Chapter 19

PERVASIVE COMPUTING: ENABLING TECHNOLOGIES AND CHALLENGES

Mohan Kumar and Sajal K Das
The University of Texas at Arlington

Reducing the complexity of daily life and enhancing human quality of life have been two of the main objectives of computing and communication technologies. Pervasive computing has emerged as a significant research area that will herald the development of user-centric and service-oriented technologies. The Internet is one important step toward making pervasive computing a reality. Through the pervasive Internet, it is possible to access information and networked services anytime, anywhere. The rapid advances made in wireless mobile communications have provided an additional degree of freedom—**mobility**—to users of computing and communication services. The ubiquitous presence of embedded devices, wearable computers, sensor networks, and radio frequency identification (RFID) tags have also made it possible to deploy computing and communication nodes and services, thus enabling pervasive computing environments that aim at providing "what you want, when you want it, how you want it, and where you want it" kinds of services to users and applications. Several important challenges need to be tackled to realize the goals of pervasive computing. In this chapter, we give an overview of various enabling technologies and enumerate some of the challenges of pervasive computing. Several ongoing projects related to this topic are also discussed.

1 INTRODUCTION AND MOTIVATION

Recent advances in computer hardware (including embedded systems), communications technologies, mobile ad hoc and sensor networks, software agents, and middleware technologies have been mainly responsible for the emergence of pervasive computing as an exciting area of research with a wide variety of applications. Pervasive computing encompasses many existing areas in computer science and engineering, such as wireline and wireless communication networks,

mobile and distributed computing, embedded computing, agent technologies, middleware, situation-aware computing, and human–computer interfaces. Pervasive computing is about providing "what you want, where you want it, when you want it, and how you want it" services to users, applications, and devices. Pervasive computing paradigms can help meet the challenges encountered in myriad applications in almost all areas of human activity – military, security, transportation, healthcare and telemedicine, crisis management, manufacturing and maintenance, education, entertainment, and others. In pervasive computing environments, hardware and software entities are expected to function autonomously, continually, correctly, and often proactively.

Various enabling technologies such as sensors (e.g., UC Berkeley Motes Sensor Network Platform), Radio Frequency ID (RFID) tags, intelligent appliances, embedded processors, wearable computers, handheld computers, and cell phones will continue to play vital roles in improving human quality of life through the advancement of pervasive computing applications. Tiny intelligent sensors have made it possible to deploy ubiquitous services and thus create smart environments. RFID tags allow subtle integration of objects (e.g., commodities in a superstore or supply-chain management, mechanical objects on an industry floor) into the computing environment. The advances in Internet technologies have allowed us to access information and services in a transparent manner. Additionally, tremendous progress in wireless communications and mobile computing have made "anytime, anywhere" computing and information availability a reality. In the following section, we give an overview of the following technologies: (1) the Internet, (2) mobile and wireless communications, (3) sensor networks, and (4) RFID technology.

The advent of advanced technologies and their associated software tools have resulted in the emergence of several applications. Consider the following three application scenarios to understand where we are headed:

Scenario 1 [15]: A car-accident victim in critical condition needs immediate attention by medical and other personnel who are in geographically distributed locations. Timely and automated actions by ambulance personnel, doctors, and hospital personnel, and their effective collaboration, are essential to save the victim's life. Devices around the victim, such as a street camera, cellular phone, and pocket PC, collaborate to exchange sensory data, recognize the occurrence of an extraordinary event (in this case an accident), and contact an ambulance service. The ambulance, upon arrival, interfaces with the hospital, medical, and other personnel to accomplish the tasks required to save the patient reliably, efficiently, and in a timely manner. In order to accomplish the life-saving mission in this context, real-time collaboration must be established dynamically and autonomously.

Scenario 2: A soldier's personal digital assistant (PDA) contains information about the terrain, strategies, vital data, enemy positions, up-to-date commands from his commander, and shared data with peers. The PDA's connection to the wireless Internet is intermittent and not continuous. It is necessary to provide the PDA with the most relevant data all the time from nearby support stations based on the soldier's current position and the events happening around him. From time to time, the soldier may request new information or advice. The soldier (and his PDA) needs to coordinate with other soldiers (via their PDAs) as well as their command center (a PC or laptop).

Scenario 3: John Smith, a medical surgeon, takes his lunch at the cafeteria. While walking to the cafeteria, he makes notes on his handheld device about a patient he just visited. It is his habit to watch live basketball games and see highlights (and scores) of finished games on his handheld device while at lunch. At the cafeteria, he receives messages and vital information from other doctors, patients, students, and nurses. He also requests the patient records system for the latest patient histories. On some days, he consults remotely with his patients: he listens to sounds and examines images and data provided by remote consultation machines, patients, and nurses. On his walk back to the clinic, he watches his daughter practice soccer at school. All on his PDA!

The aforementioned scenarios use existing basic component technologies—laptops, handhelds devices, street cameras, cell phones, car computers, image and voice recognition systems, and so forth. But the required software and middleware to enable such applications seem difficult to implement. Researchers are still discovering new mechanisms and software/middleware paradigms to glue all these component technologies together. A careful analysis indicates that the above scenarios are based on several challenging technical requirements, including intelligent proactive services; guaranteed quality of service (QoS) and availability of communication channels; adequate authentication and security mechanisms; seamless interaction among heterogeneous entities; the presence of ubiquitous computing devices in the environment; and the like. Thus, despite the advances in hardware and/or communications-related enabling technologies, pervasive computing faces many systems issues and challenges that must be tackled. Exploiting available computing devices, communication technologies, and software services *all the time* and *everywhere* to enhance the quality of human life, ensure security, and utilize resources optimally is a key issue in making pervasive computing possible. To reach this goal, we must address research challenges such as (1) heterogeneity and interoperability, (2) proactiveness and transparency, (3) location-awareness and mobility, and (4) privacy and security. We will discuss these in Section 3 of this chapter.

