CONTEXT AWARE SERVICES FOR SMART LEARNING ENVIRONMENTS

By

Kristopher Scott

A Thesis Submitted to Department of Electrical Engineering

in Partial Fulfillment of the Requirement for the Degree of

Master of Science

at

Lakehead University

Thunder Bay, Ontario, Canada

September 2009

ACKNOWLEDGEMENT

I owe a great deal of gratitude to my supervisor Dr. Rachid Benlamri. Without his guidance, this work would have never been possible.

Abstract

This thesis describes a cost-effective infrastructure and implementation for building ubiquitous learning spaces. It uses techniques from the semantic web and ubiquitous computing to build a learner-centric service-based architecture to transform existing traditional learning spaces (e.g. classrooms, computer labs, meeting rooms, and hallways) into intelligent ambient learning environments. This is achieved by blending a number of inexpensive technologies which are optimally configured to provide services that can perceive learners' location and schedule, identify current learning activity, recommend learning resources, and enable effective real-time collaboration and resource sharing between learners and their instructors. These services are semantically defined using service policies based on a shared ontology and reasoning techniques, and the system is designed to dynamically adapt to fulfill technological needs in a variety of learning situations. The proposed system uses many patterns of formal, informal, and collaborative learning supported by a range of mobile and handheld computer devices to enable better instruction in various learning setups. A prototype system is designed and built, a few learning scenarios are presented, and some test case user feedback is provided to evaluate and analyze the system.

Table of Contents

List of Figures

List of Tables

I. Introduction

With the rapid changes in information and communication technology, new practices are occurring in the teaching and learning process. The recent widespread adoption of mobile computing devices and the availability of wireless access in most learning spaces have greatly impacted the learning process. These tools are allowing learners and instructors to find new ways to communicate, collaborate, and interact. Ubiquitous access to information has helped to shift the emphasis of education away from the simple transmission of information to an active acquisition of skills and knowledge [1]. Other uses for technologies have given rise to applications that facilitate team-based, collaborative, and inquiry-based approaches to constructing knowledge [1]. Many ubiquitous learning systems are being developed to build learning environments for specific learning needs [2]-[6]. The main challenge, however, is that it is not well known how these technologies can be integrated to sustain large learning spaces that can support various learning environments (e.g. formal, informal, collaborative and virtual leaning) without major reconfiguration [7]-[9]. Research in learning space development has received less focus as opposed to research in special purpose e-learning/m-learning systems. This thesis addresses these shortcoming by investigating new ways in which on-campus spaces (e.g. classrooms, labs, meeting rooms, hallways) are augmented with new emerging communication and information technology to provide better and various learning opportunities. The system uses many patterns of formal (i.e. scheduled instructor-directed) and informal (i.e. independent self-directed and collaborative) learning supported by a variety of devices that are commonly used by learners (e.g. PDAs, iPods, cell phones, and laptops) to enable better instruction for various learning situations.

Traditionally, learning spaces (e.g. smart classrooms and computer labs) have been configured to support teaching and learning by providing rows of computers connected through wired networks in a lecture-style classroom set-up. Furthermore, smart classrooms have sometimes been criticized as wasteful misallocation of resources [10] as they include much expensive multimedia equipment while offering limited facilities as far as learners' mobility, interaction and collaboration are concerned. The uses of technology for teaching have evolved, however, and so must the design and configuration of learning spaces. Learning spaces must transform into flexible, technology-enhanced spaces where learners should be the key drivers in learning space design. This thesis suggests a cost effective architecture to transform existing learning spaces into effective spaces that enable better learning, given the resource limits of a university setting. This is achieved by using inexpensive technology consisting of infrared-based location sensors, basic networking facilities including ad-hoc and zero-configuration (Zeroconf) [11] networks, personal laptops and various handheld devices. These technologies are optimally integrated and configured to provide flexible and extensible services, making use of recent advances in wireless communications, mobile computing, ubiquitous computing, and the semantic web. Such technology has the ability to perceive learners' location and schedule, identify current learning activity, retrieve needed learning resource, share learning resources and interact with peers, provide learning recommendations, and sets alerts for various task deadlines. All of these services are semantically defined and homogeneously integrated using a shared ontology, service policies, and inference rules. Service invocation and co-ordination are triggered at run-time by context-changes in the learning environment, thus offering full context awareness and providing real-time support for various learning situations.

The main contributions of this work can be summarized in the following – (i) reliable location-awareness system to enable both learner and device mobility – (ii) infrastructure to support various real-time instructional and learning interactions – (iii) efficient integration and convergence of instructional technology and services to support both formal and informal learning in various learning environments – and (iv) ontology based model for a learning resource recommender.

The remaining of the thesis is structured as follows. Chapter Two describes background and surveys major research work in learning space design. Chapter Three describes in details the system design and architecture. Chapter Four describes three scenarios for potential use of this system. Chapter Five analyses user response in learner testing of the system. Finally, conclusions are drawn and further research is suggested.

II. Background and Related Work

As mobile and wireless technology rapidly advances, new physical and virtual learning spaces are changing the way that learners access and share resources, acquire knowledge, and interact and collaborate with each other. In these modern ubiquitous learning spaces, technology can move beyond the relatively predictable wired classroom computers and dissonant presentation systems. Learning expands to diverse embedded sensors, wireless instructional devices such as handheld computers, and a variety of interconnected technologies. These enhancements allow learning to expand beyond the classroom to labs, field-trip locations, meeting rooms, and even hallways and study areas. With all these concerns, designing ubiquitous learning spaces becomes a complex and challenging task that involves computational and learning paradigms. Context awareness, ubiquitous computing, and semantic web technologies combine with new and innovative learning and instructional interactions. This thesis proposes a solution to some of these challenges, in particular those related to learner mobility, context awareness, interaction and collaboration flexibility, and seamless resource sharing. Before addressing these concepts, however, some of the ubiquitous learning systems that have shaped this research field in the past must be surveyed.

There are many existing technologies that have the potential to be developed into powerful learning enhancement tools [12]-[17]. Sabine Graf et al. [3] have developed an infrastructure for problem-based pervasive learning environments. Their system is based on a multi-agent system architecture to integrate the various components of the learning infrastructure. In particular, the infrastructure includes a location and context awareness service, a question and answer service, an adaptive mechanism, problem-based ubiquitous learning models, social networking issues, and the evaluation of multimedia inputs. Although this system presents many features desirable in a ubiquitous learning environment, it is limited to one single learning mode, that is problem-based learning. A practical learning enhancement technology must be comprehensive as well as versatile, providing technological solutions for a wide variety of learning styles, methods, and modes.

The uClassroom [4] project is a more modern ubiquitous computing environment that is designed for school applications, created at Nagoya University of Japan. It uses context to provide optimized educational information via a web interface. While uClassroom mentions location data as part of its context awareness, it is not intended to provide any potential system for attaining this location information. Additionally, it does not involve the communication between individual systems, limiting potential applications in collaborative and peer-to-peer situations.

KERIS, a leading e-learning organization in Korea, has developed a ubiquitous smart classroom environment called u-Class [5]. u-Class combines teaching methodologies and learning resources from homes, schools, and communities through a ubiquitous, interactive network. Diverse educational materials, including electronic blackboards, electronic podiums, video lecturing facilities, magic mirrors, tablet PCs, RFID, electronic cabinets, and media books on an invisible network naturally connect the teaching activities of instructors to the learning activities of learners. Although u-Class provides an RFID attendance checking system, it is limited to closed spaces (physical classrooms) limiting the location awareness ability of the system. In addition, uClass is designed to support formal instructor-directed learning, thus limiting its use in peer-to-peer and collaborative learning situations.

Jones and Joe, at Griffith University of Australia [6], have developed a Ubiquitous Learning Environment (ULE). ULE is an adaptive teaching system using ubiquitous technology. The two main factors in ULE design are the 'what' and the 'how'. The 'what' is the model itself which resembles an interactive learning gallery and uses a wireless network with both Bluetooth and WiFi technologies. The 'how' is the inclusion of pedagogical information which is based on constructivist theory, allowing students to create knowledge from what they see, hear, read and perceive. Students using the ULE will intuitively interpret their surroundings and construct their own knowledge. Each student will carry a wireless device (PDA or mobile phone) fitted with headphones. The ULE server module tracks and locates each student within the u-space by the use of sensors. When a student approaches an object, sensors wirelessly access the intranet and ULE server module and transmit information about the object. The data is then transmitted from one of the objects to the student's handheld device. ULE provides an interesting perspective on ubiquitous learning, however it does not adapt well to enhancing conventional learning spaces and interaction styles. The system is more suitable to learning in environments such as science and technology museums and other similar setups.

Finally, Blackboard Learn [18] by Blackboard, Inc. is a leading commercial system used in many education systems. It includes web content and course management, much like the others. With the addition of extensible modules, it also supports note collaboration systems and some group chat and discussion. The Blackboard system provides the most complete education system, however its focus remains on external activity to supplement classroom learning, rather than enhance the learning experience itself. Additionally, the complexity and scope of the Blackboard system causes it to be somewhat confusing to use. Without context and location awareness, the system requires a lot of manual effort on the part of the learners and the administration.

Before suggesting a specific architecture, the requirements of learners in various learning situations were investigated and these concerns were used to inform decisions for the architectural design. This was done with the aim to bring the learner's voice into the design process. One key aspect of these findings is that new learning spaces should be designed to provide facilities that encourage collaboration and active learner participation by exploiting the new emerging technologies to their full potential. Indeed, active engagement with the learning object, whether a lecture, laboratory process, discussion, or creative medium, increases the likelihood that the learner will both retain and be able to use the acquired knowledge and skills later [2]. Another key aspect of these findings is that learners are drawn to spaces, both physical and virtual, that are open, easy to use, and stimulating. In terms of learning space design, this is reflected by three system-flexibility dimensions – (i) learning spaces should offer a variety of learning opportunities and levels of engagement (e.g. instruction spaces, seminars, labs, meetings, and virtual spaces) for both formal and informal learning and without major reconfiguration – (ii) learning space architecture should allow for integration and convergence of technologies and services seamlessly based on an invisible network infrastructure – and (iii) learning spaces should effectively support different location awareness capabilities (e.g. GPS, RF trilateration, and infrared detection). Furthermore, a learning space should enable learners to get to know each other, engage in dialogue, interact in a variety of ways, provide distributed control of content, and maintain communication and interaction across spaces without being tied to a physical location [1].

In this thesis, each of the above-mentioned requirements has been thoroughly addressed, yielding a learner-centric service-based architecture based on a hybrid peer-to-peer/client-server networking model which uses Zeroconf resource detection and web standard AJAX communication systems. Context-awareness and service adaptation, however, rely on semantic web reasoning. An ontology powers a comprehensive reasoning system for resource recommendations and learning tasks suggestion is developed based on inference rules coded in the Semantic Web Rule Language (SWRL) [19]. The system also relies on semantic web standards for user profiling by adopting the W3C Composite Capability/Preference Profiles (CC/PP) standard [20].

III. System Design and Architecture

In the development of a modern technological system for enhancing the learning experience, four goals were set, encompassing four aspects of the learning process. First, learners are becoming more mobile with their technology, and the system must be able to not only accommodate but also utilize this mobile nature. Second, mobile technology often expands beyond traditional networking and communication system into something very dissonant and complex. An education system should be able to connect learners and their teachers without any confusion or configuration. Third, there is an endless variety of teaching styles, methods, activities, and learning processes. A technological learning enhancement must be flexible enough to adapt to the teaching style of the professor as well as the learning style of the learner. Finally, the system must be seamless, inexpensive, and easy enough to use in learning situations that it actually becomes adopted by learners and teachers alike. By examining these requirements, it is possible to create an effective technological learning enhancement system.

The first aspect is the most concrete of the four, and likely the easiest to address. It calls for a system that takes advantage of the ubiquitous, mobile technologies to enhance the education system. Mobile computing platforms not only let learners work and learn from anywhere, but they also accompany learners to the places they already engage in learning activities. The system can take advantage of this mobility and the interaction with other learners in a common space by utilizing location awareness technologies. By giving a learner's mobile computer or device the ability to identify its physical context and, more importantly, other nearby learners, a whole new level of social learning interaction becomes possible. Learners in proximity can combine their learning resources (e.g. notes, lab data, etc.) to provide a more comprehensive experience, and students in lectures can connect with their professors while they are in the same lecture hall. Location awareness brings all of these possibilities and more, all automatically and without any effort from the learner.

The second aspect is also fairly concrete, however it is a little more difficult to provide a solution for. Many of the advantages from a mobile computing device in education come when it is linked to other devices and information sources. Large networks, like the kind used on campuses, can make it difficult to connect two computers together; they get lost in the vastness. Additionally, some learning spaces have no educational network available to them at all. A successful learning enhancement system requires a networking functionality that facilitates communications between learners while they work together, without all of the hassle of network management. Learners are more likely to collaborate if the technology to do so works automatically.

The third aspect deals with the uses of a technology-enhanced learning system. With an endless variety of teaching styles and learning styles in the world, a successful learning enhancement system must be able to adapt to a mode to fit any situation. The system should facilitate collaboration among various learners and their instructors. It should provide information to learners when and where they need it, and allow them to interact with the system and each other in ways that enhance their learning experiences. It should also encourage an active learning style, and encourage participation in class discussions, lectures, and activities to ensure that the learners are receiving the most out of their educational experiences.

Finally, the system should be seamless enough to integrate into existing learning experiences and provide benefits, not distractions. Technology has the potential to change learning interactions and experiences forever, but only once it becomes so invisible and natural that it is simply part of the learning process, rather than another thing to learn. The system should provide advancements to the learning experience that could not be offered by less technological systems. As well, it should integrate with other systems well to form a comprehensive learning system, rather than aim to replace educational support systems that are already comfortable for learners. The bottom line is that the learning system must be capable of supporting a learner's educational interactions in a way that is always enhancing and never detracting.

To address the problems mentioned above, the system is designed to fully automate the navigation, the interaction, and retrieval and sharing of the learning resources, giving the learners instant access to the relevant information for their current lecture, lab, or learning session. The system does this by implementing context-awareness in several forms. By knowing its own location, the identification of the learner, and the current time, a learner's computer/handheld device has everything it needs to retrieve and use all of the supporting information automatically based on current learning activity. The learner enters his or her information the first time they use the system, and the current time is available to the computer, so the location is the only context parameter that must be determined through the described system. In the Sections A through D of this chapter, the main components of the architectural design are described in detail, aiming to achieve the above-mentioned benefits of effective learning spaces.

