A FRAMEWORK FOR CASCADING PAYMENT AND CONTENT EXCHANGE WITHIN P2P SYSTEMS

GURLEEN KAUR ARORA, B.SC. (HONS)

PH.D. THESIS 2006

A FRAMEWORK FOR CASCADING PAYMENT AND CONTENT EXCHANGE WITHIN P2P SYSTEMS

GURLEEN KAUR ARORA B.SC. (HONS)

A thesis submitted in partial fulfilment of the requirements of Liverpool John Moores University for the degree of Doctor of Philosophy

November 2006

Networked Appliances Laboratory
School of Computing and Mathematical Sciences
Faculty of Technology and Environment
Liverpool John Moores University

For my Mother and Father

&

Grandfather Late Sardar Hardit Singh

15th Oct 1925 - 5th Oct 2004

ACKNOWLEDGEMENTS

First and foremost, I would like to express my deep gratitude, honour, and admiration for my supervisors Prof. Madjid Merabti and Dr. Martin Hanneghan. Words can not express the gratitude I feel for both these excellent people whose help and experience I could not have done without through this very long and arduous journey and for believing in me when I almost lost hope. I shall always be indebted to them for bringing me to this stage in my life. I feel sure that without their efforts and enormous dedication, this thesis would never have materialised. I would like to express my appreciation to my supervisor, Dr. Martin Hanneghan who has always been there ready to provide support and honest advice and to read my chapters and papers time and time again.

I would like to thank Dr. Dhiya Al-Jumeily without whose encouragement, I would not have embarked on this journey, let alone be here today.

Thanks are also due to the School of Computing and Mathematical Sciences (CMP), the Universities UK - Overseas Research Students Awards Scheme, and Liverpool John Moores University, for the financial support provided during the course of this study.

I have met some wonderful, kind and helpful people along the way, they all know who they are and how much their support and friendship means to me. I am indebted in particular to Dr. Omar Abuelma'atti, Dr. Paul Fergus, Mrs. Huma Javed, Mr. Jasim Saeed, and all my other friends and colleagues within the department for the many hours of discussions and the moments we shared during the many ups and downs of our years. Some of them are mentioned here; Mr Muhammad Arshad, Mr Faycal Bouhafs, Mr Henry Chang, Miss Aisha Khan, Dr David Llewellyn-Jones, Dr. Anirach Mingkhwan, Dr. Hala Mokhtar, Mrs. Carol Oliver, Prof. A. Taleb-Bendiab. Thanks are due to the Technicians Team for their support and help in desperate times of need.

Other friends in the UK who have been a source of immense support and encouragement need a mention here as well; Mohamed Abbas, Yosri Abbas and Souhila Serir have always been by my side during the course of my stay in the UK making it difficult to miss home.

And finally, although it might seem that I have left them to the last but that is only due to the immense amount of pride I feel in thanking my parents and sister for all their love and support through this very emotionally and physically stressful time of my life. Words can not express the gratitude and love I feel for them; their constant interest in my efforts, and faith in my abilities has proved a continual source of inspiration for me. My sister Harpreet in particular, has shown great resilience in being able to accommodate my many moments of misery and I would like to offer her my heart felt thanks for all her efforts in supporting me through my Ph.D. I would also like to thank my extended family for their patience in waiting for me to complete this important stage of my life.

Garleen K. Arora

ABSTRACT

Advances in computing technology and the proliferation of broadband in the home have opened up the Internet to wider use. People like the idea of easy access to information at their fingertips, via their personal networked devices. This has been established by the increased popularity of Peer-to-Peer (P2P) file-sharing networks. P2P is a viable and cost effective model for content distribution. Content producers require modest resources by today's standards to act as distributors of their content and P2P technology can assist in further reducing this cost, thus enabling the development of new business models for content distribution to realise market and user needs. However, many other consequences and challenges are introduced; more notably, the issues of copyright violation, free-riding, the lack of participation incentives and the difficulties associated with the provision of payment services within a decentralised heterogeneous and ad hoc environment. Further issues directly relevant to content exchange also arise such as transaction atomicity, non-repudiation and data persistence.

We have developed a framework to address these challenges. The novel Cascading Payment Content Exchange (CasPaCE) framework was designed and developed to incorporate the use of cascading payments to overcome the problem of copyright violation and prevent free-riding in P2P file-sharing networks. By incorporating the use of unique identification, copyright mobility and fair compensation for both producers and distributors in the content distribution value chain, the cascading payments model empowers content producers and enables the creation of new business models. The system allows users to manage their content distribution as well as purchasing activities by mobilising payments and automatically gathering royalties on behalf of the producer. The methodology used to conduct this research involved the use of advances in service-oriented architecture development as well as the use of object-oriented analysis and design techniques. These assisted in the development of an open and flexible framework which facilitates equitable digital content exchange without detracting from the advantages of the P2P domain.

A prototype of the CasPaCE framework (developed in Java) demonstrates how peer devices can be connected to form a content exchange environment where both producers and distributors benefit from participating in the system. This prototype was successfully evaluated within the bounds of an E-learning Content Exchange (ElConE) case study, which allows students within a large UK university to exchange digital content for compensation enabling the better use of redundant resources in the university.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

Chapter 1

1 INTRODUCTION

1.1 Preamble

In the modern age of MP3 players, podcasting [Lee Rainie 2005, Mark Shelstad 2005] and 3G mobile phones it is becoming increasingly clear that the convergence of various technologies has led to a proliferation in the many ways people access information. The world of digital media gives users greater flexibility in how, where and what information they can access. It also has an impact on the flow of information in their lives. Within this climate Peer-to-Peer (P2P) networks provide an easy and cost effective model for digital content dissemination, empowering content producers by removing their dependence on publishing houses for the distribution of their content. However, implementations of this model are flawed with content theft, free-riding[1] and lack of trust (among other problems), which detract from the benefits of this paradigm. There is need for new ventures which discourage these practices while fully utilising many positive benefits of these technologies such as low cost of entry and a natural model for resource scaling with increasing community size. The motivation for this research stems from the convergence of these realities to provide a mechanism for fair and cost effective, efficient digital content exchange for different categories of users.

The favoured economic model for the sale of traditional media services is either subscription or advertising based (where users are given access to content without a subscription because the revenue is generated by ads). In the present day, when users are increasingly discerning and demanding, commodities need to be competitively priced to attract and maintain user interest. This research addresses the need for e-commerce in the P2P domain where efficient content dissemination is balanced with the ability to create a fair market for digital goods.

This Chapter highlights the main motivation factors for this research, where the convergence of technology and market trends result in the need for the development of a delivery system to

[1] Free-riding is caused when all network participants do not contribute equally. This phenomenon in P2P networks causes them to behave as power-law networks where a few nodes are highly connected and their loss can damage the efficiency of the whole network, conversely a majority of the nodes have low connectivity and their loss should not have an adverse effect on the system performance.

address the issues pertaining to e-commerce in the P2P domain. If a mechanism is devised to provide fair content purchases in this medium it would lead to the creation of interesting and sustainable new business models and consumer/ producer markets. The main objectives of the research are listed, highlighting the novel aspects of this work. The Chapter concludes with a structure of the thesis.

1.2 The current state of affairs

In the present day, important trends in technology are converging to help drive the flow of information in users' lives. One of these is the increase in computation power as predicted by Moore [R.R. Schaller 1997]. This prediction is also mirrored in the advancement in connectivity. The increase in computing power has a corresponding effect on the speed of network connectivity [Lawrence G. Roberts 2000]. The decreasing cost of high-speed Internet connections and the availability of powerful network-ready personal computers make it possible to develop global P2P networks.

The concept of P2P computing has been around since the early days of networking when it emerged as a result of decentralising trends in software engineering intersecting with available technology. The trend towards decentralised/distributed computing was mainly inhibited by the fact that centralised systems were easier to manage.

Centralised computing relies on a central server being in-charge of and managing its clients. Although this allows for better control and management of security, in the present Internet age where more and more users are becoming part of the Internet community, the strain on these servers is increasing. Hence it makes sense to utilise all the dormant resources that are available, but not being utilised, by using the P2P paradigm which moves away from centralised computing to a specialised version of client-server computing. Hence, fully utilising the advantages of a decentralised ad hoc heterogeneous environment with increased resource availability, inherent scalability and massive cost reductions for system operation.

Development of technology is usually motivated by user requirements where user life style choices encourage research. Users choose technology that best suits their life styles and makes life easier for them. In the current atmosphere of high speed Internet connectivity, achievable through the use of fast broadband and ADSL connections and with the added connectivity made possible by the invention and wide-scale deployment of Internet ready devices such as desktop Personal Computers (PC), Personal Digital Assistants (PDA), mobile phones, laptops and tablet PCs for example, it is easy to imagine a world where people are permanently connected and can

easily communicate and interact with each other; transcending oceans and continents in a matter of seconds. This connectivity gives users the ability to locate and utilise services at their desktops and while they are on the move.

The advent of the Internet has also seen a proliferation of digital content, where anything that may be converted into a digital format has been made available over the Internet. This includes works such as books and other writings, musical compositions, paintings, computer programs and films. Usually digital content is distributed via traditional channels such as retail outlets and on media such as cassettes, CDs or DVDs. Content producers maintain relationships with traditional content distributors such as publishing houses, record labels, entertainment conglomerates and others to distribute their content. Due to the popularity of the Internet, content distributors have begun distributing content via dedicated web sites, which maintain content on centralised servers. This generates overheads that are passed on to the consumer. Hence, a digital supply chain is created and maintained.

Intellectual property rights are defined as 'the rights given to persons over the creations of their minds'[World Trade Organization 2003]. Copyright protects an area of intellectual property rights, which covers literary and artistic works along with the rights of the producers of these works. Copyright states that the producer of content deserves to get compensated for his effort. The methods of compensation may be classified into monetary and reputation-based compensation [L. Jean Camp 2002], the former involving tangible payment for the producer's effort and the latter ensuring the continued existence of the works i.e. persistence, in order for the creator to get credit for his work. Hence, copyright infringement refers to the violation of the content producers' intellectual property rights.

Currently, the issues of copyright infringement, relating to digital content distribution over the Internet, are overcome by solutions such as centralised web portals where users can download content and pay for them via established payment instruments such as credit cards. Digital Libraries, Online virtual book stores, publisher managed web sites – Journal publishers, newspapers and many other content publishers – provide access to digital content for payment, where no copyright is violated. The recent popularity of iPod devices [Beth Snyder Bulik 2004, Paola Dubini 2005] has also demonstrated the users' need for access to digital media. However, all these solutions are based on the centralised client-server model.

Centralisation forces the burden of maintenance on the provider – the strain on resources necessitates these providers be big institutions and takes the prerogative away from small entrepreneurs. P2P brings the power back into the hands of the small time producers and

encourages entrepreneurship as well as the development of new business models which encourage competitive e-marketplaces.

The Internet has become a powerful distribution channel for digital goods and services. Consequently, it has also developed into a medium for the communication of information required to complete monetary transactions. In the present day this global network is being utilised to provide P2P connectivity to users across the world, consequently making the scope of P2P communities more global and far reaching.

Content producers require modest resources by today's standards to act as distributors of their content and P2P technology can assist in further reducing this cost, thus enabling the development of new business models for content distribution to realise market and user needs [William T. Rupp 2004]. However, many other consequences and challenges are introduced; more notably, the issues of copyright violation, free-riding, the lack of participation incentives and the difficulties associated with the provision of payment services within a decentralised heterogeneous ad hoc environment. Further issues directly relevant to content exchange also arise such as transaction atomicity, non-repudiation and data persistence in ad-hoc environments.