Several projects in pervasive computing are under way in various universities and research laboratories worldwide. Section 4 discusses a few of these: the Aura project at the Carnegie Mellon University; the Gaia project at the University of Illinois, Urbana Champaign; the Oxygen project at the Massachusetts Institute of Technology; and the PICO and MavHome projects at the University of Texas at Arlington.

2 ENABLING TECHNOLOGIES

Recent years have witnessed significant progress on the technology front to improve human quality of life. The Internet, Due to its pervasiveness, is perhaps the prime contributor to this progress. Mobile and wireless communications have further made it possible to access and exploit Internet-based services anytime anywhere. Additionally, the emergence of sensor networks and RFIDs has enabled us to inject (or distribute) computing capabilities into objects (mechanical, biological, environmental, chemical, etc.) that were traditionally considered to be passive physical objects. The integration of these technologies has led to the

ubiquitous presence of computing elements and therefore the all-pervasive Internet and other network-based services.

2.1 The Internet

The Internet has indeed been a great revelation to application designers, service providers, business organizations, and individual users. The tremendous growth of the Internet is due to advances in (1) computer architectures, (2) communication networks, and (3) middleware and application software development. In addition, several technologies such as TCP/IP, Mobile IP, wireless access networks (such as GSM and CDMA) and optical communications, and MEMS (Micro Electro-Mechanical Sensors), as well as software and programming language initiatives such as Java, software agents, and middleware tools, are playing critical roles in the wide applicability of the Internet.

Today, in most homes (in the developed and developing countries), the Internet is considered to be an essential service, just like the television or telephone. For business organizations and industries, the Internet is as important as electricity or telephone service. Many of the applications and services we see today are based on the world wide web (WWW), which is a distributed repository of vast information. With a few exceptions, the Internet is mostly a source of static information that can be accessed on demand.

However, the Internet is not geared for handling dynamically changing information, and it is not a good model for addressing scalability, adaptability, and flexibility issues. Moreover, sustained collaborative interaction and performance (e.g., QoS) for the entire duration of an operation is required to meet the demands of many current applications in telemedicine, defense, transportation, manufacturing, and other areas that employ the Internet. The question is: *can the Internet model meet the demands posed by such applications*? Furthermore, in the current model of the Internet, all processing tasks are carried out at the edge of the network. The principal reasons for this situation are: (1) current solutions are application specific or reactive and thus not scalable, and (2) Internet's best effort end-to-end QoS makes no guarantees about when and whether data will be delivered at all. Therefore, there is a need for transparent but ubiquitous services that can handle dynamic information, act instantly, ensure correct behavior, make immediate decisions, and perhaps prevent undesirable events from happening.

2.2 Mobile and wireless communications

The increased demand for mobility and flexibility in our daily lives has led to the development of wireless LANs (WLANs) and cellular networks. Today WLANs can offer users high bit rates to meet the requirements of bandwidth-consuming services such as video conferences, streaming video, etc. Wireless LANs can be broadly classified into two categories: *ad hoc wireless LANs* and *wireless LANs with infrastructure*. In ad hoc networks, several wireless nodes join together to establish peer-to-peer communication. Each client communicates directly with the other clients within the network. The ad hoc mode is designed such that only clients within transmission range of each other can communicate. If a client in an ad hoc network wishes to communicate outside of the cell, a

member of the cell operates as a gateway and performs routing. They typically require no administration. Networked nodes share their resources without a central server.

In wireless LANs with infrastructure, there is a high-speed wired or wireless backbone. Wireless nodes access the wired backbone through access points that allow the wireless nodes to share the available network resources efficiently. Prior to communicating data, the wireless clients and access points must establish a relationship, or an association. In mobile systems, an ongoing connection between a Mobile Host (MH) and a corresponding Access Point (AP) is transferred from one access point to the other through a process called *handoff*. Handoff occurs during cell boundary crossing, weak signal reception, and QoS deterioration in the current cell. Present handoff mechanisms are based only on signal strength and do not take into account the load of the new cell. There is no negotiation of QoS characteristics with the new AP to ensure smooth carryover from the old AP to the new AP. Several methods have been proposed by researchers to ensure seamless handoff in mobile environments.

Since wireless devices need to be small and wireless networks are bandwidth limited, some of the key challenges to the use of wireless networks in pervasive computing environments are data rate enhancements, low power networking, security, radio signal interference, and system interoperability. Improving the current data rates to support future high-speed applications is essential, especially if multimedia services are to be provided. Data rate is a function of various factors such as the data compression algorithm, interference mitigation through error-resilient coding, power control, and the data transfer protocol. With the recent proliferation of outdoor wireless networks and the advent of Free Space Optics (FSO) or wireless optical communications, we are heading in the right direction in terms of data rate requirements. The size and battery power limitations of wireless mobile devices place a limit on the range and throughput that can be supported by a wireless LAN. The complexity and hence the power consumption of wireless devices vary significantly depending on the kind of spread spectrum technology being used to implement the wireless LAN.

A critical factor in pervasive computing is the power consumption associated with wireless communications among resource limited devices. New algorithms have been devised to conserve energy by minimizing wireless communications [20]. Further, the mobility of users increases security concerns in a wireless network. Current wireless networks employ authentication and data encryption techniques on the air interface to provide security for their users.