A. Location Awareness and Learner mobility

One of the most important factors in the automation of the classroom system is the location awareness system. Location data is used to determine the physical and social context of the learner, which is the basis for the location-aware interactions. While there are several ways to detect location automatically, as long as a common learning area can be determined the system will be able to assert its context. This is important, because it means that the system should be designed for low granularity (the location of a device within a learning space is irrelevant) but high accuracy (devices near the boundary of two learning spaces must be resolved to the correct one). Any of several sources of location information may be used to determine learner context, however some are better suited to learning environments than others.

One of these potential sources is radio frequency (RF) trilateration technology [21]. This option is a lucrative one in the location awareness field, due to the prevalence of wireless transceivers in modern mobile computers and devices. RF trilateration uses a collection of transmitters, typically existing wireless network access points or cellular communication towers. The location of these transmitters must be known precisely for the system to operate, and they are often stored in a centralized database. A matching wireless receiver in the device to be located is able to sense the transmitted signals and their respective signal strengths, or in some cases, based upon precisely accurate time stamps and "time of flight" delay times. Using the signal strength or delay data of two or more transmitters with known locations, the system is able to perform trilateration calculations to determine the approximate location of the receiver, relative to the transmitters. This relative location is queried to the transmitter location database to retrieve the actual co-ordinates. The end result is a mathematical approximation of the device's global location.

While this location detection method may be useful for gross-scale location awareness, and it requires no additional hardware beyond that which is already present for wireless networking or cellular services, it is far from ideal for interior learning spaces. Trilateration calculations that use signal strength as an indication of distance assume uniform degradation in every direction from the device to be located. This is almost never the case, especially in indoor environments. In a technology-rich indoor environment like a classroom or a lab, there are many sources of interference, attenuation, and noise. All of these RF obstacles make the signal degrade differently depending on the position and even the orientation of the device to be located. A higher rate of signal loss in one direction will make the device seem further away from that given signal source, skewing the calculations and greatly affecting the accuracy of the location results. Delay-based systems are generally designed more for cellular phone location applications, which encompass a much larger exterior area. The difference in round trip times is miniscule between locations within a building, and area also affected by the same signal obstacles as a signal strength based system. Finally, RF based systems have no sense of physical space separators, such as walls and dividers. It is extremely difficult to differentiate between users at one workstation from the adjacent workstations. Without firm divisions, the accuracy requirements for location data as context skyrockets. When attempting to differentiate between adjacent rooms, or even adjacent workstations, this inaccuracy can become a critical flaw.

Another common location detection technology is the Global Positioning System, or GPS [22]. GPS functions the same as the delay-based RF trilateration systems, but on a much larger scale. GPS uses geostationary satellites with known sky positions that transmit atomically-accurate time-stamped signals. A GPS receiver picks up these signals in the device to be located and, using the time difference and similar trilateration calculations, is able to determine its location. GPS has an accuracy advantage over RF. There are many more GPS satellites than there are geographically-mapped wireless networking or cellular transmitters, and these satellites are guaranteed to be in dramatically different directions with respect to the receiver. These factors make GPS generally accurate enough to resolve a learner's location to a room-sized area (albeit, much like with the terrestrial RF methods, without regards to workspace divisions or walls). While it may provide adequate accuracy to use as a source of location context data in outdoor learning environments, its requirement on a clear view of the sky makes it severely limited for traditional learning spaces. With a limited view of the sky, like that available through a classroom window, the system will lose the advantages in accuracy gained by the wide directional disparity in reference point. In most cases, however, there is no significant view of the sky and the system becomes unusable.

An alternative technology that is better suited to indoor spaces is infrared (IR) location detection. Infrared provides short-range, line-of-sight transmissions that are ideal for the small areas and light-blocking boundaries (i.e. walls) that exist within a building. In Active IR systems, such as ALTAIR [23] a location detection system from the Nara Institute of Science and Technology, every device to be located is equipped with a small infrared transmitter. This transmitter sends a unique IR identification signal into the room by pulsing an infrared LED in a pattern modulated by the device. To receive the signal, the room is equipped with several infrared cameras and detectors which are matched to the carrier frequency of the transmitters. If the device with the infrared transmitter is detected by several cameras, the system is able to use similar trilateration algorithms as with the RF systems to mathematically determine the device's position within the room. The process is repeated and run continuously so that the system always has accurate positioning information for every IR-equipped device within the monitored space.

While this camera and trilateration system would work rather flawlessly for the usual interior learning situations, such as a classroom environment, the hardware efficiency can still be improved. An Active IR system requires a specialized network of sensors and cameras within the room, which must be connected to a computer. This computer is required to perform the calculations to determine the locations of the contents of the room, and then to extrapolate that precise location into a given logical location expected by the system. This type of centralized location computer makes this type of system conceptually similar to a traditional client-server model in networking. With all of these calculations being handled centrally, the server would need to be performing thousands of calculations repeatedly and maintaining a database of all of these locations in a typical high school or university setting. This would require one or more server machines running just to maintain accurate location information for every connected learner. Even with a server system like that in place, the learner's device itself is not able to determine its own location in order to perform context-related actions. It is required to query the server and the calculated information must be sent back to the device to provide true location awareness. That means, in addition to monitoring and calculating the location of every device, the server also has to maintain connections to every active device to deliver location information to them. This all results in large amounts of overhead that is not at all necessary for the task, thus increasing the cost and complexity of the system.

The system described in this thesis uses a novel Passive Infrared system that addresses both of these concerns and provides an ideal solution for this kind of interior context awareness. A passive system takes advantage of the fact that each user already has a computer or similar device at their location (for the display and interaction with the system). This user-side intelligence eliminates the need for location-side intelligence, decreasing the overall cost and complexity of the system.

The system depends on the implementation of Passive Infrared beacons. These beacons are installed in common work areas such as classrooms, lecture halls, labs, libraries, and any other on-campus locations where learners may gather to collaborate. In larger rooms and halls, multiple beacons can be used to cover the entire room and act as a single logical location. For labs and similar locations with multiple work areas, beacons can be arranged to split the room into smaller logical locations. The beacons can be thought of as metaphorical signage, clearly advertising the location identification number in a way that location-aware devices can read from anywhere in the region.

Much like a light fixture, the beacons illuminate the room, but they do it in the invisible infrared spectrum. This is the same kind of light emitted from television remotes, and is completely harmless to other systems as well as to humans. The signals are stopped by obstructions or walls, which may not be ideal for some location detection scenarios, but it is a desired characteristic for room and workstation level detection. It would be impossible for a boundary condition device (such as a learner sitting against the wall of a room) to be mistakenly identified as being in the adjacent room. Separators in the physical space also function as separators in the logical location space. The microcontroller is used to modulate the light into a specific pattern, and each pattern represents a unique workspace in the institution.

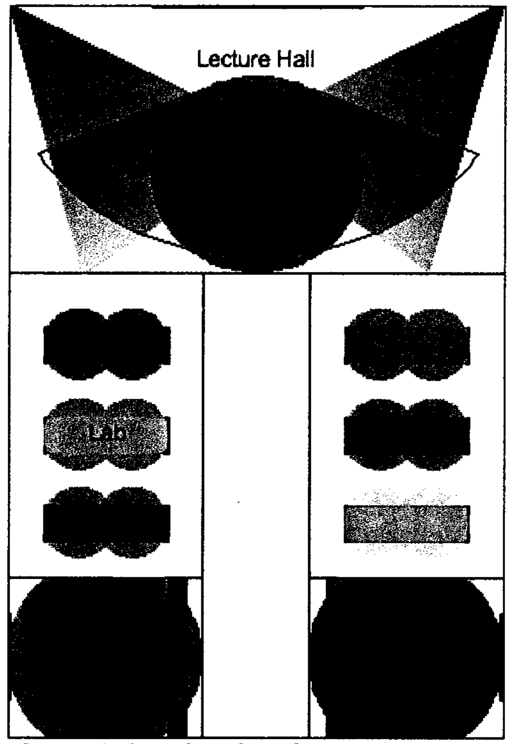


Figure 1: Sample Infrared Beacon Coverage

The beacons themselves are fairly simple devices, designed to be inexpensive and as easy to install as a light fixture. They are based on a simple power supply, inexpensive PIC microcontroller, a set of DIP switches, and one or more infrared transmitting LEDs. The microcontroller is designed to simply transmit a location identifier, predefined by the administrators using the DIP switches, whenever transmission is triggered. This triggering system had to be designed to allow for the synchronization of multiple beacons in a large learning location where one beacon could not provide adequate coverage.

Several methods exist to provide synchronization in pulsed signal systems. One of the methods considered involved a wireless or wired sync signal. A wire or radio frequency signal could be used to create an IP or other protocol-based connection between beacons in larger locations. This network connection could also be used to administer the system remotely, changing location identifiers without the need for DIP switches on the units themselves. While it did provide that small advantage, this method seemed extremely superfluous for the task required of it. It would require the installation of cabling and infrastructure throughout the learning space, as well as networking and communication hardware and firmware on every beacon. This would increase the costs of system implementation significantly, counter to the design goals of low-cost and simplicity.

One synchronization technique that was thoroughly considered is inspired by nature, known as coupled oscillation [24]. Often referred to as "firefly synchronization", this is the same technique used by male fireflies to flash in unison, and it is also responsible for many other biological functions [25]. This method requires no intercommunication between beacons, and can allow the system function autonomously. Each beacon cycles its location signal at a standard rate; the frequency of the signal is unified. Beacons would also require infrared receiver hardware, although this hardware is very low cost and not a prohibitive factor. When a beacon detects the transmission of a location identifier by a nearby beacon, its own transmission cycle is phase-shifted by an amount mathematically related to its current cycle's phase. This causes a phase shift, in an attempt to match the phase of the adjacent beacon. This process continues and propagates across all of the beacons in a region until they are all phase-matched and synchronized. The result is a unified infrared pulse, made up of the signals from all of the individual beacons combined.

The firefly synchronization method is a very good option for the system. With tuned phase-shifting algorithms, the beacons can synchronize quickly and accurately enough to provide unified location identification coverage. One major advantage of this system is frequency independence. The identification signal can be of any arbitrary length, as long as all of the beacons that have to synchronize use the same signal. This also allows the use of standard infrared encoding techniques, and simplifies the reception and decoding of the identifier signal. With arbitrary lengths, each bit can be well defined and the padding time between repetitions can be extended. These factors both increase accuracy in receiving the signal and decoding the location identifier at the user end. There are a few disadvantages that make this process less than ideal, however. The most obvious is the startup delay. When the beacons are first powered, they are not necessarily synchronized. The overall system of beacons needs to transmit and interact for a period of time before they unify on a common phase. Any devices attempting to receive the signal during this incoherent period will receive erroneous or inaccurate results. Additionally, the system will be very susceptible to interference from other infrared devices. Any stray infrared signal from a remote control, irDA device, or even from sunlight may cause the beacons to attempt to resynchronize. They will phase shift to adjust to the stray signal and the entire array will lose coherence.

The beacons use a third method for synchronization, which is based upon the AC power grid that each of the beacons is attached to. The power supply of a beacon based on a simple step-down transformer with a bridge rectifier to convert the 110V/60Hz AC mains power (or 220V/50Hz mains in some regions) to a 5V DC supply for the circuitry .

It also, however, contains an additional diode-based circuit to extract a 5V/60Hz trigger pulse from the AC signal.

This trigger pulse is used to start the transmission of the location identification signal. Beacons in the same physical location are ideally (and most likely) connected to the same electrical circuit, or at least on the same phase of the multi-phase power system. This means they will all be triggered by the same pulse, all simultaneously. The mains-based pulse trigger system provides the advantages and reliability of the networked beacon system described above without any of the hardware or infrastructure costs. The network cabling between the devices is already present to power the devices, and the network signal is derived from the electrical energy already present in that cabling. The result is an array of infrared beacons that are synchronized to fire in unison, collectively covering the learning space with location data.

The system uses an 8-bit Manchester-style encoding. Manchester encoding is common amongst infrared remote control devices as it requires no reference level, as infrared is an non-electrical connection. Logical LOWs are signaled with an ON-OFF signal transition, and logical HIGH with an OFF-ON. The signal itself is modulated by a 40kHz carrier. This is also common amongst infrared devices, as IR receiver modules use integrated band-pass filters to eliminate unmodulated and differently-modulated IR noise.

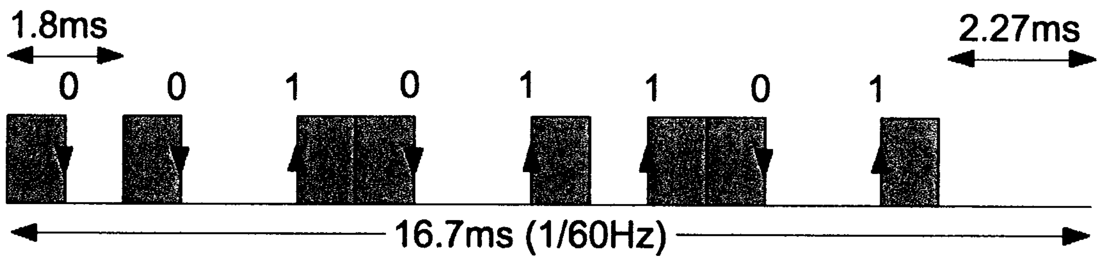


Figure 2: Manchester-like Encoded IR signal

One disadvantage to the mains-triggered system is that each location cycle is time-limited to a 1/60Hz period of 16.7ms (50Hz systems have slightly more time, with 20ms). If the signal takes longer than one cycle to transmit, the system will not be able to effectively synchronize. This limitation limits the number of bits that can be transmitted with the standard Manchester encoding used in IR. By narrowing the separation time between pulses to its minimum, it was possible to fit a full 8 bits in the allotted time and still maintain the rest of the Manchester specifications. This allows for 256 distinct learning areas. This should be enough locations to allocate for an entire physical location, like a building. For particularly large or complex institutions where more than 256 locations are required, this data may be supplemented by subnets, server allocations, and other traditional network-based allocations, all invisibly to the user. As long as no two locations are identical by more than one location detection system, there will be enough information to resolve the learners logical learning area and connect them to their fellow learners in proximity.

On the reception end, the location identification signals are detected by sensors attached to the learners' personal computers or mobile devices. The devices receive the

8-bit IR signal sent by the room beacons and, without any calculations or references, immediately have the logical location identifier for the room or workstation. This way, the system always knows its precise logical location, and is able to use that information to retrieve information relevant to that context, and connect to other systems in the same context, all automatically.

