the end user.*

*From the professor point of view, Learn HVAC needs to make more data readily available to the user in order to solve the most basic of engineering laws. Take the First Law of Thermodynamics for example, and let's say a professor wants to assign a simple energy balance problem on the fan. The software in its current state doesn't give you the air flow velocities, pressures on either side or even the net work per unit mass flow exerted by the shaft of the fan. When you click on the fan, professors want to see: (1) a dotted line outlining the boundaries of the area being examined, (2) an equation below defining the energy balance, (3) arrows going in or out of the system to show whether work is being added or extracted, and finally (4) all the raw variables to plug into the equation. Professors could even hide one of these variables and let the student solve for it themselves. However, without these quantifiable values, the software has little value in teaching the fundamentals of engineering at the university level.*

*Lastly, from a perspective HVAC designer's point of view, the software needs to have more variable options in the layout of the system and a cost measure to easily differentiate between designs. What I mean by more options is, for example, a tool bar that allows the user to add a number of different heat sources to the room or change between cooling coils of varying cost and efficiency. Heat sources could include typical office equipment like computers, printers, coffee machines and refrigerators. Through trial and error, the user would learn that the fridge is a huge energy consumer, which translates to X amount of dollars in a particular season. Designers need to have these options to gain the experience that, for example, the inexpensive Cooling Coil Model A works better in milder climates, whereas the high cost and highly efficient Model B works best in hot desert climates. Just as important as the options, the software needs quantify the cost factor behind the HVAC system. By inputting local energy rates for various tiers of electricity, the software could calculate the dollar value behind cooling a building in any season for any period of time. This cost factor drives innovation, and designers need to understand the trade-offs between initial cost and efficiency. All in all, the Learn HVAC software provides a valuable resource for teaching about HVAC systems, but needs to add a number of new features in order to be useful at a University level.*