2.3 Sensor networks

Sensor networks enable us to observe and interact with the physical world in real time and, allow users to monitor the environment, and also to take appropriate actions. Such pervasive instrumentation will be of great value in a range of applications such as security, telemedicine, transportation, crisis management, etc. Thus, sensor networks readily extend to monitoring interactions among hardware and software entities in ubiquitous computing environments. The sensor nodes and their networks are expected to provide sensory services to applications/users continually and autonomously. However, sensor nodes are often low-resource

devices with limited CPU power, memory, battery power, and low bandwidth wireless communication channels. Therefore, it is extremely important for sensors to conserve their energy (battery power) in order to prolong their active longevity as well as the lifetime of the entire network. In a sensing application, the observer is interested in monitoring the behavior of a phenomenon under some specified performance requirements (e.g., accuracy or delay). In a typical sensor network, the individual sensors sample or gather local values (measurements), aggregate them in a meaningful way, and then disseminate information as needed to other sensors and eventually to the observer. The measurements taken by the sensors are mostly discrete samples of the physical phenomenon under observation, subject to individual sensor measurement accuracy as well as its location.

Although sensor networks share many of the challenges of traditional wireless networks, including limited energy and bandwidth and error-prone channels, communication in sensor networks may not typically be end-to-end. More specifically, the function of the sensor network may be to report information regarding the observed phenomenon to an observer who is not necessarily aware of the network infrastructure and individual sensors as an end point of communication. Furthermore, energy in sensor networks is more severely limited than in other wireless networks due to the nature of the sensing devices and the difficulty in recharging their batteries. The energy constraint in sensor networks indeed imposes serious challenges in hardware design as well as in communication protocols.

In a pervasive computing framework, tracking of objects (e.g., persons, goods, chemical and biological agents) is extremely important and can be facilitated by using smart devices such as active and passive sensors, motion detectors, RFID tags, digital camera, surveillance equipment, and so on. Such a framework deals not only with the information captured by task-specific sensors, but also with that handled by deployable networks of heterogeneous MEMS (micro-electro-mechanical systems) multisensor nodes (e.g., portable optical or chemical sensors) that communicate via wireless RF and are connected to the Internet backbone. Data coming from these sources need to be aggregated after appropriate transformation and then stored in a specialized server for intelligent decision making. The major design and research challenges in sensor networks include (1) power conservation of mobile sensors, (2) coding and compression of multimedia signals, (3) data fusion to reduce data communication complexity, (4) cooperation among heterogeneous sensor nodes, (5) flexibility on the security level to match the application needs so as to conserve critical resources, (6) scalability, self-organizing, and self-learning of sensor nodes, (7) trust and security decisions based on the utility for the application, keeping mobility and volatility as transparent as possible, and (8) protecting the network from external and internal intrusion.

There are multiple ways for a sensor network to achieve its accuracy and delay requirements, and a well-designed network should meet these requirements while optimizing the energy usage and providing fault tolerance. By studying the communication patterns systematically, the sensor network designer should be able to choose the infrastructure and communication protocols that provide the best combination of performance, robustness, efficiency, and low cost of deployment.

Applications such as sensor fusion, simulation, and remote manipulation, allow users to "see" composite images constructed by fusing information obtained

from a number of sensors. Thus, sensors might be viewed as offering network-based services that can be browsed by authorized users. The network may participate in synthesizing the query (for example, by filtering some sensor data or aggregating data). Nodes along the path can take an active role in information dissemination and processing. In this respect, sensor networks are similar to an active network. Application-specific in-network data processing is essential to maximize the performance of sensor networks.

2.4 RFID technology

Radio frequency identification (RFID) is an automatic data capture (ADC) technology that comprises data tokens/tags and mobile scanners/readers equipped with an antenna. The reader detects the presence of an RFID tag within its range. The frequency varies from very low (10–30 KHz) to very high (30–300 GHz). The RFID tags are attached or embedded in objects and programmed with data that identify the object. RFID tags can be read only or a read/write type. The emergence of RFID tags has created an opportunity to enable large numbers of passive objects with no embedded computing resources to be identified and tracked in a networking environment. For example, the nuts and bolts required to assemble a machine part on a manufacturing floor can be tracked with the help of such tags. The use of proxy agents and surrogate services makes it possible to incorporate passive physical objects into any pervasive computing environment. The major challenges include (1) the development of middleware for incorporation of tagged objects into the computing environment, (2) exploiting RFID tags to extract context information, (3) provisioning services, and (4) combining RFID tag information with other sources of information.

2.5 Middleware technologies

Traditionally, agents have been employed to work on behalf of users, devices, and applications [3]. In addition, agents can be effectively used to provide transparent interfaces between disparate entities in the environment, thus enhancing invisibility. Agent interaction and collaboration are critical to the development of an effective middleware for pervasive computing. Software agents in the middleware can be deployed to overcome the limitations of hundreds and thousands of resource-limited devices.

Service discovery is described as the process of discovering software processes/agents, hardware devices, and services. The role of service discovery in pervasive computing is to provide environment awareness to devices and device awareness to the environment. Service provisioning, advertisement, and service discovery are the important components of this module. Although service discovery in mobile environments has been addressed in existing work, service discovery in pervasive computing is still in its infancy. Existing service discovery mechanisms include JINI and Salutation, as well as the International Naming System (INS) [1].

Several new embedded devices and sensors are being developed in the industry and in research laboratories. The architecture of the Berkeley sensor motes and the TinyOS operating system are very good examples of devices and technologies

developed for use with embedded networked sensors. The challenge here is to design devices that are tiny (disappear into the environment), consume little or no power (perhaps powered by ambient pressure, light, or temperature), and communicate seamlessly with other devices, humans, and services through a simple all-purpose communication protocol.

