SEMI AUTOMATED BARTERING OF DIGITAL GOODS AND SERVICES IN PERVASIVE ENVIRONMENTS

By

Olga V. Ratsimor

THE UNIVERSITY OF MARYLAND, BALTIMORE COUNTY
BALTIMORE, MARYLAND

Abstract

The vision of mobile personal devices querying peers in the environment for information such as local restaurant recommendations or directions to the closest gas station, or traffic and weather updates has long been a goal of the pervasive research community. However, considering the diversity and the personal nature of devices participating in pervasive environments it is not feasible to assume that these interactions and collaborations will take place with out strong motivating incentives.

We propose development of a bartering communication model that incentivizes collaborations in pervasive environments by supporting peer to peer bartering for digital goods, services and information. To demonstrate feasibility of our approach, we propose to develop a middleware framework that will employ bartering protocols and strategies that will be context sensitive and allow taking advantage of personal relationships that exist between the owners of the devices that operate in pervasive environments. We propose to develop *value based service descriptions* to express the perceived valuation of information and services in a personalized manner. Our proposed bartering model will also rely on *value based descriptions* to efficiently control information and services that leave and enter each device. In particular, we plan to investigate the *value in use* and the *value in exchange* facets. To expand the space of possible deals during bartering, we propose to exploit social relationships between the owners of mobile personal devices that populate pervasive informants.

Contents

Chapter 1

Introduction

The most profound technologies are those that disappear. They weave themselves into the fabric of everyday life until they are indistinguishable from it." - Mark Weiser[52]

With the explosion of wireless technologies[48], it is becoming clear that mobile computing is slowly but surely becoming a dominant new culture[41]. People have quickly become more and more reliant on the flexibility and autonomy that mobile devices can provide. Mobile users no longer can imagine their lives without their cell phones, PDAs, MP3 players, digital cameras, etc. This mobile culture is in a fluidic state of nonstop evolution. New hardware, new software and new services are being developed and broadly used. Old applications and old services are being transformed to keep up with this mobile evolution. In addition, a new class of digital goods and services is starting to evolve. Distribution and sales of cell phone ring tones, MP3 music files, podcasts, mobile games, and electronic coupons is developing into a separate industry. Mobile users equipped with personal devices are actively accruing theses goods and services. Many users are motivated by a goal of personalization of their off-the-shelf equipment. Others are looking for entertainment in a compact, pocket-sized form that can be ubiquitously present with them throughout the day. As this mobile revolution unravels, it is becoming clear that the next big step to truly embed pervasive computing in to everyday lives of regular individuals is seamless, automated, personalized and context aware collaborations between these personal devices. The vision of mobile personal devices querying peers in the environment for information such as local restaurant recommendations or directions to the closest gas station, or traffic and weather updates has long been a goal of the pervasive research community. However along the way of envisioning these interactions and collaborations, we have lost track of reasons that would prompt personal devices that have limited resources to cooperate with their peers in the environment. Personal devices have technical limitations such as limited computing power, limited battery life etc. and most important of all, they are designed to serve the needs of a single individual.

1.0.1 Problem Statement

Considering the diversity and the personal nature of devices participating in pervasive environments, is the current model of altruistic ad hoc collaborations still the most effective model to motivate collaborations? Is it valid to assume that devices in pervasive environments will collaborate regardless of their technical limitations? Is there a mechanism that will incentivize personal mobile devices to interact and cooperate to enhance the level of service offered to their users?

Consider the following scenario:
Sam and Ellie are undergraduate students in UMBC. Both Sam and Ellie are taking "The Introduction to Computer Science" class. Though they are registered for different sections, both of their classes are covering the same material. Sam missed one of the classes and he is interested in getting the notes for that lecture. Ellie has the lecture notes that Sam is looking for but she is reluctant to give them away. After all, she has put a lot of effort in composing those notes. Sam proposes an idea that would motivate Ellie to give him a copy of the notes. He suggests that in exchange for the much needed notes, he will share with her a useful technique that his lecturer taught during the last lecture. Ellie is intrigued. She has been having hard time solving one of the homework problems and she feels that this technique could help her with solving that problem. Sam and Ellie exchange lecture notes for the technique description. This collaboration proved to be beneficial to both of the students.

This social collaboration scenario is not far from the desired collaborations that have been envisioned in pervasive environments. Could we apply this economic bartering model to stimulate collaborations in pervasive environments?

1.0.2 Our Hypothesis:

Interactions and collaborations that are driven by self motivated economic goals will incentivize devices to collaborate and improve the over all quality of collaborations.

1.0.3 The Proposed Approach

To address the above described challenges, we propose development of a bartering communication model that incentivizes collaborations in pervasive environments by supporting peer to peer bartering for digital goods, services and information. To demonstrate feasibility of our approach, we proposed to develop middleware framework that will employ bartering protocols and strategies that will be context sensitive and allow taking advantage of personal relationships that exist between owners of the devices that operate in pervasive environments. We propose to develop *value based service description* to express the perceived valuation of information and services in a personalized manner. Proposed bartering model would rely on *value based descriptions* to efficiently control information and services that leave and enter each device. In particular, we plan to investigate the *value in use* and the *value in exchange* facets.

- *Value in Use* reflects the perceived significance, importance and usefulness that a device/user receives from this service. Different users might have different perception of the *value in use* for the same service. Thus, *value in use* is a highly personalized facet of electronic goods and services.

- *Value in Exchange* reflects the potential value of the service against any other service in *value-for-value exchanges. Value in exchange* of a good or a service is not dependent on its perceived value in use. In essence, a good or a service could have no personal use or value to a particular user but it could prove to be very valuable to another user or device. By trading away a good or a service, a user or a device can receive indirect satisfaction by acquiring some other needed good or service. Use of *value in exchange* allows context aware environmental management of pervasive environments.

Value based service description would be proactively used by our framework to efficiently manage services on the device. Most valuable services would be kept close to the foreground while less value services would be pushed into background. Finally, low value services would be removed from the device by being traded away or just simply deleted.

Proposed collaboration approach has a set of issues that are linked to the nature of bartering. The most prominent issue that has an effect on bartering is the need to have a *coincidence of double wants*. This in essence means that in order to have a successful trade each of the bartering parities must desire goods and services that the other offers. This limits the space of possible trade deals.

To expand this space of possible deals during bartering, we propose to exploit role based relationships such as social relationships or trust based relationships between devices. We categorize social relationships between devices into five categories. *Self Devices* - are personal devices that belong to a single user. *Sibling Devices* - are personal devices that belong to a single user. *Friendly Devices* - are devices that belong to a set of different users that form a tight circle of family and friends. *Stranger Devices* - are devices that just meet and are most likely will never come in contact again. *Unfriendly Devices* - are devices that are actively uncooperative. Proposed middleware will employ bartering strategies that take advantage of these relationships. During bartering, the level of cooperation would be related to the social relationships between devices.

As part of the proposed work, we plan to investigate effects of Digital Rights Management (DRM) on the bartering collaboration model and effects of DRM on proliferation of goods and services in pervasive environments. DRM mechanisms that restrict duplication of the goods and services can hinder exchanges. In case of DRM duplication restrictions, the user and in essence their devices are required to permanently give up the rights of ownership to the good or service during the exchange. Hesitation involved in relinquishing the rights followed by the confiscation of the good or service will affect the dynamics of bartering exchanges and slow down proliferation of goods and services within pervasive environments.

1.0.4 Summary

To finalize, the key contributions of our research is development of bartering models as collaboration management mechanisms for personal mobile device in pervasive environments and development of the management framework that will employ valuation based description of electronic goods and services. Our research approach is to develop bartering protocols, implement a prototype system that will employ these protocols.

The rest of the proposal is organized as follows. In Chapter 2, we discuss related work that had been done in the relevant fields. In Chapter 3, we describe our preliminary work that we have done in the aria of pervasive computing. In Chapter 4, we motivation behind our work and describe challenges that are importantand need to be addressed in our proposed work. In Chapter 5, we in detail describe the proposed approach and state the expected contributions of our proposed approach. In Chapter 6, we conclude our discussion by presenting our research plan and our research timeline.

Chapter 2

Background and Related Work

2.1 Introduction

In this chapter, we survey related work, both to point out contributions of previous researchers and to place our proposed contributions in the proper context. We organize this survey around the main themes of our research:

- Pervasive Environment,
    - Context-aware Computing,
    - Context Mediation,
    - Ambient Services,
    - Mobile P2P Computing
- Incentive Mechanisms in P2P Systems
- Mobile Commerce
- Electronic Goods and Services
- Bartering

2.2 Pervasive Environments

Pervasive environments are populated with mobile personal devices such as cell phones PDAs and laptops. These devices host personal applications such as calendars, contact managers and business type applications. But in addition to traditional personal applications, a new set of mobile personalized entertainment services are emerging and rapidly moving onto personal mobile devices. Explosive growth of computational capabilities, dramatic increase in demand for communication and information services and development of integration technologies has resulted in the development of new and effective pervasive computing and communications paradigms. Technologies such as Bluetooth [1] and 802.11[2] enable and empower development to an entire class of applications that allow users to interact, collaborate and share information in a seamless and spontaneous way.

2.2.1 Mobile P2P Frameworks

In the following sections, we describe frameworks and architectures that were built to facilitate peer to peer interaction in pervasive environments. In addition to the frameworks described below, as a part of our preliminary work, we have developed the *Numi* project. The *Numi* framework employs collaborative agents to facilitate p2p data routing to enrich service environments for mobile devices that operate in infrastructure-based wireless networks. Detailed description of this project can be found in Chapter 4.

Infostations

The Infostations project experimented with the notion of data hoarding in mobile environments. They argued that even though cellular voice and data networks facilitate "any time, anywhere" connectivity, they are expensive and offer low bandwidth. Infostation networks[40] have often been suggested as a viable alternative to meet the needs of mobile applications. An infostation network consists of a set of towers offering short-range high bandwidth radio coverage. They offer high-speed discontinuous coverage, which is inherently low cost. Network access is available to users that are passing in close proximity to an Infostation. In this sense, the infostation is similar to a base station coupled with an information server such that the base station provides the network connectivity while the information server handles the data requests. A mobile device thus experiences areas of connectivity (when close to an infostation) and areas of disconnection (when there is no infostation nearby). Specialized data link protocols have been suggested for allowing devices to communicate with such Infostations [16].

MoGATU

MoGATU is a lightweight peer-to-peer data management architecture for pervasive environments. The MoGATU framework facilitates serendipitous querying and data management in mobile ad-hoc environments [32] [33]. The MoGATU project regards devices as semi autonomous entities which are guided in their interactions by personal user profiles and the context that devices operate in. MoGATU uses a contract-based transaction model. Information about users is described in the personal profile that are represented in a rich semantic language. Each of the objects is described in terms of "beliefs", "desires", and "intentions". MoGATU introduces data-based routing algorithms and semantic-based data caching and replication algorithms. These algorithms enable mobile devices to utilize their data-intensive vicinities. MoGATU devices also use automated interactions in attempt to obtain data relevant to the user's "intentions" and "desires" which are encoded in the user profiles. MoGATUs transaction models is based on contract net principles for peer-to-peer interaction.

Proem

Proem is a peer to peer middleware framework that allows deployment of mobile and ad hoc applications [26]. The key goal of Proem is the support of wearable communities [15]. Wearable communities emerge when sufficient number of people that use wearble computing devices come together and use their devices to communicate and interact with one another [24]. The framework was developed to augment face to face interactions that happen between people that come in contact with one another. Proem relies on personal area networks to create a "digital sphere". When individuals come in close proximity to each other, their "digital spheres" overlap and this enables devices to communicate. During this stage, devices are free to exchange information and access each others services. This communication can potentially enhance social interactions between individuals and further promote face to face contact. Once the devices move apart, the connection is severed and the spheres detach and end communications. Match making, friend discovery, and file sharing are the sample applications that can be built on top of Proem. The Proem middleware consists of three main components: an application runtime environment, a set of middleware services, and a protocol stack [25]. The application runtime environment in essence is a peerlet engine running Proem compliant applications called peerlets. Peerlets employ an event based model and can be added to the peerlet engine and removed from the engine at runtime. Proem employs a set of protocols to facilitate interactions: transport, presence, data sharing, and community building protocols are used to facilitate interactions between devices. Proem also employs a set of managers to provide basic services for presence management, profile management, data space management and community management. It also provides mechanisms to log encounters and propagate events.

2.2.2 Context Awareness

Context awareness is a powerful concept that empowers many applications that wound not otherwise be able to operate in pervasive environments. Context aware applications exploit environmental factors such as: time, user's location, current and future events, other users in the environment etc. Describing context, evaluating it, identifying the most relevant information and then reasoning over it has been a focus of many research projects. The aspect of filtering through the contextual information is the most relevant aspect for our proposed approach.

One of the subfields of context aware computing is the use of ambient intelligence [45]. It exploits location relevant contexts. Ambient services are linked to the surrounding physical environment. They have geographical boundaries of relevance and utility. Ambient services are highly applicable in the domain of location based mobile commerce [28]. An example of a mCommerce ambient service is a mobile advertisement service that distributes ads to potential customers that are a short distance away form the store. This type of context dependent advertisement is very effective since it is highly relevant to consumer's proximity to the store.

Contextual mediation[12] can also be used to filter out contextual noise that overwhelms a mobile device that operates in pervasive environments. Contextual mediation [11]is a form of application aware adaptation that is used to manage contextual information requests. During a contextual mediation process, an application selects the most relevant and the most appropriate subset of available contextual data and delivers it to the user. This contextual data is described using semantic representation and a set of attributes. The utility of the data varies between "no interest" to "most interesting". Preferences that reflect users interests are used to narrow the set of contextually relevant data.

myCampus Framework

myCampus framework [44] is one of the more prominent and well developed projects that incorporates and robustly exploits context wariness in pervasive environments. *myCampus* framework uses *eWallet* management component that is running on uses personal devices. Users personal resources are modeled as semantic web services. Personal and contextual resources and services are represented in using ontologies. Task specific agents can use *eWallet* to reason over the context and application specific reasoning. *myCampus* framework gives special attention to the aspects of privacy for context sensitive applications. *myCampus* project allows automated management of various context dependent applications such as scheduling of a meeting or privacy sensitive inquiry about locations of another user.