The primary clients for the system are portable notebook computers which use a USB interface to communicate with the sensors. The sensors are designed to be extremely simple and low-cost, with a per-unit production cost of $5 to $10. The sensors are powered by the USB bus and are designed to operate in a plug-and-play fashion without any specific drivers using the Human Interface Device (HID) protocol [26]. The devices themselves are very simple, consisting of a PIC microcontroller with integrated USB module and a standard infrared transducer which is frequency-matched to the carrier of the IR signal. The transducer itself filters ambient infrared noise from the room, and the microcontroller detects the repeating signal and decodes it into an 8-bit binary value. This value is relayed to the computer where it is received by the software and used to connect or create the local learning network.

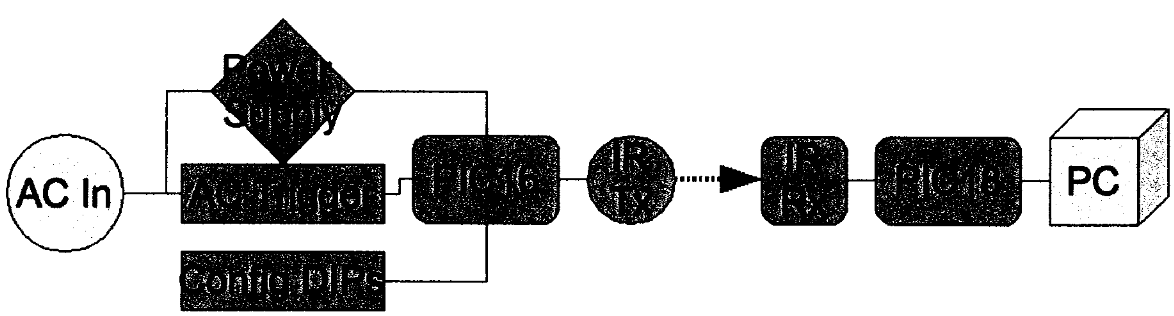


Figure 3: Block Diagram of a Single Beacon and USB Receiver

B. Infrastructure for Flexible Learner Interaction

Once the context is determined, the system establishes a network with the other learners in the same location. Depending on the type of system implemented and the current learning context, different types of connections may be required. In traditional computer labs, established networks are relatively simple to maintain. With known, static hardware, the network is architected in a structured top-to-bottom fashion. As computers move out of the classrooms and into less traditional learning spaces, however, new challenges are presented in constructing and managing connections between computers. Learners have their own computers and mobile devices which move with them throughout the environment. The system needs to be able to adapt so that the appropriate connections and interactions can always be made. A lecture or other one-to-many situation would benefit most from a formally structured network setup. Group work, like lab lessons or collaborations, requires a slightly less structured, informal type of connection. Sometimes there is a need to collaborate in an unscheduled on-the-fly manner with a group of learners, and this type of situation is also possible using the same system. Our system is designed to allow freedom in learning style and situation while maintaining a consistent user experience, with the constant goal of maximizing learning time and minimizing effort.

One way to fulfill the need for a dynamic network has been developed and it is known as Agent technology. Agent frameworks, like JADE [27], are designed to allow for peer-to-peer (P2P) communications between different nodes in a network. Agents are self-contained P2P nodes which connect with other agents to exchange information directly, without an intermediate server system. These agents can be the learners' computers, instructors' computers, or other classroom computers. When a system starts up, the agent framework in the software connects and registers itself on a network-wide Agent Management Service (AMS). This central server architecture is used at the initial connection time to manage and co-ordinate agents. As other agents connect, they check with the AMS to determine the current network situation, including the presence of other active agents, and then connect to other agents that provide their required services. The agent framework negotiates and connects to other agents in a P2P fashion; the AMS is still used to negotiate and connect additional agents, however it is not used in the transmission of data. The agent system is designed primarily for semantic applications. Information is exchanged between agents in a special format known as Semantic Language (SL). SL is designed to be an agent-specific communication format for use with semantic web technologies and ontologies. It contains syntax specific to semantic events, and when connected, the agents exchange information directly as events happen.

While an Agent system provides a standard and protocol for P2P communications and transactions, it is not necessarily ideal for the nature of interactions required in a dynamic learning environment. Agent systems require a central, known server to act as the AMS. While servers are generally available in a school environment for other uses, and indeed required for some advanced functions of the system , the requirement on a server for a P2P connection limits the system's versatility. To be able to adapt to the variety of physical environments and logical contexts that constitute learning spaces, it would be advantageous to adopt a more peer-centric networking system. As well, the system relies on an uncommon message format, the Semantic Language. While SL might be ideal for systems based purely on ontologies and specific Agent frameworks and libraries, it is very limited in both scope and utility. The system uses ontologies to map course information and provide a comprehensive learning enhancement, but there are many more aspects to the system than can be effectively mapped in ontologies. For this information (ranging from real-time course data to notes taken by learners) the ontological event model would have to be adapted and modified, adding complexity and delays in the overall system. Additionally, the Agent system does not allow for the integration of existing systems and resources. The Agents must only communicate with other Agents using their specific protocols and encoding. This would eliminate the use of any existing complementary learning systems alongside the Agents, and would decrease the available resources and learning enhancements that are available on the local network and on the wider internet.

With the wide adoption of HTTP based semantic formats like RDF (Resource Description Framework) [28], it is more beneficial for a system to adopt an existing common language. RDF is a format based on XML that allows the same level of semantic description as SL, but in a medium that is more portable and compatible with common web systems. This allows for interactions with existing web-based information sources and devices without translation or conversion. At a lower level, while Agent technologies like JADE provide utility in certain spaces and situations, the Agent system is not the best P2P concept to adopt for all of the dynamic learning situations that the described system seeks to enhance. Below is described an alternative approach to developing a hybrid peer-to-peer/client-server model for enabling flexible interaction and service creation on the top of an invisible reconfigurable network model.

1) Service Creation

Rather than employing proprietary and specific protocols and frameworks, this system uses standard web server technologies to share information between peers and to the central servers. Using modern bi-directional web techniques like REST [29], it is possible to transfer messages to and from HTTP servers with standard protocols. Bringing the client-server frameworks of the web to a peer-to-peer system presents some interesting challenges, but the advantages of a traditional, standards-based technology are plentiful.

Web interactions usually involve a client, like a web browser, and a server located in a centralized location. Even smaller internal intranet connections within a building or organization require a central server. This system looks to eliminate the requirements for a server, and instead embeds server functionality into every client. Frameworks, like those available from the CocoaHTTPServer project [30], package the functionality of a web server into Object Oriented classes. This allows for the trivial inclusion of web serving capabilities into a regular user application, and it allows the server to be created and destroyed just as simply as any other object.

When a user connects to a network and their computer or device is able to establish its location, it immediately checks the network for the availability of servers at that location. It first checks for a conventional central server. Traditional servers are used for support information, non-realtime aspects of the system, and for user authentication and network management. The peer then checks for existing peer servers (or "supernodes") in the current location. Depending on the results of these checks, the peer will instantiate a server object and promote itself to a local supernode. This means that the peer has server capabilities, hosting the web-style network in that location. The result is a small location-based network created on-the-fly, without any permanent or existing architecture.

During the development of this system, Opera Software introduced a new experimental project known as Opera Unite [31]. Unite functions as a feature of the latest beta version of the Opera browser, and it is the first known browser to include web service capabilities. The goal of the project is to offer users the ability to share their own content with others by hosting it on their own computers. Opera uses an online username system to provide a dynamic DNS system and maintain a consistent URL for a user's browser-server. Although it is possible to create custom Unite modules to overcome some of these limitations, the overall system is designed toward media and content sharing from one user to several others, maintaining the client-server mentality of the web. Each user essentially has their own personal website hosted from their browser, and other users may browse. The system discussed in this thesis connects all of the users to a single unified web application, and allows them to operate on it together simultaneously. This slight shift in perspective provides a significant shift in utility, as this unified system provides a more coherent sharing experience for information and contextual data, and is much closer to the P2P mentality.

There is a similar project based around a Firefox add-on that provides web serving capabilities called POW, which stands for Plain Old Webserver [32]. The POW solution provides a much more compelling environment. POW was designed to allow for personal web applications, designed to be served and run on the same machine, without external connections or users. It is worth noting, however, that this is the goal of the project, and the local connection limitation is arbitrary and not required. This system, therefore, could be used as a supernode in a learning environment, and host a P2P session amongst a collection of learners. This is where the advantages of the described system begin to show. Primarily, Firefox and the POW extension have no support for user context and location. The system described in this thesis is designed to interact with hardware sensors to determine the precise room or workstation of the computer and its user. The POW system has no such support, and users must enter a specific URL to connect to a POW server. As well, there is no support for the automatic negotiation of the location-based network. The described system uses a set of configuration indicators, after which it will start the server, if necessary. Subsequent learners will automatically be connected to the existing server, something that cannot happen with the POW system. Without any automatic negotiation and connection capabilities, Firefox POW still falls short for the described application.

By adapting web technologies into the core of the system, it allows the system to be expanded in several ways. Firstly, it facilitates the integration of existing web services and systems into the classroom. These existing services may include e-learning systems previously adopted by the institution, web resources created or discovered by the professor, or even from educational and academic services and backend systems that offer their data in a machine-readable formats. Resources and functionality related to the system can be seamlessly integrated into the interface of the classroom system and even work automatically with the location and context awareness systems. Because the peers all display standard HTML pages and include the ability to parse and display XML data, any web-standards-based web resource is available to every peer without adaptation or transposition into another format. As well, classroom resources can be easily integrated into the system. In a science lab, for example, a computer connected to measuring equipment need only be able to share those results in a web-standard format for them to be integrated into the rest of the technology of the learning space. The system brings modern web technology into a real-time context-aware application, and in effect brings the entire potential of the academic web into its comprehensive grasp.

Along with adding the capabilities to work with third-party services and web sites, the strong emphasis on web standards adds another incredibly valuable feature: the ability to work with third-party clients. With a stand-alone or proprietary system, the level of standards compliance is not an issue. As long as the system functions as intended the system may be designed and implemented in any way possible. When designing a system to be universal and easily adopted, however, it becomes more beneficial to increase the userbase. Since information is already transferred in HTML, XML, and Javascript calls, it is trivial to also extend support to other clients and indeed other devices where the fully featured peer software is not available. The user interface of the system and all of the functionality is created to be rendered just as easily in a browser as it is in the peer software. This functionality could also be used by stationary classroom devices, like display systems. As long as the projected image can show the session's dynamically-created web page, it may function as a fully capable client. Optimized for the WebKit browser engine, the system also has potential for non-computer mobile devices. Most modern mobile smartphone operating systems (other than Windows Mobile) use WebKit in their browser applications. Using custom javascript and CSS information from these browsers, the system is even able to present to them CSS-modified pages relative to their screen size. All of this is done without adding any additional software or hardware to the system: everything is already inherently extensible.

It is worth noting that third-party servers that contribute information to the system are not considered peers in the overall P2P architecture employed by the system, and neither are third-party web clients and browser systems. This provides an interesting dynamic twist on the traditional client-server and peer-to-peer system. The peers themselves are both independently-functional clients AND servers, with the possible expansion to other pure clients and pure servers to expand capabilities and accessibility. The overall makeup of servers, peers, and clients can be seen below in Fig. 4.

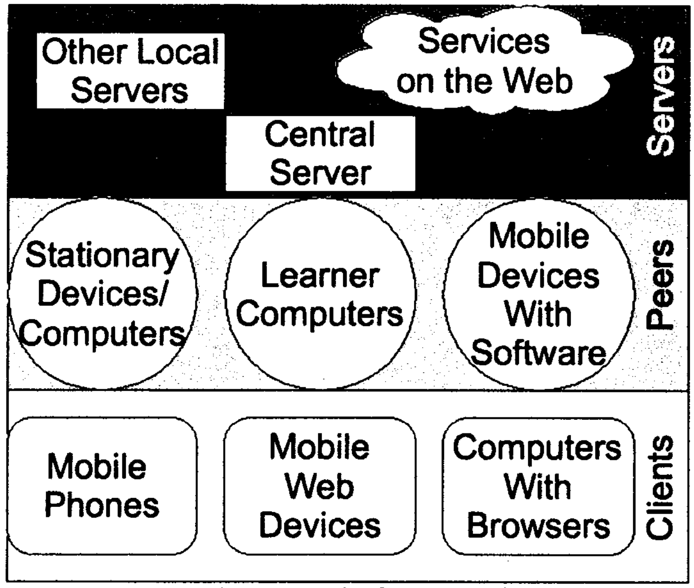


Figure 4: Pure Servers, Hybrid Peers, and Pure Clients

2) Service Publishing and Discovery

Rather than using a centralized service directory, the described system uses a decentralized service publishing and discovery technology known as Zeroconf (Zero Configuration) [11]. Zeroconf is an open-source project, based on a form of DNS-based Service Discovery (DNS-SD) that takes advantage of the broadcast capabilities in the TCP/IP stack.

Zeroconf and DNS-SD are closely related to Multicast DNS (mDNS) [33]. In a traditional Domain Name System (DNS), a central server contains all of the host names and associated network IP addresses for nodes in a given network. When a user requests a resources at a given URI, a request is sent to the known DNS server. The DNS server resolves the requested URI into a specific IP address where that resource is known to be found. In mDNS, the central server is eliminated. Instead, URI requests are sent to a broadcast IP, an address on the local network that transmits messages to every node. It is worth noting that this broadcast IP is generally non-routable, meaning that hosts and services are only available for local connections. The node which is intended to respond to the given URI will respond, and all other nodes ignore the request. This allows the system to automatically allocate domain names without a central server to manage them. Zeroconf takes the same concept but instead applies it at a higher level, to available services. It is a service discovery system and it uses the same type of broadcast messages to advertise the availability of services it offers using other protocols (in this case, an HTTP service).

When a peer initializes, it first detects its location identifier from the location system. It uses this location ID as its service name when performing a Zeroconf request. Like the mDNS system described above, a request is broadcast for a server with the same location ID. If no server exists (and, if a central server is authenticated against, if the peer it authorized) the peer promotes itself to supernode status, enabling its own server capabilities (as described in the previous section). The new supernode then advertises its own HTTP services on the Zeroconf system using the location ID, and any subsequent peers will be able to detect them. When a peer receives a broadcast response from an available supernode to its Zeroconf query, it then opens communications and resolves the address, port, and connection information from the supernode. It can then use this information to open a more traditional communication connection to the supernode and transfer data to and from its peers.