1.3 P2P as a model for content dissemination

The widespread acceptance of Napster[2], Gnutella [Matei Ripeanu 2002b], Freenet [Ian Clarke 2002] and other Peer-to-Peer initiatives for content dissemination have made it obvious that P2P is a viable model for content delivery. Even the entertainment industry (Warner Bros.) [Gary Gentile 2006] recognises the potential of P2P networks as a reliable and cost effective method for movie and media distribution. These initiatives use the Internet as the medium of transport, but instead of the traditional client-server model they use P2P technology to deliver content. Figure 1.1 illustrates the topological difference between the two models. With inherent features such as decentralised control and the distribution of information across peers rather than on central servers, P2P has distinct advantages over the client-server model. As content is distributed across peers there is no single point of failure, so content is always available; each peer brings extra information to the pool and the sum of the parts gives greater benefits to the system than the whole. Also, peers can be lightweight, e.g. mobile phones or personal digital assistants (PDAs), further widening the scope for content delivery and content accessibility. The barriers to starting and growing such systems are also low, since they usually don't require any

[2] Napster, "The Napster Homepage," Available at http://www.napster.com

special administrative or financial arrangements, unlike centralized facilities[Hari Balakrishnan 2003].

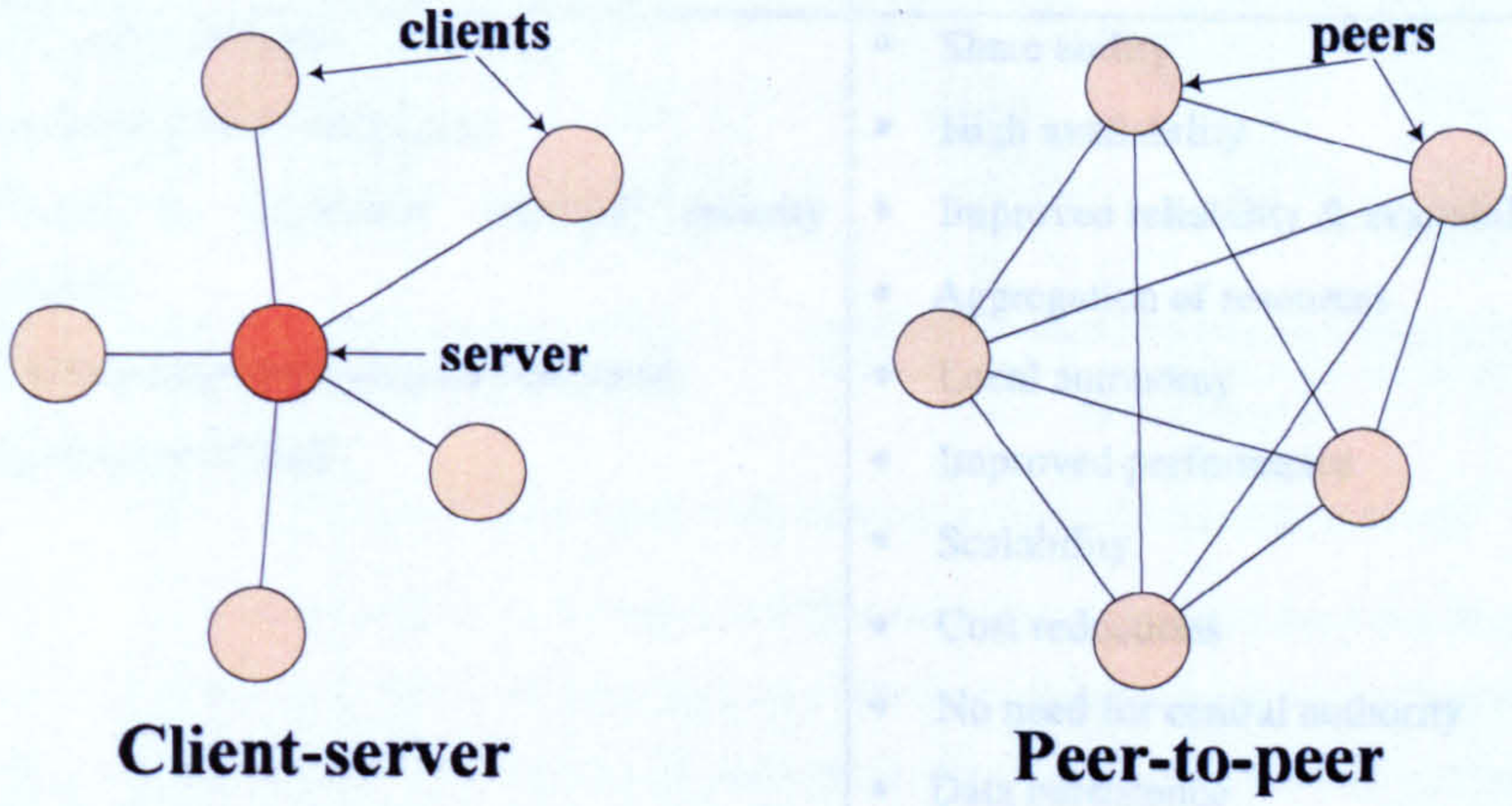


Figure 1.1: Client-Server (centralised) Vs Peer-to-Peer (decentralised)

On the other hand, P2P content sharing networks have their disadvantages; Table 1.1 lists the common technological limitations of P2P systems. However, other less obvious problems of P2P systems do exist which have a social impact on the P2P community. The primary among them are content theft through piracy and free-riding [Eytan Adar 2000].

Analysis by Adar and Huberman [Eytan Adar 2000] showed that almost 70% of Gnutella users did not contribute to the system and that almost 50% of the query results came from the top 1% of sharing hosts. One explanation for the high degree of free riding may be the high cost of contribution in regard to network bandwidth. Current system practices to overcome this problem are to deny service to non-contributing users. However, another negative impact of free-riding to the overall community is the fragmentation of the network which can reduce system efficiency thus detracting from the benefits of the P2P system.

In a P2P system user community, reputation also has a social context apart from the technical issues. Trust in decentralised and more or less anonymous systems has to be based on an ad-hoc reputation mechanism, like peers earning reputation by being a source of high quality material. The heterogeneous nature of the environment makes it difficult for the system to have a persistent and concise global view of the entire network, especially in the case of pure P2P networks. Hence, previous relationships with participating nodes are difficult to maintain. Therefore lack of trust in the system and its users is a common problem in P2P systems.

Table 1.1: P2P limitations and advantages

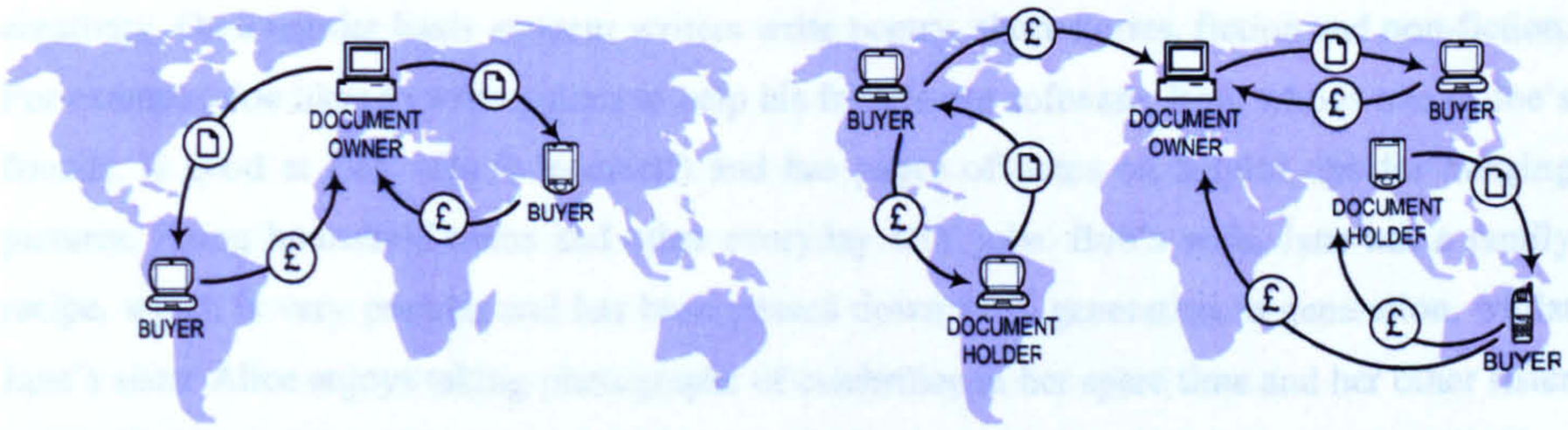
Limitations	Advantages
• Community reliance • Increase in system complexity • Difficult to implement standard security protocols • Novel management techniques required • No guarantee of QoS	• Share ability • High availability • Improved reliability & availability • Aggregation of resources • Local autonomy • Improved performance • Scalability • Cost reductions • No need for central authority • Data persistence

1.4 Challenges for this research

We envision a system where users may come together to form ad hoc communities that can legally share content on the move. At present P2P technology allows users to form ad hoc communities to perform useful functions, but it does not cater for the legal exchange of content for payment in a truly decentralised manner. Hence, in our producer-centric content dissemination scenario (§1.5 pg 7) Alice, Bob, Joe and Jane would all be a part of a community that would allow them to share their content.

This could lead to a complete new set of business models for digital content distribution which could change the way people buy and sell digital content like books, music, movies, software, newspapers, journals, magazines etc. People could create content and, using the power of P2P, distribute (Figure 1.2) it from their desktops at no extra cost. The network would ensure that we get paid for our content every time it is used or propagated. The removal of the overheads of maintaining servers and websites would make digital content available at cheaper prices and the competition created within the network would keep prices competitive while adding value to our day to day activities.

Figure 1.2a shows an initial state where the document owner sells his document to buyers across the world, over a period of time (Figure 1.2b) these buyers become document holders who assist in the dissemination of that document, all the while ensuring that the owner is compensated.



a) Paid digital document exchange at time T_0 where the owner receives the payment

b) Paid digital document propagation across the global network at time T_1 where the owner and the holders receive payment

Figure 1.2: Equitable Digital Content Distribution across a Global P2P network

A number of research projects have engaged in P2P computing and most have been focused on efficient resource location and load balancing; very few have addressed the need of payments in P2P systems or considered the components required to allow payments in a P2P environment. This body of work aims to address these shortcomings.

In summary,

- An excessive amount of idle resources are being wasted at people's desktops. The use of P2P technologies can overcome this problem by aggregating these resources and utilising them for the benefit of the peer community.
- Online sales involving micropayments have risen and will continue to rise [David Greer 2004].
- The subscription model is not suitable – although research companies predicted a rise in subscriptions they do not reflect the latest trend in the increase in pay-per-view models for e-commerce.

In this situation the P2P model is ideal to provide an economic method for content distribution for personal publishing as well as for large publishers.

The major challenges for this research include:

- Managing and protecting intellectual property rights(IPR)
- Insecurity of the P2P environment
- Dynamic and ad hoc nature of the P2P environment

1.5 A producer-centric content dissemination scenario

The vision for this thesis may be summed up thus; in everyday life people capture their intellectual capital, in the form of solutions to day–to–day problems or simply through their

creativity. On a regular basis amateur writers write poetry, short stories, fiction and non-fiction. For example, Joe likes to write guides to help his friends use software. Bob, who is one of Joe's friends, is good at DIY (Do It Yourself) and has pages of notes on helpful tips for hanging pictures, fixing household items and other everyday DIY jobs. Bob's wife, Jane has a family recipe, which is very popular and has been passed down from generation to generation, whilst Jane's sister Alice enjoys taking photographs of celebrities in her spare time and her other sister Annie is a financial analyst who compiles reports on the current state of the stocks. All these people can potentially benefit from the content they generate, they should be able to publish their content and get compensated for it. However, all this content goes undocumented because it is too expensive or difficult to publish.