Understanding of device and network technologies is important to create a seemingly uniform computing space in heterogeneous environments. The backbone network will probably continue to be the Internet for some time. The challenge is to overcome the Internet's end-to-end architecture and at the same time to allow flexible interactions among network devices, services, and users.

Mobile computing devices have limited resources, are likely to be disconnected, and are required to react (transparently) to frequent changes in the environment. Mobile users desire *anytime anywhere* access to information while on the move. Typically, a wireless interface is required for communication among the mobile devices. The wireless interface can be a wireless LAN, a cellular network, an ad hoc network, a satellite network, or a combination thereof. Techniques developed for routing, multicasting, caching, and data access in mobile environments should also be extended to pervasive environments. A pervasive environment comprises numerous invisible devices, anonymous users, and ubiquitous services. Development of effective middleware tools to mask the heterogeneous wireless networks and mobility effects is a challenge.

Provisioning uniform services regardless of location is a vital component of mobile computing. The challenge here is to provide context-aware services in an adaptive fashion, in a form that is most appropriate to the location as well as to the situation under consideration.

3 CHALLENGES OF PERVASIVE COMPUTING

3.1 Heterogeneity and interoperability

Today's computing world is replete with numerous types of devices, operating systems, and networks. Cooperation and collaboration among various devices and software entities is necessary for pervasive computing. At the same time, the overheads introduced by adaptation software should be minimal and scalable. While it is almost unthinkable to have homogeneous devices and software, it is, however, possible to build software bridges across various entities to ensure interoperability. But then the following question arise: How many such bridges should we create? What about the overheads introduced by the bridges? The Oxygen project [23] envisions the use of uniform hardware and network devices to enable smooth interoperability. The limitations of low resource hardware can be overcome by exploiting the concepts of agents and services. The challenge is to develop effective and flexible middleware tools that mask uneven conditions and to develop portable and lightweight application software.

Network QoS for delivering information and QoS for provisioning services are critical to pervasive computing. For example, in Scenario 1 described earlier, if real-time collaboration is necessary between the ambulance personnel and the doctors in the hospital, multimedia streaming over heterogeneous communication

systems must be realized. Such streaming must meet stringent QoS requirements, or else it will be useless. Defining QoS for pervasive computing applications will be a significant challenge to meet. Pervasive computing environments will definitely require service providers to address QoS issues.

3.2 Proactivity and Transparency

The development of computing tools such as the handoff operation in mobile systems have been, in general, *reactive* or *interactive*. On the other hand, the "human in the loop" has its limits, since the number of networked computers will surpass the number of users in the near future. Users of pervasive computing applications may wish to receive "what I want" information and services in a transparent fashion. A thought-provoking paper on active networks [21] envisions a majority of computing devices in the future as being proactive. Proactivity can be provided by the effective use of overlay networks. Active networks can be used to enhance network infrastructure for pervasive computing, ensure network management on a just-in-time basis, and provide privacy and trust [11].

Today, most computing and communication services are also reactive in nature. Most proactive services available today are usually obtrusive and often useless (the online paper clip and pop-up messages are good examples). Ideally, proactive services should be user/application specific and unobtrusive, and must ensure efficient utilization of resources. These requirements can be best described by considering our Scenario 3. Firstly, Dr. Smith's profile must be on his cafeteria's server so that the server can send appropriate information to his device. For example, the server can receive the doctor's schedule a priori from his PDA and determine whether he would be consulting with his patients, his students, or his colleagues. If his schedule is not busy, he may be interested in receiving news or music. Video streaming presented by the proactive server in the cafeteria to the doctor's handheld computer must meet certain QoS requirements in terms of resolution and clarity, brightness, timeliness, etc. In Scenario 1, the cell phone, the PDA and the camera all observe the occurrence of extraordinary events, interact, and make proactive decisions. It will be necessary for users to negotiate QoS to suit their profiles and applications. For example, in Scenario 3, the challenge is how to define proactivity in general and how to tailor proactivity to specific users.

Associated challenges include (1) how to leverage research work in the areas of situation-aware computing, device/user/application profiling, and software agents to enhance proactivity in existing computing devices and (2) how to exploit active network technology to overcome the end-to-end to limitations of the Internet to provide just-in-time services? Profiling can be effectively employed to ensure appropriate proactive services in pervasive systems [12].

3.3 Location Awareness and Mobility

Models of twenty-first century *ubiquitous computing* scenarios [22] depend not only on the development of capability-rich mobile devices (such as web-phones or wearable computers) but also on the development of automated machine-to-machine computing technologies, whereby *devices interact with their peers and the networking infrastructure, often without explicit operator control*. To emphasize the

fact that devices must be imbued with an inherent consciousness about their current location and surrounding environment, this computing paradigm is also called *sentient* [12] or *context-aware computing*. "Context-awareness" is one of the key characteristics of applications under this intelligent computing model. If devices can exploit emerging technologies to infer the current activity state of the user (e.g., whether the user is walking or driving, or whether he/she is at office, at home, or in a public environment) and the characteristics of their environment (e.g., the nearest Spanish-speaking ATM), then these devices can intelligently manage both the information content and the means of information distribution.