2.3 Incentive Mechanisms in P2P Systems

A number of peer to peer file sharing systems, have been developed in recent years. The more prominent examples of music file sharing systems that have proven to be immensely popular are *Napster*, *Gnutella* and *KaZaA* [3, 4, 5]. Participants of these networks can offer songs from their musical collections for others to download. Participants can also search and upload songs from other peers on the network.

However, majority of the peer to peer systems in their original implementations suffer from the free rider problems. Many of the participants did not contribute resources to and only consumed them [8]. A number of approaches have been developed to encourage contributions to collaborations in peer to peer file sharing systems. The approaches involved currency based solutions, and payment based incentives and the use of credit systems.

Below we describe several research projects that address issues of free rider problems in peer-to-peer systems.

MojoNation

MojoNation[18] was one of the earlier designs that addressed the issues of free riding, freeloading and denial of service attacks in peer to peer systems. It also offered mechanism for balancing resource supply and demand. *MojoNation* followed a free economy approach of using internal currency called Mojo. The *MojoNation* barter system combined a "*digital token*" micropayment system with *peer-to-peer microcredit*. Initially each joining user was given an amount of currency. The user could spend the currency on purchasing the files from other peers. Users earned *Mojo currency* by sharing their disk space, processing power, bandwidth etc. All of the transactions were cleared by the centralized authority of a central bank.

Karma

Karma is a framework for P2P resource sharing[51]. The *Karma* project addressed issues of free riding in p2p applications through the use of a decentralized currency. Decentralization is one of the key features of *Karma*. This project employs a peer-to-peer scheme for tracking karma currency transfers. This approach protects the systems against malicious attempts to corrupt and alter the currency balance in the system. *Karma* also utilizes a secure exchange mechanism that ensures that participants can not counterfeit karma currency. This project also addresses issue of inflation and deflation to regulate the supply of *karma currency* on the network. These mechanisms provide significant improvement over the unregulated approach of *MojoNation* system. Similar to *MojoNation*, *Karma* employs a reward mechanism to incentivize peers in the system to contribute their resources.

N-Way Exchange Based Approach

[9] explored an alternative bartering approach to resolve the free rider problems. Users directly traded resources between themselves. Transfer priority is given to the users that conduct exchanges rather than simple consumption. Simple consumption is permitted only if there are no peers that are willing to participate in the exchange. The authors also explored n-way exchange mechanisms. In the n-way exchanges users formed rings of N peers. Each peer is served by its predecessor and service the successor in the ring.

Our Proposed Approach

Unlike our proposed approach, above described systems did not consider value of the entities that are being exchanged. These approaches also did not take into consideration social relationships between the users of the system.

2.4 mCommerce

Commerce has long been a critical component that stimulates collaborations[41]. Economically motivated collaborations use incentives of pay off to promote trade and other interactions and collaborations. Our research, primarily focuses on collaborations in pervasive environments. Thus, the most relevant form of commerce is mobile commerce. There is a grate diversity of mobile applications that fall the under umbrella of mobile commerce [49]. Mobile financial applications, mobile advertising, mobile inventory management, product location[46], proactive service management[50], wireless business re-engineering, mobile actions, mobile entertainment, vehicular mobile commerce[47], are all considered to belong to the mobile commerce domain [43, 49].

We have done significant amount of preliminary work in the development of mobile commerce middleware that operate in pervasive environments. In particular, we have developed the *eNcentive framework* which focuses on the issues of delivering mobile advertisements to the users in pervasive environments[37]. We have also developed the *Agents2Go* framework that supports location-dependent service discovery in mobile electronic commerce environments[36]. We also have developed the *Allia* framework that uses alliance-based service discovery to discover mCommerece services in adhoc environments[35, 39]. Detailed description of theses projects can be found in Chapter4.

A framework for mCommerce depends on four general levels of operations: wireless network infrastructure, mobile middleware, wireless user infrastructure and finally, mCommerce applications. Below, we describe a set of mobile commerce middleware projects and implementations are most relevant to our proposed research.

EasiShop - Context Aware Shopping

The *EasiShop* framework[21, 22] explores the concept of cross merchant product comparison shopping through the use of personal mobile agents and virtual market places. The framework assists consumers with their shopping experience. It envisions consumers that are quipped with personal devices that are running *EasiShop Shopping Agent*. The *Shopping Agent* is a mobile agent that is responsible for managing the shopping list and finding the best suited products that match consumers preferences.

As, consumers walk by the retailers locations, thier *Shopping Agents* enter the *EasiShop Catchment Zones* where they come in contact with the retailers *Retailer Agents* that is hosted by the wireless device in the shops. If a *Shopping Agent* identifies a product that a consumer is interested in, then the interaction is moved to the *Market Place*. *Market Place* acts as a forum where consumers and retailers can interact with one another independent of their physical movements. *Market Place* categorizes products into relevant sets which are represented as *stalls*. At a particular *stall* a consumers *Shopping Agent* has a chance to cross-compare the products and their prices. This enables the *Shopping Agent* to make informed decisions and derive recommendation that could be passed to the consumer.

iClouds - Mobile Advertising System

iClouds[17] is a framework that was developed to support dissemination of advertisements to mobile devices. This framework envisions formation of *information clouds* when several mobile devices come in proximity of each other. Devices are enabled to exchange electronic advertisements that are supplied by various merchants. Anonymous bonus points are used to reward active users that propagate advertisements. Devices keep track of users interests in data structures referred to as *iLists*. *iHave list* contains ads that user has and *iWish list* contains the list of types of ads that user is interested in. During encounters, devices exchange their *iLists*. If a match is found, the ad is moved to the device that desires the ad. The ads propagate from a device to a device creating a chain. The ad keeps information about every device that participated in its propagation. If the ad materializes into a purchase, then every device that assisted with propagation is rewarded with bonus points. The bonus points are managed by the *Mediator*. The *Mediator* in essence is a central database that coordinates merchants and customers. To protect customers privacy dynamic network data and encryption of application layer information is provided during the building of the chain.

Our Proposed Approach

We propose to exploit economically motivated incentives to promote collaboration in pervasive environments. Our approach relies on bartering as the exchange mechanism. Economical motivated mobile personal devices are driven to exchange goods and services in an attempt to increase their personal net worth and provide needed goods and services to their users. Personalized valuation of goods and services is at the center of our approach. We also propose to employ personalized bartering strategies that acknowledge and exploit personal relationships between users of the devices. Detailed description of our approach can be found in Chapter 5.

2.5 Digital Goods and Services

MP3s, podcasts, ring tones, screen savers, mobile games, wallpaper, video clips and electronic coupons are invading everyday lives of average consumers. Digital goods, services and information are becoming common mainstream entities. The use of digital goods is becoming part of daily routine for many people. Currently, for a small fee, average consumers can easily acquire any one of the above listed digital goods, store it on a personal device and have a change to enjoy it at any moment they whish. Some goods, such as ring tones, wallpaper and screen savers are used by many individuals as a statement of their individuality and personality. Other digital goods and service such as games, MP3s, podcasts and video clips bring entertainment and leisure. Development of this new class of goods and services has resulted in the development and emergence of a new digital economy. Buying ring tones and MP3s is becoming a matter of social norm. Users can log in into one of hundreds of websites, listen to hundreds of previews of numerous types and various genre of ring tones and just for a couple of dollars they can buy one and install it their cell phone in a matter of seconds.

However, behind all the fun of immensely popular digital goods and services, lies the serious and complex nature of digital goods and services. Formally, a digital good can be defined as: *"a payoff-relevant bitstring that affects the utility of or a payoff to some individual in the economy"*[34]. Digital goods are discrete since they are distributed and acquired in integer amounts. Unprotected digital goods can be easily duplicated and distributed. A copy of a digital good is usually another independent digital good. Digital goods are non-rival goods which means that consumption of this good by one individual or a device does not diminish the amount that is available to others. A digital good is an experience good. In order to get an understanding of quality and content of the goods, one needs to gain access to that good. Digital goods can be fragile and non fragile [34]. A fragile good is a good that will significantly denticulate in value if even small portion of the good is lost. A non fragile good is a good that will not significantly suffer in economic value even if a small portion of that good is lost. All these aspects of the nature of the digital goods have an effect on the distribution and acquisition mechanisms and strategies employed.

This dynamic nature of the new digital economy does not come with out controversies. At the early acceptance stages, sharing of MP3 files was a common practice. Music industry claims that is lost millions of dollars due to the violation of the of the copy rights of the songs that were freely distributed by users of p2p systems such as Napster and Gnutella. This controversy brought into focus issue of high costs involved in production of the original first copy of a digital good.

2.5.1 Original Cost of Digital Goods

Information, digital goods and services are inherently expensive to produce. On the other hand duplication and reproduction of already existing information, good or service quite inexpensive. For example, if a device wants to produce a traffic report for a particular stretch of road, it needs to collect a substantial amount of sensory data from that stretch of road, analyze it and may be even compare it with previous records. Finally, it needs to assemble the data and convert it into a report format. This device needs to spend a considerable amount of computing and battery power while it's sensors are collecting and processing the data and compiling it into a concise report. On the other hand, if the device wanted to share this information with another device, all it needs to do is to clone this information and transmit it to the other device. Clearly, the cost of producing an additional copy is insignificant compared to the "sunk cost" of the production of the first original copy. In fact, there is no limit to the production of additional copies. If a device can produce one copy, it can produce hundreds of copies. In addition, devices that receive the copies of information can themselves make duplicates and start to distribute this information. This distribution will generate fast growing competition amongst sellers. The end result of this replication and competition will push information price to zero. This clearly contributes to the diminished importance of original cost during valuation of services and data. In pervasive environments, production of the first good is notably more expensive since, the devices that frequently operate in these environments have very limited resources and computational power. To address the information price issue, devices could employ Digital Rights Management (DRM) mechanisms. These mechanisms restrict duplication and potentially slow down the effect of the rapid and uncontrolled information distribution. Unless DRM mechanisms are employed the price of services, information and digital goods in the environment will eventually be pushed to zero.

2.5.2 DRM - Digital Rights Management

Recent controversial developments in p2p file sharing applications exposed complexities involved in managing and distributing electronic goods and services. In addition to that, these controversies brought to the forefront, the importance of Digital Rights Management (DRM) mechanisms. DRM mechanisms are designed to protect the rights of the creators of original goods and service. Typically, DRM mechanisms incorporate encryption, conditional access, copy control mechanisms, and media identification and tracing mechanisms such as water marks [42].

2.6 Bartering

Bartering is one of the forms of trading [27]. It is also frequently referred to as a "pure exchange". Bartering is a method of trading services and information directly for one another (without the use of money or other similar unit of account or medium of exchange). In a barter exchange, one good is traded directly for another. Other forms of trading are: bargaining, bidding, auctions, clearing and contracts [31]. The process of bartering shares many common principals of other trading techniques. In bartering, sellers of a good or service are worse-off when there is a large number of other sellers of a similar good or service. Buyers of a good or service will prefer to have as many sellers as possible for the good or service they are buying. Buyers of a good or service prefer to have as few other buyers as possible. Sellers of a good or service would like to have as many buyers for their good or service as possible.

Bartering is an ancient form of trading. It is widely believed that it superseded use of money and currency. Bartering is still commonly used in various social and economical interactions particularly when the infrastructure that facilitates currency based exchanges fails. Recently, during the first few days in the aftermath of hurricane Katrina that hit the city of New Orleans, when communications were down and banks were not operational, local residents and business owners conducted bartering transactions to acquire needed resources.

Bartering has proven to be a reliable tool of exchange when the conducting transactions become unreasonably expensive. The absentce of exchange infrastructure or high costs of conducting transactions can be a motivation factor to barter. This is frequently the case in pervasive environments. Access to bank accounts from a personal device such as a cell phone for a micropayment is not a very complex process. However, the ratio of the transaction cost to the average amount involved in payments is very high for many average consumers. In fact, micropayment mechanisms have long been criticized for high transaction cost and high consumer anxiety [30]. Bartering can alleviate some of the transaction costs in pervasive environments. A user of a mobile personal device seeking particular MP3 would rather give one of his or her MP3s as a payment and not deal with conducting a monetary transaction that would involve bank transfers or a credit card transaction.

Clearly, barter exchange has a set of issues that can hinder a successful exchange. One such issue is *double coincidence of wants*, another is potential of asymmetric knowledge of the quality goods that are being exchange.

Double Coincidence of Wants

One of the critical issues that impacts success of bartering is *double coincidence of wants*. The concept of *double coincidence of wants* is the key behind the traditional economic definitions of a bartering process. The phrase "*double coincidence of wants*" was used in Jevons (1875)[20]. *"[T]he first difficulty in barter is to find two persons whose disposable possessions mutually suit each other's wants. There may be many people wanting, and many possessing those things wanted; but to allow of an act of barter there must be a double coincidence, which will rarely happen."* The translation of this concept into the bartering in pervasive environment is the following. Suppose *Device-1* that has a *Service-A* and wants *Service-B* meets another *Device-2* that has *Service-B* and wants *Service-A*. This convenient coincidence allows *Device-1* and *2* to exchange services and satisfy each others needs and wants. General belief in conventional economics is that this type of coincidence is very uncommon and thus making it difficult to trade.

Our Proposed Approach

We propose relaxation of *double coincidence of wants* through use of methods and strategies that can be employed in pervasive environments. To increase the space of possible deals, we propose to exploit social relationships between the owners of the personal devices that operate in pervasive environments. These methods and strategies are discussed in Chapter 5.

Asymmetric Knowledge

Sellers of the good or service have full knowledge of the quality of that good or service. On the other hand, a buyer has only partial knowledge that has been provided by the seller. This unequal and asymmetrical knowledge of the quality of goods and service that are being exchanged adds complexity and anxiety which can influence the exchange.

Our Proposed Approach

We propose to exploit social relationships of the owners of devices as a guarantee of the quality of service. Devices that belong to users that know each other are unlikely to intentionally mislead and misrepresent the quality of the good that they are offering. Clearly if a relationship is not established, then there is a risk of abuse of asymmetrical knowledge. However, in pervasive environments, an argument can be made that devices that come in contact in the pervasive environment most likely belong to individuals that have some social link to one anther. It is also reasonable to assume that these devices will come in contact again. This close knit, personal nature of pervasive environments will discourage abuse of asymmetry of knowledge of quality of good or service.