Although there are services in place that allow users to publish their content on the Internet for example blogging or podcasting [Lee Rainie 2005, Mark Shelstad 2005], at present they bear the overheads of managing a website, possibly paying for a domain name and not necessarily benefiting from the use of their intellectual property. But imagine if these people could publish their content by just utilising the power of their desktops without any other overheads or the dependence on large publishing and distribution concerns. Every time Joe wrote a tutorial on how to use any software he could immediately distribute it on the Internet and for every time it got used Joe would receive a small payment. Regardless of where other people found Joe's tutorial on the Internet and used it, Joe would still receive his payment. Joe would not have to manage his website or use a server anywhere to host his content and the sum of the small payments could potentially multiply into larger amounts and be a source of income at very low initial cost.

1.6 Aims and objectives

The aim of this research is to provide a system which enables the fair exchange of digital content for payment, while exploiting the economic advantages of using a P2P network for content distribution. This system should also ensure that content owners are always compensated and copyright is not violated.

To achieve this, it is necessary to fulfil the following objectives:

1. Understand current P2P technologies and their working.
2. Review payment techniques and models while studying the nature of e-commerce in the P2P domain.
3. Define the requirements of a Payment Instrument pertaining to this research.

4. Develop a model to ensure content owner and distributor compensation and satisfy intellectual property rights protection requirements.
5. Develop a method to maintain transaction data to conform to economic practice.
6. Develop a method to ensure payments (from compensation) are securely handled.
7. Design a system to realise the developed content and payment exchange model in the P2P domain, without detracting from the advantages of the P2P domain.
8. Evaluate this model against current models.

The focus of this research is on e-commerce in the P2P domain, where the difficulties of payment for digital content in true P2P networks is addressed. The main focus is on the issues pertaining to:

a. ensuring content owners are fairly compensated for their content every time the content is distributed i.e. copyright is upheld,
b. fair exchange of digital content for payment,
c. non-repudiation of a transaction,
d. maintenance of transaction data, and
e. dealing with unknown peers in a dynamic environment.

Although many of these issues have been addressed for e-commerce using the client-server model, the impact of these issues on the P2P domain makes this a very interesting area of research. Hence, the nature of the subject area necessitates an understanding of various technologies and research areas which include P2P technologies, copyright protection, economic models, payment mechanisms, e-commerce, security techniques, trust issues, service-oriented architectures, databases and others.

The focus of this research will be on content exchange P2P networks (described in Chapter 2); where, as in KaZaA, a user has the ability to search for content and download it directly from the other peer where content is found. At present user decisions are based on the connection speed of the uploading peer and how highly rated his content is. As part of this research the user shall be informed of the content's value as well.

The research that surrounds the area of watermarking techniques, replication and redemption (conversion) of tangible payments from digital payments is not the focus of this research; although this work does consider the application of such techniques and utilises them.

1.6.1 A definition of digital content

Within the context of this thesis digital content may be defined as any content which has been encoded in a format that may be processed by a computer; consequently, data in the form of photographs, text, electronic books, newspaper articles, games, graphics, 3D images, spatial models and maps, music, film, sound, and applications (software) may all be categorised as digital content.

This content maybe utilised via computers, television, radio, CDs, DVDs, handheld mobile devices including cell phones and digital media players – and any other carriers of information that arrive in the future.

The features of digital content are that it:

- can be created, manipulated, duplicated and re-used with ease;
- can be easily transmitted and accessed globally – anywhere, anytime;
- may stimulate new ways of social networking e.g. blogging and interactive television;
- can make connections between different content that were not previously possible or obvious;
- can be aggregated into powerful data sets of information to generate new knowledge.

Within the context of this research a single digital content item is static during its lifetime within the system. Any subsequent alterations will create new content. Newer versions of the altered content can co-exist within the system with their own individual properties.

Another attribute of any content is its ownership, for the purpose of this research the individual/institution that places the content within the system is presumed to be its legal owner. The complexity of multiple ownerships and how the various owners interact with each other in the real world is outside the scope of this research.

1.6.2 Methodology adopted

The methodology used to conduct this research can be divided into two distinct parts. In the first part, domain analysis within a domain engineering framework was used to understand the problem area and the different techniques and tools utilised within the P2P domain. This allowed us to define a clear hypothesis and hence derive a set of requirements for the overall research project. These requirements were derived in an incremental fashion through exploration of the research domain and literature review. Once the initial requirements analysis

had been performed and a high level solution derived, the resultant system was designed and implemented using software engineering principles following the stages of an iterative process which reflects common engineering practice such as unified process where extensive use of UML was undertaken to capture the various stages of the design of the system.

1.7 Novel aspects of this work

The contributions to knowledge through this research are:

- A Cascading Payment Model (CPM) for the fair distribution of royalty and commission in exchange for content. This model mirrors real life economic models and ensures that the content owner is always recognised as the intellectual property right owner, while the distributor is compensated for his participation in the content value chain. We have published a paper [Gurleen Arora 2003] discussing the various components of this model.
- The use of a user-centric approach for intellectual property rights owner identification, where every owner has a unique identity in the system linked with his global identity. This identity forms the basis for cascading payments and the protection of the payments. We also mobilised the intellectual property rights owner identity as opposed to the use of a centralised point of management of Intellectual Property Rights, to facilitate copyright protection. A paper was published [Gurleen Arora 2005a] which demonstrated the working of these protocols.
- The use of an Overlay Network as a collective Bank for the system ensures that the P2P network can function in a truly P2P manner while still following the rules of commerce. The network of bank peers work in the same plane as our normal peers but act as decentralised third parties to validate a transaction and maintain records for the purposes of non-repudiation. They also assist in the fair exchange of content for payment while maintaining the atomicity of the transaction. These contributions were published in a journal paper [Gurleen Arora 2005b] describing the performance of the transactions involving bank peers. A paper was published [Gurleen Arora 2006] which demonstrated the working of the bank protocols.
- A Cascading Payment Content Exchange (CasPaCE) framework that supports the Cascading Payment Model and the Overlay network of Bank Peers for the P2P domain has also been developed. Here we used a service-oriented architecture to design the CasPaCE framework which enabled the creation of an open and flexible framework. This allows inter-operability and paves the way for the application of the CPM to other digital commodities such as services and functionality. This also allows the creation of a service oriented framework which allows the development of different applications

where intellectual property rights can be maintained for different problem domains not just content exchange. A UML data model which describes the various components of the framework adds value to this contribution.

1.8 Summary and thesis structure

This Chapter has discussed the current trends in digital content distribution over the Internet, particularly through the use of P2P networks. It highlighted that P2P technology provides an excellent vehicle for the distribution of digital content at low entry-level cost to the content producer and distributor obviating the need for publishing middlemen. This is primarily due to its inherent features of resource redundancy which enables high availability, aggregation of resources, cost reduction and scalability as opposed to the traditional client-server based content distribution model. However, since P2P content sharing networks are flawed with free-riding and copyright infringement we concluded that there is a need for new economic models, in this domain, to ensure copyright protection and empower users. This Chapter also listed the scope and objectives of this research along with our contributions to knowledge.

In the following Chapters the reader is given a better understanding of what P2P means in the present day, the issues and challenges involved in performing commerce in this arena, while still maintaining the inherent features of P2P.

The thesis is organised as follows, Chapter 2 defines the term P2P in the present day context; the metrics that can be used to classify P2P systems are discussed and it also highlights the main application areas of this technology. It lists the technologies that are currently used to enable e-commerce. Peripheral technologies that are used in conjunction with P2P technology to perform different functions are also described here. It also lists the different research areas which have a bearing on this work.

Chapter 3 gives a breakdown of requirements that have to be fulfilled to develop a system which facilitates equitable digital content exchange in the P2P domain thus achieving the objectives discussed in Chapter 1. These requirements are classified on the basis of the various areas this system has to work across. The various components of the solution are described along with the relevant terminology. Further it introduces the concepts and models developed as a solution to fulfil these requirements and address the issues which are raised. These concepts will be used to construct a framework which will satisfy the objectives of this work.

Chapter 4 focuses on the design methodology used to describe the system and its deliverables. As part of the solution description the formal design of the CasPaCE Framework is presented (using UML) detailing the various components of the framework and the overall interactions between them. The system data model is also presented here.

Chapter 5 discusses the implementation decisions made for the realisation of the framework, the various technologies used and their impact on the overall system. The testing methodology used is also discussed in this Chapter. We discuss the various test scenarios used to perform integration testing. During the course of the implementation certain design issues were revisited as a result of unit testing, these issues and their implementation solutions are described here.

Chapter 6 demonstrates the application of our framework in the E-learning Content Exchange (ElConE) case study and its evaluation. Each component of the system, as described in Chapter 3, is evaluated against related work. The performance evaluation for the system pertaining to a transaction is also described here. This Chapter also discusses other application areas where our framework may be utilised to improve application performance and create new models for equitable digital content exchange.

Finally, Chapter 7 presents the concluding remarks to this body of work and summarises the findings of this thesis. Further it lists future enhancements to the framework followed by the Appendices which include detailed design notes.

Chapter 2

2 BACKGROUND AND RELATED WORK

This Chapter gives the reader an understanding of P2P technology from its basic features and evolution into what P2P means in the present day. The different classifications of P2P technology and its application areas are discussed followed by a review of the various research disciplines which have a bearing on this research.

2.1 Peer-to-Peer (P2P) technology

The term Peer-to-Peer has been around since the early days of networking where Peer-to-Peer implied a point-to-point communications model. Although the traditional model of the Internet was client-server, where servers hosted information and services and served requests made by clients over the Internet, the communication between the Internet nodes was Peer-to-Peer or point-to-point.

Initially Peer-to-Peer was the term given to a point-to-point communications model, where all peers were equal and any peer could initiate a communications session [Claudia Leopold 2001]. In current day usage this term also refers to a class of applications, systems or infrastructures that adapt this communications model to perform critical functionality.

A number of definitions have been proposed to define what P2P is today:

Shirky[3] [Clay Shirky 2001] broadly defines P2P as:
"a class of applications that takes advantage of resources -- storage, cycles, content, human presence -- available at the edges of the Internet. Because accessing these decentralized resources means operating in an environment of unstable connectivity and unpredictable IP addresses, P2P nodes must operate outside the DNS system and have significant or total autonomy from central servers". We consider this as a broad definition because Shirky bases it on a generic litmus test which asks the following questions - 1) Does it treat variable

[3] "What is P2P ... and what isn't?",(2000) Available at
http://www.openp2p.com/pub/a/p2p/2000/11/24/shirky1-whatisp2p.html?page=2

connectivity and temporary network addresses as the norm, and 2) does it give the nodes at the edges of the network significant autonomy? If the answer to both of those questions is yes, the application is P2P. If the answer to either question is no, it's not P2P.

As defined by the Peer-to-Peer working group[4]:
"...peer-to-peer computing is the sharing of computer resources and services by direct exchange between systems. These resources and services include the exchange of information, processing cycles, cache storage, and disk storage for files. Peer-to-Peer computing takes advantage of existing desktop computing power and networking connectivity, allowing economical clients to leverage their collective power to benefit the entire enterprise."