Location awareness has been perhaps the most widely investigated context, since the current (or future) location of users strongly influences their information needs. Applications in computing and communications utilize such location information in two distinct ways [9]:

Location-Aware Computing. In this category, the information obtained by a mobile device or user varies with location changes. The most common goal on the network side is to automatically retrieve the current or anticipated neighborhood of the mobile user (for appropriate resource provisioning), while on the device side, the typical goal is to discover appropriate local resources. As an example of this category, we can consider the case where mobile users would be automatically provided with local navigation maps (e.g., floor plans in a museum that the user is currently visiting), which are automatically updated as the device changes its current position. For example, in Scenario 1, knowledge of the accident location is critical to providing appropriate responses—to direct the ambulance or to provide network connections via available wireless routers. Similarly, in Scenario 3, appropriate information can be sent to the doctor's device from the closest server if his current location is known or can be predicted.

Location-Independent Computing. In this case, the network endeavors to provide mobile users with a set of consistent applications and services that do not depend on the specific location of the users or on the access technology employed to connect to the backbone information infrastructure. Information about the user's location is required only to ensure the appropriate redirection of global resources to the device's current point of attachment. Such applications are not usually interested in the users' absolute location but only in their point of attachment to the communication infrastructure. An example is cellular telephony, where mobility management protocols are used to provide a mobile user with ubiquitous and location-independent access.

While location-independent computing applications have a fairly mature history, location-aware computing is still at an early stage. Innovative prototypes of location-aware computing environments are still largely experimental and are geared towards specific target environments. The location support systems of different prototypes, as a result, have been mostly autonomous and have always remained at the disposal of the system designers. It is, however, important to realize that the full potential of location-aware computing can be harnessed only if we develop a *globally consistent location management architecture* that caters to the needs of both location-aware and location-independent applications and that allows the retrieval and manipulation of location information obtained by a wide variety of component technologies. This is an important challenge, since

location-aware and location-independent applications typically face significantly different *scalability* concerns. In general, location-aware applications do not pose many scalability issues, since they primarily involve local interactions. However, scalability is a critical concern for location-independent network services, which must support access to distributed content by a much larger user set.

In [9], we surveyed the various ways in which context-aware pervasive computing applications are likely to exploit and manage location information, and we used this understanding to debate whether a *universal location management infrastructure* should store location information in a topology-dependent (symbolic) or topology-independent (geometric) format. A detailed analysis of both location-aware and location-independent applications reveals three important points: (1) different systems and prototypes use a wide variety of location resolution technologies, (2) a significant number of location-based applications are primarily interested in resolving the location of a mobile node only relative to the connectivity infrastructure, and (3) obtaining geographical location coordinates requires varying levels of hardware that are absent in many pervasive devices. It seems more preferable for the universal location management infrastructure to manipulate location information in a structured, symbolic form. In cases where geographical coordinates are needed, these may be obtained through the use of access-specific technologies or via appropriate mapping. In the following, we enumerate the objectives of pervasive computing from the viewpoint of the desirable features of a universal location management infrastructure. In particular, we believe that *location prediction, location translation, signaling optimality, and location privacy* are four "must-haves" in a practical pervasive computing infrastructure.

Recall that the basic goal of pervasive computing is to develop technologies that allow smart devices to automatically adapt to changing environments and contexts, making the environment largely imperceptible to the user. The set of candidate applications and their underlying technologies is, however, anything but uniform! Developing a uniform location management infrastructure is thus a challenging task. In the following, we identify five location-related features that a universal architecture must support.

3.3.1 Interoperability across Multiple Technologies and Resolutions

Current prototypes for pervasive applications typically choose a specific location tracking technology that is suitable for their individual needs. Uniform location management architectures must be capable of translating the location coordinates obtained by such systems into a universal format that can be utilized by various application contexts. For example, cellular-based mobile communications will primarily need to resolve the location of a mobile device only up to the point of network attachment. Fleet management and tracking applications may, however, require explicit geometric information. The mobility management infrastructure should thus be capable of efficiently translating such location information between different representations, and also at different granularities (e.g., mobile commerce applications advertising e-coupons may not be interested in the precise hotel room where a given user is located).

3.3.2 Prediction of Future Location

Predicting the user's future location is often the key to developing smart pervasive services. For example, the ATIS active database can be triggered more intelligently by predicting the most likely routes and by warning the client about adverse road conditions along those routes. Prediction of an individual's future position in the indoor office can be very helpful in aggressive teleporting (e.g., supporting follow-me applications). In addition to this explicit service-oriented need for prediction, there is also an implicit need for predictive mobility tracking from the viewpoint of network infrastructure. In several location-independent computing scenarios, the network must meet stringent performance and latency bounds as it ensures uninterrupted access to global information and services, even as the users change their locations. For example, in order to provide quality of service (QoS) guarantees for multimedia traffic (such as video or audio conferencing) in cellular networks, appropriate bandwidth reservations must be made between the hand-held terminal and the serving base station (BS), as well as between the BS and the backbone network. To meet strict bounds on the handoff delay, the network must also proactively reserve resources at the cells where the mobile is *likely* to move. Since many of the tracking technologies do not themselves offer such predictive capabilities, the infrastructure must be capable of constructing such predictive patterns based on the collective or individual movement histories.

3.3.3 Location Fusion and Translation

In certain pervasive computing scenarios, location tracking is achieved through the combination of multiple technologies and access infrastructures. For example, an office application can resolve the location of a user at different levels of granularity using different technologies. As an example, the specific building could be identified through the current wireless LAN cell where the mobile currently resides, whereas an additional ultrasonic system (such as Cricket) [17] may be used to identify the precise orientation and room location of the mobile user. Since the user's complete location reference is obtained only by combining these distinct location management protocols/systems, our global location management framework must efficiently *fuse and merge location information* from two or more distinct network technologies.

The intelligent management of vertical (or intersystem) handoff, on the other hand, often requires the ability to *translate* the mobility and location-related information from one frame of reference to another. For example, when a user switches from a wireless LAN to an overlaid personal communication systems (PCS) network, the system must be able to translate the mobility patterns and location prediction attributes from one system to the other, independent of the representation format utilized by each individual network.