This Zeroconf system, combined with the application-space HTTP server, has several advantages over other P2P connection methods and styles. The most prominent of these advantages is the independence from any central server. Most P2P systems rely on a central server, or at the very least a known address for an existing peer [34]. This is how P2P networks are able to maintain coherence and connectivity among all peers. Due to the context and location dependence of this system, however, peers will always be assumed to be within physical proximity to each other. They are also assumed to be on the same local network, and so routing complexities and previously-known IP addresses do not become an issue. The single supernode for a location may actively advertise its existence on the local network without fear of saturation or interference. In fact, it is asserted that the broadcast messages from mDNS-type systems like Zeroconf are of the same order of magnitude as standard ARP packets that are continuously sent on TCP/IP networks without issue [33]. Also, as mentioned above, the nature of these type of broadcast messages prevents them from being routed to external networks, making this decentralized discovery possible without conflicting with other locations. As an added advantage, this also eases the distribution of location identifiers. Traditional web and network traffic can continue to be routed normally while location-specific networking is isolated by the routing level. As long as there are no two identical location identifiers within the same routed subnet, a unique location-based network grid will be created for each location ID as nodes attempt to connect. The result is the automatic assembly and management of web-style connections, all without the requirements that come with managing a static server.

3) Adaptable Network Architecture

The realtime classroom system is designed around a Hybrid Supernodal P2P architecture. To understand this networking architecture, it makes sense to look at it from back to front. P2P networks transfer data directly between nodes (or "peers") without a dependancy on an intermediate server. In a pure P2P system, every node is treated equally, and each one has the capacity to function autonomously and connect to other peers. A pure P2P system, thus, creates a meshed network layout (see Fig. 5). Peers connect to several other peers, and relay messages between peers without a direct connection. This architecture makes sense for situations where every peer has an equal level of significance and an equal potential for sending and receiving data. In most learning situations, however, not every node is an equal peer: there tends to be one "teacher" providing most of the data and many "learners" receiving that provided data. A pure P2P system would include additional overhead and convolution to this primarily unidirectional data flow, and is therefore not the best solution for a learning environment.

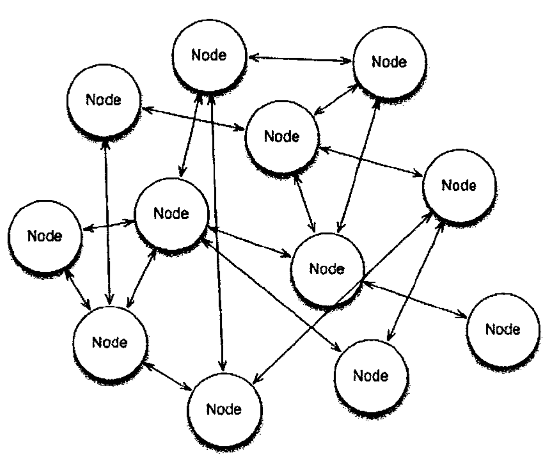


Figure 5: Pure P2P Architecture

There are variations on the P2P concept that provide a better fit for this type of learning environment. The best way is to adopt some of the one-to-many structure of a client-server system while still maintaining the autonomy and self-contained nature of P2P. The system, therefore, uses a Supernodal architecture (see Fig. 6). The supernode concept was developed by KaZaa Error: Reference source not found for their P2P filesharing application, and the same technology was migrated to the Skype [36] Voice-over-IP system to ease connections between nodes blocked behind firewalls. Every peer in a supernodal system includes the capabilities to act either as a standard client-type peer or a supernodal server-type peer, with the ability to switch between these modes. This returns some hierarchical structure to the P2P system while still maintaining peers by definition. By implementing full supernodal capabilities in every peer, it is possible for the network to recover and maintain the P2P advantage of a distributed system. If one supernode disconnects for any reason, the session is maintained and another peer may simply be promoted to be the new supernode, continuing the learning session without missing anything.

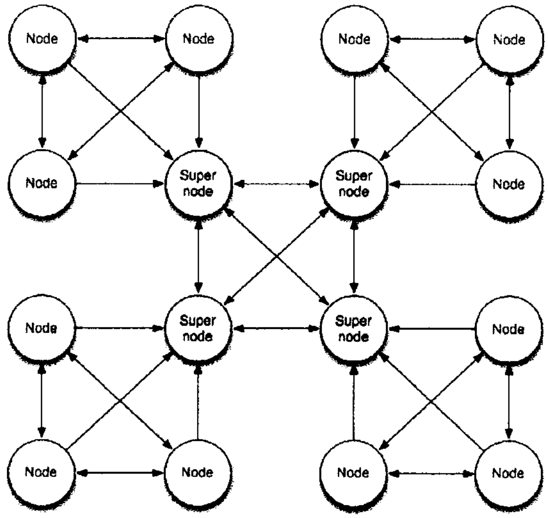


Figure 6: Supernodal Architecture

The final classification of the location-based network is that it is a Hybrid network. A hybrid P2P network is one that still maintains a server for some elements of the network traffic. Generally speaking, a hybrid network server is used during the initial connection process, for various reasons, and then neglected in favour of P2P communications as nodes become connected. In this location-based network system, the central server is a non-essential part of the architecture that provides several services and functions. If present and available, it provides user authentication and permissions, location-based network management functions, and recommendation and support information on request of the peers. It is important to note that no persistent connection (or, indeed, no connection at all) to the central server is required for the P2P system to function. While it may help in negotiating small subnetworks, the peers have enough connection logic to handle connection management autonomously. See Section D. Learning Task Recommender for more information on the additional support roles of the central server.

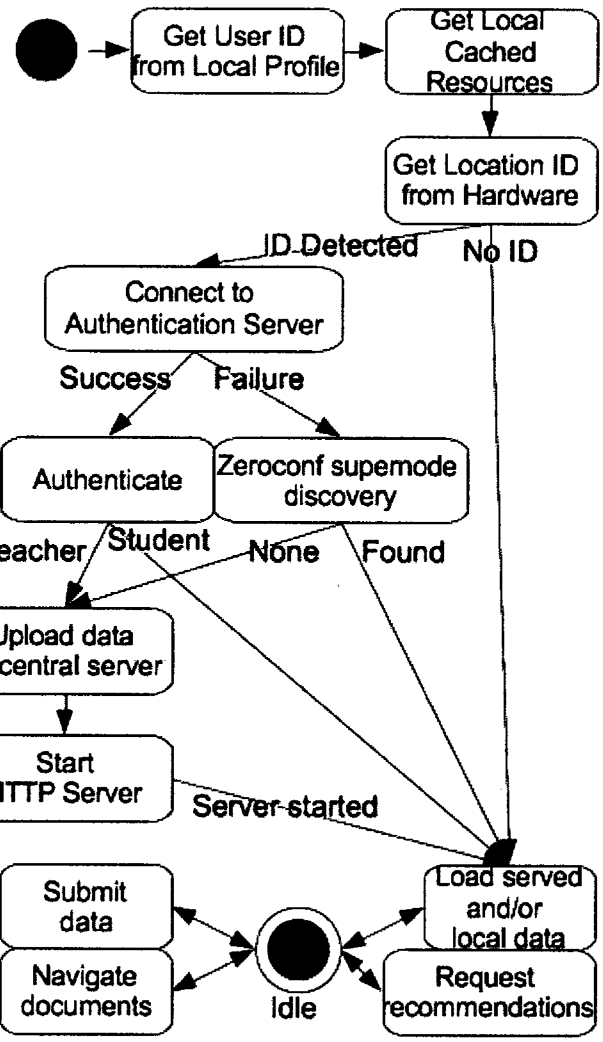


Figure 7: System Setup Process

The best way to examine the versatility of the network is to look at the various modes and configurations in which it can operate. Depending on the type of system implemented and depending on the current learning context, different types of connections may be required (see Fig. 7). A lecture or other one-to-many situation would benefit most from a structured network setup. In this situation, the location, content, participants, and session leader are all predetermined and registered with the central server. In this system, this type of situation is known as a Formal learning session.

Group work, like lab lessons or collaborations, requires a slightly less structured, more communal type of connection. This is a kind of context where the location and content is known but the participants and leader may vary between schedule time and the event time, or even during the session itself. This is known as an Informal learning session. Sometimes there is a need to collaborate in an unscheduled on-the-fly manner with a group of learners, and this type of situation is also possible using the same system. In this case, there is no prescheduled session on the central server at all, and in fact the server is not even required. This is known as an Ad-Hoc Collaborative learning session. Our system is designed to allow freedom in learning style and situation while maintaining a consistent user experience, with the constant goal to maximize learning time and minimize effort. Below describing the way the network infrastructure dynamically adapts itself to the current learning environment. The system senses the current learning context and automatically reconfigures itself to provide the best learning setup and the most efficient interaction mode. It should be noted that the network reconfiguration process remains invisible to the user.

a) Formal Learning

After establishing its context, the learner's computer attempts to establish its current situation and the network appropriate for it. It starts by assuming a formal structured setup, as would be useful in a lecture. The computer connects to a centralized school server and provides its location and context information (see Fig. 8). The server then checks with its authentication information to determine the role of the connecting peer within the context. Possible roles would be, for example, an instructor, a teaching assistant, or a student. The server determines if there is another user in the same location who is scheduled to lead a lesson, or another user with higher authority. If a location leader exists, this information is sent back to the learner. The learner's computer uses the Zeroconf system to detect the existing session leader on the network. It then connects to the leader's computer, which is functioning as the location's supernode, directly via the HTTP connection. If the given user is to lead the lecture themselves, the system allocates them as the location's leader and their own computer is set up as a local supernode and broadcasts its availability using the Zeroconf system. When other learners enter the same location, the central server notifies them that a location supernode exists, and the computers connect to that machine and begin to receive the lecture data and exchange learning information (see Fig. 9).

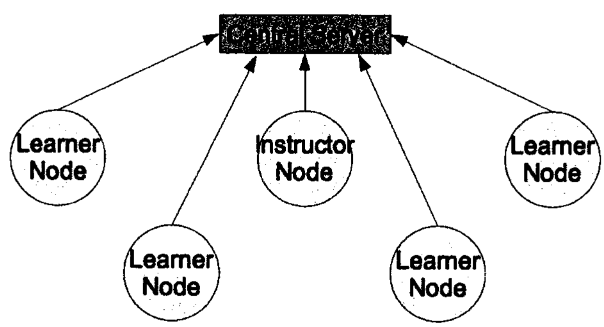
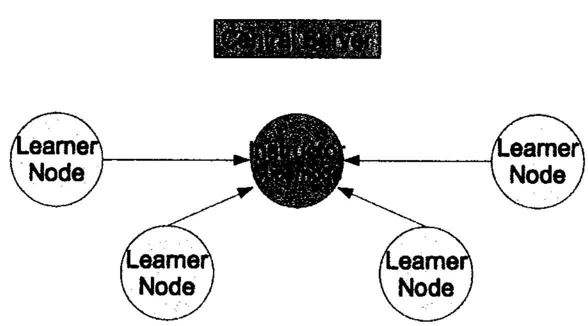


Figure 8: Learners Authenticating for a Formal Session



9: Instructor Authenticated as Supernode

A formal learning session includes several other nuances that affect the functionality of the system. Formal sessions are most likely a one-to-many situation, where one instructor is leading a lecture or class (primarily providing information) and many learners are participating in the class (primarily receiving information). In this situation, the software is configured to provide the professor with universal control over the slide progression, in order to lead the lesson uninhibitedly. Students are free to take notes with the note function, and these notes are tied to their contextual slides. The chat functionality may be disabled, limited to instructor-learner only, or fully functional, depending on the personal preference of the session leader. Question functionality is enabled for use by learners for submitting questions to the instructor, and these questions may be anonymous at the discretion of the learner. The system may also be used for learner survey and quizzes, and these answers are aggregated by the instructor of the class. All of these configuration choices optimize the system for a traditional lecture-style learning situation.

b) Informal Learning

If the central server has no formal structured lecture scheduled for that particular context, the system will function autonomously, without any configuration or intervention of the learner. Without a pre-established leader, the first learner to authenticate in a given location context is arbitrarily designated to be the session leader (see Fig. 10). Their computer acts as the location's supernode, and all subsequent learners to arrive in the same location will detect their system and connect to it (see Fig. 11). Informal sessions include open lab sessions, where learners work together in a specific space on a specific task, but without a predetermined leader. Any session specifically linked to a certain location and context, with access to the central server, is considered an informal session.

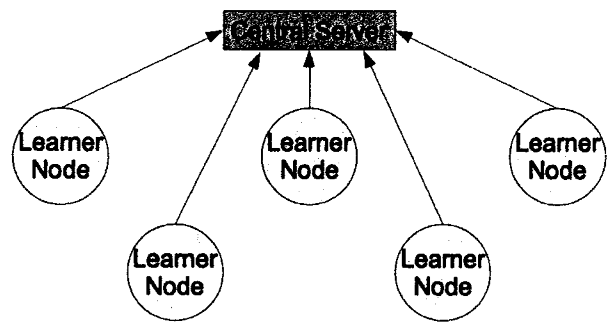


Figure 10: Learners Authenticating for an Informal Session

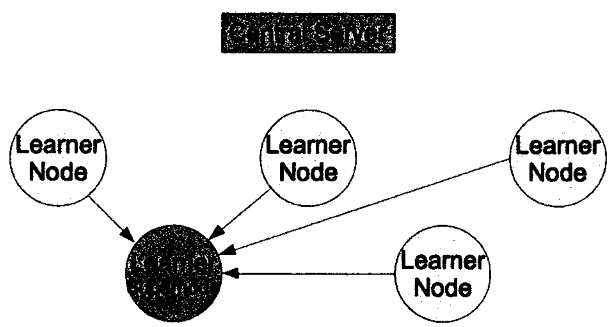


Figure 11: First Learner Selected as Supernode

The systems knows, from the central server's response, that the situation is different from a one-to-many lecture. Even though one of the nodes is established as a supernode for networking purposes, all learners in an unstructured session are treated equally, and may collaborate and work together seamlessly. Because there is no formal lesson leader, there is no individual to manage the permissions of the features of the session, and therefore the system is unrestricted and unauthenticated. Users may cycle through the main content pane at will, and they may chat and share data freely. Since informal sessions tend to be contained in smaller workstation-sized locations, such as in a lab, it makes sense to have the entire location "on the same page", literally. If an informal session takes place in a large location, it may be preferable to configure the system to disable synchronization of the main content pane, allowing learners to work at their own pace while still enabling content sharing and chat features. If a lecturer or other leader arrives in a location that currently contains an Informal session, they are authenticated as the leader and can take control of the session, converting it to a Formal session if they so choose.

c) Ad-Hoc Collaborative Learning

Due to the autonomous, peer-to-peer aspect of the learning network, the system is able to expand beyond traditional lessons and environments, even beyond the access to the central server. If the system is unable to detect the central server, it can operate in an entirely ad-hoc P2P fashion (see Fig. 12). The system will attempt to establish its context, using infrared beacons if they are available, or using other location awareness technologies such as RF trilateration or GPS (resolved to a general room-sized radius), if available. If absolutely no location data is available, the user may choose to specify a location, chose a random location identifier (which must be shared with participants via other means), or simply consider the entire available wired or wireless network to be a single location. This aspect is particularly useful for small off-campus locations such as a home or coffee shop, and multiple ad-hoc learning sessions may exist on the same physical network. Due to the unroutable nature of the Zeroconf broadcast messages, even an unconfigured collection of devices on the same wireless access point will be able to detect and connect to each other. The computer will search the network using Zeroconf for a supernode created by another learner in the same location. If one exists, the computer will discover it using Zeroconf and connect to it directly (see Fig. 13). Since there is no server authentication at all in this situation, there is no designated leader to an ad-hoc session and all learners may collaborate equally. This also means that there is no method of authenticating participants, to easy connectivity, however a simple session-wide password system is feasible to protect the location-based network and the shared information within. If the peer arbitrarily designated as the supernode must disconnect from the network, the other nodes detect the loss of connectivity and a randomized re-allocation of the supernodal status occurs automatically.