Androutsellis-Theotokis and Spinellis [Stephanos Androutsellis-Theotokis 2004] define P2P as : "distributed systems consisting of interconnected nodes able to self-organize into network topologies with the purpose of sharing resources such as content, CPU cycles, storage and bandwidth, capable of adapting to failures and accommodating transient populations of nodes while maintaining acceptable connectivity and performance, without requiring the intermediation or support of a global centralized server or authority."

Hence, we can describe P2P as a set of protocols which provide essential functionality for direct exchange between systems and their resources.

Two computers are considered peers if they communicate with each other and can perform similar roles i.e. make and serve requests. For example, a desktop computer in an office might communicate with the office's mail server; however, they are not peers, since the server is playing the role of server and the desktop computer is playing the role of client. However, if one desktop computer was providing a service to a mobile phone as well as utilising the office's mail server to receive email then the desktop could be considered a peer capable of both providing services as well as requesting them.

Present day characteristics of P2P include networks that mainly comprise of desktop PCs that can make and serve requests. These computers are on the 'edge' of the Internet and they usually work outside the Domain Name System. There is no centralised control; hence the individual nodes are autonomous systems. But as more and more hand-held devices are made Internet

[4] Peer-to-peer Working Group, "Peer-to-peer Working Group website - What is Peer-to-peer?," Available at http://www.p2pwg.org/whatis/index.html,

capable, it is soon becoming apparent that mobile phones, PDAs, laptop computers etc. will play a significant role in P2P networks [Richard Gold 2001].

Put simply, Peer-to-Peer computing is the sharing of computer resources and services by direct exchange between systems. These systems could be PCs, set-top boxes, mobile phones or Web-connected databases and the resources and services include the exchange of information, processing cycles, cache storage, and disk storage for files. Peer-to-Peer computing takes advantage of existing desktop computing power and networking connectivity, allowing economical clients to leverage their collective power to benefit their entire community. In the Peer-to-Peer world, any machine could upload or download information to or from another, as opposed to the more rigidly hierarchical model of individual computers downloading information from dedicated Web servers as is the norm on the Internet.

Based on our understanding of P2P we propose the following definition:
P2P technology is any network system, architecture, protocol/ protocol suite that enables devices with a digital heartbeat to share resources which include but are not limited to information, content, processing power, storage and services. These devices are equal in role (i.e. make and serve requests) but may possess a varied degree of resources thus forming heterogeneous networks. These autonomous devices can provide a plethora of services only limited by the resources available to them.

Recently the resurgence of interest in P2P, generated by the advent of file-sharing applications such as BitTorrent, Napster and Gnutella, has also highlighted the major advantages of P2P such as decentralised control and the distribution of information across peers rather than on central servers. P2P has distinct advantages, for content distribution, over the client-server model especially in the present day.

As technology and information have increased so has the need for people to access this ever-increasing information. In the current situation, it is more advantageous to have content distributed across peers where there is no single point of failure, so content is always available. Each peer brings extra information and resources to the pool and the sum of the parts gives greater benefits to the system than the whole. This is because P2P systems exhibit massive resource redundancy and consequently increased resource availability, these features can be combined as a virtual resource for systems who may not possess these resources locally but can use it from the collective system. Peers can be lightweight (PCs, mobile phones, PDAs) and the environment caters for intermittent connectivity. A large proportion of the Internet is made up of individual users that connect via intermittent dial-up connections. As connections to the

Internet gain in speed and become affordable to a larger spectrum of users, P2P allows for greater availability of information, assisting in the better delivery of digital content.

On the other hand P2P technology does have its disadvantages; it is possible that resources may not be available at any given time or even if they are available they may not be reachable. The general and effective solution to both these problems is redundancy, whereby resources are replicated [Kavitha Ranganathan 2002]. The more live nodes that can provide a resource, the more chances there are of it being available and reachable at any given time. Another solution to this problem lies in the development of more robust P2P lookup algorithms [Antony Rowstron 2001a] which guarantee the location of resources.

Common challenges of the P2P research field include:

- Scalability, which is the issue of ensuring that the system can accommodate a large number of nodes and grow without affecting its performance.
- Resilience, which is the ability of a system to recover from failures. In P2P systems this resilience is mainly dependent on the continued availability of resources. Hence if some resource is made unavailable an alternative should be available. Replication of resources is the common solution to this problem.

In summary, Peer-to-Peer became popular because it provided an economical solution by utilising inexpensive computing power, bandwidth and storage; all present and under utilised on the Web. Many applications have been motivated by this premise. The main applications and uses of P2P are described below along with the challenges they present for our research.

2.2 A taxonomy of P2P technology

There are different criteria that may be used to categorise P2P systems[Krishna Kant 2002]. The following sections will describe the various metrics that are used to categorise these systems and how these classifications are interrelated (Table 2.2).

2.2.1 P2P models

P2P networks may be classified on the basis of the connection and routing (lookup and discovery) methodologies they use. Based on this criteria they conform to one of three types of models: the Centralised P2P model, Pure P2P model or the Hybrid P2P model. Each of these models uses different techniques to resolve the lookup and discovery issues in P2P systems.

In the Centralised P2P model (Figure 2.1), peers connect to a central server (or set of servers) which assist the peer to locate other peers. Once other peers have been discovered the communications between the peers are carried out directly without the interaction with the central server(s). The Napster [Clay Shirky 2001] music-sharing application conforms to this model and most presence management and Instant Messaging (IM) P2P applications conform to this model as well, where connection information is retrieved from a central server but connection is maintained directly between the peers. This model has its advantages since resources can be located quickly and efficiently, the search can be exhaustive and users can be monitored since they would be registered in the system. On the other hand, the system is vulnerable to censorship and technical failure primarily in the form of a 'denial of service', where the server may be too busy to cater to requests or it may have out of date information. Hence there are scalability and resilience issues with this model which are balanced by the ease of maintenance and management of these types of systems.

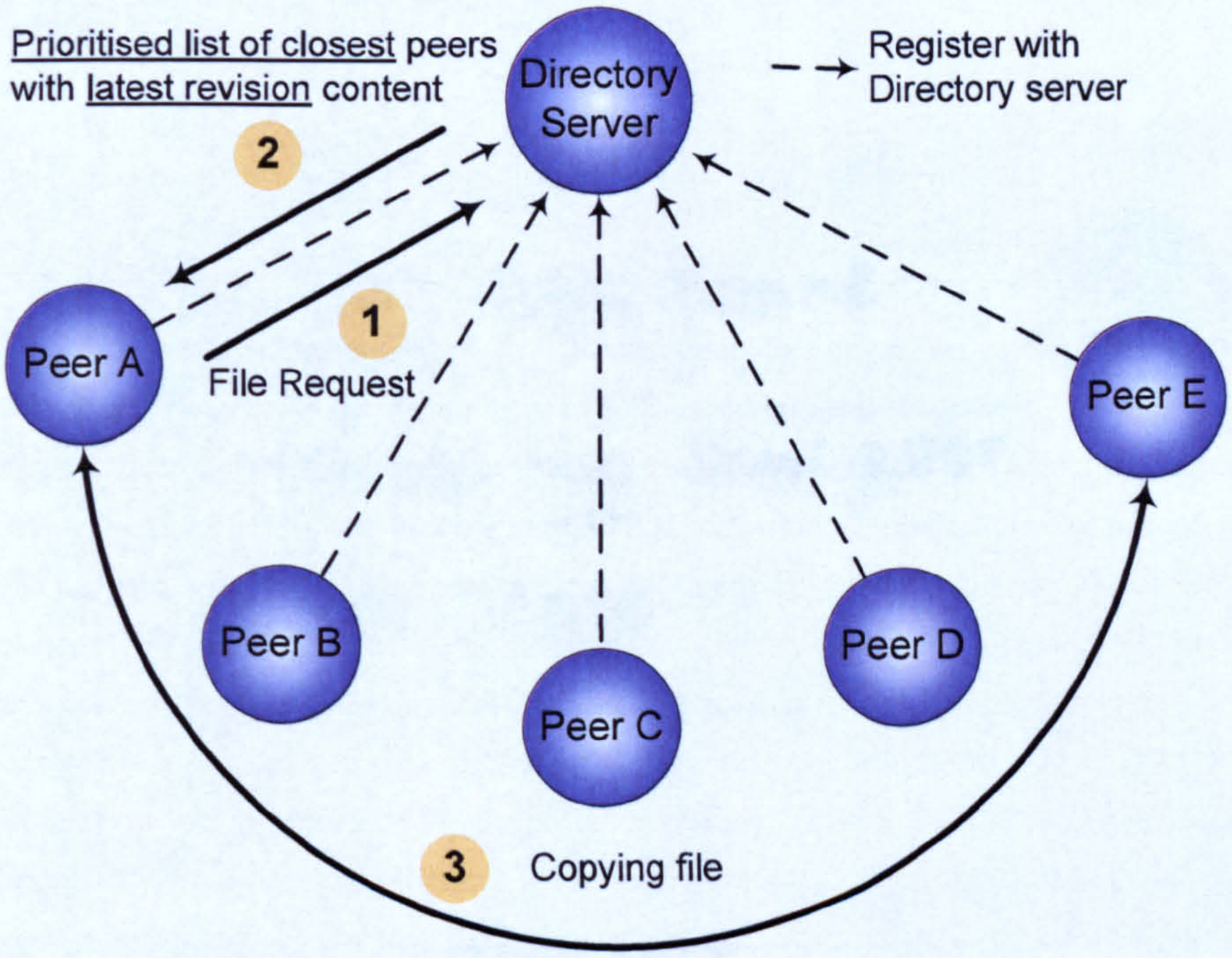


Figure 2.1: Napster's centralised P2P model

The Pure P2P model (Figure 2.2) does not use any centralised servers to assist in peer discovery; instead it relies on the cooperation between peers to assist in peer discovery by exchanging location information between them. In this model, a peer would connect to the network and 'discover' other peers using location information gathered from a previous connection and at the same time inform other peers of its own existence. The applications which

DIAGRAM ON THIS PAGE EXCLUDED UNDER INSTRUCTION FROM THE UNIVERSITY

use this model are usually setup to look at bootstrap servers when they come online for the first time. Gnutella and its variants use this methodology and hence may be classified as pure P2P networks. Once the location information is gained the communication between the requestor and responder is direct. These networks have the major advantage of being completely decentralised and hence scalable, they also avoid the 'single point of access' problem and are fault-tolerant. However, they tend to be inefficient in the lookup process as it can be slow as well as traffic intensive. There is also the issue of a limited lookup horizon, where the resource may be available in the system but the lookup procedure can not locate it. The original Gnutella protocol has been modified to overcome these issues, mainly by including caching techniques, however these systems take some time to become efficient as there is an initial learning curve involved and discovery of resources is still not guaranteed. Although scalability is not an issue here, the lack of 100% guarantee of finding resources can be considered a shortcoming of these systems by creating a false sense of lack of resources. But in systems where resource location is not of paramount importance this model works well and has a low cost of entry and maintenance.