2.7 Summary of Related Work

In this chapter, we describe related work to provide the foundation that is necessary to describe our research approach and our expected contributions. We discuss the unique nature of pervasive environments and devices that operate in them. We also describe p2p systems and their approach to addressing the free riding problems during resource sharing. We describe mCommerce approaches to stimulate collaborations in mobile pervasive environments. We also describe and discuss aspects of digital goods and services. In this chapter, we also introduce and motivate the use of bartering exchanges. We also examine the issues that effect the success of bartering. We lay the foundation necessary to motive the use of bartering to stimulate collaborations in pervasive environments.

Chapter 3

Motivation

In the last few decades, we have witnessed dramatic changes in computing technology and people's perception of this technology. Many recent technological advances have managed to deeply embed themselves into every day lives of regular individuals. For instance, small personal devices are no longer perceived to be luxury articles but are thought of as an every day necessity. Visions of these personal devices coming together and forming pervasive computing environments have been a focus of many prior and ongoing research projects. Majority of the work has focused on establishing communication between these devices, discovering services on these devices, embedding personalization and context awareness into the services and their management. However, amidst this important work, there have been no attempts to investigate approaches that would incentivize much discussed peer to peer collaborations. In this proposal, we argue that an alternative approach is necessary to stimulate peer to peer collaboration. We propose to use a bartering based communication model to promote exchange of digital goods and services in mobile pervasive environments.

3.1 Basic Sample Scenarios

To motivate our research consider the following scenarios.

Scenario A:

Consider a computing environment with five mobile devices: *D1*, *D2*, *D3*, *D4* and *D5*. Each of the devices has a set of services that it is interested in acquiring and a set services that it is willing to offer to others. For example, device *D1* is interested in acquiring *Song-A* and could offer *RingTones-B*, *C* and *D*. One option is that all of the devices satisfy each others needs and blindly and generously offer up their available resources to each other. Another option is that devices could possibly sell needed services to one another. And third option is that devices attempt to exchange a service for a service in order to address their needs and wants. Issues associated with the first two options are discussed in a later section. Lets look in detail at the third option.

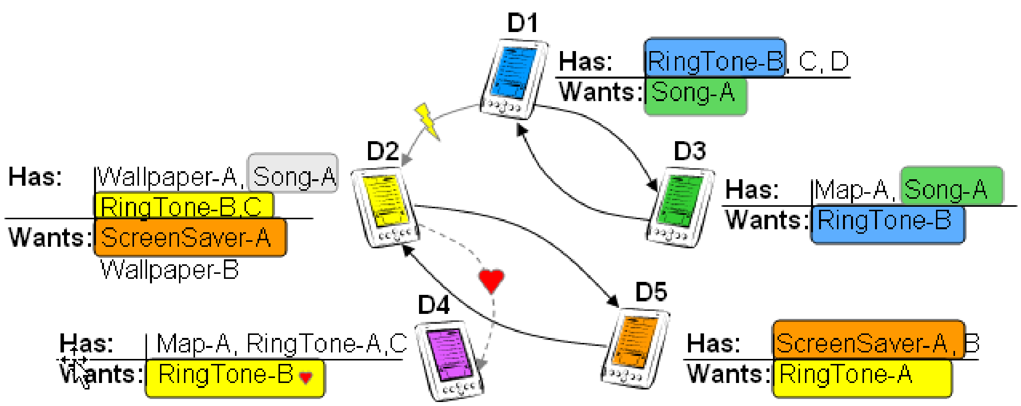

Figure 3.1: Basic Bartering Scenario

- Suppose, device *D1* is interested in acquiring *Song-A* (Figure 3.1). *D1* discovers that *D2* and *D3* are the devices that could potentially offer the song it is interested in. It starts by contacting *D2* and proposing an exchange of *Song-A* for a *RingTone-B*. Device *D2* responds with a rejection. *D2*'s reason for rejection is that it is not interested in ring tones. After analyzing the reason for rejection *D1* terminates negotiations with *D2*. *D1* attempts to approach *D3* with same proposal. *D3* is interested in one of the ring tones that *D1* has to offer. *D3* responds with a positive reply. *D1* and *D3* conduct a transaction and exchange *Song-A* and *RingTone-B*.

- **Counter Proposal Scenario:** Suppose, device *D2* is interested in acquiring *ScreenSaver-A*. It discovers that the only device that is capable of offering the much desired *ScreenSaver-A* is *D5*. *D2* composes an offer that gives a list of choices that *D5* can have in exchange for *ScreenSaver-A*. The first choice is

Wallpaper-A. The second choice is *Song-A*. The third choice is *RingTone-B* and the fourth choice is *RingTone-C*. *D5*, after receiving the proposal, composes a counter proposal which suggests a trade of *ScreenSaver-A* for two ring tones, *RingTone-B* and *C* After consideration, *D2* agrees. *D2* and *D5* conduct a transaction and exchange *ScreenSaver-A* or *RingTone-B* and *C*.

- **Relationship Scenario:** Suppose, device *D2* and device *D4* are owned by two sisters (that like each other). Both devices are aware of this relationship. Now suppose, device *D4* is interested in acquiring *RingTone-B*. *D4* determines that *D1* and *D2* have this ring tone. *D4*, by default, will prefer to barter with the sibling device. *D4* composes proposal requesting *RingTone-B* and in exchange offering either *Map-A* or a *RingTone-A* or *C*. *D2* is not interested in maps or ring tones. *D2* is looking for *Wallpaper-B* which *D4* does not have. However, *D2*, instead of sending a rejection to *D4*, considers the relationship and offers *D4* much desired *RingTone-B* with out anything in return.

Scenario B: Investment Scenario

Devices could also acquire services that they are not planning to personally use. These services could be treated as an investment and could be bartered off at a later time.

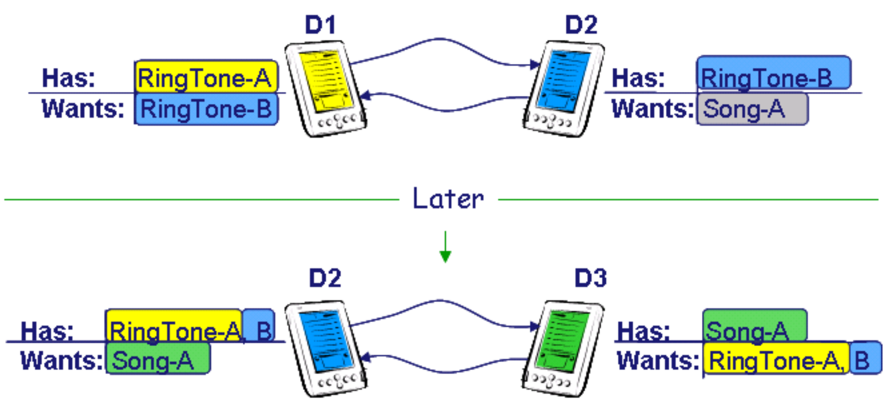

Figure 3.2: Acquiring Services for Later Resale

Suppose, device *D1* is interested in acquiring RingTone-B (Figure 3.2). It discovers that the only device that is capable of offering this ring tone is *D2*. Unfortunately, *D2* is not personally interested in *RingTone-A* that *D1* is offering in return. However, despite personal disinterest in the *RingTone-A*, *D2* identifies this ring tone to be in great demand by other devices. So, *D2* looks at *RingTone-A* as an investment that could be later cashed out for another service that *D2* needs. Thus, *D2* accepts the proposal from *D1* and exchanges *RingTone-B* for *RingTone-A*. So now, *D2* has *RingTone-A* and *B* but it is still looking for *Song-A*. At a later time, *D2* meets *D3*. *D2* identifies that *D3* has *Song-A* that *D2* was looking for some time. *D2* composes a proposal that suggests an exchange of *Song-A* for *RingTone-A* or *RingTone-B*. *D3* analyzes the proposal and responds with counter proposal suggesting an exchange of *Song-A* for the two ring tones. *D2* after consideration decides to go through with the exchange. Clearly, *D2*'s the investment into *RingTone-A* paid off. *D3* did not agree to exchange *Song-A* for just one ring tone.

Scenario C: Similarity Scenario

Devices could also barter based on the attributes or features of a service or information. This provides additional flexibility during the exchange and allows to increase the space of possible deals.

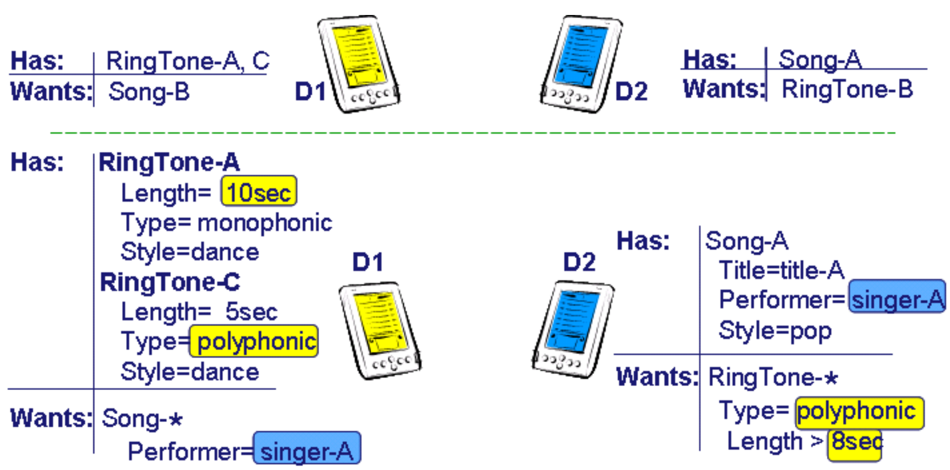

Figure 3.3: Attribute Based Bartering

Suppose, device *D1* is interested in acquiring *Song-B* (Figure 3.3).Unfortunately, this particular song is not available in the environment. Similarly, *D2* is searching for *RingTone-B* which is also unavailable. These devices have a choice. They can continue to look for a perfect match or they can look for similar services. Suppose, *D1* and *D2* decide to look for similar services. *D1* identifies that the most important feature of the song was it's performer, *Singer-A*. So it starts to look for songs by that performer. Meanwhile, *D2* determines that is interested in polyphonic ring tones that are longer than 8 seconds. So *D2* starts to search for the ring tones. Clearly, relaxation of the constraints increases the space of possible deals. *D1* discovers that *D2* has a *Song-A* performed by *Singer-A*. *D1* composes a proposal to exchange *RingTone-A* or *RingTone-C* for *Song-A*. *D2*, upon receiving this proposal determines that neither *RingTone-A* or *RingTone-C* matches all of the attributes *D2* was looking for. *D2* decides to further relax its requirements and ignore the length of the ring. *D2* accepts the proposal from *D1*. *D1* and *D2* exchange *Song-A* for *RingTone-C*.

To realize the above described scenarios, we need to address a set of challenges that will affect peer to peer collaborations in pervasive environments.

We argue that a bartering based communication model is a viable approach to incentivize collaboration and exchange of digital goods and services. Therefore, in the subsequent parts of this chapter, we will present issues that need to be addressed to develop a bartering model that will be able to represent complexities of trading digital goods and service in pervasive environments.

3.2 Challenges of Collaborations in Pervasive Environments

3.2.1 Free Riders in Pervasive Environments

In the previous chapter, we have discussed prior and ongoing research in cooperation and collaboration of peers in conventional peer to peer settings. Similar to conventional peer to peer systems, pervasive environments best function when devices agree to collaborate and coordinate with each other. One of the common visions of pervasive environment interactions is devices asking one another to share their goods and services to enhance the social experience of their owners. Request for traffic updates, restaurant recommendations, search for cheap gas station are all frequently discussed applications of pervasive environments. Parallels can be drawn between collaborations involved in these applications and collaborations in peer to peer music file sharing systems. However, unlike conventional peer to peer systems, pervasive environments are populated with small personal devices that are tightly constrained in their resources and computing power. Inefficient and uneconomical collaborations are far more damaging to the well being of the individual devices and to the computing environment as a whole. Analogous to the conventional peer to peer systems, collaborations in pervasive environment are not immune to the problem of free riders. In fact, the limited computing power and the personal nature of the devices that are involved in the collaborations further stimulates these devices to disregard ongoing partnerships and cooperative teamwork efforts in the environment. An example of free rider behavior in pervasive environments is a device that, in an attempt to conserve it's computing resources and power, is ignoring and dropping service requests and other communications from other devices while sending out service discovery requests for its own benefit. If a large enough percentage of devices sabotage collaborations, then this incapacitates the computing environment, leaving other cooperative devices potentially isolated and unable to complete their tasks. Free riders are very taxing on collaborative computing environments. To deal with this issue, environments need to have strong incentives that would attract devices with limited resources and limited computing power into honest personally meditated collaborations. The challenge lays in design and development of a communication management component that is capable of identifying incentives and exploiting them to derive the benefit. In order to achieve that, management components on these devices need to be able to compute benefits that they would receive from collaborations, so that devices are not unnecessarily overtaxed and over burdened. Incentives need to be embedded into the communication model to ensure that collaborations are done out of self beneficial economically motivated goals.

3.2.2 Information Noise Makers

Pervasive environments and personal devices are not exempt from spamming and other heavy communication services. Such services could be regarded by many others in the environment as useless noise makers. Intense environmental noise has a strong potential of hindering collaborative efforts in the environment by overwhelming the devices with extra information. In such “noisy” environments, devices must be capable of filtering out communications and interactions that are perceived to be of no interest or importance and still be able to participate in relatively useful information exchange and discover desired services. The challenge lies in development of a communication management component that is capable of identifying importance of the communications in a personalized manner. In essence, a communication management component needs to be able to determine the value of incoming data and offered services and make a decision of how critical is this interaction to the state of the device. Acquisition and retention of digital goods and services should be done on the basis of perceived value and desire and not on the basis of availability. Employing this mechanism would help keep devices clear out the content that is perceived to be unimportant. This will be achieved by controlling accumulation of "noisy" information and irrelevant digital goods and services on the devices and keeping to a minimum. In addition to that, this valuation approach will result in environmental hygiene since the services and goods that are not in demand will not be able proliferate throughout the environment.