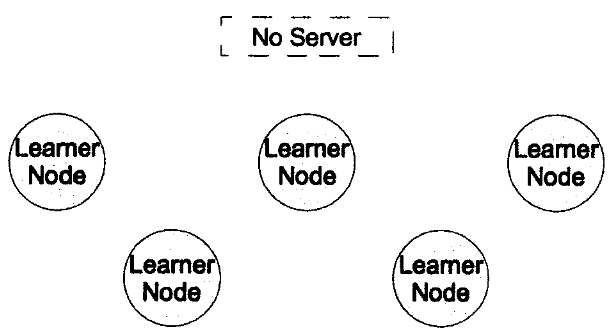


Figure 12: Ad-Hoc Session Without a Server

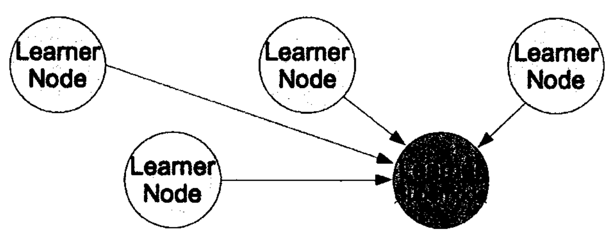


Figure 13: Random Node Assigned as Supernode

Ad-Hoc sessions are especially useful for group work, collaborating on projects, or in group study sessions. The system has no means of authentication or configuration, so all of the interactive aspects of the system remain unlocked. Lecture data is shared from any of the learners to the supernode, which then distributes it to the rest of the peers. This allows the lecture notes to be shared to a learner who was absent for a session, for example. Users notes, usually personal, may also be shared between users. The notes are collaborated into one document, maintaining all of their metadata and slide-based organization. The learners are then able to review their collaborated notes together, along with the relevant slides, providing a comprehensive account of the information obtained in class.

4) Web-Based Realtime Communication

The system is designed to implement as much of the functionality for a regular, non-supernodal peer as possible using web technologies. In fact, all of the interaction that a learner has with the system is done via an embedded web browser frame within the application. The system uses concepts and technologies developed for rich web applications in a traditional internet-wide client-server system and applies them to a local network-based peer-to-peer architecture. This is a novel approach to application development, and it provides some unique benefits and challenges. On top of that architecture, the system uses several modern web technologies that allow for a smooth user experience and rich information management.

The basis of the system is the integrated in-application web server, a feature which can be enabled and disabled by the software freely depending on the requirements. This radically changes the current architecture of the web, which is designed around centralized servers hosting applications and content. Relatively simple clients, such as web browsers, make connections to that central server and interact with its hosted content. This system radically changes that model, and adapts the fundamental web system to a P2P architecture. When a learner launches the application, a series of tests and connection attempts are made (see Section B. Infrastructure for Flexible Learner Interaction for more information about the service initialization and discovery process). If there is no existing supernode found, the peer activates its internal web browser. The primary Mac OS X version of the application uses the CocoaHTTPServer [30] to embed web serving capabilities into a regular application, providing a simple Object Oriented method for starting and stopping the service capabilities. Similar things can be done in Windows-based object oriented languages like C# (for example, SimpleHTTPServer [37])This is a paradigm shift from traditional web servers like Apache [38], which require system-wide setup and configuration. This is an important change, since a web server without any system-wide setup can function invisibly to the user, without modifying his/her configuration at all. The result is a complete learning session hosted on a server that seems invisible.

To provide communications between learners, web technology developed for interactive web applications provides impressive groundwork. The entire system is based primarily on AJAX, which stands for Asynchronous Javascript And XML. AJAX is a fundamental concept that revolves around the separation of a static page and its dynamic content. AJAX sites instantiate a javascript XMLHttpRequest object with certain parameters describing its content and destination. These parameters may include: the type of request (GET, POST, etc.), the passed variables and their names, the destination server, and most importantly the "onreadystatechange" function. This request is then transmitted to the destination server (or, in the case of this system, the local supernode) where data is processed remotely. Since AJAX is an asynchronous system, local javascript is free to run and the learner may still interact with their computer while the request is pending. When the results are ready, the server returns them and the function defined by onreadystatechange is executed. This function can read the data from the XMLHttpRequest response properties and, using additional javascript, modify the contents of the page without reloading the entire page. AJAX is important for web applications because it allows information to be stored in the page's current state (this state is maintained in the Document Object Model space, or DOM). Only the parts of the page that require updating are changed, and the rest is maintained.

AJAX is an impressive technology for changing the contents of a page, or transferring data for a web application, on-the-fly. It still, however, requires that the information request be initialized by the user. In a system designed for realtime synchronization of information, a push-type transmission is more beneficial than a fetch-type. The system uses a novel variation on AJAX that provides push functionality without any additional overhead. The concept is known as "long polling", or more distinctly as "Comet" [39] since its behaviour mimics that of a comets long elliptical orbit. On the peer node side, a Comet request functions identically to a normal AJAX request. Thanks to the asynchronous nature of AJAX, even a long delay between request and response is feasible without degrading the performance or blocking the execution of other javascript functions. On the supernode side, several things are happening. Firstly, the supernode is configured to spawn a new thread for each connection made. Not only does this allow the system to scale to various multi-core systems, but it also allows individual threads to be slept without interrupting the rest of the system. The Comet request thread checks the values of all of the push-worthy data items. It then runs a sleep loop on the connection thread until one of the push-worthy data items changes. To recap, while the data remains unchanged, the peer node is free to function as normal while still waiting in the background for the return of the Comet data. The supernode sleeps idle threads, eliminating their resource expenses, until a time when some push-worthy data is changed. These are two key characteristics to the Comet system as they minimize or eliminate the strain on any node or supernode until the point when the systems are refreshing their data content. Once a piece of data is changed, the response is assembled by the supernode and returned to the learner's node. Local javascript on the learner's side parses the returned data and modifies the contents of the page to reflect the new data. All of this may take place over a period of seconds or minutes, depending on the rate of the change of push-worthy content like slides, chat, and questions.

5) Device Connectivity and Integration

Existing technology for learning spaces includes stationary computers and projection systems. While the described system can create a complete learning environment without any of these existing systems, it is designed to work together with them. A classroom computer will have a static location identifier. When activated, it connects to the local supernode. In a lecture situation, there is likely to be a computer connected to one or more projectors. This computer will automatically connect to the supernode created by the instructor's computer. Using the same minimal interface mode that the instructor uses, the projector displays the lecture notes and slides. Just as the slides are synchronized to the learners in the class, the slides are also synchronized to the projection system and displayed in real-time. This gives the lecturer the freedom to use their own computer without the hassle of setting up the projection system. As well, it makes it possible for any participant to lead the lecture. Once they are given control of the learning space, the system distributes their resources and slides to the other learners, and to the in-class projector, in real time. This type of "handoff" may be useful for informal learning sessions, where learners would want the ability to display their own content on the main projected screen and to share it with the other learners.

This sort of automatic client connections can also be extended for laboratory spaces, or other locations where interfacing to special hardware is required. A lab computer connected to the specific machinery can act as a static peer and provide resources to learners in the same learning space. A software link must be created between the learning software and the equipment (via a plugin or utility). Then, much like the projector, the static peer can connect to an existing area supernode. In a lab situation peers are given privileges to contribute information, so the stationary computer is able to send its information directly to the learners. Not only does this provide the advantages of the projector system, in that learners are able to use their own computers without making a physical connection, but it allows the technology to be easily taken advantage of. The lab hardware is already connected and communicating with the static lab computer, no complicated setup is required to utilize that equipment.

The system also allows for integration with mobile devices, for situations when a full computer may not be practical. Since the entire system is based off of web standards like HTTP and XML, any mobile device with a web interface is technically capable of interacting with lesson information, recommendations, interactions, and everything that the system has to offer full computer users. Without the IR-based location awareness, the automation of the system is limited, but the functionality remains for advanced users who are comfortable with technology. In addition to standard web technology, all of the frameworks and components used in the Mac OS X version of the learning client are portable to the iPhone OS for instance. This means that a complete fully-functional peer, including the ability to act as a supernode, is possible for users of the iPhone and iPod Touch. For example, with the introduction of iPhone OS 3.0 in the summer of 2009, allocations for third-party hardware make IR location detectors viable [40]. It would be possible to create a location sensor for the iPhone and have a complete, fully-functional learning enhancement device. A professor may come into a lecture with only their smartphone and have the ability to conduct a technologically-enhanced lecture to hundreds of students. These capabilities would not be possible without the consideration of standard web and networking technologies, and really showcase the power and potential of this learning technology.

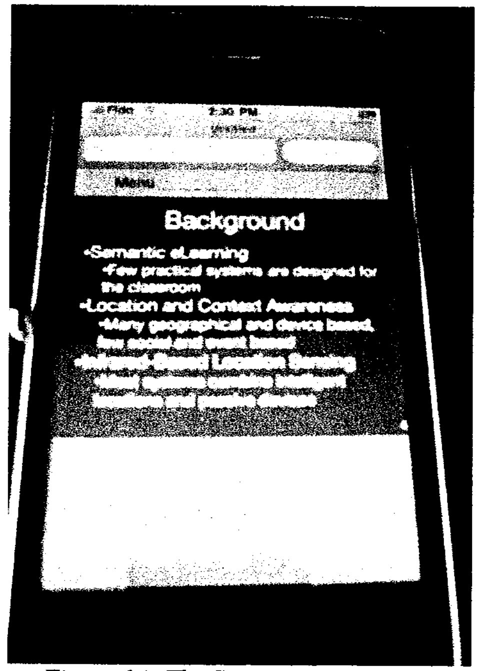


Figure 14: The System Running on a Mobile Client

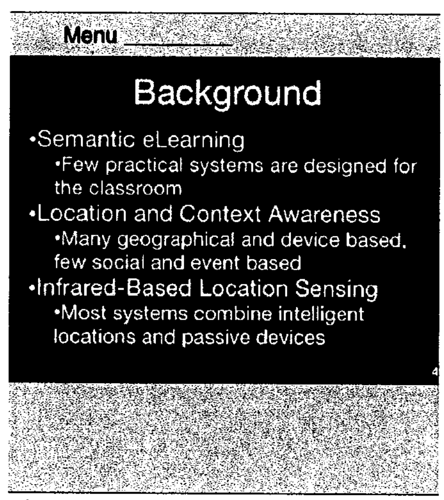


Figure 15: The Interface Resizes to Fit the Device

C. Resource Sharing and Collaboration

Once established, these automatically-configured networks can be used to provide advantages, enhancements, and assistance in any learning environment. The system is designed with two key adjectives in mind: seamless and real-time. The act of setting up the network autonomously and connecting learners of a similar context already puts the system above e-learning systems designed for use outside of the classroom in terms of seamless operation. The functionality of the system, once it is connected to its peers, demonstrates the importance of a real-time classroom system in enhancing the learning experience.

1) Real-Time Resource Sharing

The primary and most obvious learning resource for a one-to-many lecture situation is to provide lecture slides and related notes from the current lecture (see Fig. 16). It is common for professors to create a slideshow presentation to accompany their lecture. This is the same kind of information that students would normally obtain by navigating through a series of pages on the web or in an existing education system, if it is made available at all. In this learning enhancement system, learners are automatically connected to their professors by their common location and context. Notes and slides are transferred directly to learners at the time that they need them: right during the lesson. The system goes beyond this resource-delivery system, as well. Rather than simply delivering a slideshow file to learners, the system provides an interactive interface for displaying the slides as the lecture progresses, changing them in real-time and synchronized with the session leader. The professor imports their existing slideshow files into the system, and the system converts the slides into a series of images, capable of being delivered and displayed as part of an HTML file. When the session commences, the slide resources are sent to the supernode, where they are pushed to the connected learners incrementally using AJAX Comet techniques over a standard HTTP connection. By using standard HTML, it is possible to style and modify the content using Cascading Style Sheets (CSS). This allows the display of information to be variable and platform-dependent, while the content and functionality of the system remains platform-agnostic. In general, the system provides course content just as well as other education content management systems. It goes beyond these systems, however, but providing the content in real-time and automatically, pushed to learners right when they need it. This makes the presence of technology in the classroom even more useful than ever, while simultaneously simplifying a task that, without this system, takes time and attention away from the content itself.