Figure 2.2: Gnutella's pure P2P model [Karl Aberer 2001a]

The Hybrid P2P model gains location information by cooperation between peers as well as from previous knowledge of the resources' location, they do not rely on central indexing servers but on knowledge gained from previous participation in the network or knowledge inherent in their routing algorithm (Freenet - Figure 2.3). Systems like PAST[Antony Rowstron 2001b], Scribe

[Miguel Castro 2002] and others [Sylvia Ratnasamy 2001] conform to this model, as the location of the resource is related to the resource's name and regular querying of the network results in regular updates to the routing tables. These systems overcome the issues of limited lookup horizons because discovery is guaranteed due to the nature of the underlying routing protocols. These routing protocols (CAN [Sylvia Ratnasamy 2001], Chord [Ion Stoica 2001], Tapestry , Pastry [Antony Rowstron 2001a], Kademlia [Petar Maymounkov 2002] and Viceroy [Dahlia Malkhi 2002]) rely on the use of distributed hash table (DHT) abstractions as a method for lookup and data location. These techniques allow the P2P system to remain decentralised by removing their reliance on centralised index servers and provide better guarantees for node lookup as compared to broadcasting or indexing. This model works best for systems where resource location is of paramount importance and by far these systems are the most efficient while being scalable and resilient.

Figure 2.3: Freenet's hybrid P2P model [Ian Clarke 2001]

Structured lookup algorithms provide better guarantees for resource location once it has been placed in the system. Here there is well defined information available regarding other nodes in the system. On the other hand symmetric lookup algorithms are more resilient as they do not depend on hierarchical sources for lookup information; instead all nodes are treated as equal and are aware of their immediate neighbours. Broadcast techniques are used to keep this information current. DHT based lookup solutions tend to be the most robust as they are both structured as well as symmetric. They have the added benefits of scalability, low latency lookups, ease of routing table maintenance, efficient handling of peer arrival and departure as well as even distribution of node indices (keys) among network members. Table 2.1 illustrates the

classification of commonly used lookup algorithms in the P2P domain into symmetric and structured algoithms.

Table 2.1: Types of lookup algorithms & the techniques they use

Symmetric Lookup algorithm	Structured Lookup algorithm
Broadcast	Indexed
Hierarchic broadcast (using super peers)	Hierarchical indexed
Broadcast using unstructured tables	DHT
DHT	

2.2.2 Name based classification

Other P2P classifications may be based on the type of location metrics used to form the network, where the resource naming technique distinguishes between the networks. This basically leads to two types of networks, Structured P2P networks [Hung-Chang Hsiao 2003] (like Pastry and Tapestry) and Unstructured P2P networks (like Napster and Gnutella).

Structured P2P networks use structured resource naming techniques to assist in the location of nodes or content in the P2P network. Whereas, Unstructured P2P networks rely on the content discovery technique to locate content, the content itself is inserted/placed locally and propagated in the system by virtue of its popularity. Content is not named in a structured way; instead naming techniques are determined by the user instead of the network. Consequently these networks provide keyword searching, most content is typically replicated at a fair fraction of participating sites, and the node population is highly transient. For instance content in KaZaA and Napster is inserted and named by the user, but PAST [Antony Rowstron 2001b] inserts content based on a hashed value which is placed on nodes that are numerically closest in Node Id to the numerical File Id. These names are unique to the resource and are generated based on a common network policy. For instance, in Freenet [Ian Clarke 2001] a hash of the content is used to name the content, this is generated by the local Freenet servent based on Freenet's naming policy, the content is then stored on nodes whose node ids belong to the same name space and are nearest numerically (Figure 2.3).

2.2.3 Application based classification

Based on our definition of present day P2P technology, P2P systems may also be classified by the primary functionality they provide. Of the many uses of P2P the most prominent are

content-sharing applications such as Napster, Gnutella (along with its variants), Freenet and many others. These applications are used to share digital content. Other than content-sharing applications, P2P is also being utilised to assist with Distributed Computing, Instant Messaging and Collaboration. Some of the more recent uses of P2P also include P2P radio [Howard Wen 2002], where P2P technology is being used to broadcast radio over the Internet and webcasting.

Table 2.2: P2P technology classification interrelation

* Hybrid

		Connection, lookup & discovery		Resource naming technique	
Application Group	**Application Name**	**Centralised P2P**	**Pure P2P**	**Structured**	**Unstructured**
Content Sharing					
	Napster	✔			✔
	Gnutella		✔		✔
	Freenet		✔	✔	
Distributed Computing					
	SETI	✔		✔	
	Parabon	✔		✔	
Collaboration & Communications					
	MSN messenger	✔			✔
	Groove	✔		✔	
	Jabber	✔			
Multimedia Streaming					
	AllCast	✔			
	P2P radio				
P2P Frameworks					
	JXTA	✔ *	✔ *	✔	
	.NET				

The following sections describe the various P2P application areas on the basis of how they handle the following criteria: node discovery, routing, scalability, anonymity, complexity and resource access or compensation criterion.

2.2.3.1 Content management

P2P content management systems take into consideration the special requirements for managing digital content, which includes sharing, archiving and filtering or mining content. In this respect they tend to utilise the advantages of P2P networking to aggregate the resources of multiple nodes to provide increased bandwidth for content sharing and aggregated storage for archiving and persistence of content. However, they have to address the issues pertinent to the management of digital content and the sharing of information.

Content sharing P2P networks provide peers the ability to share files of different digital formats such as audio, video, text, games and software, although some systems restrict the types of files that may be shared based on their underlying system objectives. For instance Napster only allowed the sharing of music files. This served two purposes, firstly it reduced the security risk

by reducing the chances of malicious content on the network and it served its underlying objective of providing an economical music sharing network.

The number of servers that could be maintained to manage user logins and content indexing limited Napster's scalability. But Gnutella, Freenet, PAST and others have been proved scalable [Rüdiger Schollmeier 2002] by virtue of the properties of their underlying routing algorithms. Although there was a popular belief that Gnutella's routing algorithm limited the search horizon, the protocol has since been modified [Matei Ripeanu 2002a] to include concepts such as search and routing information caching, the networks have also included the use of super-nodes[5] that provide extra functionality to assist in the efficient working of the network.

Routing and location protocols such as Achord [Steven Hazel 2002] and Freenet [Ian Clarke 2001] provide anonymity to publishers and requesters of content by using the mechanism of tunnelled connections. The main objective of these protocols is to give users the ability to freely publish content without the threat of identity disclosure. These protocols would also allow for the creation of applications where user anonymity was the primary concern and persistence of data to prevent censorship is one of the main motivations for these research initiatives.

We can thus conclude that different P2P file-sharing initiatives have been designed to serve different purposes. For instance file-sharing P2P variants, including Freenet (freenet.sourceforge.net), Morpheus (www.musiccity.com), and MojoNation (www.mojonation.net)[6], address dissimilar challenges and systems such as Oceanstore [John Kubiatowicz 2000] and PAST are designed to reflect true file systems behaviour and are used for archiving data to ensure its persistence. Opencola's main objectives are filtering and mining of data. On the other hand Gnutella's Kazaa system is used for content exchange alone. Freenet (freenet.sourceforge.net) provides decentralized anonymous content storage protected by strong cryptography against tampering. Morpheus provides improved search capabilities based on metadata embedded in common media formats. The latest version of Morpheus (beta 4.0)[7],

---

[5] Super-nodes are specialised peers with more resources and hence more responsibilities in contemporary P2P systems.

[6] Since the beginning of this research, MojoNation's work has been incorporated into Allmydata Inc.'s shared storage product (http://www.allmydata.com/).

[7] This version of Morpheus was motivated from a need to standardise P2P file sharing networks and hence it worked by creating a search facility which plugged into various underlying P2P systems. At the time of its release there were concerns over the legality of P2P file-sharing networks, Morpheus claimed

allows users to search across different P2P file-sharing networks, based on the Gnutella protocol (such as KaZaA, iMesh, eDonkey and many others). Projects such as Freenet and Publius [Marc Waldman 2000] provide anonymous publishing of content for storage and distribution via a P2P network. Although Freenet achieves its primary goal of anonymous publication, it does not have any mechanism to guarantee the persistence of all content [Hari Balakrishnan 2003], therefore unpopular content may disappear from the system. Freenet can't always guarantee the most efficient location of content either. Lime Wire (www.limewire.com) and MojoNation's storage system addresses the issue of resource allocation and access control. Whereas Lime Wire tackles these issues by allowing users to restrict downloads based on the number of files that a requesting client is sharing with the network, MojoNation takes this model one step further and incorporates a system of 'currency' that users earn by sharing resources that they can then spend to access resources. It uses an artificial 'currency', called Mojo, to enforce resource sharing. MojoNation addresses the challenges of both motivating its users to contribute and also reducing the cost of contribution. The cost of contribution is reduced by using a technique called 'swarming' – which splits each complete download into smaller units which are then uploaded by different peers. In this way each uploader's contribution is reduced to such an extent that it does not have an adverse effect on their system and consequently encourages participation. A similar technique is used by BitTorrent [Johan Pouwelse 2004] to split the load of a single download across a number of peers by splitting the file into small chunks. Torrent based systems further improve download efficiency by using the concepts of *seeds*, *downloaders* and *trackers*. *Seeds* possess complete files whereas *downloaders* are peers who possess some parts of a file but can act as sources of uploading these parts while downloading the rest. *Trackers* keep track of the various *seeds* and *downloaders* that possess a file and act as an index for future downloads.

2.2.3.2 Collaborative computing & communications

Of all the research done in the P2P domain, the majority of innovations have been focused on the content sharing application of P2P. Through all this, the area of collaborative P2P computing has received very little attention and consequently most P2P collaborative and communicating protocols tend to use the same techniques.

We can divide collaborative computing and communications applications into two sub categories; presence management and instant messaging.

Presence management

The Internet Engineering Task Force's (IETF) Session Initiation Protocol (SIP) [IETF 2002], is a signalling protocol for Internet conferencing, telephony, presence, events notification and instant messaging. SIP was developed within the IETF MMUSIC (Multiparty Multimedia Session Control) working group, with work proceeding since September 1999 in the IETF SIP working group. SIP is a text based protocol similar to HTTP and SMTP, used for initiating interactive communication sessions between users. The media types carried in the session are also varied and include voice, video, chat, interactive games, and virtual reality. This protocol handles the initiation, management and termination of sessions where the control over the session is passed to the participants of the session instead of being maintained by a central switch.

SIP is an Application Layer communications protocol which was extended to create the SIP for Instant Messaging and Presence (SIMPLE) protocol. SIP is being used extensively today to support a broad range of voice, instant messaging and presence-based services over mobile, wireline and IP-based networks e.g. IM applications such as GoogleTalk and MSN messenger.

Instant Messaging

ICQ[8] [Mirabilis Ltd. 1996] was the first legacy Instant Messaging (IM) client which was followed by many imitators such as MSN messenger, AOL messenger and Yahoo Messenger. However, they all ran different communications protocols, with different registration procedures and IM clients. Hence, they all developed into separate communities. Jabber[9] tried to reverse engineer all these protocols and create gateways to connect to all the IM services using XML streaming, allowing users to interact across the IM networks. However, the internet service providers resisted this effort, e.g. AOL blocked Jabber clients.

The communication protocols for all IM services work in a similar fashion, where a network of centralised servers assist users during log in and for maintaining presence information (i.e. when a member is online or offline). Once this information has been passed to the clients all messaging is carried out directly between the clients. Current IM services also provide additional services such as voice conferencing, video conferencing, file exchange, interactive environments and many others that assist in collaboration. In these terms, IM networks are classified as Pure P2P networks, as initial lookup and discovery is centralised, but the actual communication is P2P. This is the case, because using the centralised P2P model gives

[8] Mirabilis Ltd., "ICQ Home Page," Available at http://web.icq.com/

[9] Jabber Software Foundation, "Jabber Org Website," Available at http://www.jabber.org/

maximum benefits for management of these types of networks. Service providers want the ability to manage connections and control access to their network which is done through centralised registration. In some instances IM is provided as an added bonus of becoming a service subscriber. Consequently, in these applications scalability issues are addressed by service providers by throwing more resources at the system since service providers do not lack resources and recuperate costs through advertisement and subscription models, usually the service providers main business interest lie in other areas and provide messaging functionality as a side benefit. Network resilience is similarly addressed and managed by administrators.