3.2.3 Unpredictable Quality of Services and Goods

When devices collaborate and communicate, they acquire and collect information, goods and services. Depending on the environment and context, it is possible that received data and services are not what were originally expected. Upon acquisition, a device might identify the good or a service to be useless and there is no point it keeping and maintaining this good or service. Similarly, it is possible that when a good or a service was acquired, it was believed to be a very useful, important, and a valuable item. However, over a period of time, it became clear that the potential usefulness or need for this good or service was overestimated. Maintenance and upkeep of this good or a service is consuming limited valuable resources. Yet another possibility is that an obtained good or service was acquired for a short term use. Once the intended time expires, the device should be capable of disposing of the service in an optimal way. Another possibility is that the service is valuable only in a particular rarely occurring context, which is not expected to occur in the near future. Once again, the device should be able to identify such goods and services and possibly dispose of them in an optimal fashion. On the other hand, devices should be able to identify and protect goods and services that are perceived to be very valuable but are rarely used. Challenges in realizing this functionality lie in identifying the perceived quality and value of goods and services that are sorted and maintained by the device. This requires a management component that is capable of tracking and evaluating status of goods and services on the devices. Farther more this management component should be able to identify the reasons that contribute to this perception. In addition to that, once a good or a service is deemed insignificant, there are several logical approaches that could be taken to free the device form this good or service. One simple approach is to remove the service and cut the losses of resources that are being expended on maintenance of this good or a service. Another alternate approach is to try to sell the service or to trade it away in exchange for another good. In essence, this approach could be titled as "one mans trash is another mans treasure". The challenge lies in determining which of the two approaches to take. In order to pick the optimal approach, a device needs to be aware of the external demand for the service in question. This requires evaluation of the needs of the pervasive environment.

3.2.4 Anonymous Collaborations

Conventional views of mobile devices in pervasive environments assume that all of the communications and collaborations are anonymous. Social relationships between the owners of the devices are not propagated into of the collaboration and communication models that are employed. There is a number of reputation and trust based mechanisms that are used to identify cooperative and uncooperative devices in the environment. Though these mechanisms are powerful, they are disconnected from the real human social relationships. Such relationships are significant since they can be used to explain and justify the levels of cooperation that personal devices exhibit during peer to peer collaborations. There are a set of tools such as LinkedIn[6] and Orkut[7] that represent such social relationships. Removal of anonymity from collaborations brings additional flexibility to these strategies in pervasive environments. Collaboration strategies between "friendly" devices can be more relaxed since there is the factor of long term relationship between the devices. The challenge lies in developing mechanisms that allow personal devices to follow and execute personalized collaboration strategies that reflect existing social relationships between their owners. The communication management component needs to be able to reason over the relationship status and incorporate the characteristics of the relationship into the communications and negotiations.

3.3 Possible Solutions and Approaches

The following is a description of potential solutions that could be used to address the above described challenges of collaboration management in pervasive environments.

3.3.1 Solution A: Traditional Context Aware Personalized Approach

Use of personal profiles and context aware principals and mechanisms is one of the possible approaches to address the above described challenges. Context aware mechanisms as location and time based data management can be used to identify the importance and perceived value of the proposed collaborations. Personal user profiles can be used to complement context aware management of data and collaboration by further personalizing peer to peer interactions. However, even though these mechanisms are powerful and effective in representing personal and contextual preferences and settings, they lack factors that would stimulate and motivate collaborations in pervasive environment. With out the necessary incentives, substantial and meaningful collaborations will not be successful especially in the environments that are highly prone to abuse and obstruction from free riders. In addition to that, current approaches of context aware computing and personal profiles do not address issues of representing aspects of the perceived value of goods and services on the devices and in the pervasive environments. This limits the capability of the devices to identify goods and service that are perceived to be critical to the users and environment in which their personal devices operate. This is particularly important in highly dynamic environments where perception of value rapidly changes.

3.3.2 Solution B: Monetary Value and Abstract Currency

Another possible approach that could address some of the challenges satiated above is a computational economy approach. In this approach, every digital good, service or information is assigned a dollar value. Other currency, including internal abstract currency can be used to implement this approach. Devices that are interested in collaborations such as trading of goods or service can purchase the items from one another. This economic approach to collaboration provides the flexibility of pricing every collaborative task. This form of pricing the data would create a miniature economy within computing environments. It will also ensure that the environment is not abused by free riders. Every piece of data will have a clear cost associated with acquiring it. Devices would acquire only the data that they would truly need or be willing to pay for. This will reduce accumulation of unnecessary data on a personal device. Unfortunately, economic approach that involves currency requires a set of mechanisms that address issues of payment processing. Micropayment mechanisms could be used to finalize transactions. However, micropayments have a number of well known shortcomings like "mental transaction cost" and "user anxiety". These shortcomings will impact the computational microeconomic environments. Micropayment drawbacks will also apply to designs that use abstract currency. In addition, the microeconomic approach that uses abstract currency needs to have mechanisms that handle issues of inflation and deflation of that currency. Though this economic approach addresses the problem of free riders and noise makers in the computing environment it does not consider the issues of dynamic changes of perceived value of data. Traditional microeconomic approaches also do not take into consideration issues of context aware and personalized interactions and social relationships of device owners.

3.4 Proposed Solution

We propose a middleware framework that employs a computational economy approach of bartering of digital goods, services, information and resources to address the above described issues of collaborations in pervasive environments. The economic nature of the bartering approach addresses the issue of free riders in the environment. Economically motivated devices participate in exchanges and collaboration and use their resources, goods and service as payments for the services provided by their peers. This economic motivation will also control collection of unnecessary information thereby reducing the noise in the computing environment. In order to provide mechanisms to allow fair and balanced bartering, our approach relies on determination of perceived value of goods and services. We propose to develop mechanisms that will provide means to describe the aspects of valuation of goods and services. These descriptions will provide a personalized way to address perceived valuations and this distinguishes our approach from the approaches that use currency as a valuation mechanism.

There a number of issues that affect bartering. One of the more prominent and characteristic issues associated with bartering is the problem of *coincidence of double wants*. In fact, this problem is one of the major drawbacks of bartering interactions in any type of environment. Basically, in order to have a successful trade, the participants need to desire each other's goods or services. This constraint, in its strictest form, dramatically limits the pool of possible solutions. However, to promote bartering in pervasive environments, we propose to use a set of mechanisms that can broaden the set of possible solutions to facilitate a trade. We propose the use of social relationship dependent strategies to improve the set of available options of goods and services during bartering. We plan to relax the constraints of equality of values of goods and services during the exchange with friendly devices. Incorporation of this relationship based bartering strategy will help to widen the set of possible solutions since devices have an option of being more lenient and not driving hard bargains during an exchange with a friendly device.

We also plan to use investment based strategies. Devices acquire service not for personal or internal use but for future trade, which also further widens the set of possible trades and exchanges.

3.5 Expected Contributions

Below is a condensed list of expected contributions of the proposed research. Detailed description can be found in Chapter 5.

- Development and implementation of a framework that employs bartering to incentivize collaborations.
- Development and implementation of an Intelligent Bartering Reasoner.
- Development of value based descriptions for electronic goods and services.
- Development and implementation of context based personalized bartering protocols / strategies that employ:
  - Relationship Networks
  - Valuation-Based Bartering

- Demonstration of P2P environment that incentivizes collaboration between peer devices.

3.6 Summary

In this chapter, we describe the challenges of collaborations in pervasive environments. We motivate our approach by describing a set of basic, incentivized collaboration scenarios. We describe a set of challenges that affect traditional pervasive environments. We also discuss alternative solutions to deal with the described challenges. We also compare our proposed approach with conventional approaches. And finally, we conclude with a discussion of challenges that need to be addressed to ensure optimal approach to manage collaborations in pervasive environments.

Chapter 4

Preliminary Work

We now present our preliminary work in the aria of pervasive computing and mobile commerce and peer to peer communications. We describe our *Numi* framework which facilitates collaborative data management in a network of InfoStations. We them describe our most resent project called *eNcentive* which was build on top of *Numi*. *eNcentive* explores ideas of intelligent marketing in mobile peer-to-peer environments. We then describe our projects *Allia* and *Agents2Go*. In *Allia* project we investigate alliance based service discovery in ad-hoc environments for devices with varying capabilities and policies. Finally, *Agents2Go* project looks an infrastructure for location-dependent service discovery in m-commerce environment. For each project we provide motivation, describe design and architecture and discuss key aspects of functionality that are important to our proposed work.

4.1 Numi Project

We begin description of our preliminary work by introducing the project *Numi*[38, 23]. *Numi* is a framework that is capable of supporting combined infrastructure and ad hoc peer-to-peer computing. We built a working prototype of our frame work with a sample music delivering application built on top of this framework. We also simulated the proposed design. *Numi* also serves as a platform for the *eNcentive* framework that is described in detail in the following section.

4.1.1 Numi Framework

We have developed a framework for collaborative mobile data management in InfoStation networks. Users with mobile devices (with limited memory capacity) move through geographical regions characterized by islands of cheap high-speed network connectivity (InfoStations) separated by regions of no network access. Users that meet in these areas of no network access can participate in ad-hoc collaboration with peers passing by. The *Numi* project aims to study the feasibility of using such peer nodes to carry data for a node that is currently in a region with no network access. Servers (on the InfoStations) predict users' needs and mobility patterns. Using this information, these servers utilize other devices (with spare capacity) moving in the appropriate direction to deliver needed data to the appropriate users.

4.1.2 Numi Architecture

The network model that we considering are islands of high-speed wireless connectivity surrounded by regions of low or no network access. We envision that devices that are within these islands have access to an infrastructure component (access point) while in surrounding areas, only ad-hoc communication is possible between neighboring peer devices. The key components of our network include *Service Portals* (*SPs*), *Mobile Hosts* (*MHs*), and *Services*. *SPs* are infostations offering high-speed network connectivity and hosting services that can be used by nearby *MHs*. These *SPs* are connected by high speed links to the rest of the wireline network as broadband connectivity has become cheaper and more ubiquitous. The SPs use their wireless capabilities to interact with MHs that are in range and use their wireline connectivity to communicate among themselves.

This model of disjoint areas of coverage is quite realistic. In fact, increasing popularity of community networks and their commercial deployments (Starbucks offer connectivity in their kiosks and in most metropolitan areas, one encounters a Starbuck every few blocks thereby creating a network of pockets of network access separated by a distance of a few blocks) is adding more credence to the viability of this network model. Our *SPs* are more intelligent than conventional infostation systems and can predict a user's future data needs and through collaboration with other SPs in the network, data can be scheduled to be piggy backed on other devices to support this device in a completely distributed manner.

Mobile Hosts (*MHs*) are wireless mobile devices that can communicate both with SPs (infrastructure mode) and neighboring *MHs* that are within range (ad-hoc mode). These *MHs* travel through the geographical area populated with SPs. Our network can be thought of as comprised of two distinct types of zones: landing zones and transit zones. A landing zone is essentially an island of connectivity around a *Service Portal* limited only by that Portals wireless range. An *MH* can communicate with an *SP* when it is in a landing zone. In a transit zone, an *MH* can communicate only with peer *MHs* that are within its immediate neighborhood. Furthermore, we assume that MHs move along predetermined routes (like highways or based on their personal information such as appointments, place of work and habits etc.). An *MH* can request services from SPs that it encounters as well as other MHs. We assume a heterogeneous mix of mobile devices in our network with differing capabilities. Devices also vary in their level of participation in our system and the amount of resources that they are willing to share. Services are actual applications that are hosted on SPs and available to the MHs. In this sense, our SPs can be considered as a combination of access point (for providing network support) and application server for hosting popular services. We envision the usage of semantic languages like RDF or DAML to efficiently describe a service so that it can be made available to MHs that may have a need for them (Newspaper service, stock quotes, local interests guide etc.).

4.1.3 Design, Implementation and Prototype

To study the viability of our approach, we have designed and built Numi, a framework for supporting infrastructure coordinated ad hoc collaborative applications. The Numi framework can be run both on a mobile device and a Service Portal. Services can be implemented to run on top of our platform and take full advantage of the available ad hoc and infrastructure mode communication support. We designed Numi with the goal of reusability. Numi is essentially a set of agents and an agent runtime that could be used on both *MH* and SP. By abstracting functionality into distinct agents, our framework is highly modular and loosely coupled. It is possible to pick and choose agents that a device needs to run thereby allowing different configurations of our framework (a lighter configuration may be more suitable for a cell phone while a laptop could support a much heavier configuration). Services on top of Numi are also implemented as agents. Service providers can implement agents conforming to our specifications and these agents can be seamlessly introduced into a network. Service agents offer services that a user would need (a music service for example) while framework agents handle the lower level tasks that are needed for our framework.

The above figures illustrate typical configurations for Numi on SPs and MHs. A Heartbeat Generator Agent is responsible for broadcasting device presence messages. These heartbeats are broadcast periodically (every t seconds) and are used for identifying other devices that are in range. The SP heartbeat includes a unique platform identifier and a URL pointing to the list of services that are currently available to the user at that SP. The *MH* heartbeat contains a unique platform identifier, type of device and the route that this *MH* is traversing.

A *Location Monitor Agent* is responsible for location dependent issues. On an MH, this agent identifies whether or not that device is currently in a landing zone or a transit zone (based on whether or not a portal presence message has been received in the last t seconds). Also, this agent tracks the other peer devices that are currently in the neighborhood (based on the peers' device presence message).

A *Message Handler Agent* is responsible for handling the messaging needs of the framework. Agents on our platform use this component to send and receive messages to other agents on the same or on different platforms asynchronously. Messages are routed using a combination of agent identifier and platform identifier. Each message in our system contains a source, a destination, a message type and a message content. Our Message Handler also allows agents to register for specific types of messages. In this case, whenever the Message Handler receives a message that is not addressed to a distinct agent on that platform, that message can now be routed to agents that have registered for that message type (to handle advertisements for example).

A *Logger Agent* records every interaction that takes place on the local device. This includes user interaction with a specific service, messages that pass though the local Message Handler, peer encounters, peer queries for service etc. An SP uses these logs collected on an *MH* to extrapolate useful information such as service usage patterns, queries issued by other devices that this *MH* has encountered etc. On MHs with limited capacity, this agent may minimize logging or not log at all.

A *Task Scheduler Agent* is responsible for scheduling prescribed tasks at various times. These tasks could be one-time tasks that need to be executed at a set time or repetitive tasks that occur at fixed durations.

A *Data Handler Agent* is used for transferring data volumes between MHs and between an *MH* and an SP. The agent is required to implement a reliable protocol to exchange data volumes. Currently, our framework uses the FTP protocol.