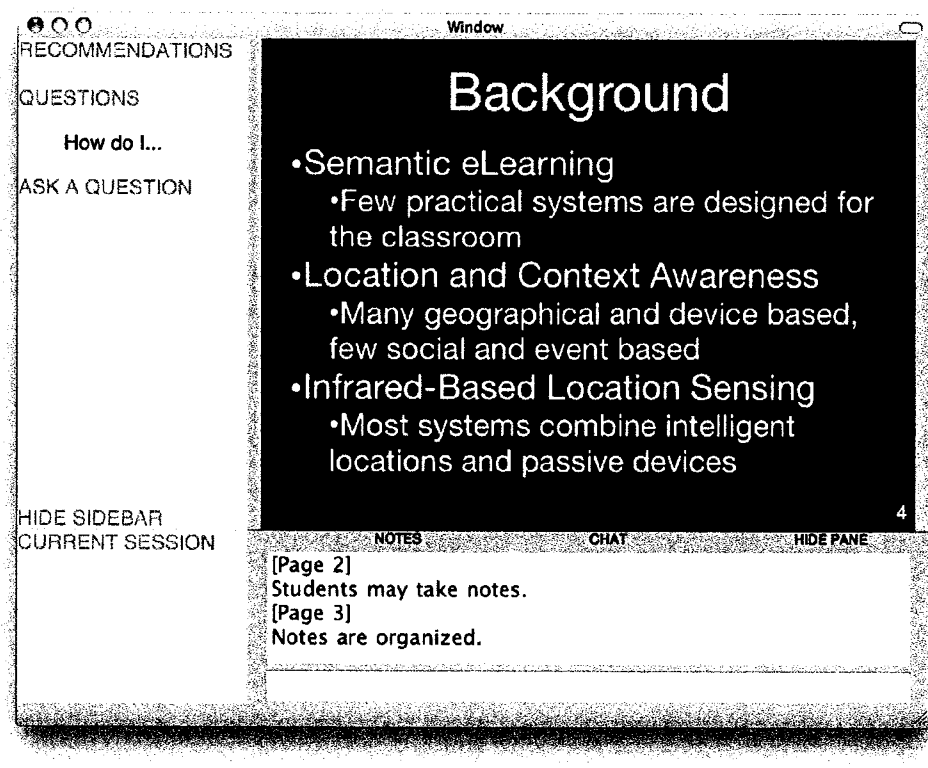


Figure 16: Standard User Interface

2) Personal Notes and Information

With lesson content delivered and displayed in real-time, there are new and innovative ways that this information can be used to enhance the learning experience. As the lesson progresses, learners have the ability to make personal notes. These notes are entered into the same web-based interface that displays the rest of the lesson content. The notes are stored locally in memory, and each note contains contextual metadata. When a note is taken, it is stored with the time of the note as well as the currently displayed content and slide (see Table 1). They are displayed with their relevant slide, and are organized to follow the flow of the lesson notes. This provides several advantages. Primarily, when reviewing notes at a later time, it allows the learner to view their own notes alongside the professor-provided lesson notes and slides, providing a comprehensive and context-rich set of personal notes for the learner. This limits duplication of notes taken by the learners, allowing them to concentrate on the lesson as it occurs rather than transcribing notes that are already provided by the professor. When a session finishes, the notes are stored to the local storage, and (depending on the configuration and the session mode) notes may also be synchronized to the central server. This method of semantic, comprehensive note-taking is one of several ways that the system is designed to use technology to simplify the learning process.

Table 1: Notes Stored in XML Format with Context Information

XML Data	<note> <title>An example note</title> <content>Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua.</content> <context> <created>20090501100230</created> <course>ENGI3050</course> <lecture>5</lecture> <resource> <file>chapter5.pdf</file> <page>15</page> </resource> </context> </note>

3) Learner Interactions and Questions

While much of the utility of the system comes from the professor and to the learners, facilitating learner to instructor interaction is also very important in the design and development of a learning enhancement system. Bidirectional communication opens up the possibilities for even more interaction and comprehension during a learning session, and increases the engagement of the learners in the course content.

In addition to taking personal notes, learners may submit questions to the session leader. Questions may be submitted with the learner's name, or they may be submitted anonymously. Anonymous question submission helps to alleviate nervousness and hesitation amongst shy learners, and promote interactive learning for everyone in the class without social pressure. These questions, like the personal notes, are encoded in

XML format and include metadata about the time of the question as well as the related lecture slide. When a question is submitted, the session leader receives a subtle indication that there are questions waiting for them (see Fig. 17). Depending on the teaching style and personal preferences of the professor, they may choose to address the questions immediately or to wait until an appropriate point in the session and avoid disrupting the flow and continuity of the lesson. If they decide the latter, the process of answering questions is greatly improved over traditional, non-enhanced learning sessions. When the session leader selects a question to address, the slide that was visible during the submission of the question is shown along with the question itself. This provides context for the question to help the instructor fully understand what is being asked as well as helping them answer the question by proving the relevant information.

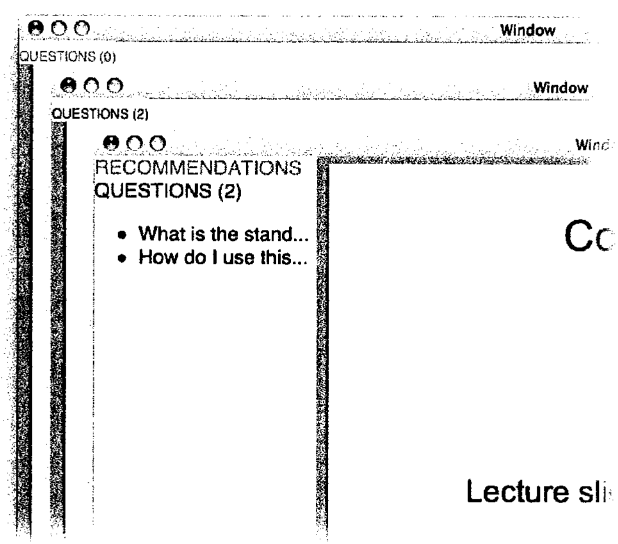


Figure 17: The Question Process

4) Peer Collaboration

The learning enhancement system provides the greatest advantages in sessions where information should be shared not only from a single instructor to many learners, but also when learners are participating and sharing information with each other. These kinds of interactions are especially useful in Informal and Ad-Hoc learning sessions where, without an assigned leader, learner collaboration is especially important. Learners may collect information and participate in a real-time chat with others within the same session (see Fig. 18). Much like the questions and personal note systems, chat data is also tagged with context about the time and current visible slide, or other content that is being displayed in the main content pane. This sort of data can be used to assemble a complete outline of the learning session, including all of the comments and information shared between the participants with reference to the time at which they were shared. While the physical proximity of the learners may make this seem redundant (after all, the participants are, by design, all in the same location), the ability to share data, URLs, and other text snippets with participants can make the system even more powerful. Additionally, recording information along with the main content is useful for later review, providing a comprehensive account of the learning session. With all of this collaboration and group participation capabilities, even private study session can be enhanced by a context-aware learning enhancement system.

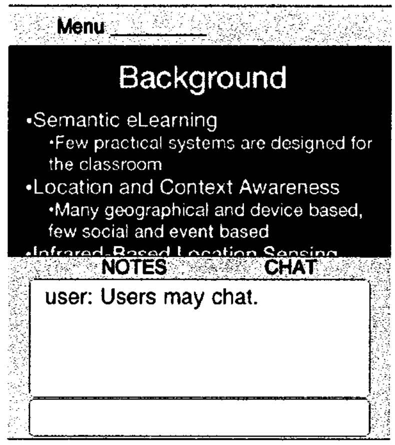


Figure 18: Chat Interface As Displayed on a Mobile Device

D. Learning Task Recommender

The system is designed to provide a more technologically-rich learning environment, but it also takes the experience beyond enhancing a lesson, and offers additional information to enhance an entire learning career. It does this by offering suggestions and related information via a sophisticated rule-based engine. This system runs on the central server of the institution, and combines information from individual courses and lectures into an overall learning ontology. When a professor or administrator registers a course with the system, he or she may include metadata and keywords to classify the course. Whenever a lecturer or tutor first connects to authenticate as a session leader with the central server, the lecture slides and lesson information are uploaded to the server and categorized automatically, based on the course and the lecturer's profile. This categorization is all done using established ontological standards, such a CC/PP Document standard [20]. This standard allows for the inclusion of the data required for categorization. Finally, as learners connect to a session, their context information is used to update their profile on the central server as well. These profiles are also stored in a CC/PP standard, the standard for Consumers.

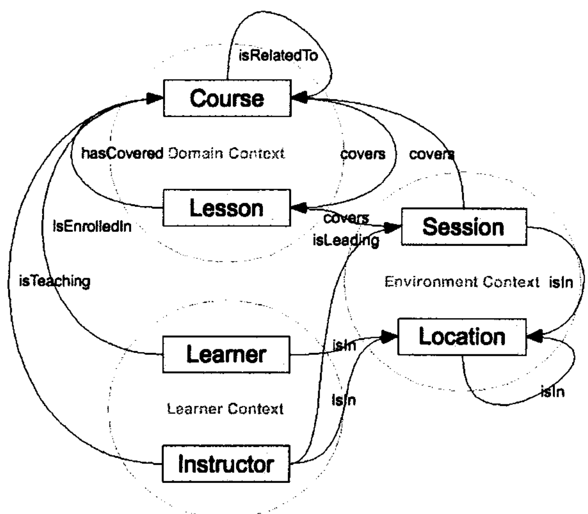


Figure 19: Simplified Learning Ontology

This information is all evaluated using a series of SWRL semantic rules by the server, and relationships between courses, lectures, and learners are inferred (see Table 2). During a lecture, or at any time when a learner is accessing the system, he or she may request additional information and recommendations. Using their lecture history and the profile created by the system, the central server evaluates the recommendation rules and creates a list of recommendations of additional information, lectures, and courses. This information can be specific to a lecture or note that the student is currently accessing, or more general course-level recommendations for planning an academic career. Due to the system's focus on in-class technology, it has a major advantage over other systems by including the specific academic profile of the learner, down to the individual session level. This lets the system provide the best related information based specifically on a learner's experiences.

Table 2: SWRL Rule Determining Learner Location

SWRL Rules	Learner(?a) ∧ Location(?b) ∧ Session(?c) ∧ Lesson(?d) ∧ Lesson(?e) ∧ Course(?f) ∧ isIn(?a, ?b) ∧ isIn(?c, ?b) ∧ covers(?c, ?d) ∧ isRelatedTo(?d, ?e) ∧ covers(?f, ?e) ∧ isEnrolledIn(?a, ?f) → isRecommended(?a, ?e)

Once formed, the recommendations are sent to the learners using their standard XML formatting. This is one of the areas that the web-standards based system has a large advantage over similar technologies. Communications with the recommendation engine are client-server, and the information is delivered in RDF-formatted XML, however the local clients are able to style that information and include it in the interface without any conversion or computation. All of the data received by the learner's computer is web standard data, and easily combined into one homogeneous interface.

These recommendations and profiles bring the described system beyond the traditional learning environment and really show the power of including technology in every aspect of the learning process. Since the system has profiles built for courses and students, providing recommendations provides a greater enhancement to the overall learning experience while maintaining the focus on automation and ease-of-use.

IV. Case Studies

A. University Lecture Hall

The most obvious usage for this system is in a university lecture hall. Many university students have their own laptop computers, and their computers become location-aware with the addition of a small USB infrared receiver. Upon entering the lecture hall and launching the student software, the computer uses the IR receiver to detect its current location. Ideally, the student will already have their user ID information stored. The client sends a request with this location and user information to the school server, which also has information about the lecture schedule. The server detects that a scheduled session exists for that student in that location, and instructs the student's computer to act as a standard node. The student's computer uses Zeroconf to connect to the professor's computer, acting as the location's supernode. As the professor leads the class in the lecture, the students are able to follow along on their own clients, as well as on the projected screen of the professor's client. If one of the students has a question, they can simply press a button on their own client and submit it to the professor without interrupting the lecture. A small indicator appears on the professor's screen, and the time and presentation slide number when the question was submitted is recorded. The professor may chose to answer the question immediately, or return to it later in the lecture. When the lecture finishes, the professor indicates the dismissal on the client, and the server records the session's progress. The material covered is marked as completed for the course, so that the next lesson is automatically selected at the beginning of the next lecture. The profiles of the present students are also updated, so that the server can provide for them accurate recommendations in the future.

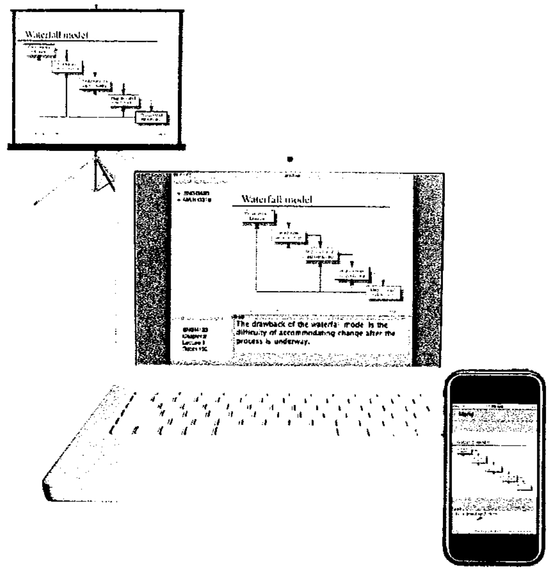


Figure 20: Example Lecture Setup

B. High School Lab

A high school science lab is another excellent application for this ubiquitous learning technology. In a high school situation, students are less likely to have their own portable computers, so the school could offer laptops already equipped with location detection devices for the students to use. In a lab situation, it makes sense for the class to be separated into logical groups. This means that the lab could be separated into different locations with different infrared identifiers for each workstation. The infrared beacon would be situated so that it illuminates the workstation itself, and any IR-aware systems placed on the workstation would join the small workstation network. The system connects to the central server. This server determines that there is no scheduled leader for that particular workstation location, and assigns the first active student to be the supernode for that location. The server also retrieves specific information about the current lesson or lab that is assigned to that specific workstation. As the system knows that there is no assigned leader, all peers are treated as collaborators, and the supernode allows input from all students equally. Students in the same workstation subnet can record data on their own clients, and the data from all of the clients at a workstation will be combined automatically, keeping lab information co-ordinated and organized. At the end of the lab session, the teacher sends the dismissal signal to the server. The server saves the lab data, linking it to each of the participating students' profiles. It updates the lesson data, course data, and student experience data for future recommendations and processing, just like the lecture situation.