The use of P2P collaborative computing has also had an impact on the education sector where these technologies are being used for online learning environments such as Helpmate [Kevin Curran 2002], which was implemented for the Physics department at Coleraine University, and Edutella [Wolfgang Nejdl 2002]. These applications combine the content sharing and collaboration aspects of P2P technology to provide teachers and students with access to learning material in an efficient and cost effective manner while removing the need for access to a centralised website.

2.2.3.3 Distributed Computing (Hardware resource sharing)

Distributed computing initiatives like SETI@home[10] (setiathome.berkeley.edu) [Eric Korpela 2001] did not start as P2P initiatives, however their working can be described in terms of P2P functionality, in so much as the clients receive pieces of 'work' from the SETI@home servers and only contact the server when the 'work' has been processed. Hence, the computers work autonomously at the edge of the Internet and provide a distributed resource (computing cycles), which combines to benefit a community. Similarly, other projects like Distributed.net (distributed.net) and companies such as United Devices (www.ud.com) also provide distributed computing solutions, which use the resources on idle desktop PCs to accomplish a task. SETI uses the idle resources of computers to analyse signals it receives from outer space, in its Search for Extraterrestrial Intelligence (SETI). Work at Stanford University [Vijay Pande 2000, Keri Schreiner 2001] focuses on the use of distributed computing to understand protein folding and aggregation to understand protein related diseases. Parabon Computing's [Parabon Computation 2000] Pioneer Platform enables the distribution of work tasks to computers to compute data to find a cure for cancer. They also market their product to other companies to fulfil their intensive computation needs.

[10] Seti@home website: http://setiathome.ssl.berkeley.edu/

The working of all these initiatives is comparable and follows the same centralised P2P model, where packets of work are handed out, by central servers, to individual devices which process them and return the results to their central server. These servers combine the results and determine the next set of work packets to be distributed.

In this domain, one of the major concerns is whether participants process/compute the pieces of work in the correct fashion [David Molnar 2000]. Participation incentives in distributed computing mainly focus on reputation based incentives such as acknowledgement of a user's contribution by mentioning his name on the website or compensation in terms of redeemable coupons at partner sites. Even in a situation like this, where the rewards are intangible, the system is prone to divergences affected by users. Providing contributors with a tangible payment for services rendered would be ideal in this situation.

In MojoNation, Mojo[11]s can currently only be converted to karma (goodwill or strength), i.e. to "buy" priority or for improved performance. Real money may be introduced later and the system is capable of supporting both e-shopping and pay-per-view solutions. The Mojo strategy was used as a participation incentive to balance the production and consumption in a P2P system by creating a market for resources like bandwidth, storage, computation.

2.2.3.4 Gaming

Although gaming can be classed as a subset of group communication and collaborative computing, it has been placed under its own section primarily because of the current increase in gaming and the revenues generated from it thus making it an interesting area with its own specific requirements and challenges for the future.

The requirements for gaming can be classified into two categories based on the mode of play/distribution of the game. Some games are developed for online group communication where online presence is required to play the game across a gaming community. On the other hand with the increasing popularity of gaming on hand held devices it is becoming increasingly common to be able to share games as standalone modules which can be run locally. Based on these categorisations, the role of the P2P system may be one of games delivery or one of group communications. In either case there is a need for managing revenues, particularly in the form of micropayments, in this domain.

[11] At the time of the writing of this thesis the Mojo Project has been suspended.

2.2.3.5 Multimedia Streaming

Various initiatives [Damien Stolarz 2001, Florian Unterkircher 2002, Nezer J. Zaidenberg 2002] started using P2P technology to perform multimedia streaming. The most popular application was to P2P radio and web casting. This technology works by distributing the workload of the main server across participating peers. These peers are then used as broadcast nodes to stream the content. Using the benefits of P2P technology in this instance avoids the problem of flash crowds[12] and bandwidth costs by distributing the load across a number of nodes thus increasing the effective bandwidth and reducing the load on a single node. These initiatives also had to address the interesting problem associated with steaming, particularly maintaining QoS.

The web casting model is centralised P2P because the central server determines which nodes could provide the best streaming facility to a new joining node. It also maintains an index for newly joining nodes to use to locate its nearest upstream broadcast peer. Allcast[13] is a web casting initiative that is functioning as a company with a patent for P2P streaming. The architecture proposed by Unterkircher and Welzl [Florian Unterkircher 2002] addresses the particular issues with P2P networks of the lack of persistence of nodes (transience of nodes), the limit on uplink bandwidth due to heterogeneity of the network and the inefficiency of using repetitive connection requests as a method for traversing firewalls and NATs. As in other P2P systems the solution to these problems are data redundancy (replication) and caching.

The nature of this domain necessitates the use of the centralised model for node selection so as to retain control over the content and to maintain its integrity. The need to maintain basic QoS also motivates the use of the centralised model.

2.2.3.6 Web search

P2P technology has also been used for the development of Web search engines. At initial conception InfraSearch used P2P technology to provide a distributed search engine facility; it was included into Sun's JXTA search project in February 2001[14]. Now JXTA's search utility is based on the same concept [Sherif Botros 2001]. Other distributed search engines have also

---

[12] 'Flash crowds' is a term given when a web site (networked resource) catches the attention of a large number of people, and gets an unexpected and overloading surge of traffic. This surge of traffic can cause Denial of Service.

[13] http://www.allcast.com

[14] http://www.oreillynet.com/pub/d/234

been created by using the Gnutella protocol[15], which cached the results across a set of hubs. Most P2P content storage and caching systems use some form of search, these search techniques can be applied to form distributed web search applications.

2.2.3.7 P2P frameworks and initiatives

P2P has been identified as an empowering technology for the enterprise [Tom Curran 2001], where its features of integration between different platforms, distribution of resources across peers nodes, ease of use, scalability and others make it ideal for collaboration between and within enterprises.

Groove Networks[16] provides a collaboration platform for the enterprise, where inter and intra-company collaboration is possible. The platform allows users to have personal and shared spaces, which they create for collaboration purposes. Members from within the organisation or outside may be invited to these shared spaces to work on documents online or offline, services such as document synchronisation, chatting and file-sharing are provided to facilitate collaboration. As one of the main advantages of Groove is its ability to allow secure collaboration, the creators have devoted a great deal of effort into the various aspects of security required. Groove [Jon Udell 2001] bases its establishment of trust on the premise that most collaborators have had previous contact either in person or via voice conversations, hence when they create a collaboration space they need to only ensure that the messages they receive are actually from the person who it says they are from, i.e. authenticate the message sender. Groove uses different sets of keys to enable secure collaboration between participants across organisations (intra-organisation) and inter-organisation collaboration. Microsoft[17] has recently acquired the Groove platform [BBC News 2005] to consolidate its collaboration products targeted at large and small organisation to enable borderless collaboration on joint projects.

Jabber has been classified under the IM section above but since its early conception, the Jabber Protocols have been accepted as an approved IETF standard for instant messaging and presence management technology known as Extensible Messaging and Presence Protocol (XMPP) [Peter Saint-Andre 2002, Peter Saint-Andre 2004]. Now Jabber technologies provide functionality beyond IM, which allow developers to create applications for message exchange in different scenarios. In particular the XMPP defines the method for exchange of structured XML elements

[15] http://www.dcs.gla.ac.uk/~iraklis/fyp_report/node19.html

[16]Groove Networks, "Web Services and Peer Services," http://www.groove.net/pdf/web-peer_services.pdf

[17] http://www.microsoft.com/presspass/features/2005/mar05/03-10GrooveQA.mspx

between networked endpoints. Jabber servers may be run to form P2P communities for organisations or for personal use.

JXTA[18] is an initiative, started under the auspices of Sun, to allow the development of P2P applications and services. The JXTA Framework provides base P2P functionality and caters for the deployment of services for added functionality as required by P2P applications. Whereas, the Groove and Jabber protocols were developed primarily as a solution for collaboration, JXTA's creation was motivated by a need for common standards for the development of P2P networks that could serve different purposes. Other frameworks that provide support for the creation of P2P applications include Microsoft's .NET [Curt Simmons 2002], which uses Extensible Markup Language (XML) and SOAP (Simple Object Access Protocol) to provide support for Web Services (described below) .

Although most of the applications mentioned so far have a primary function and hence have been classified under a specific category, it is important to mention that most of these applications in the present day provide additional functionality that may overlap the other categories. For instance, almost all popular file-sharing applications allow a limited degree of Instant Messaging between members. Similarly, Instant Messaging applications allow file-sharing on a one-to-one basis, limited only by file size. Collaboration platforms have to, by virtue of their existence, provide both file-sharing and one-on-one communication between clients while providing work archiving tools.

2.3 Overlay Networks

An overlay network may be defined as a logical structure which abstracts the physical connectivity of the underlying layer. They are created to address a specific service need in the network. For instance the Internet is an overlay network that connects various small networks (LANs, WANs and MANs) together. Similarly, P2P networks can also be defined as overlays.

Overlays are commonly used to address routing, addressing, security, multicast and mobility issues. There have been many initiatives [Nicholas J. A. Harvey 2003, Ben Y. Zhao 2001, Ben Y. Zhao 2002] in the P2P domain which have been implemented as overlays to address these very issues.

---

[18] Project JXTA, "The Jxta Homepage," Available at http://www.jxta.org

P2P routing and location infrastructures such as CAN[Sylvia Ratnasamy 2001], Chord [Ion Stoica 2001], Pastry [Antony Rowstron 2001a], Skipnet [Nicholas J. A. Harvey 2003] and Tapestry [Ben Y. Zhao 2001] allow the formation of P2P overlays, which may be utilised to create different P2P storage applications [Sameer Ajmani 2002, Frank Dabek 2001a, Antony Rowstron 2001b]. The same routing infrastructures may also be used to provide different functionality; for instance in Scribe[Miguel Castro 2002] the Pastry routing protocol is used to provide a multicast infrastructure for applications that require the ability to multicast requests and services. The PAST[Antony Rowstron 2001b] storage utility on the other hand uses the Pastry routing protocol to create a large-scale persistent data storage utility where the routing protocol guarantees the location of data. This also demonstrates the fact that different P2P services may be used in different combinations to provide specific functionality.

Overlay networks thus overcome the need for constant hardware upgrades in the underlying network structure. Hence, multiple overlays may coexist to provide a range of functionality to a P2P system. We conclude that conceptually we could have different overlays forming which would perform different functions and yet these overlays exist within the same system. Other overlays could perform trust management, group management, system monitoring, accounting and data persistence to support the main functionality of a particular application domain.

2.4 Digital content & copyright infringement

2.4.1 Digital content

As defined earlier in section 1.6.1, digital content may include various types of data and exhibit certain properties, paramount of which is its ability to be replicated with no loss in quality. This replication assists in the content's indiscriminate copying and propagation, which can lead to loss of revenue for the digital content owner or producer. The popularity of the file-sharing P2P applications such as Napster and KaZaA has demonstrated that replicated digital content is as good as content purchased from traditional vendors. While this property aids in the efficient replication and delivery of digital content via the P2P networks it also leads to copyright violation on behalf of the content owner as highlighted by Dong et al. [Ying Dong 2002] and proclaimed by the global media industry [Lee S. Strickland 2003].