Portal Service Agents run on top of our Numi platform on SPs and offer services to a user. These include services such as a music jukebox, newspaper service, stock quote service etc. These agents conform to the Numi agent specification and have access to different features offered by the platform. Service Agents at an SP actively wait for a user's request for a service. When a user requests a new service, the Service Agents determine the amount the data that user needs and capacity available on the device. Using this, the SP downloads an initial set of data to the device and informs the next SP on the devices route. This can be generalized to notify the next n SPs on a devices route. (Current implementation supports only one hop scheduling). The Service Agents at the subsequent portals determine when the device is likely to need data and take proactive measures to get the data to this device through other carriers. It is also possible that a *MH* was given enough data to last until the next SP but was moving slower than usual. In case a device does not show up at a landing zone at it's prescribed time, the Service Agents at that SP then actively starts looking for other devices whose routes indicate that they are heading in the direction of that *MH* and attempt to use them to deliver the next data segments. These Service Agents continue to track the *MH* until it arrives and a new set of data. Once this is done, the SP Service Agent notifies the next SP(s) on the route to schedule future data for this MH.

A *Node Service Agent* runs within Numi on an *MH* offering a service to the user. Whenever a device enters a landing zone, the Node Service Agents on that *MH* receive data updates from their respective SP Service Agents. An *MH* Service Agent also monitors service data usage and detects when the service is running out of data. When this occurs, the Node Service Agent publishes queries in its neighborhood to obtain the next set of data needed to keep that service running. Service Agents on other devices that have this data acknowledge these queries and using the Data Handler Agents, data can be exchanged. These impromptu ad hoc collaborations have a high probability of succeeding, as a needy device's neighbors have probably been pre-equipped with this needed data by an adjacent SP. In some cases, neighbors cannot handle these queries. However, since the Logger is logging these interactions, a neighboring device reaching an SP can trigger this SP to attempt to deliver the data to the *MH* that initiated the query.

A *Service Manager Agent* is responsible for managing Service Agents on a platform. The SP Service Manager hosts the Service Agents representing services available at that SP. This Manager monitors system usage by each Service Agent including statistics like the amount of memory used, running time, messaging overhead incurred etc.

Portal - MH Interaction

An *MH* that comes in range of a Portal communicates with the Portal to obtain data. Based on this interaction, SPs can track the service usage patterns for a user as well as the proposed route for the user. This information is used by the Portal to intelligently transfer only as much data as needed by the device until it reaches the next Portal. Also, Portals know about the impending data needs of other nodes that this *MH* is likely to meet during its travel. By strategically downloading right amounts of data, any additional memory that this *MH* has can be utilized to route data for other nodes.

Portal - Portal Interaction

Once a Portal services a Particular MH, it informs its neighboring Portals of this *MH* and its future data needs. Other Portals can then determine if this *MH* has enough info to last till it arrives at the next Portal or if proactive action is required by the Portals to transfer data to this MH. Through this interaction, Portals perform distributed scheduling to intelligently transfer data to even nodes that are not in the immediate range. Currently we are considering that a Portal interacts with its 1-hop neighbors to intelligently service MHs. This can be generalized to n-hop neighbors.

MH - MH Interaction MHs traveling in a transit zone can form ad-hoc communities with neighboring peers. In the event of needing data, an *MH* can query its neighboring peers to see if they have the data that is needed. If a peer has the desired data, the peer notifies the *MH* and the data can be transferred. If no peer responds, the *MH* continues its querying process and as its neighborhood changes, other peers get involved in this collaboration. The process continues until a response is received or the *MH* arrives at a landing zone (where the SP can definitely provide the required data). It is through this interaction that a *MH* can obtain data that has been routed to it by a neighboring Portal that predicted this impending data need.

4.1.4 Prototype Implementation

We have implemented a prototype of our framework using Java programming language. We installed our platform on three PCs and three iPAQs. The PCs run the SP platform and the iPAQs run the *MH* platform. All devices used were equipped with 802.11b[7] wireless LAN cards. The iPAQs were running the Jeode Embedded Virtual Machine. Each SP also runs a Tomcat Apache servlet engine. To simulate the mobility of the devices (moving in range and out of range of each other) we divided each transit zone into non-overlapping cells. Each cell has a unique cell ID. MHs are able to communicate with each other only if they are in the same cell. Messages have been augmented to carry a cell ID. Since we are using 802.11, broadcast messages will be heard by all devices. However, the message handler filters out all messages that do not match a device's current cell ID. By using this notion of cells, we can simulate neighborhoods and by changing a MH's cell ID, its neighborhood can be changed thereby simulating movement. We have developed an additional simulation component called the Mobility Coordinator. Using this, control messages can be sent to any *MH* to change its current cell ID. As a sample service, we have built a working music application that allows a user to listen to his favorite MP3 playlist on his PDA. On startup, the application downloads only a few songs on the playlist. As the user listens to his songs, previously heard songs are removed from the device and replaced with his other favorites.

4.2 eNcentive Project

We continue description of our preliminary work by introducing our most recent and still ongoing project eNcentive[37]. Primary goal of eNcentive project is to explore the aspects of information and services distribution, exchange and valuation in m-commerce environments. Currently, eNcentive provides a support for a limited query routing and processing, some semantics-based data caching, a limited notion of user profiles, and it achieves some measure of pro-active profile-initiated interaction. eNcentive is an initial prototype for our proposed work and we present progress to date. We will use eNcentive as a testbed for implementing and validating our findings and design choices. In this section, we present the current status of the framework and point out its important and interesting features.

Typical Scenario Consider a user equipped with a PDA running an eNcentive platform traveling though a geographical region populated by businesses (restaurants, cafes and drycleaners) that are actively broadcasting coupons, advertisements and other promotions to attract extra business. As the user passes by these businesses, the eNcentive platform running on the user's PDA actively collects/caches these coupons and promotions. These advertisements and coupons can be redeemed by this user at a later time at that business location or at any other service provider who honors those promotions. Alternatively, the user can employ the eNcentive platform to become a distributor of these coupons, promotions and advertisements. In this case, the platform starts to actively advertise coupons to other eNcentive peer platforms that the user passes by along the way. The peer platforms can cache these distributed advertisements and later redeem them with the business that honors these advertisements or likewise become another distributor. Thus a coupon can be passed from a user to another to yet another user before it is redeemed. To keep track of this chain, every coupon contains a list of platform IDs of every eNcentive platform that ever distributed this coupon. When a user decides to redeem a coupon and presents it to the business, the business after honoring the coupon stores the list of the platform IDs for future reference. Every participating business can choose to reward its most effective distributors with additional discounts or other rewards. For example, when a user redeems a coupon, the business can check its list of recently redeemed coupons and see how may times that user's eNcentive platform ID appears in those lists. The business can then reward the eNcentive user with additional discounts or upgrade the product or service the user was buying. This interaction thus is mutually beneficial to both the businesses and the users that distribute coupons on behalf of these businesses.

4.2.1 eNcentive Architecture

eNcentive framework[16] was developed to explore the aspects of information distribution in m-commerce environments. eNcentive was built as an application on top of an in-house agent framework Numi which is described in Section 4.1. eNcentive employs an intelligent peer-to-peer marketing scheme, by providing users the capability to collect promotional information disseminated by hotspots on behalf of small businesses and merchants. Users can propagate these sales promotions and discounts to other users in the ad hoc network. Participating users benefit from such circulation since merchants and small businesses that originally created the promotions reward the active distributors with additional promotions and other compensations. In the similar way, (by the same token) participating businesses and merchants benefit from this promotion distribution as they effectively reach a wider set of potential customers in their local neighborhoods. Our framework also offers advertising businesses and other information providers a mechanism for targeted promotions, which help them adapt and modify promotional information in more customized and personalized way.

Consider a user equipped with a PDA running an eNcentive platform traveling though a geographical region populated by businesses (restaurants, cafes and drycleaners) that are actively broadcasting coupons, advertisements and other promotions to attract extra business. As the user passes by these businesses, the eNcentive platform running on the user's PDA actively collects/caches these coupons and promotions. These advertisements and coupons can be redeemed by this user at a later time at that business location or at any other service provider who honors those promotions. Alternatively, the user can employ the eNcentive platform to become a distributor of these coupons, promotions and advertisements. In this case, the platform starts to actively advertise coupons to other eNcentive peer platforms that the user passes by along the way. The peer platforms can cache these distributed advertisements and later redeem them with the business that honors these advertisements or likewise become another distributor. Thus a coupon can be passed from a user to another to yet another user before it is redeemed. To keep track of this chain, every coupon contains a list of platform IDs of every eNcentive platform that ever distributed this coupon. When a user decides to redeem a coupon and presents it to the business, the business after honoring the coupon stores the list of the platform IDs for future reference. Every participating business can choose to reward its most effective distributors with additional discounts or other rewards. For example, when a user redeems a coupon, the business can check its list of recently redeemed coupons and see how may times that user's eNcentive platform ID appears in those lists. The business can then reward the eNcentive user with additional discounts or upgrade the product or service the user was buying. This interaction thus is mutually beneficial to both the businesses and the users that distribute coupons on behalf of these businesses.

4.2.2 Design and Implementation

The eNcentive framework is an agent based framework implemented as an application running on top of the Numi framework. Numi acts as a underlying foundation since it provides a communication layer for eNcentive that can seamlessly work for both infrastructure-based and ad hoc wireless networks. eNcentive reuses much of the existing functionality provided by Numi. Device and infrastructure discovery, location management, data communication and messaging, application management and logging are provided by Numi framework agents. eNcentive framework runs on both mobile devices and advertisers' portal (portal are access points managed by application servers). There are two configurations for the eNcentive framework: eNcentive Mobile Node Configuration and eNcentive Advertiser Configuration. Our modular design allows us to reuse same agents to implement these functionalities in both configurations.

eNcentive Advertiser Configuration

The Advertiser Configuration is set up at the location of the business involved in marketing. It can be set up at the individual stores and shops or it could be used by an aggregate of multiple businesses. For example, a shopping mall could run a single instance of this configuration on behalf all participating stores in that mall. The configuration consists of two eNcentive agents: eNcentive Ad Maker Agent and eNcentive Marketing Agent running on top of Numi Portal Configuration. The eNcentive Ad Maker Agent provides advertisers with interfaces that allow them to create promotions and coupons. Promotions in eNcentive are objects or intelligent agents that encapsulate marketing details like discount information, promotion creation time, start time, expiation time, replication policy and a reward model that will be used as an incentive to active participants. The eNcentive Ad Maker Agent facilitates creation of e-promotions. It is also responsible for setting of replication policy. Once an e-promotion is created, it is given to eNcentive Marketing Agent. This agent is responsible for scheduling of advertisement release. The Marketing Agent is essentially in charge of the advertising policy. It specifies how frequently an e-promotion should be broadcasted by the advertising platform. This allows periodic and controlled injection of marketing information into the network. The Marketing Agent is also responsible for determining which targeted advertisement should be supplied to the users interested in targeted promotions.

eNcentive Mobile Node Configuration

Broadcast advertisements and promotions are picked up by the mobile devices running eNcentive Mobile Node Configuration. This configuration consists of two eNcentive agents running on top of Numi Node Platform: eNcentive Marketing Agent and eNcentive Ad Manager Agent. To reuse the modular functionality of eNcentive, we utilized the Marketing Agent used in eNcentive Advertiser Configuration. The job of Marketing Agent running on Mobile Node Configuration is to distribute advertisements that are approved by the local Ad Manager Agent.

The eNcentive Ad Manager Agent performs a number of tasks on the mobile node. This agent collects, organizes and maintains promotions. The collection function of the agent is linked to the user profiles and current user context. Once Numi platform notifies Ad Manager Agent about incoming advertisements, the agent consults the user profiles and makes a determination whether to collect or to ignore the advertisement. Current implementation of our framework employs relatively simple profiles described in XML. We are currently working on a more sophisticated implementation were profiles are implemented in RDF[5]. The eNcentive Ad Manager Agent also maintains and organizes already collected promotions. The promotions are kept in a lightweight data structure that is linked to the user interface. The promotions are also categorized based on their start and end times. Our data structure also insures that the information is displayed in the appropriate order. The Ad Manager Agent also is able to organize promotions by location. In particular, the Ad Manager agent can notify users that he or she is in range of a business that accepts the coupons held by the user. Thus, if the promotion is about to become valid, a user has a choice of remaining in his current location and being able to take advantage of the promotion.

The eNcentive Ad Manager Agent is also in charge of responding to the peer requests. That is, if a device A is interested in the promotion that is being advertised by the device B then Ad Manager Agent of A request the Ad Manager Agent of B to forward the promotion. When the request is granted the Ad Manager Agent of B inserts B's platform ID into the requested promotion and then forwards this to A.

Current design of the eNcentive framework employs a push model. The advertisements are actively broadcast through out a network. This is clearly not the only model that can be used. A pull model can also work well in mobile peer-to-peer environment. Mobile devices that are interested in the marketing information can query other peer device in the neighborhood. Other hybrid approaches are also possible.

4.2.3 Functionality

eNcentive investigates issues of dissemination of information in ad hoc environments within hotspot networks.

4.3 Agents2Go Project

Agents2Go System[36] is a location aware, distributed system that allows mobile users to request and receive various services information that is of most relevance to their current geographical location. Mobile users will not be burdened with extraneous information for services in remote locations. Also, the Agents2Go System allows service providers to supply dynamic service updates. This dramatically improves the value of the service for the providers and gives users more refined service information. Our implementation currently deals with restaurants, but it could be easily updated to work with other location specific services.

4.3.1 Agents2Go Architecture

The Agents2Go System, illustrated in Figure. below , is composed of several components: the PalmApp, the Agents2Go Server, the Locator, the Agents2Go Information Repository, the restaurant Brokers and participating Restaurant Agents.

4.3.2 Design and Implementation

In this section we describe each part of the Agents2Go framework design, and explain numerous implementation choices we maid and we also briefly mention on other possible alternate designs.

PalmApp

The PalmApp is the end user interface to the Agents2Go System. This component runs on the user's PDA equipped with a CDPD modem. Essentially, it is a generic "form visualizer" that is independent of the system functionality. To reduce complexity, in-house markup tags are used to specify the layout and components of the form. The PalmApp captures a user request, converts it to an appropriate format, and then forwards that request to the Agents2Go Server. The PalmApp also handles the responses from the Agents2Go Server and presents them to the user.