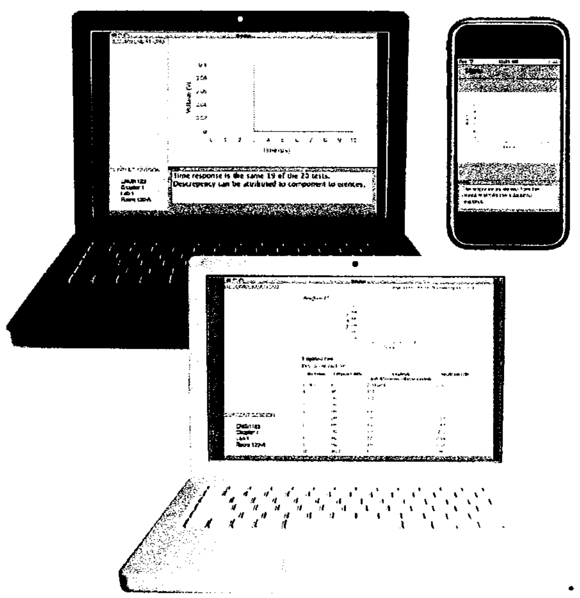


Figure 21: Example Lab Setup

C. *Office Meeting*

This system is very versatile, and although it is designed for school environments, it will work just as well in many different situations (for example, managing information in meetings in an office environment). Currently, when employees gather in a conference room with their laptop computers, those systems are not any more associated than when those employees are anywhere else in the building, even though they are physically in the same room. By equipping these machines with infrared receivers, and setting up one or more location beacons to cover the conference room, the systems now have the ability to associate with each other. Without a central server, this system will act as an ad-hoc network. The first peer to fail to detect the server will allocate itself as the location supernode. Other peers will detect the supernode, know that it is for the location based on its advertised location information and their own IR location identifier, and connect. Since ad-hoc systems have no way of determining hierarchy or authentication, all peers in a meeting are treated as equals, much like the lab situation examined above. That way, anyone who has an idea to share may do so, and all of the others in the room can see it on their own screen. When the session is concluded, the data is saved by each computer on its own local storage. Participants may revisit this data in the future on their machine, and review their own notes taken during the meeting along with any shared information from others.

D. Medical Care Facility

Another example of an alternative use for the system is in a medical situation, such as a hospital or care facility. Medical professionals can be issued handheld computing devices, complete with infrared location detection hardware, and patient rooms are equipped with infrared location beacons (one per bed). Infrared technology is benign and causes no interference to medical equipment, in contrast to some RF-based location technology which might cause issues and complications. Upon entering a patient's location, the doctor or nurse receives a patient profile from the central server, and an Informal session is created around their handheld device. If wireless networking is also restricted, due to the RF radiation, the system may simply use this location data to retrieve patient information from a local, synchronized database on the device itself. This patient profile would contain all of the medical information that is normally contained in physical files and on a patient's chart. Strict laws guide the protection of patient information, and the system uses modern web encryption and secure connections to ensure that patient profiles remain confidential. As well, location-based security measures such as local data encryption could be put into place to ensure that patient data is only available from within the medical facility. The creation of the Informal location-based session provides the facility to share and examine patient information, test results, and other medical data with any other doctors or nurses who may also be present at that patient's bed. This is one more way that the information sharing and collaboration capabilities, along with the location and context awareness, can expand beyond traditional learning situations.

V. Experimental Results

A. Student Response and Feedback

To test the validity, reliability, and utility of the system, a small focus group was set up in a classroom on the Lakehead University campus. A small group of students were given an opportunity to test the system and provide feedback in the form of a user survey. In addition, the system was benchmarked with a classroom full of computers, to test the strain that would be put on the individual computer that is selected to act as a supernode.

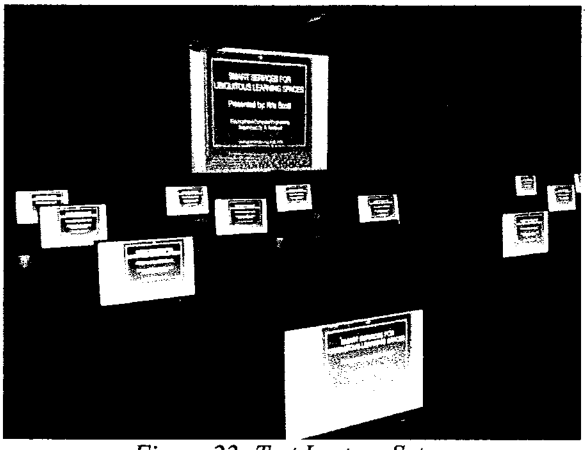


Figure 22: Test Lecture Setup

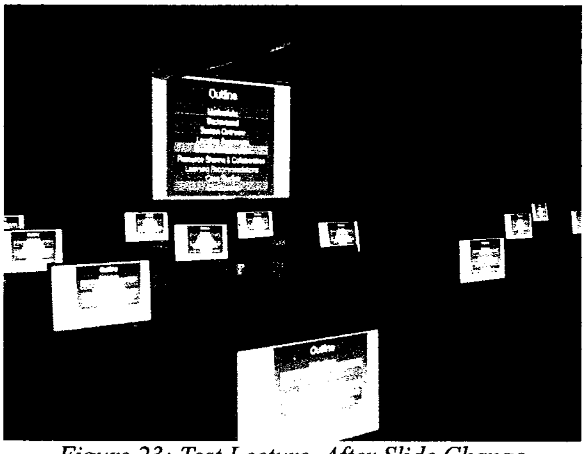


Figure 23: Test Lecture, After Slide Change

The focus group was given a questionnaire before testing the system, to gauge their experiences with classroom technology. All of the learners had participated in lecture-style learning before. On average, the group found lecture learning to be satisfactorily useful, however most found lectures could be more enjoyable. Making the learning process more enjoyable for students is a big advantage for conveying information, as it keeps the students interested in the session. More than half of the group had used remote e-learning solutions before, and they found them to be, on average, about as useful as lectures, however slightly less enjoyable. A physical learning experience is often more engaging than a remote learning experience, and engaging students is a primary goal of the described learning enhancement system. Finally, every member of the group had participated in a learning session in a "smart classroom", indicating technologies such as cameras, microphones, multiple projectors, and internet access. Although the group agreed that the technology was not too difficult to use, they also indicated that the professors do not actively use the technology available to them. A learning enhancement system for the classroom is only as effective as its use, and by making the system extremely easy, the described system is designed to increase the currently unsatisfactory use of technology in modern lectures and classes.

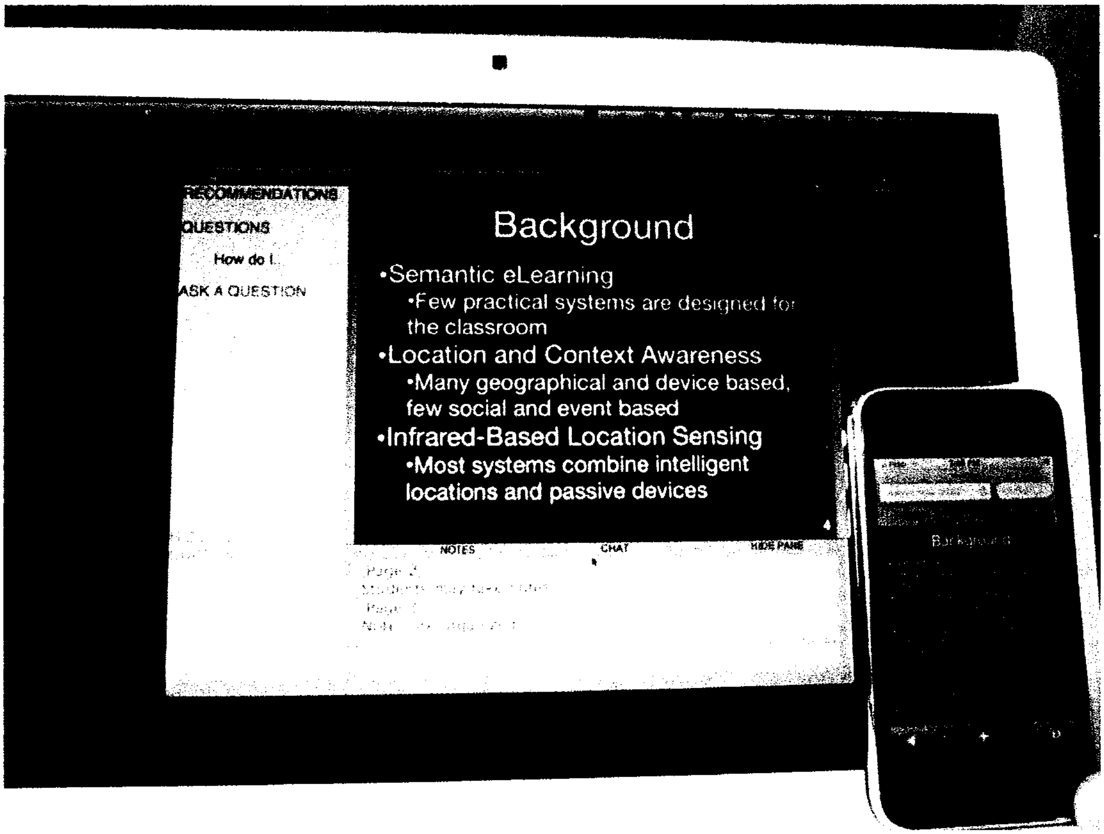


Figure 24: System Functioning on a Notebook Computer and a Mobile Device

The focus group indicated a strong need for smarter technology in learning spaces, with emphasis on student interaction, participation, and note-taking during lectures. They also voiced some concerns about technology being introduced into classrooms and lecture halls. There were some concerns about distraction and participation in the course content with the increased technological interaction, and about the under-utilization of classroom technologies. Our system is designed to decrease distraction and maximize adoption by minimizing the effort required to use the system. Automation lets the software configure and operate itself, allowing the users to focus on the content. One subject pointed out the importance of catering to different learning styles. According to [41] there are 3 main learning types: Cognitive, Affective, and Psychomotor. Cognitive learning promotes active participation and peer support. The system provides this with its question system. Affective learning concentrates on discussion of information in a group. The location-based chat provides this. Finally, Psychomotor enables learning by demonstration. The slide syncing feature allows the professor to easily lead the class in demonstrations, catering to this. One student also voiced concerns about security of the lecture content. By basing the system on web standards, it is possible to use standard web encryption to ensure that data remains protected. Ad-Hoc situations do not benefit from the authentication system of the other two configurations, however content changed while working in an Ad-Hoc session is not transferred back into the system; course data only flows from professors and session leaders to learners.

The focus group was then given a sample session with the learning enhancement system, and asked some questions about their reactions. Firstly, the entire group agreed that the system primarily functions as a way to keep students engaged with the course content while minimizing the amount of tedious note-taking. The majority of the group also agreed that the context-aware note system is the best feature of the system, providing contextual information for learning sessions. They were more than satisfied with the ease-of-use of the system, and the group agreed that the system supplies the course content in a way that is not distracting.

The synchronized content slides and context-aware notes were both found to be the most useful features of the system, with the question submission system also very highly rated. The least useful, as defined by the group, is the location-based chatroom. The chat function is not considered to be a vital feature in situations where the participants are in the same room, as vocal communication is more direct and more convenient, posing less of a distraction. The chat function is designed to be an additional means to share text-based content with a group, and may prove to be more useful in smaller groups and sessions than in the test-case lecture situation.

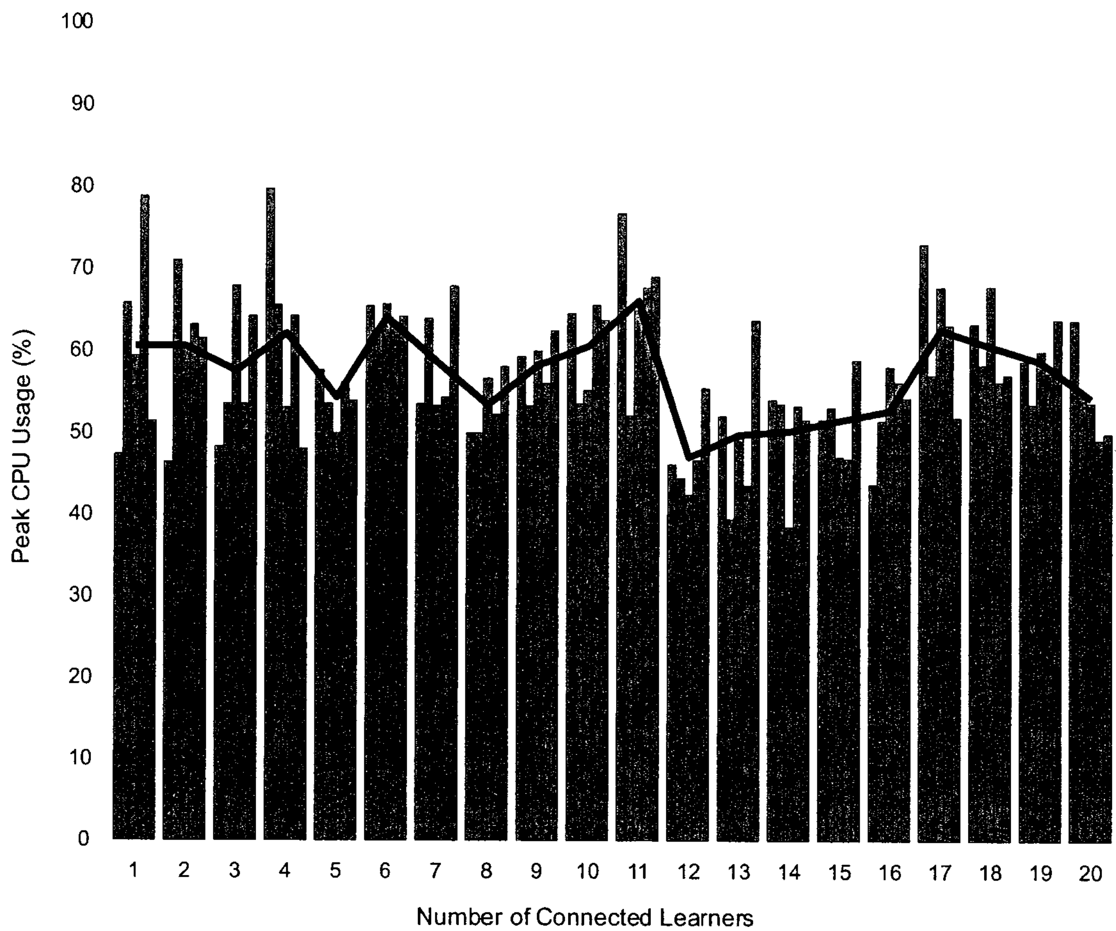
Of the three main learning session types that the system is designed for (lecture, lab, and group study), the focus group found the lecture situation to be the most useful, and the lab situation (where work is often individual) to be the least. Lectures and learning sessions that include large numbers of students show the most potential for technology, as they are the situations where everyone in a location is working on the same content at the same pace. Interestingly, group study was considered to be just as compelling as a lecture in terms of the use of the system. The group was interested in the potential applications of the system in non-classroom learning situations that still maintain a physical proximity, such as private study groups and collaborative projects.