3.3.4 Scalable and Near-Optimal Signaling Traffic

The desire for provably optimal location update and paging strategies in cellular networks is not new. There has indeed been a great deal of work on efficient location management strategies. The world of pervasive devices is soon expected

to see a quantum jump in the number of mobile nodes (from millions of cell phones to billions of autonomous pervasive devices) and an even greater variation in their capability (such as power or memory). We must therefore develop efficient and near-optimal signaling mechanisms that minimize any unnecessary signaling load on both the devices and the networking infrastructure.

3.3.5 Security and Privacy of Location Information

Security and privacy management is a key challenge in pervasive networking environments. Notwithstanding the availability of advanced devices and location resolution technologies, users will not embrace a pervasive computing model until a scalable infrastructure is in place to appropriately protect such location information. The problem is not one of simply making such location information either visible or invisible to specific networks; we must allow the user to dynamically configure the scope of location visibility, possibly in multiple representation formats, to individual pervasive services and applications. For example, a user may wish to expose his precise GPS coordinates to emergency response applications (such as 911) but only a much coarser view (say at a granularity of 20 miles) to automobile insurance companies trying to monitor his driving profile. Alternatively, the user may want to specify his network point of attachment (symbolic information) but not his precise in-building location (geometric coordinates) to a pervasive enterprise application.

In a series of works [4, 5, 10, 18], we have developed an *information-theoretic* framework for effective location prediction with optimal signaling cost in wireless mobile networks. In particular, we have shown how the LeZi-update algorithm [4, 5] uses an adaptive learning technique to optimize the signaling associated with location update and paging in a symbolic domain. By treating the movement of a mobile device as a sequence of strings generated according to a stationary distribution, our novel algorithm is able to efficiently store a mobile's entire movement history and also to predict future locations with asymptotically optimal cost. A symbolic representation of location data allows the management infrastructure to deal with an extremely heterogeneous set of networking technologies that possess a wide variety of underlying physical layer and location sensor technologies. Indeed, the ability to accommodate *device heterogeneity* and *technological diversity* is a key to the success of a universal location management scheme. Moreover, we have shown that symbolic information is more amenable to storage and manipulation across heterogeneous databases, and can be exploited to provide necessary functions such as location prediction, location fusion, and location privacy. We have also designed a "hierarchical LeZi-update" algorithm that permits efficient translation of location profiles between heterogeneous systems [13].

4 PERVASIVE COMPUTING PROJECTS

4.1 Aura

Designed for distraction-free pervasive computing, the Aura project [24] focuses on human attention, thus creating an environment that adapts to the

user's context and needs. To accomplish this goal, the Aura research spans various individual technologies such as task-driven computing, energy-aware adaptation, intelligent networking, resource opportunism, multifidelity computation, nomadic data access, wearable computers, wireless communication, multimodal user interface adaptability, data and network adaptability, software composition, proxies/agents, collaboration, and smart space. Underlying this diversity, Aura applies two broad concepts, namely, proactivity and self-tuning. Proactivity is a system layer's ability to anticipate requests from a higher layer, whereas self-tuning allows layers to adapt by observing the demands made on them and adjusting their performance and resource usage characteristics accordingly.

The Aura architecture includes already developed but much modified systems such as Odyssey [16] and Coda [14], and other new system components such as Spectra and Prism. Odyssey provides resource monitoring and application-aware adaptation, and Coda supports nomadic, disconnectable, and bandwidth-adaptive file access. Spectra is an adaptive remote execution mechanism that uses contexts to decide how best to execute the remote call. Prism is a new system layer that is responsible for capturing and managing user intentions. Prism, also called the *task layer*, sits above individual applications and services but below the user, providing high-level support for proactivity and self-tuning.

To amplify the capabilities of a resource-limited mobile client and thus to improve user experiences, Aura applies *cyber foraging*. The idea is to dynamically augment the computing resources of a wireless mobile computer by exploiting a wired hardware infrastructure. A surrogate (hardware in the wired infrastructure) assists the mobile computer temporarily. Cyber foraging helps define many challenges such as proactivity for tracking user intent, adaptation for matching the demand and supply of a resource, context awareness to modify its behavior based on the user's state and surrounding, and balancing of proactivity and transparency.

4.2 Oxygen

The goal of the Oxygen project [23] is pervasive human-centered computing based on bringing abundant computations and communications as pervasive and free as air naturally into people's lives. This approach combines integrated user and system technologies that make it easier for people to do more by doing less, wherever they may be.

System technologies include devices, networks and software. Devices technologies provide intelligent space through environmental devices (E21s) that are embedded in homes, offices, and cars to sense and support a local-area computational and communication back-plane. Handheld devices (H21s) are person-centered devices equipped with perceptual transducers, and they can reconfigure themselves through software into many useful appliances in response to speech commands. Flexible, decentralized networks called N21s connect dynamically changing configurations of self-identifying mobile and stationary devices to form collaborative regions. The Oxygen software architecture can adapt to changes, as it relies on control and planning abstractions that provide mechanisms for change.