Agents2Go Server

The Agents2Go Server is the component that handles messages to and from a PalmApp. User queries are forwarded to the Locator, and the corresponding responses are forwarded back to the PalmApp. Upon receiving a "form request message", the Agents2Go Server reads the requested form from a file. If the desired form cannot be located, a suitable error form is sent back to the PalmApp. Once the desired form is located, the Agents2Go Server uses a lookup table to map the specified cell tower id (obtained from the request message) to its neighborhood name. This neighborhood name is inserted into the location field of the form that needs to be displayed to the user. This neighborhood name, which can be changed by the user, is inserted into location field of the form displayed to the user. Thus a user, regardless of his/her current location, can find information about any participating region. This is encapsulated in a "response form message" and sent back to the PalmApp. When the Agents2Go Server receives a "form data message" from a PalmApp, it forwards it to the Locator. Other alternative designs could be used, a "form data message" could be forwarded to the corresponding Broker. Once the Agents2Go Server receives a response from either the Locator or a Broker, it generates the corresponding "response form message" and sends it to back the PalmApp.

A2G Locator

The Locator is the component that receives requests from the Agents2Go Server, determines which Broker is responsible for the area from which the request originated and then forwards the request to that Broker. The Locator dynamically builds and maintains a table that maps geographical areas to Brokers. The Locator listens on a well defined port for registration messages from Brokers. This registration consists of a port on which the Broker will accept requests forwarded by the Locator, and the geographical area for which this Broker is responsible. Upon receiving a request form the Agents2Go Server, the Locator looks inside the request string and extracts the point of origin information. This information is used to determine the designated Broker. The Locator then forwards the request to that Broker. If the Locator is unable to locate a suitable Broker for the given request, the Locator sends a "broker not found" message back to the Agents2Go Server. A reliable communication channel is maintained between the Agents2Go Server and the Locator for all message transfers.

A2G Broker

The Broker is the component of the system that maintains information about restaurants in its designated geographical region. The Broker processes requests from a user and generates suitable responses. These requests are forwarded to the Broker from the Locator and the generated responses are sent to the Agents2Go Server for forwarding to the requesting PalmApp.

The Agents2Go System partitions participating restaurants into sets based on the geographical region in which these restaurants are located. These sets are called coverage regions. Each coverage region is assigned a unique name and is serviced by a designated Broker. This Broker is responsible for generation of replies for requests pertaining to its coverage region. Our current Broker implementation allows grouping of several geographical regions or partitioning a single geographical region to construct a coverage region. Every Broker in the Agents2Go System is also associated with a specific Agents2Go Information Repository. An Agents2Go Information Repository is a set of databases that contain information about participating restaurants in a Broker's coverage region. Restaurant information like name, address, cuisine etc. of all participating restaurants in that coverage region is distributed among these databases. This information can be classified a static, since it rarely changes. The Broker is also responsible for frequently changing restaurant information like waiting times and promotions. This kind of information can be classified as dynamic.. This separation of dynamic and static information reduces the number of messages that is exchanged between the Agents2Go System components.

The Restaurant Agent

The Restaurant Agent is the component of the system that resides and runs at the location of the participating restaurant. This component allows a restaurant host to update dynamic information such as waiting times, promotion information, etc. geographical area in which the restaurant resides. If the restaurant is located in a cell overlap region, which is managed by several Brokers, then the update message is sent to every Broker that manages the overlap region. The update message contains the restaurant id, and other relevant information like the value of the wait time for table for two, the wait time for table for four, the wait time for table for six, etc. Each update message also contains a timestamp specifying the creation time. The Broker, upon receiving an update message, extracts the relevant values from the message and inserts these values into the appropriate row of its "waiting time" table.

4.4 Allia Project

Allia framework we explored alliance based service discovery in ad-hoc environments for devices with varying capabilities and policies. The Allia framework uses peer-to-peer caching and policy-driven agent-service discovery to facilitate cross-platform service discovery for an eCommerce environment [**?**, **?**]. The main goal of platform formation is to provide an agent better access to services in the vicinity (to facilitate interoperability). Our solution describes a policy-based distributed architecture towards achieving best-effort service discovery in ad-hoc networks for mobile commerce environments.

Alli's vision is that each participating device will be able to run a lightweight version of the platform components like yellow pages and white pages service. Putting an agent platform on extremely resource-constrained devices might overload such devices. It is essential for each device to have a set of bare minimum platform components like *AMS* and *DF* so that it can host an agent platform and operate even in regions of disconnections or stay independent. An agent in that case does not have to depend on other peer devices to host itself. The *Main Container* of LEAP [10] that contains the *AMS* and the *DF* runs on standard laptops and 3870 iPAQs. Looking at other devices with lesser resources and smaller footprints, we see that they have the capability to host advanced runtime environments (like Java Virtual

Machine). For example, most i-mode phones support Java Micro Edition[19], Mobile Information Device Profile [29] or Connected Limited Device Configuration [13] and have capability to store around 12Mb of static memory space like Sony Ericsson P800. Some phones have a run time heap of 180K (e.g. Samsung x350 series). While this is less than what the current version of LEAP on Personal Java [14] requires (around 700K of dynamic memory), it is well within the limits that will be feasible on even cell phones shortly considering the rate at which the capabilities of these phones are increasing. Cell phones can be considered one of the least resource-rich devices amongst the plethora of heterogeneous mobile devices existing around us. In addition to these platform components, the device will be able to host at least one agent. The main purpose behind formation of compounds was to enable agents on a device to better utilize services/agents in its vicinity. Our approach concentrates on providing a solution to this issue without imposing the dependence that devices sharing a distributed compound will have on each other. The yellow pages service component (*DF*) registers only the services that are hosted on that local platform. The white pages service component (*AMS*) registers only the agents that are hosted on the local platform. Each device has a policy that reflects the device capabilities, user preferences, application specific settings, etc. The policy governs the way this device is going to present itself to the other platforms/devices in its vicinity. It also describes the way in which the device takes advantage of resources/services in other platforms in its vicinity.

In our approach, every node advertises its services to other nodes in its vicinity in accordance with the local policy. These advertisements are broadcasted. On receiving an advertisement, an agent decides based on its policy, whether to cache it or reject it. We introduce the concept of "alliance" of a node. An *Alliance* of a particular node is a set of nodes whose local service information is cached by this node. Thus, a node explicitly knows the member nodes in its *alliance*. However, each node does not know the alliances in which they are members. Whenever a node leaves a certain vicinity, and enters a new vicinity, it constructs its own *alliance* by listening to advertisements. It also becomes a member of other *alliances* by advertising its local services. Its exit from other *alliances* is passive since the nodes governing these *alliances* detect its absence and remove the node from their *alliances*.

The local policy of a node dictates the ways in which the node wants to advertise itself to peer nodes in its vicinity. The rate of advertisements is also controlled by the policy. The policy can specify algorithms to allow dynamic adjustments of advertisement rates based on mobility of nodes in a neighborhood. Policies also determine the number of members that a node can have in its *alliance* (by limiting the number of remote advertisements this node can cache). An *alliance* of a node can span across multiple hops. The advertisements in that case also are sent across multiple hops.

When an agent needs to discover a certain service, it first looks at its local platform to check whether that service is available. On failure, by looking at its own cache, it checks the members of its *alliance* to discover the service. If the service is still unavailable, the source platform tries to broadcast or multicast the request to other *alliances* in its vicinity. The local policy determines the method (broadcasting or multicasting). Multicasting is used to prevent broadcast storms in the network. We use policy-based multicasting where the node multicasts the request to other nodes in its vicinity where there are greater chances of obtaining the service. A node on receiving a service request chooses, based on its policy to either process it or drop this request.

Our flexible approach towards *alliance* formation does not have the overhead of explicit leader election. Every device in this environment is self-sufficient. However, it utilizes resources/services in the vicinity whenever they are available. Dynamic network topology changes are automatically reflected in the *alliances* that are formed. A node does not need to register or deregister with the neighborhood alliances when it changes its location. Our policy-based approach towards *alliance* formation buys us the advantages of a compound-based agent formation mechanism. Using such policies, more traditional compounds like those utilizing explicitly elected leaders can also be achieved. With the help of policy, our architecture can also take user preferences into consideration. This is a major advantage since users of mobile commerce applications need the ability to control the ways in which their own resources are utilized.

4.5 Preliminary Work Summary

In this section, we have presented at a set of projects that we have worked on that are related to the proposed research. We have described Numi project, eNcentive porject, Agents2Go project and Allia Poject.

Chapter 5

Proposed Approach

In this chapter, we present our proposed approach to address the set of challenges described in the previous chapter. We propose development of a middleware framework that manages digital goods and services in pervasive environments. The framework incentivizes collaborations by supporting peer to peer bartering for digital goods and services. Devices that are driven by self motivated economic goals will be capable of trading and exchanging their goods and service in a personalized and context aware manner. Our proposed middleware framework will employ bartering strategies and protocols that will allow taking advantage of personal relationship that exist between devices in pervasive environments. This is a departure from the conventional exchanges that occur in traditional peer to peer communications frameworks. We will employ valuation based description mechanisms to evaluate digital goods and service. This will optimize exchanges and discourage free riding behavior which is frequently exhibited in conventional peer to peer systems.

5.1 Bartering Model

The proposed bartering model will support context-based, personalized, continuous bartering with peers for obtaining new goods and services and for lending/selling/giving out local goods and services.

To increase the pool of potential solutions to the bartering problem of *coincidence of double wants*, our bartering model will employ a set of strategies that will exploit on device to device relationships. Through this, our bartering model departs from the traditional p2p approach of anonymous exchanges and ad hoc altruistic collaborations.

Our bartering model will rely on the proposed valuation description model to allow devices to reason over the fairness of potential exchanges.

5.1.1 Valuation of Digital Goods and Services

At the center of our design lies valuation of digital goods, services and information. We categorize valuations into two key dimensions: *Value in Use* and *Value in Exchange*.

- ***Value in Use*** - is the value of the particular electronic good or service for the particular user or device.
- ***Value in Exchange*** - reflects the potential value of the service or good against any other service in *value-for-value exchanges*.

Value in Use reflects the perceived significance, importance and usefulness that a device/user receives from a particular service. In essence, the service value is determined by evaluating how this service satisfies the needs of the device and it's user. Different users might have different perception of the *value in use* for the same service. Thus *value in use* based description is a highly personalized description of services and information. Each device in the environment will form a value-based view of its collection of services. This approach places personalized valuation at the center of our service management middleware design.

Value in exchange reflects the potential value of a service against any other service in *value-for-value exchanges*. Value in exchange of a good or a service is not dependent on its perceived value in use. In essence, a good or a service could have no personal use or value to a particular user but it could prove to be very valuable to another user or device. By trading away a good or a service, a user or a device can receive indirect satisfaction by acquiring some other needed good or service. Use of *value in exchange* allows context aware environmental management of pervasive environments. Devices that are driven by self motivated economic goals will be moved to obtain goods and services that are in demand in their computing environment. These devices will proactively identify digital goods and services with a high *value in exchange* index. Such goods and services will thrive in computing environments based on their demand.

On the other hand, goods and services that are not in demand and hence have a low value in exchange index will automatically be cleansed out from the environment. This proposed economically driven collaboration approach will ensure context aware interactions between devices in pervasive environments that aim to maximize the utility factor of these environments.

5.2 Aspects of Value

There are a number of aspects that factor into determining the value for a specific service. These aspects are discussed below.

- ***Systemic Value*** represents the essence of the digital good or service. For instance, the essence of printing service is its printing functionality. On the other hand, color printing is an additional attribute that enhances the value of this service but does not alter the *systemic valuation* of this service. So if the user was simply interested in printing a document and did not have a preference of quality, then the color attribute would be irrelevant to the valuation of the printing service. Systemic value reflects the importance of the service being present on the device.

- ***Liquidity*** reflects the ease with which a service can be acquired or exchanged for any other service without a significant loss in value. *Simplicity* and *similar value* are the key factors of liquidity. During bartering in pervasive environments, a service or a good is considered *highly liquid* if it can be exchanged frequently by many devices for services of a similar value. Services with higher *liquidity index* would be in greater demand in the pervasive environment and could be used by devices as exchange tokens.

- ***Context Based Value*** is the value that describes importance of the service, information or an electronic good in a particular context. For example, in the context of traveling, the digital map service is more valuable asset than a cooking recipes catalog service. If a service is used in a number of contexts, then it has a high *contextual index*. Similar to *liquidity based* valuation, services that have high contextual index could evolve into a token for exchanges and bartering in pervasive environments.

- ***Original Cost*** of a service or information reflects resources or money that had to be spent to initially acquire that service. Original cost of obtaining a service or information is not given a substantial importance during the valuation process particularly in determining value in exchange. Importance or usefulness of a service or information is not necessarily connected to the initial cost. For example, a service or information could have been very expensive to acquire but turned out to have very little utility value. As mentioned earlier in Chapter 2 information and digital goods are inherently very expensive to produce but cheap to reproduce. Thus if DRM mechanisms are not employed, prices of digital goods and services eventually come down to zero. However, if digital goods and services are bartered for another set of goods and services, there is no consistent force that pushes down the prices.

- ***Historic Valuation*** refers to the temporal variations in the value of a good or service that have been observed in the past. This information can be used to extrapolate future expected valuation for a given service. *Original Cost* is the first record of the historic value for a given service. The relationship between *Original Cost* and *Historic Valuation* is highly dependent on the nature of the service and context. *Historic Valuation* is a measure that is based on past experiences with this service or similar services and goods. *Historic Valuation* subdivides into two categories of *personal experiences record* and *external recommendation record* which includes experiences that other devices shared during the collaborations.

- ***Store of Value*** refers to how consistently can the service or a good retains its value. Value of the services, information and goods can depreciate over time. Service and information could have expiration time and date. DRM mechanisms could limit replication of an item. Similarly a good or a service can appreciate over time. If an electronic good or a service is some form of memorabilia it could become more valuable. Another example appreciation of a digital good is a song that was acquired before it became a top hit once it becomes a new "must have" song, its value correspondingly increases.

- ***Transaction Cost*** describes costs involved in discovering the good or a service, bargaining for that item and transferring data and verifying the correctness and finalization of transaction.