2.4.2 Copyright infringement

Copyright infringement of digital media was common before the advent of P2P file-sharing applications, Napster just made it easier for users to get music into their computers or personal media players in a ready and easy to use format through one global music search directory. The

large volumes of music mp3s being exchanged via Napster combined with a fall in music sales caused the media industry to take notice. However, it would be erroneous to suggest that the fall in sales is a direct consequence of the use of P2P file-sharing applications.

As stated earlier (§1.2, pg.2), copyright infringement refers to the violation of the content producers' intellectual property rights. In this case the traditional method of owner recompense is via a system of royalties, where the owner receives a percentage of the value of the goods and the middleman receives a commission. Traditionally, when people exchange goods for money, this is referred to as monetary recompense. Other features of copyright relate to reputation (whereby the owner deserves credit for his work), archiving, persistence and others which are detailed in [L. Jean Camp 2002]; for the purpose of this research, only the issue of monetary recompense is of concern.

2.4.3 Digital Rights Management (DRM)

One of the most popular solutions, to combat copyright infringement, adopted by digital content owners is Digital Rights Management (DRM) [Peter Biddle 2002, L. Jean Camp 2002] technology. This technology prevents copyright violation primarily by denying access to digital content. These technologies have been developed to enable secure distribution - and more importantly, to disable illegal distribution - of paid content over the Web.

DRM technologies work by either binding content to a particular device or limiting the number of times it can be accessed (Windows Media Player) or the time-span within which it is accessible (Adobe eBook). DRM products allow users to set their own access rules and encryption policies, some allow administrators to control whether content can be read, printed or even shared.

The down side to DRM [Gary Marshall 2006]is the:

- Dependence on centralised servers, where content has to be checked against a centralised library to ensure it has been legally purchased.
- The license keys are not transferable between devices; hence if a user has purchased content to play on his Windows Media Player on his PC, he often cannot play it on his PDA without purchasing a new license.
- DRM technologies work on the objective of tamper resistance of copyrighted material and do not give flexibility of use to the end user.

These technologies do not solve the problem completely because there still exist portals on the Internet that provide free access to digital content [Peter Biddle 2002] termed the Darknet. At the moment most DRM technologies look towards binding content to proprietary systems, thus inhibiting interoperability and wide access to all digital content. For instance recent DRM technologies developed by Microsoft [John Borland 2003], limit access to digital content to Microsoft software only, consequently there has been a call for the development of an interoperable DRM standard [John Borland 2003], while this is still under consideration it is not the best solution as presently this encourages people to resort to illegal methods of digital content access. Another downside to DRM is that these technologies do not benefit the individual small producers of content who may not have a relationship with traditional content publishers or distributors.

Although P2P content sharing networks have become popular and demonstrated that they are an ideal method of content distribution at low cost, in recent years they have displayed certain limiting characteristics termed free-riding [Eytan Adar 2000] whereby all users do not equally contribute to the network and hence expose the network to failures or fragmentation. Here these networks conform to the power-law theory where a small number of nodes provide most of the content. Free-riders generally desist from contributing content to the system and only download content. This has a negative impact on the overall network as it makes it weak and susceptible to attacks as network topology can be spoofed. It also reduces the chances of finding content as data persistence is affected due to the problem of limited search horizons. Since a few nodes become the suppliers of a majority of the content in the network it exposes the individual nodes to legal action[19]. Users free-ride for various reasons, primarily 'greed'; uploading content takes up bandwidth which users would rather use to increase the speed/number of personal downloads. It is also done to avoid legal action[20] as content providers are targeted for hosting copyrighted content.

As identified by Lui et al. [S. M. Lui 2002], rewarding individual nodes in the system would act as a participation incentive and overcome free-riding. Hence, we infer that enabling a fair payment model in this environment would solve the problems of free-riding as well as ensure that the content producers' copyright is not violated.

---

[19] Legal action in this scenario refers to a target industry identifying the system as a source of illegal activity. E.g. Napster was identified as a system encouraging content theft and copyright infringement by the RIAA and consequently taken to court.

[20] The other side to this argument states that there may be cases where the legal action is unfair, in these cases the primary aim and motivation for these systems is to prevent censorship and promote free-speech.

A solution for distributed DRM has been implemented by Cox in the MyBank [Brad Cox 2002] application which allows users to distribute their digital content via the Internet and get compensated every time that content is used. The application locally debits the users' check-book every time they use some copyrighted content and redeems the payment, when the user goes online, by synchronising with the MyBank servers. This system conforms to the usage based charging model. Although users' copyright is upheld, this solution does not compensate intermediate users' in P2P networks whose resources may be used to propagate the content. Hence, it is not an ideal solution for the P2P environment where free-riding is a common vice and fair participation incentives are few.

The problem with usage based charging models is that they do not conform to consumer expectations [Deirdre K. Mulligan 2003]. In a traditional transaction, a consumer would purchase a hard copy of a book and use it as and where he pleases e.g. lend it to a friend. Although, the motivation behind DRM was to protect copyright it has an adverse effect on consumer experience primarily because it gives copyright owners the right to dictate the conditions of use of the commodity. On the one hand DRM allows for the development of new business models because the owner can dictate the usage terms of his content; however the downside is that access to works is restricted or even denied in some cases. DRM technologies have no mechanism in place to actually determine whether there is infringement or if a work is being used fairly (e.g. for study, research or reporting). In some cases these piecemeal rights do not conform to the rights conferred on users under the Copyright Act and are not directly translated to non-digitised works.

To ensure that content owners receive recompense for their content, they need to get monetarily compensated. These monetary payments need to cascade back to the owner every time the content is propagated - a system of royalties. At the same time to ensure that intermediary distributors are encouraged to propagate this content it is important to ensure they receive a commission. By applying this system of recompense for copyright protection we can utilise the advantages of P2P technologies to develop new models for content distribution. The introduction of payments in the P2P arena raises challenges, which shall be described in the next section.

2.5 E-commerce

One definition of E-commerce refers to all retail sales to consumers where transactions and payments are carried out over an open network e.g. the Internet. There are many other aspects to E-commerce dependent on the context of the transaction (e.g. B2B, B2C, C2C, B2G, G2C and

many others), the communications channels being used and the applications or technologies being used to conduct the transaction. The *system* refers to the whole infrastructure including the transport medium (networks), the software components (applications and protocols), the participants and the commodity for sale. The *payment instrument* refers to the actual payment technique used such as the debit card, credit card, cash (Fiat money[21]), money order, bank draft, cheque and others. The major *players* in an e-commerce transaction are the buyer, the seller, the commodity and the payment system. In popular e-commerce systems these can be the business (B), consumer (C), government (G). The overall carrier of the transaction would be the underlying network, in the context of this research the P2P network over the Internet. Major e-commerce systems can be classified into B2B, B2C, C2C, B2G, G2C and many other combinations of the main players, where these classifications are used to identify the scope of the players in the system. This has an affect on the type of payment instrument to be used as well as the payment system to be used.

However, the aim of this research is to focus on small producers and consumers, hence traditional e-commerce models, though well established, do not cater for the needs of these users. These users need payment instruments which are easily manageable, globally applicable and yet do not incur high usage fees. To this end a brief description of payment methodologies is provided in the following section.

2.5.1 Electronic payment techniques

As the Internet became a popular distribution channel for digital goods and services, it also developed into a medium for the communication of information required to complete monetary transactions. The popular model of the Internet became client-server; consequently all digital payment schemes were developed to conform to this model. In the present day the popularity of P2P networks has demonstrated a need for payments, which are feasible in the P2P environment. The following sections shall describe some of the different payment schemes and whether they may be used in the P2P environment or not.

2.5.1.1 Macro and Micro Payments

Macropayments refer to payments that amount to large values (from £10 to several million), which can be processed by traditional payment instruments such as credit cards, bank orders,

---

[21] Fiat Money is money that a government has declared to be legal tender, despite the fact that it has no intrinsic value and is not backed by reserves. Most of the world's paper money is fiat money.

and cheques. These payments are large enough in value to justify the cost of processing of the transaction and one of the participating entities volunteers to bear this cost.

Micropayments on the other hand deal with payments of very small amounts, such as sub-units of a £1. These payment values are small enough that the cost of the transaction to process them would have to be very small to justify the use of a traditional payment instrument other than cash.

In terms of electronic micropayment schemes in use in distributed computing, some are defined on the basis of non-fungible payments and others as fungible payment systems. A non-fungible micropayment system deals with work carried out instead of a tangible payment that can be redeemed for monetary gain or used to pay for other services. These types of micropayment schemes are generally used to gain access to resources or limit overloading of a system by the time interval it takes the requestor to complete the piece of work (POW). Although these types of schemes are widely used in P2P systems [Roger Dingledine 2001], the class of micropayments of interest to this research are referred to as digital cash schemes. Both classes of micropayments shall be described in some detail in the following sections.

2.5.1.2 Electronic Cash

Electronic cash payment schemes like VisaCard and Mondex [Felix Stalder 1999] require the use of smartcards which hold monetary value. The downsides to these are that system users need to purchase smartcards from traditional vendors (shops) and require extra hardware (card readers) to use the system. The money earned by using this system can not be used to directly[22] reimburse users utilising different systems. Hence, a payment may only be used to pay one person; the same payment cannot be transferred to another user.

Digicash's Ecash [Daniel Minoli 1997], uses a system of anonymous e-coins to pay for services, where the identity of the buyer is anonymous to the seller and the electronic cash bears the properties of real cash. However, the problem arises with the need for online verification of coins via a trusted third party in this case the Ecash server that resides with the bank.

Netbill [G. Winfield Treese 1999] is another electronic payment system, where payment is made for encrypted digital content and once the payment has been validated the decryption key

[22] There exist intermediary service providers such as PayPal who could enable transfer of money indirectly.

is passed to the buyer of the content, again this system requires the use of a trusted third party (the Netbill server). Netbill does take care of the delivery of the decryption key if the connection between buyer and vendor is lost for any reason as long as the payment has been processed and the vendor passes the confirmation to the buyer. The use of a single centralised trusted third party is not feasible in a P2P scenario as it creates a bottleneck and a single point of failure, since P2P content sharing systems scale to millions of users there is potential for the trusted third party to be constantly unavailable.

Tewari and O'Mahony [Hitesh Tewari 2003] present a lightweight payment scheme based on unbalanced one-way binary tree (UOBT) hash chains to pay nodes in an ad hoc network for the service of relaying packets between the source and destination. The use of these hash chains is convenient, as they do not require online verification by a broker or bank. The design of the payment scheme is independent of the underlying transport protocols and does not require the online presence of a trusted third party such as a bank or broker to pay nodes in changing paths. It relies on smartcards for the purpose of identification of participating nodes.

Blaze et al [Matt Blaze 2001] describe another payment scheme that addresses the issue of offline payment verification which is a problem inherent in the hash chain based PayWord [Ronald Rivest 1996] system. The benefits of the scheme are that it does not rely on the use of trusted hardware. However, it relies on the use of trust metrics to compute risk of allowing the user access to limited types of transactions. This risk assessment is used to generate credentials which ensure that there are negligible chances of benefiting from uncollectible or fraudulent transactions.

2.5.1.3 Licensing schemes

Licensing schemes allow users access to content or services on production of a valid license. These licenses may be valid for a life-time, hence require a one off payment on purchase, or they may be valid for a shorter period of time where users have to renew them on expiry if they wish to continue access to the service. Subscription based payment systems, could be likened to licensing schemes.

Perry and Williamson describe a licensing system that monitors the distribution and payment of e-books [Russell Perry 2002] which may be on any device. Again the down side to this system is the dependence on central servers to authenticate licenses and store licenses, and individual content producers do not get compensation for their work. Content producers still need to register with the server maintainer to ensure he gets compensated for his work.