Oxygen's user technologies directly address human needs. These are perceptual technologies such as the spoken language and visual interaction, and other

user technologies including knowledge access, automation, and collaboration that help users perform a wide variety of tasks they want to accomplish in the ways they would to do them like. Speech and vision technologies enable the user to communicate with Oxygen as if they are interacting with another person, thus saving time and effort. Multimodal integration increases the effectiveness of these perceptual technologies. Knowledge access supports improved access to information customized to the needs of people, applications, and software systems. Automation offers natural, easy-to-use, customizable, and adaptive mechanisms for automating and tuning repetitive mundane information functions and control of the physical environment. For example, Oxygen allows users to create scripts that control devices such as doors or heating systems according to their tastes. Collaboration forms spontaneous collaborative regions that accommodate the needs of mobile people and computations and maintains the collaboration context using knowledge access and automation. It provides support for recording and archiving speech and video fragments from meetings, and for linking these fragments to issues, summaries, keywords, and annotations.

4.3 PICO

The Pervasive Information Community Organization (PICO) is a middleware framework that enhances existing Internet-based services [15] with the goal of meeting the demands of time-critical applications such as telemedicine, military, and crisis management. PICO provides automated, continual unobtrusive services and proactive real-time collaborations among devices and software agents in a dynamic heterogeneous environment. PICO deals with the creation of mission-oriented dynamic computing communities that perform tasks on behalf of users and devices autonomously. It comprises two basic building blocks: software entities called *delegents* (intelligent delegates) and hardware devices, called *camileuns* (connected, adaptive, mobile, intelligent, learned, efficient, ubiquitous nodes). The concept of PICO extends the current notion of pervasive computing, that is, that computers are everywhere [22]. Its novelty lies in creating communities of delegents that collaborate proactively to handle dynamic information, provide selective content delivery, and facilitate application interface. In addition, delegents representing low-resource devices have the ability to carry out tasks remotely.

In general, the devices in a pervasive environment provide the services of which they are capable. However, it is necessary to capture the device characteristics in terms of hardware and software and the services they can provide. In the PICO framework, a camileun captures the functional entities of a device. It is an abstract logical representation of a device and provides a link between a device and delegent(s). A camileun is described by the tuple of $C = <C_{id}, F, H>$ where C_{id} is the camileun identifier, F is the set of functionalities, and H is the set of system characteristics.

A delegent provides encapsulation, interface, delegation, adaptation, and manageability for the camileun, user, or application with which it is associated. A delegent *encapsulates* one or more functional units of a camileun. A delegent is goal directed and works by itself or in a community. A delegent responds to sensory inputs, events in the community, and events within itself, and takes

appropriate actions based on a set of rules. Delegents work in a community environment where they interact with other delegents and their environments. The modeling of delegents is described here. Delegents not only make camileuns *adaptive* to their surrounding environments but also *condition* them to overcome uneven capabilities of various collaborating camileuns. The set of operational rules, R, defines how a delegent responds to events when it is in a certain state. The operation rules include community engagement, communication, and migration of delegents. Events can be internal or external to the delegents. Each rule consists of a pair of conditional facts and actions.

A community is a collection of one or more delegents working together towards a common goal. Communication and collaboration are essential to the operations of a community, which provides a framework for collaboration and coordination among delegents. A delegent provides a common *interface* to communicate or collaborate with other delegents. Devices are capable of providing services to users and applications. The PICO concept allows the representation of various devices through their respective delegents, who collaborate with each other to provide integrated services.

Communities in PICO are formed either statically or dynamically. Static communities, also called *service provider communities*, are created to provide services in various applications. Dynamic communities are created in response to the occurrence of extraordinary events in the environment. Once a community is formed, its delegents collaborate to carry out the goal of the community.

4.4 MavHome Smart Home

In [7], we defined a smart environment as one that is able to acquire and apply knowledge about its inhabitants and their surroundings in order to adapt to the inhabitants and meet the goals of comfort and efficiency. These capabilities rely upon effective prediction and intelligent decision making with the help of such technologies as robotics, wireless and sensor networking, mobile computing, databases, machine learning and multimedia technologies. With these capabilities, a smart home can adaptively control many aspects of the environment such as climate, water, lighting, maintenance, and multimedia entertainment. Intelligent automation of these activities can reduce the amount of interaction required by the inhabitants, reduce energy consumption and other potential wastages, and provide a mechanism for ensuring the health and safety of the environment occupants [6].

In the MavHome project [8], smart home capabilities are organized into an agent-based software architecture that seamlessly connects needed components while allowing improvements to be made to any of the supporting technologies. The technologies in the MavHome are separated into four cooperating layers. The *physical layer* contains the environment hardware, including devices, transducers, and network equipment. The *communication layer* exchanges information between agents. The *information layer* collects information and generates inferences useful for making decisions. The *decision layer* selects actions for the agent to execute. The MavHome software components are connected using a CORBA interface.

Because controlling an entire house is a very large and complex learning and reasoning problem, the problem is decomposed into reconfigurable subareas or

tasks. Thus, the physical layer for one agent may in actuality represent another agent somewhere in the hierarchy, which is capable of executing the task selected by the requesting agent.

Perception of the state of the smart home is a bottom-up process. Sensors monitor the environment (e.g., lawn moisture level) and, if necessary, transmit the information to another agent through the communication layer. The database records the information in the information layer, updates its learned concepts and predictions accordingly, and alerts the decision layer to the presence of new data. During action execution, information flows top down. The decision layer selects an action (e.g., run the sprinklers) and relates the decision to the information layer. After updating the database, the communication layer routes the action to the appropriate effector to execute. If the effector is actually another agent, the agent receives the command through its effector as perceived information and must decide upon the best method of executing the desired action. Specialized interface agents allow interaction with users, robots, and external resources such as the wireless network or the Internet. Agents can communicate with each other using a hierarchical flow. As compared with other projects related to smart homes, MavHome is unique in combining a multitude of technologies from artificial intelligence, machine learning, wireless mobile networking, sensors, databases, robotics, and multimedia computing to create a smart home that acts as an intelligent agent.