- ***Maintenance Cost*** describes the costs associated with keeping goods services and information up to date. For example, a cell phone with limited memory capacity storing a large, high resolution photograph is incurring a significant maintenance cost since very high percentage of its memory is occupied by this photograph. At the same time the same photograph on a laptop has significantly lower maintenance cost.

- ***Attribute Based Valuation*** represents importance of the attributes or functionalities of the service. In essence, each service has a set of attributes or functionalities that it can perform. Service attributes and functionalities would have a value function that reflects the perceived significance of that attribute or functionality. For example if a device has a printer/fax/scanner service it might assign a higher value to the printer functionality, lower value to the fax and scanner functionality.

- ***Composability Index*** reflects on how many other services that a particular service be composed with. It represents how *"handy"* the service is. For example, a spell checker service can be used in wide range of applications. In case of valuation of the service for exchange, *composability index* is proportional to the number of the services in the environment that the service could be composed with in that environment. In the pervasive environments, services with high *composability index* would be in greater demand, since they are more likely to be needed to realize more services that can be composed on the fly.

- ***Frequency of Use Valuation*** reflects the frequency with which the service or a good is being used on the device. A service that is used more often has a higher value.

5.2.1 Valuation Model

Each service on the devices will be evaluated in the terms of its perceived *value in use* and it's perceived *value in exchange.*

Value in Use Value of the service for internal use of the device is represented in the following aspects of the valuation:

- *Systemic Value*
- *Frequency of Use Valuation*
- *Context Based Value*
- *Attribute Based Valuation*
- *Maintenance Costs*
- *Store of Value*

Value in Exchange

Service value in exchange is represented by the following aspects that have environmental and communication related influences.

- *Liquidity*
- *Transaction Costs*
- *Service Maintenance Costs*
- *Store of Value*
- *Composability*

5.3 Relationship Based Bartering

To expand the space of possible deals, we propose to exploit social relationships between devices. We categorize relationships between devices into five categories.

- *Self Devices* - are personal devices that belong to a single user. These devices have different capabilities and purposes but together they form a personal computing environment. Examples of *self devices* would be: John's cell phone, John's PDA, John's laptop. Bartering between *self devices* would be motivated by optimization and personalized load balancing of tasks and resources.

- *Sibling Devices* - are devices that belong to a set of different users that form a tight circle of family and friends. Devices belonging to theses users typically have same higher level goals but might have conflicting sub goals. Example of *sibling devices* would be: John's cell phone, John's sister's PDA, John's best friend's laptop.
- *Friendly Devices* - are devices that belong to different users but frequently come in contact and collaborate with one another. An example of *friendly devices* would be PDAs that belong to John and Bob who are coworkers who come in contact with one another three times a week. Level of *friendship* and collaborative interactions can vary from device to device and from user to user.
- *Self Devices* and *Siblings Devices* are special cases of friendliness. Friendly devices could also further divided into subcategories and priority levels.
- *Unfriendly Devices* - are devices that are actively uncooperative. These devices have existing relationships of conflicting interactions. An example of unfriendly devices will be rival competitors that are actively trying to sabotage each others states and interactions.
- *Stranger Devices*- are devices that just meet and are most likely never to come in contact again. Example of such devices would be two PDAs discovering each other in a shopping mall, or in an airport or on a highway.

Our proposed middleware will employ bartering strategies that take advantage of these relationships. During bartering, the level of cooperation during *value for value exchanges* is directly tied to the relationship between the devices. In all of the strategies, regardless of the relationship between devices, the priority is given to the exchanges of services of even or close values. The strongest *sibling* relationship is given the largest leeway in disparity of valuations. *Siblings* could potentially provide the services to each other and receive no other service in return. However, priority is given to service exchange and not to gifts. *Siblings* also could agree for compensation at a later time and basically enter into a “long term” contract. Strategy for exchanges between strangers is much tighter. Thresholds of tightness are personalized for each device. Trust and Role based relationships could also be used to determine the level of cooperation.

One approach of bootstrapping our relationship model is to explicitly hard code relationships for every participating device. Another approach is to use already existing social networks such as LinkedIn[6] Orkut[7] to initialize the relationship network on the devices. Our work will not focus on bootstrap mechanisms. Our approach assumes that the devices are already armed with the relationship network information that could be used for reasoning during bartering. Similarly, though it is not part of our proposed research, our framework could be used as a feedback mechanism for relationship network maintenance.

Our proposed approach is different from other trading protocols and systems where all participants are treated equally as untrusted and unfriendly strangers. There are frameworks that differentiate between interacting partners based on the previous cooperation {ICDCS04}. Our proposed design does not restrict relationship to the levels of previous cooperation. Our proposed framework allows devices to explicitly identify relationship. This functionality is important since it does not necessarily require use of learning techniques and allows explicit closeness between *Self Devices* and *Sibling Devices*. In essence *Self Devices* and *Sibling Devices* "can do no wrong".

5.4 Bartering Structures

Our proposed middleware uses three basic structures to facilitate bartering.

- ***p****Have* – is a list of goods and services that a device has. This list is only for internal device use. Services that are on the ***p****Have* list are not for trading. If the service is falling out of favor it could be moved to ***i****Have* list.

- ***i****Want* – list of goods and services that a device desires and is actively searching for. The intensity of the search is represented by *focus factor*. On the ***i****Want* list, goods and services with high *focus factor* are aggressively searched for. Personalized thresholds can be set to identify the activity level. When a service is acquired, the record is moved to either the ***i****Have* list or ***p****Have* list depending on the intended purpose of the newly acquired good or service.

- ***i****Have* – list of goods and services that a device is willing to disclose. Digital goods and services on this list can be used as payment in bartering transactions. In addition, ***i****Have* services can be actively distributed by driving the device to actively advertise the precents of the good or service. *Focus factor* is used to represent the activity level. On the ***i****Have* list, goods and services with high *focus factor* are aggressively advertised for distribution. Goods and services that have low *focus factor* are reluctantly used as payment during bartering.

Determination of the *focus factor* is outside the scope of the proposed research. Our proposed research simply provides the mechanism and incorporates it into the management decision process. *Focus factor* for a service on the ***i****Have* and ***i****Want* list can be explicitly specified by the user. *Focus factor* potentially could also be derived from user's personal profile. *Value in use* and *value in exchange* could also play a role in determining the *focus factor*.

Every service that is listed in the ***i****Want*, the ***i****Have* lists, is listed along with it's value based description. The description is composed of *value in use* description and *value in exchange* description. Both of the descriptions are discussed in detail in Section 5.2.

5.5 Bartering Protocol

There are two key actors in our bartering protocol: the *initiator* and the *participant*. The *initiator* is a device that acts on the goal of acquiring a service or on a goal of selling or actively distributing a good or a service. The device initiating bartering due to the goal of acquiring a service or a good is executing the *pull based communication model* variation of the bartering protocol. The device initiating bartering due to the goal of distributing of selling of a service or a good is executing the *push based communication model* variation of the bartering protocol. A *participant* is a device that replies to the initiators inquiry.

The basic bartering protocol starts when the *initiator* broadcasts the *discovery message*. In case of a *pull based model*, the *discovery message* contains an inquiry about a good or a service that is being sought out by the *initiator*. Devices that are capable of offering the good or service respond with a *reply to discovery message*. In case of a *push based model*, the *discovery message* contains solicitation for acquisition of a service or a good that the device is selling or distributing. Devices that are interested in acquiring this good or service send a *reply to discovery message*. Regardless of the communication model once the *initiator* receives the *reply to discovery message*

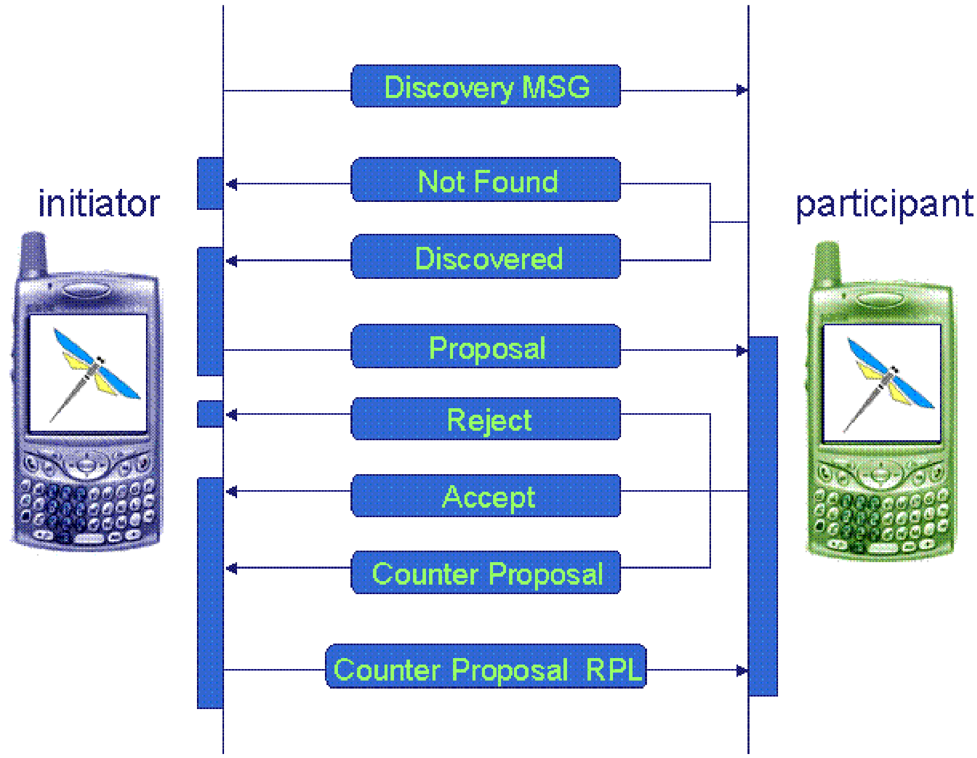

Figure 5.1: Bartering Protocol

it composes a *proposal* and sends it to the *participant*. The *proposal* is composed by the *Bartering Reasoner*. If multiple *participants* respond to the *discovery message*, the *Bartering Reasoner* uses the *relationship network* to determine the sequence in which the *participants* should be approached and bartered with. The structure of the proposal is dependant on the communication model. For the *pull based model*, the *proposal* is composed of the detailed definition of the good or service that is being sought out and a set of goods and services that is being offered as payment. The determination of what services could be used as payment is done by the *Bartering Reasoner*. For the *push based model*, the *proposal* is composed of the detailed definition of the good or service that is being offered and a set of goods and services that it can accepted as payment. The description of the proposal compositions can be found in Section5.6. The *proposal* is sent to the *participant*. The *Bartering Reasoner* on the participant's side analyzes the proposal and replies with either a *rejection message*, or an *acceptance message* or a *counter proposal message*. The description of how *Bartering Reasoner* arrives to the response can be found in Section 5.6. In case of the rejection, the *initiator* terminates the bartering protocol with the participant (in regards to this particular good or service) The *initiator* uses *Bartering Reasoner* to determine the next set of actions. If the reply to the proposal is an accept then the devices conduct the transaction. If the response is a *counter proposal* then the *initiator's Bartering Reasoner* evaluates the *counter proposal* and acts in accordance to the evaluation. *Bartering Reasoner* evaluation details can be found in Section[].

5.6 Bartering Reasoner

To manage bartering communications and reason over the strategies and relationships and valuations that need to be applied we propose to develop and incorporate into our proposed framework, a component called *Bartering Reasoner.* The *Bartering Reasoner* will be responsible for producing decisions that will affect the path of collaborations and bartering.

In essence, Bartering Reasoner will be answering a set of questions that could potentially arise during collaborations and bartering.

Every collaboration or bartering will be initiated by one of the two following questions: "***What do I want?***" or "***What am I offering?***". The first question belongs to the *pull based communication model.* It will be asked by a device that is looking to acquire a service or a good. *Focus factor* from the ***i****Want list* can be used to identify such a service or good. The second question will be asked by the device that is soliciting an acquisition of a good or a service that it owns. *Focus factor* from the ***i****Have list* can be used to identify such service or good. Both of these questions will be trigged by an external element, for example a user of the device. The *discovery message* will be sent out to the devices in the environment to identify possible bartering partners.

Responses to the discovery message that was sent out by the initiator will come from a set of devices in the environment. The *Bartering Reasoner* is responsible for answering the question of "***Who do I want to barter with?***". The *Bartering Reasoner* can employ a set of strategies to determine the sequence of bartering opponents. For example, Bartering Reasoner can use *relationship network* to identify the devices that are most friendly and approach them first.

The *Bartering Reasoner* is responsible for the *proposal* composition. The proposal will be composed once the device identifies the potential bartering partners. The proposal consists of two critical components. If the *pull based communication model* is used, then the first component is the good or a service that device would like to acquire. The second component is the set of goods or services can be used as payment. Thus, another important functionality of the *Bartering Reasoner* is to answer the question of "***What do I offer in return?***". The reasoner makes this determination by comparing the perceived value of the service that it is about to acquire and the perceived value of the service that it is planning to use as payment. Determination of perceived values is accomplished by identifying perceived *values in use* and identifying perceived *values in exchange* for both sets of services. If the *push based communication model* is used, then the first component is the good or a service that the device would like to dispose of. The second component is the set of goods or services can be used as payment for the service that is being offered. It is possible that this set is empty which will imply that the service it being gifted away. It is possible that the a device will be willing to take anything else in place of the offered good or service, this can be represented by a wild card.

The *Bartering Reasoner* is responsible for determining the response action to a received proposals. Thus, yet another functionality of Bartering Reasoner is to answer the question of "***Do I accept, Do I reject, or Do I counter-propose?***". As in case of proposal composition, the decision is made based on the perceived valuation of the goods and services that are involved in the exchange. Valuation is tied to the determination of the perceived *value in use* and the perceived *value in exchange*.

The *Bartering Reasoner* is in charge of accepting the received proposals sent by peer devices. So, the answer to the question of "***Do we have a deal?***" is also the responsibility of the *Bartering Reasoner*. This decision is also reached by evaluating the perceived v*alue in use* and perceived *value in exchange* of the services and goods in the proposal.

The *Bartering Reasoner* is responsible for composition of the counter proposal "***What is my counter-proposal***?" The counter proposal is composed when a device identifies a marginally better deal. The determination of the better deal is done by examining the proposal and evaluating the services and goods that compose the proposal. Perceived v*alue in use* and perceived *value in exchange* are used in of the decision-making process.