Overall, the response to the system was very positive. The group could easily see the need for improved classroom technology, and could appreciate this learning enhancement system and the solutions that it provides.

B. Supernode Scalability

In addition to these user feedback test, some technical tests were conducted to investigate the viability of the system on a class-wide scale. The largest concern for the system is the scalability of the the supernodes. Acting as a peer-server for a given location, supernodes have to be able to maintain connections to all of the other peers in a session. It is important to ensure that the addition of additional users does not jeopardize the functionality or response of the technology. Doing so would distract from the utility of the system.

A supernode was set up in an Ad-Hoc session. The first machine to connect to the network was a 2GHz MacBook running the learning enhancement software. Being the first connector, and not detecting an active server, the supernode functionality was activated in the software. At this point, an increasing number of clients were connected, and 5 standard slide-changes were performed at each point. This data was recorded using Instruments, a benchmarking and analysis tool that is included with the Apple XCode IDE [42], and the results can be seen below. Note that this is peak CPU usage, at the very point when the slide is changing for every peer simultaneously.



25: CPU Usage with Increasing Connections

There is no appreciable trend in CPU usage as the number of users increases. With a greater number of peers, the peak processor usage still remains within an acceptable 60% range. As tested, the system is configured to push updates to peers at a maximum rate of 0.25 seconds per peer. That means that any given peer is never more than 0.25 seconds out of sync with the session leader. With all 20 peers connected, some experimentation was done to modify this update cycle, however there was no significant change in the overall performance or system utilization with this update change. It would be possible to configure this parameter to adjust with the number of connected peers, if a greater number of clients (hundreds) caused the system to slow down, but for a general 20 student class session, the system performs without any delay.

The system is able to scale so well due to its significant use of threading. Every peer connection is allocated its own processor thread, segmenting the processor load into small pieces. Threading allows the system to take advantage of multiple cores and multiple processors. As personal computers become more advanced, this type of application will scale easily to take advantage of the CPU power efficiently. As well, these threads are well managed. Threads are idle until new information to push becomes available. Sleeping the threads minimizes their CPU load, decreasing the overall utilization. At the point when some information is changed, the threads re-activate, push their information to their respective peers, and terminate. A new thread is started for by each peer every time the previous one is terminated, and this new thread is again slept until it is needed. Functioning this way not only minimizes CPU load, but also decreases network traffic, thereby increasing overall system efficiency.

VI. Conclusions and Future Work

Despite advances in mobile, interactive, and ubiquitous technology, the adoption of instructional technology and learning aids in traditional learning spaces has been lackluster. The system described in this thesis is designed from the ground up to be affordable to implement, easy to operate, and most importantly, it is designed to keep the focus on the learning material by offering flexible and efficient learning interactions. It includes a cost effective room-level location detection system based on infrared light. This provides context awareness and allows the system to dynamically adapt based on its surroundings. The proposed learning infrastructure is based on a versatile and automated network architecture which can function both with a central server and completely independently on a peer-to-peer communication basis. Finally, the system provides a variety of information sharing, collecting, and collaboration capabilities based on semantic web technology. It is designed to enhance traditional learning situation with modern instructional technology including handheld devices. The proposed system has the potential to expand beyond indoor learning space through the use of mobile technology and other communication systems, thus providing a completely comprehensive learning experience.

This learning enhancement system was designed to provide a truly useful, practical, and realistic way to incorporate technology into traditional social learning situations. In doing so, however, it also facilitates the expansions of the learning experience beyond these spaces. There are several ways in which the system could be expanded to enhance other parts of the learning experience, based on content and information collected for and collaborated during learning sessions. One such way is in the expansion of lesson plans and event schedules. As instructors and administrators plan tests, assignments, and other course milestones, they may input them into the system and have them uploaded to the central server. Using semantic reasoning, the system can notify learners of these upcoming events when they log on to the system. At this point, the learners may opt for external reminders of upcoming deadlines. These reminders may include emails or SMS messages to mobile phones. Reminder events may be triggered by the time and date on the central server, and it dispatches the notices to all of the participants to which they apply. Since all of this information is stored in the same server that contains the usual class information, it is all semantically linked. Reminders can contain links to related classroom information, and they will be specific to that learner's profile and experience. By integrating these reminders into the same system that learners use in class and while studying, the entire system remains unified and it makes it much more likely that the reminders will be used. This is merely one example of the advantages of integrating technology into the learning experience at every level.

In conclusion, the system succeeds at its goals to incorporate technology into the classroom in a way that is both seamless and comprehensive. In doing so, it helps to address the issues with ubiquitous technological adoption in traditionally non-technical situations as well as providing practical solutions to these issues. This work can provide a basis with which to finally create smart learning spaces that will actually be used by learners and instructors alike.

References

[1] C. Johnson and C. Lomas, "Design of the Learning Space: Learning and Design Principles", *EDUCAUSE Review*, 40(4), 2005, pp.16-28.

[2] M. Ratto, B. Shapiro, Troung, and G. Griswold, "The activeclass project: Experiments in ecouraging classroom participation," Computer Supported for Collaborative Learning, 2003. [Online]. Available: http://www.cs.ucsd.edu/~wgg/Abstracts/activeclass-cscl03.pdf [Accessed: May 20, 2009].

[3] S. Graf, K. MacCallum, T. Liu, M. Chang, D. Wen, Q. Tan, J. Dron, F. Lin, N. Chen, R. McGreal, Kinshuk, "An Infrastructure for Developing Pervasive Learning Environments", Proceedings of the Sixth Annual IEEE International Conference on Pervasive Computing and Communications PerComp'08, Hong Kong, March 17-21, 2008, pp.389-394.

[4] S. Kajita, K. Mase, S. Jang, M. Ueda, Z. Yu, and N. Lin, "uClassroom: Expanding Awareness in Classroom to Ubiquitous Teaching and Learning Using Eclipse RCP", *EclipseCon 2007*, Santa Clara, California, USA, March 5-8, 2007.

[5] K. Kim, "Ubiquitous Learning Supporting System for Future Classroom in Korea", in K. McFerrin et al. (Eds.), Proc. of Society for Information Technology and Teacher Education International Conference, Las Vegas, USA, March, 2008, *pp. 2648-2657.*

[6] V. Jones and J.H. Jo, "Ubiquitous learning environment: An adaptive teaching system using ubiquitous technology", Proc. of the 21st ASCILITE Conferences, Perth, Australia, December 5- 8, 2004, pp.468-474.

[7] H. Peng, C. Chou, and C.Y. Chang, "From virtual environments to physical environments: Exploring interactivity in ubiquitous-learning systems", *Educational Technology and Society*, 11(2), 2008, pp.45-66.

[8] A.J. Milne, "Entering the Interaction Age: Implementing a Future Vision for Campus Learning Spaces", *EDUCAUSE Review*, 42(1), 2007, pp.12-31.

[9] H.J. Strauss, "The Future of the Web, Intelligent Devices, and Education", *EDUCAUSE Review*, 42(1), 2007, pp.32-37.

[10] C.C. Van Den Blink, "Uses of Labs and Learning Spaces", *EDUCAUSE Quaterly (EQ), 32*(1), 2009, pp. [Online]. Available: http://www.educause.edu/EDUCAUSE+Quarterly/EDUCAUSEQuarterlyMagazine Volum/UsesofLabsandLearningSpaces/163858 [Accessed: May 20, 2009].

[11] E. Guttman, "Autoconfiguration for IP networking: enabling local communication", *IEEE Internet Computing*, 5(3), 2001, pp. 81-86.

[12] J. Barbosa, R. Hahn, S. Rabello, and D. Barbosa, "LOCAL: a Model Geared Towards Ubiquitous Learning", *ACM SIGCSE Bulletin*, 40(1), 2008, pp.432-436.

[13] Y. Rogers, S. Price, C. Randell, D.S. Fraser, M. Weal, and G. Fitzpatrick, "Ubi-learning integrates indoor and outdoor experiences", *Communications of the ACM*, 48(1), 2205, pp. 55-59.

[14] Y.E. Shih, and D. Mills, "Setting the New Standard with Mobile Computing in Online Learning", *The International Review of Research* in Open and Distance

Learning, 8(2), 2007.
[15] T. Wu, T. Yang, G. Hwang, and H. Chu, "Conducting Situated Learning in a Context-Aware Ubiquitous Learning Environment", Proc. of the Fifth IEEE International Conf. on Wireless, Mobile, and Ubiquitous Technology in Education, Beijing, China, March 23-28, 2008, pp. 82-86.
[16] H. Ogata, "Computer Supported Ubiquitous Learning: Augmenting Learning Experiences in the Real World", Proc. of IEEE International Conference on Wireless, Mobile, and Ubiquitous Technology in Education, Beijing, China, March 23-28, 2008, pp. 3-10.
[17] T. Chen, C. Chang, J. Lin, and H. Yu, "Context-Aware Writing in Ubiquitous Learning Environments", Proc. of the Fifth IEEE International Conference on Wireless, Mobile, and Ubiquitous Technology in Education, Beijing, China, March 23-28, 2008. pp. 67-73.
[18] "Blackboard Learn Platform," Blackboard Inc. United States, 2009. [Online]. Available: http://www.blackboard.com/Teaching-Learning/Learn-Platform.aspx [Accessed: May 18, 2009].
[19] L. Horrocks, P.F,H. Boley, S. Tabet, B. Grosof, and M. Dean, "SWRL: A semantic web rule language combining OWL and RuleML", W3C Member Submission, 21 May 2004, http://www.w3.org/Submission/SWRL/#2 [Accessed: May 20, 2009].
[20] C. Kiss, "Composite Capability/Preference Profiles (CC/PP): Structure and Vocabularies 2.0," W3C Working Draft 30 April, 2007. [Online]. Available: http://www.w3.org/TR/2007/WD-CCPP-struct-vocab2-20070430/ [Accessed: May 20, 2009].
[21] N. Brachet and F. Alizadeh-shabdiz. "Methods and systems for estimating a user position in a WLAN positioning system based on user assigned access point locations." U.S. Patent 7 471 954, December 3, 2008.
[22] "What is GPS?," Garmin, Ltd., 2009. [Online].Available: http://www8.garmin.com/aboutGPS/ [Accessed: May 18, 2009].
[23] M. Sakata, H. Sasaki, M. Imura, Y. Yasumuro, Y. Manabe, and K. Chihara, "Active IR-Tag User Location System for MR Collaborative Environment," Nara Institute of Science and Technology, Japan, 2004.
[24] S. Strogatz and I. Stewart. "Coupled Oscillators and Biological Synchronization," *Scientific American*, December, 1993. Available: http://www.math.oregonstate.edu/~gibsonn/Teaching/MTH323-001S09/Supplements/coupled.pdf [Accessed: September 5, 2009].
[25] A. Tyrrell, G. Auer and C. Bettstetter. "Firefly Synchronization in Ad Hoc Networks." [Online]. Available: http://www.bettstetter.com/publications/tyrrell-2006-minema-firefly.pdf [Accessed: September 5, 2009].
[26] HID Protocol, "Device Class Definition for Human Interface Devices (HID)," USB Implementers' Forum. 2001. [Online]. Available: http://www.usb.org/developers/devclass_docs/HID1_11.pdf [June 1, 2009].
[27] F. Bellifemine, G. Caire, A. Poggi, and G. Rimassa, "Jade: A white paper," *EXP in Search of Innovation*, vol. 3, no. 3, pp. 6-19, 2003. [Online]. Available: http://jade.tilab.com/papers/2003/WhitePaperJADEEXP.pdf [Accessed: February

18, 2009].
[28] Resource Description Framework (RDF) specification, 2004. [Online] available http://www.w3.org/RDF/ [Accessed: December 3, 2008].
[29] R. Costello. "Building Web Services the REST Way." 02 August, 2002. [Online]. Available: http://www.xfront.com/REST-Web-Services.html [Accessed: April 28, 2009].
[30] "CocoaHTTPServer," Google Code. 2009. [Online]. Available: http://code.google.com/p/cocoahttpserver/ [Accessed: April 6, 2009].
[31] "Opera Unite", Opera Software USA, 2009 [Online]. Available: http://unite.opera.com/ [Accessed: September 13, 2009].
[32] D. Kellogg. "Plain Old Webserver." [Online]. Available: http://davidkellogg.com/wiki/Main_Page [Accessed: September 13, 2009].
[33] S. Cheshire and M. Krochmal. "Multicast DNS," Apple, Inc. September 2009. [Online]. Available: http://files.multicastdns.org/draft-cheshire-dnsext-multicastdns.txt [Accessed: September 10, 2009].
[34] "Bootstrapping – LimeWire," Lime Wire LLC. 2009. [Online]. Available: http://wiki.limewire.org/index.php?title=Bootstrapping [Accessed: September 19, 2009].
[35] "FAQ: Supernodes," Sharman Networks. 2006. [Online]. Available: http://www.kazaa.com/us/help/faq/supernodes.htm [Accessed: September 10, 2009].
[36] S. Guha, "An Experimental Study of the Skype Peer-to-Peer VoIP System", Cornell University [Online]. Available: http://saikat.guha.cc/pub/iptps06-skype/ [Accessed: September 10, 2009].
[37] J. Pobar, "Simple HTTP Server in C#," C# Help [Online]. Available: http://www.csharphelp.com/archives2/archive360.html [Accessed: April 6, 2009].
[38] The Apache Software Foundation. [Online]. Avalable: http://www.apache.org/ [Accessed: September 10, 2009].
[39] R. Gravelle, "Comet Programming: Using Ajax to Simulate Server Push," Webreference, WebMediaBrands Inc. [Online]. Available: http://www.webreference.com/programming/javascript/rg28/ [Accessed: August 4, 2009].
[40] "iPhone SDK 3.0 – Accessories," Apple, Inc. 2009. [Online]. Available: http://developer.apple.com/iphone/program/sdk/accessories.html [Accessed: August 28, 2009].
[41] J. Ross-Kerr, M. Wood, Ed., Canadian Fundamentals of Nursing, 3rd Edition. Toronto: Elsevier Canada, 2006.
[42] "Mac OS X Leopard – Developer Tools – Instruments, " Apple, Inc. 2009. [Online]. Available: http://www.apple.com/ca/macosx/leopard/developer/instruments.html [Accessed: September 13, 2009].