5 CONCLUSIONS

In this chapter, we have presented an overview of the enabling technologies for the emergence of pervasive computing and communication infrastructures. While the Internet will perhaps continue to be the backbone of pervasive computing, the tremendous advances in wireless mobile communications allow the creation of ubiquitous networks with very little effort and insignificant cost Moreover, wireless communications offer users the luxury of mobility and provide connectivity on the move. Sensor networks, RFID tags, and embedded devices also help in the deployment of environments that are replete with computing and communicating services. Heterogeneous devices and networks, interoperability among disparate entities, and mobility and security will continue to challenge pervasive computing researchers.

REFERENCES

[1] W. Adjie-Winoto, E. Schwartz, H. Balakrishnan, and J. Lilley, The design and implementation of an intentional naming system, ACM SIGOPS Operating Systems Review, *Proceedings of the Seventeenth ACM Symposium on Operating Systems Principles*, *33*(5), 186–201.

[2] G. Banavar, J. Beck, and E. Gluzberg (2000): Challenges: An application model for pervasive computing, in *Proceedings of 6th Annual International Conference on Mobile Computing and Networking (MOBICOM 2000)*, pp. 266–274, Boston, MA, USA.

[3] P. Bellavista, A.Corradi, C. Stefanelli (2000): A mobile agent infrastructure for the mobility support, *Proceedings of the 2000 ACM Symposium on Applied Computing*, pp. 239–245.

[4] A. Bhattacharya and S. K. Das (1999): LeZi-update: An information-theoretic approach to track mobile users in PCS networks, *Proc. 6th Annu. ACM Int. Conf. on Mobile Computing and Networking (MobiCom)*, pp. 1–12.

[5] A. Bhattacharya and S. K. Das (2002): Lezi-update: An information-theoretic framework for personal mobility tracking in PCS networks, *ACM/Kluwer Wireless Networks Journal, 8*(2-3), 121–135.

[6] D. J. Cook and S. K. Das (2003): Health monitoring in an agent-based smart home, International Conf. on Aging, *Disability and Independence (ICADI)*, Washington, Dec.

[7] D. J. Cook and S. K. Das (eds) (2004): *Smart Environments: Architectures, Protocols and Applications*, John Wiley, to appear.

[8] S. K. Das, D. J. Cook, A. Bhattacharya, E. O. Heierman, and T.-Y. Lin (2002): The role of prediction algorithms on the MavHome smart home architectures, *IEEE Wireless Communications* (Special Issue on Smart Homes), *9*(6), 77–84, Dec.

[9] S. K. Das, A. Bhattacharya, A. Roy, and A. Misra (2003): managing location in 'universal' location-aware computing, *Handbook of Wireless Internet* (Eds. B. Furht and M. Ilyas), Chapter 17, pp. 407–425, CRC Press.

[10] S.K. Das and C. Rose (2004): Coping with uncertainty in mobile wireless networks, Proceedings of 15th IEEE International Symposium on Personal, *Indoor and Mobile Radio Communications (PIMRC)*, Barcelona, Spain, Sept (Invited Paper).

[11] W.M. Farmer, J.D. Guttman, and V. Swarup (1996): Security for mobile agents, issues and requirements, *Proceedings NISSC'96 National Information Systems Security Conf.*, pp. 591–597, Baltimore, MD, October.

[12] A. Hopper (1999): Sentient computing, The Royal Society Clifford Patterson Lecture, http://www.uk.research.att.com/~hopper/publications.html.

[13] R. Kambalakatta, M. Kumar, and S. K. Das, Profile based caching to enhance data availability in push/pull mobile environments, *International Conference on Mobile and Ubiquitous Computing*, MobiQuitous 2004, Boston, August 22–25, Boston, USA.

[14] J.J. Kistler and M. Satyanarayanan (1992): Disconnected Operation in the Coda File System, *ACM Trans. Comp. Sys. 5*(1), February.

[15] M. Kumar, B. Shirazi, S. K. Das, M. Singhal, B. Sung, and D. Levine (2003): Pervasive Information Communities Organization PICO: A middleware framework for pervasive computing, *IEEE Pervasive Computing*, 72–79.

[16] L.B. Mummert, M.R. Ebling, and M. Satyanarayanan (1995): Exploring weak connectivity for mobile file access, *Proc. 15th ACM Symp*. Op. Sys. Principles, Copper Mountain Resort, CO, December.

[17] N. B. Priyantha, A. Chakraborty, H. Balakrishnan (2000): The Cricket location-support system, *Proceedings of the 6th Annual International Conference on Mobile Computing and Networking*, August, pp. 32–43.

[18] A. Roy, S. K. Das, and A. Misra (2004): Exploiting information theory for adaptive mobility and resource management in future cellular networks,

IEEE Wireless Communications (Special Issue on Mobility and Resource Management), Aug, to appear.
[19] M. Satyanarayanan (2001): Pervasive computing: vision and challenges, *IEEE Personal Computing*.
[20] E. Shih, P. Bahl, and M.J. Sinclair (2002): Wake on wireless: an event driven energy saving strategy for battery operated devices, *Proceedings of the 8th Annual International Conference on Mobile Computing and Networking*, pp. 160–171.
[21] D.L. Tenenhouse (2000): Proactive computing, *Communications of the ACM* 43:5 (May).
[22] M. Weiser (1991): The computer for the 21st century, *Scientific American*, *265*(3), 94–104.
[23] http://oxygen.lcs.mit.edu/
[24] http://www-2.cs.cmu.edu/~aura/
[25] http://www.cse.uta.edu/~pico@cse
[26] http://ailab.uta.edu/mavhome/