5.7 Expected Contributions

The expected contributions of the proposed research are the following:

- Development and implementation of a framework that employs bartering to incentivize collaborations.
- Development and implementation of an intelligent Bartering Reasoner.
- Development of value based descriptions for electronic goods and services.
- Development and implementation of context based personalized bartering protocols / strategies that employ:
  - Relationship Networks
  - Valuation-Based Bartering
- Demonstration of P2P environment that incentivizes collaboration between peer devices.

5.8 Summary

In this chapter, we described our proposed research approach which involves development of middleware framework for peer to peer collaborations in pervasive environments. Proposed middleware will manage digital goods and services and supports device to device service and good exchanges and other collaborations by utilizing bartering techniques. In this chapter, we also described a set of mechanism and protocols that our middleware will employ to ensure optimal and fair interactions.

Chapter 6

Research Plan

Our research plan consists of performing the following tasks in order to achieve the goals that are identified in this thesis.

1. Development of value based descriptions for digital goods, services and information.

2. Development and implementation of a management component that employs these descriptions.

3. Development and implementation of a bartering support engine.

4. Implementation of bartering protocols and strategies.

    - Analysis of effectiveness of these protocols and strategies.
    - Incorporate and study effects of DRM.

Before we start the work on the above described tasks, we will develop an interactive environmental emulator that is capable of supporting emulation of bartering and collaborations of dozens of peers that are running our framework. In particular, we plan to support two types of configurations for peers.

- The first configuration is the fully automated prototype. Fully automated platforms will move along preprogrammed paths and will be capable of bartering with preprogrammed plans for buying and selling electronic services and goods. These platforms will provide background for the second configuration which is semi automated.

- The semi automated prototype will be controlled by a user in real time. This configuration of the prototype will allow users to choose a movement path and will permit a manual relocation from one cell to another. Bartering will also be semi automated. User will have an option to manually control *focus factor* of goods and services on ***i****Wish* and ***i****Have lists*. User will also have a chance to monitor bartering. User will have control over the final say in acquiring and selling of electronic goods and services.

Portions of the emulator functionality are already built and functioning as part of research work which was described in the preliminary work chapter.

Task A:

We will develop value based description that would be used by the emulator to facilitate the value for value bartering. In particular, we will focus on development of *value in use* and *value in exchange* descriptions.

Task B:

We will also implement a management component that will use these descriptions to support service bartering and other interactions between devices and also to organize devices and to control information pollution.

Task C:

We will develop and implement a bartering reasoner that will assist our proposed framework in the following tasks:

- The task of selecting the appropriate bartering peers.
- The task of composing the proposals.
- The task of determining the response to received proposals, in particular:
  - The decision of accepting, rejecting or counter proposing.
- The task of composing counter proposals.

Task D:

We will implement bartering strategies and study their effectiveness.

- We will start with implementation of role based negotiations. First, we will explore effects of social relationships. In particular, we will first implement *friendly* type relationships and study on bartering outcomes and how it effects service proliferation in pervasive environments. We will follow up with other relationships to demonstrate the effects of social relationship diversity on the proliferation of goods and services and on the levels of collaborations in pervasive environment.

- We will also implement contextual strategies of bartering where time and location of a bartering transaction will play a role. The effectiveness and efficiency of outcome will be the goal of this activity.
- We will study the effects of Digital Rights Management (DRM) restrictions on barter models and proliferation of electronic goods and services. We will also look at how bartering models can support and encourage use of DRM.

6.1 Success Metrics

Currently, there is no complete work that is similar our proposed research. Therefore, completion of design, development, testing and refinement will be the extent of our work. We believe that our proposed research will become complete once the goals which we have divided into above described tasks have been accomplished. In addition to that, we expect that our final work will be scalable enough to support several hundred devices.

6.2 Time Line

We have arranged our schedule along 15 month timeline. The end points of tasks are estimations and are flexible. The endpoints could be adjusted due to an internship. Many of the items in the time line are very much dependant on each other. Progress of any one of the tasks will affect the time line for the following task. Nonetheless, we will work on the tasks in the sequence that is listed below. The last stage of our work will be mainly focused on testing, refinement and finalization of our research. We will conclude our work with the writing of the PhD dissertation and PhD defense.

Task	Description	Completion Date
Task A	Emulator for pervasive environment	Nov 05
Task A	Value in Use & Value in Exchange	Dec 05
Task B	Value Based Management Module	Jan 06
Task C	Initial version of Bartering Reasoner	Feb 06
Task D	Role Based Bartering Strategies	Mar 06
Task D	Contextual Bartering Strategies	Apr 06
Task C	Bartering Reasoner	Jun 06
Task D	Effects of DRM on Bartering	Sep 06
	Testing & Refinement & Finalization	Oct 06
	Writing PhD Desertion	Dec 06
	Defense	Jan/Feb 07

Bibliography

[1] Bluetooth White Paper, WWW, http://www.bluetooth.com/developer/whitepaper/.

[2] IEEE 802.11TM WIRELESS LOCAL AREA NETWORKS, WWW, http://grouper.ieee.org/groups/802/11/.

[3] Napster, World Wide Web, http://www.napster.com.

[4] Gnutella, World Wide Web, http://www.gnutella.com.

[5] KaZaA, World Wide Web, http://www.kazaa.com.

[6] LinkedIn, World Wide Web, http://www.LinkedIn.com.

[7] Orkut, World Wide Web, http://www.orkut.com.

[8] Eytan Adar and Bernardo A. Huberman, *Free riding on gnutella*, First Monday (2000).

[9] Kostas G. Anagnostakis and Michael B. Greenwald, *Exchange-based incentive mechanisms for peer-to-peer file sharing*, ICDCS '04: Proceedings of the 24th International Conference on Distributed Computing Systems (ICDCS'04) (Washington, DC, USA), IEEE Computer Society, 2004, pp. 524–533.

[10] F. Bergenti and A. Poggi, *LEAP: A FIPA platform for handheld and mobile devices*, presented at ATAL, 2001.

[11] Dan Chalmers, *Contextual mediation to support ubiquitous computing*, Ph.D. thesis, Imperial College London, 2002.

[12] Dan Chalmers, Naranker Dulay, and Morris Sloman, *A framework for contextual mediation in mobile and ubiquitous computing applied to the context-aware adaptation of maps*, Personal Ubiquitous Comput. **8** (2004), no. 1, 1–18.

[13] Connected Limited Device Configuration, http://java.sun.com/products/cldc/.

[14] PersonalJava Application Environment, http://java.sun.com/products/personaljava/.

[15] Steve Fickas, Gerd Kortuem, Jay Schneider, Zary Segall, and Jim Suruda., *When cyborgs meet: Building communities of cooperating wearable agents*, October 1999.

[16] G. Wu and C. W. Chu and K. Wine, J. Evans and R. Frenkiel, *Winmac: A novel transmission protocol for infostations.*, 1999.

[17] Andreas Heinemann and Tobias Straub, *An anonymous bonus point system for mobile commerce based on word-of-mouth recommendation*, Proceedings of the 19th Annual ACM Symposium on Applied Computing (ACM SAC 2004) Nicosia, Cyprus, 2004.

[18] Autonomous Zone Industries, *MojoNation*, http://www.mojonation.com/.

[19] Micro Edition Java 2 Platform, `http://java.sun.com/j2me/`.

[20] William Stanley Jevons, *Money and the mechanism of exchange*, D. Appleton and Company, 1875.

[21] K. Keegan and G. O'Hare, *Easishop: Context sensitive shopping for the mobile user through mobile agent technology*, 2002.

[22] Stephen Keegan and Gregory M. P. O'Hare, *Easishop: Enabling ucommerce through intelligent mobile agent technologies.*, Proceedings of 5th International Workshop, Mobile Agents for Telecommunication Applications, MATA 2003, Marakech, Morocco, October 8-10, 2003, Proceedings, 2003, pp. 200–209.

[23] S. B. Kodeswaran, O. Ratsimor, A. Joshi, T. Finin, and Y. Yesha, *Using peer-to-peer data routing for infrastructure-based wireless networks*, IEEE International Conference on Pervasive Computing and Communications (PerCom 2003) (Fort Worth, TX, USA), March 2003.

[24] Gerd Kortuem, *A methodology and software platform for building wearable communities*, Ph.D. thesis, University of Oregon, 2002, Adviser-Zary Segall.

[25] Gerd Kortuem., *Proem: a middleware platform for mobile peer-to-peer computing*, SIGMOBILE Mob. Comput. Commun. Rev. **6** (2002), no. 4, 62–64.

[26] Gerd Kortuem, Jay Schneider, Dustin Preuitt, Thaddeus G.C. Thompson, Stephen Fickas, and Zary Segall., *When peer-to-peer comes face-to-face: Collaborative peer-to-peer computing in mobile ad hoc networks*, Proceedings 2001 International Conference on Peer-to-Peer Computing, Aug, 2001.

[27] Ting-Peng Liang and Jin-Shiang Huang, *A framework for applying intelligent agents to support electronic trading*, Decis. Support Syst. **28** (2000), no. 4, 305–317.

[28] Seng Wai Loke and Arkady Zaslavsky, *Integrated ambient services as enhancement to physical marketplaces*, HICSS '04: Proceedings of the Proceedings of the 37th Annual Hawaii International Conference on System Sciences (HICSS'04) - Track 9 (Washington, DC, USA), IEEE Computer Society, 2004, p. 90284.3.

[29] Mobile Information Device Profile (MIDP), http://java.sun.com/products/midp/.

[30] Andrew Odlyzko, *The case against micropayments*, Financial Cryptography 2003 (Minneapolis, MN), Springer Verlag, February 2003, Lecture Notes in Computer Science.

[31] Santiago Ontanon and Enric Plaza, *Cooperative case bartering for case-based reasoning agents*, CCIA '02: Proceedings of the 5th Catalonian Conference on AI (London, UK), Springer-Verlag, 2002, pp. 294–308.

[32] F. Perich, S. Avancha, D. Chakraborty, A. Joshi, and Y. Yesha, *Profile driven data management for pervasive environments*, 3th International Conference on Database and Expert Systems Applications (DEXA 2002) (Aix en Provence, France), September 2002.

[33] Filip Perich, *On peer-to-peer data management in pervasive computing environments*, Ph.D. thesis, University of Maryland, Baltimore County (UMBC), 2004, Adviser-Anupam Joshi.

[34] Danny Quah, *Digital goods and the new economy*, CEP Discussion Papers dp0563, Centre for Economic Performance, LSE, March 2003.

[35] O. Ratsimor, D. Chakraborty, S. Tolia, D. Kushraj, A. Kunjithapatham, G. Gupta, A. Joshi, and T. Finin, *Allia: Alliance-based service discovery for ad-hoc environments*, 2002.

[36] O. Ratsimor, V. Korolev, A. Joshi, and T. Finin, *Agents2go: An infrastructure for location-dependent service discovery in the mobile electronic commerce environment*, ACM Mobile Commerce Workshop held in conjunction with MobiCom 2001. (Rome, Italy.), July 2001.

[37] Olga Ratsimor, Tim Finin, Anupam Joshi, and Yelena Yesha, *encentive: a framework for intelligent marketing in mobile peer-to-peer environments*, ICEC '03:

Proceedings of the 5th international conference on Electronic commerce (New York, NY, USA), ACM Press, 2003, pp. 87–94.

[38] Olga Ratsimor, Sethuram Balaji Kodeswaran, Anupam Joshi, Timothy Finin, and Yelena Yesha, *Combining infrastructure and ad-hoc collaboration for data management in mobile wireless networks*, Workshop on Ad-hoc Communications and Collaboration in Ubiquitous Computing Environments, Nov 2002.

[39] Olga Vladi Ratsimor, Dipanjan Chakraborty, Anupam Joshi, Tim Finin, and Yelena Yesha, *Service discovery in agent-based in pervasive computing environments*, Journal on Mobile Networking and Applications (2003), Special issue on Mobile and Pervasive Commerce.

[40] R.H. Frenkiel, B.R. Badrinath, J. Borras, and R. Yates,, *The infostations challenge: Balancing cost and ubiquity in delivering wireless data*, IEEE Personal Communications **7(2)** (2000), pp.66–71.

[41] Howard Rheingold, *Smart mobs: The next social revolution*, Perseus Publishing, 2002.

[42] William Rosenblatt, Stephen Mooney, and William Trippe, *Digital rights management: Business and technology*, John Wiley & Sons, Inc., New York, NY, USA, 2001.

[43] Norman Sadeh, *Mobile commerce: New technologies, services and business models*, Wiley, 2002.

[44] Norman Sadeh, Ting-Chak Chan, Linh Van, OhByung Kwon, and Kazuaki Takizawa, *Creating an open agent environment for context-aware m-commerce*, Agentcities: Challenges in Open Agent Environments (LNAI, Springer Verlag) (Burg, Dale, Finin, Nakashima, Padgham, Sierra, and Willmott, eds.), 2003, pp. 152–158.

[45] Nigel Shadbolt, *Ambient intelligence*, IEEE Intelligent Systems **18** (2003), no. 4, 2–3.

[46] Upkar Varshney, *Location management for mobile commerce applications in wireless Internet environment*, ACM Transactions on Internet Technology **3** (2003), no. 3, 236–255.

[47] Upkar Varshney, *Vehicular mobile commerce*, Computer **37** (2004), no. 12, 116–118.

[48] Upkar Varshney and Ron Vetter, *Emerging mobile and wireless networks*, Commun. ACM **43** (2000), no. 6, 73–81.

[49] Upkar Varshney and Ron Vetter, *Mobile commerce: Framework, applications and networking support*, Mob. Netw. Appl. **7** (2002), no. 3, 185–198.

[50] Upkar Varshney, Ronald J. Vetter, and Ravi Kalakota, *Mobile Commerce: A New Frontier*, Computer (2000), 32–38.

[51] Vivek Vishnumurthy, Sangeeth Chandrakumar, and Emin Gün Sirer, *KARMA: A secure economic framework for peer-to-peer resource sharing*, Proceedings of the Workshop on Economics of Peer-to-Peer Systems (Berkeley, California, USA), June 2003.

[52] Mark Weiser, *The computer for the 21st century*, (1995), 933–940.