2.5.1.4 E-commerce protocols

Many electronic payment schemes exist to facilitate e-commerce via the Internet [Efraim Turban 1999]. Invariably they either rely on the presence of online verification by a centralised third party or on inbuilt security mechanisms to discourage misuse. In the case of electronic payments that imitate real cash, the coins have to be checked for double spending and have to rely on complex security procedures to ensure they may not be forged. Secure communications protocols like EFT (Electronic Funds Transfer)[Efraim Turban 1999], SET (Secure Electronic Transaction) [Giampaolo Bella 2002] and SSL [Stephen Thomas 2000] allow secure online monetary transactions between buyer and seller, where debit or credit cards may be used to pay for services without compromising the users' details. Hence, they all have to rely on the use of central servers maintained by the payment instrument issuer. In pure P2P networks (§2.2.1 pg 17) the presence of centralised servers takes away the benefits of the P2P environment and also is an impediment to scalability. Every time the network becomes larger, the number of servers required to guarantee quality of service has to increase.

2.5.1.5 Methodologies in use

The first P2P file-sharing network to propose payments for content was Napster, after it lost it's battle in court against the Recording Industry Association of America (RIAA), Napster proposed the use of a subscription service [Clare Saliba 2000] which would take care of the copyright violation issue and the RIAA would receive royalties for all music content exchanged via the Napster servers. Currently Napster 2.0 [Napster 2003] offers users the ability to pay for individual songs ($0.99) or for the complete album ($9.95), effectively becoming a virtual music store, but still using P2P technology. Barring this, the user may choose to pay a monthly subscription, which is marginally lower than the amount paid for an album. Not to exclude children or students from participating in the network, Napster has also provided the ability to pay for songs without the use of credit cards and users in America can buy pre-paid cards – the Napster Music Card (which work similar to calling cards) to pay for music. The distinct disadvantage with this method is that it centralises the system for the purpose of validation and storage of payment details, regardless of the payment instrument being used. Also, it does not cater for the pay per access or pay per view scenarios where the value of the individual item is not as much as the subscription, consequently the occasional user does not benefit from the subscription. This system does not reflect the true value of the items for sale either.

In terms of P2P file-sharing networks that use a form of payment to either regulate access to the network or prevent 'free-riding' [Eytan Adar 2000], initiatives like The MojoNation Project[23] proposed the use of Mojo tokens or Mojos [Declan McCullagh 2000, Marc Waldman 2001] as a means for gaining access to the network as well as payment for content. Mojos could be earned by providing storage in the network or running a search engine to assist other users to search the network. The project has since been developed into a company that provides P2P based distributed backup storage services. PAST [Antony Rowstron 2001c] proposed the use of smart cards to prevent free-riding on the P2P network, where your smart card tracked the amount of storage you were entitled to based on the amount of storage you were contributing to the network. BitTorrent's robustness is attributed to its use of strong incentives [Bram Cohen 2003] which discourage free-riding by regulating the users' download limit against their upload contribution and the use of non-trivial file piece selection algorithms. However, the classic problem here is the inability for new members to join the system if they do not have substantial contributions.

The PPay [Beverly Yang 2003] payment scheme was designed specifically for the P2P domain; however it still relies on the online presence of coin owners or coin brokers for the purpose of authorisation of a coin transfer.

Summary

As has been illustrated in the preceding sections, majority of the digital payment instruments require the presence of online third parties to complete the transaction. In a small number of instances this verification can be performed offline, however the third party is still required to ensure impartiality and discourage collusion in a transaction. On the basis of these observations it can be concluded that to enable users to pay for content in the P2P domain it is not sufficient to apply existent payment schemes to the domain but that there is need to ensure that the infrastructure can be modified to accommodate the requirements of payment schemes to maintain their integrity.

---

[23] HiveCache Inc, "The MojoNation Project," HiveCache, Inc., Available at http://www.mojonation.net/mojo.html, (Accessed: 14 November 2005).

2.6 Security & trust

Security issues in computing may be classified into the following categories - authentication, integrity, confidentiality and non-repudiation[24]. Napster used centralised servers to provide authentication, where the server let you login and used your connection information in conjunction with your login Id to give you access to the rest of the network. Similarly, most Instant Messaging applications also use centralised servers to manage authentication, the users' integrity is attached to their respective IDs. But in more decentralised P2P systems authentication, integrity, confidentiality and non-repudiation become issues that rely on trust. Due to the dynamic nature of the environment this becomes a complex issue with no trivial solution.

Generic trust building mechanisms used in e-commerce are the use of third parties or reputation management systems. Online auction websites like Ebay allowed users the ability to complain against misbehaving participants, but as resolution of complaints is a long and tedious process, users have resorted to personal mediation/or interaction to evaluate trust at a personal level. Hence, currently large money transactions are usually completed with the assistance of a third party like PayPal or either the buyer or the seller volunteers to assume the risk in the transaction. Whether a user decides to assume risk is based on a personal choice which is aided by ebay's reputation management system which allows transaction participants to leave feedback about a buyer or consumer.

2.6.1 Security

Various tools have been developed to assist developers in securing their applications. Among these the most relevant to this research are smartcards, encryption and hashing techniques.

2.6.1.1 Smartcards

Smartcards [Carol Hovenga Fancher 1997] are credit card sized devices that incorporate one or more integrated circuit (IC) chips. They possess memory, a card operating system and optionally one or more co-processors which can execute programs (chip card applications) and complex arithmetic operations. Microprocessor based smartcards are commonly used to perform digital signatures, authenticate users for access control purposes, and encrypt or decrypt messages. Other smartcards which are mainly used for storage i.e. memory cards only posses

---

[24] The word *repudiation* means refusal to acknowledge a contract or debt. We will frequently use its antonym, non-repudiation, in discussions in this thesis.

some electronic logic components. Smartcards may be considered as one of the securest storage media today.

Smartcards require specialised hardware at the device end to be operable. In the P2P environment they are used to assist with access rights, where the amount of storage allocated to a peer is stored on the peer's smartcard [Antony Rowstron 2001c]. This technique is also used as a fair participation mechanism within this system.

This research focuses on the use of financial incentives as a means to overcome copyright infringement and free-riding within the P2P content sharing environments. The use of smartcards is one method of storing electronic money which would be mobile as well as local to the peer.

2.6.1.2 Encryption

Encryption is the technique, in cryptography, used to convert data in plaintext into ciphertext which can only be deciphered by the intended recipient. There are two main techniques for encryption: Symmetric and asymmetric encryption techniques.

Symmetric [Peter Thorsteinson 2003a] or private-key encryption algorithms use the same secret key to encrypt and decrypt data. These include the obsolescent Data Encryption Standard, the Advanced Encryption Standard, as well as RC4.

Asymmetric [Peter Thorsteinson 2003b] or public-key encryption algorithms include RSA and ElGamal which are based on the principle of generating a related key pair, where one is used to encrypt plaintext and the other is used to decrypt the ciphertext. Public-key encryption can be used for authentication, confidentiality, integrity and non-repudiation. However, DSA (Digital Signature Algorithm) [William Stallings 2006] is less flexible since it can be used for digital signatures and not for confidentiality or symmetric key exchange. Public-key cryptography obviates the need for participants to share secret information (keys) via some secure channel.

Each technique has its pros and cons, whereas symmetric algorithms are faster and provide much greater security than asymmetric algorithms for a given key size. On the other hand public key cryptography is more feasible in environments where participants are not acquainted with each other. The shelf life of an asymmetric key pair is longer as opposed to that of a symmetric key. In spite of these disadvantages, neither of these techniques may be replaced by the other; instead they are applied complementarily to maximum effect (e.g. protocols where these

algorithms are used together PGP [Network Associates Inc. 2000], IPSec [William Stallings 2006], SSL [Stephen Thomas 2000]).

IPSec is an IETF standard that provides authentication, integrity, and privacy services at the datagram layer, allowing the construction of virtual private networks (VPNs). The SSL protocol, developed by Netscape, provides authentication and privacy over the Internet, especially for HTTP (Hypertext Transfer Protocol) which is an application layer protocol.

It is also generally true that asymmetric algorithms tend to be much slower and less secure than symmetric algorithms for a comparable key size. To be effective, asymmetric algorithms should be used with a larger key size, and, to achieve acceptable performance, they are most applicable to small data sizes. Therefore, asymmetric algorithms are usually used to encrypt hash values and symmetric session keys, both of which tend to be rather small in size compared to typical plaintext data.

Public Key Infrastructure (PKI) is most relevant to this research as there are two different keys generated via a common algorithm where one half of the key pair (Public Key) can be exposed to the public while maintaining the secrecy of the second half (Private Key). These key pairs are authenticated by a certificate authority and can be used in digital signatures as well as to encrypt documents.

2.6.1.3 Hashing

Hashing techniques are used to disable the corruption of data en-route and to detect tampering. Algorithms such as SHA1 and MD5 are public one-way hash functions which take a variable-length message to produce a fixed-length hash. It is computationally infeasible to discover the original message from its hash value. One of the common uses of hashing is to speed up a public key digital signature system, where the one-way hash of a long message is computed and signed instead of signing the message itself.

Hashing is commonly used in P2P systems such as Freenet [Ian Clarke 2002] to generate unique file identifiers of each content which is being shared. In Freenet this naming technique assists in the protection of free speech as well as an aide to the underlying routing and location mechanism of the system.

2.6.2 Security mechanisms used in P2P systems

A P2P system like PAST [Antony Rowstron 2001c] uses a combination of smartcards and PKI [Network Associates Inc. 2000] to handle authentication; whereas Groove Networks' [Jon Udell 2001] collaboration platform relies on a combination of PKI and other encryption techniques to ensure that messages are being received from a trusted entity. However, the creators of the Groove Collaboration platform base initial development of trust on a prior knowledge of or relationship with the people they will be collaborating with. Although this assumption suits the Groove collaboration environment, it does not apply to all P2P networks that work in an environment where members do not necessarily know each other or even where the other members may be.

Security considerations in P2P networks, can be divided into two categories [C. Wang 2002] – infrastructure security and application security. Deciding who gets access to a resource and whether the node gaining access is authentic or not, are application level security concerns. These usually deal with authentication/identification, authorisation and access control. These are defined at the application level. Whereas, the availability of the system and how secure it is to use (integrity) are infrastructure security concerns and include integrity of the infrastructure and its availability.

2.6.3 Trust & P2P systems

In a decentralized P2P electronic community, peers often have to interact with unknown or unfamiliar peers and need to manage the risk involved with these transactions without the presence of trusted third parties or trust authorities. There exist various aspects of trust [Tyrone Grandison 2000], and it varies based on the context of the scenarios. For instance, trust in the context of networked and distributed computing systems: it is the remote system as well as the interactions over underlying services such as communication services that need to be trusted. In another context such as a financial transaction a person may only be trusted to deal with financial transactions less than a specific amount.

Intel [Cecily Barnes 2001] developed an API called the Peer-to-Peer Trusted Library, which assists in the development of P2P applications which require peer authentication, secure storage, encryption and digital signatures.

As mentioned earlier there are different aspects of trust based on context, projects such as Publius, Free Haven [Marc Waldman 2001], Mojonation all tackle these aspects of trust in

different ways. Publius believes if content is tamper-resistant and censorship proof then it can be trusted. Free Haven uses a trust network, which creates a trust relationship between the publisher and the server the content is stored on, along with reputation in the network to manage trust. Mojonation uses the concept of Mojos that control access to the system.

Work is being carried out in various projects [Karl Aberer 2001b, Ernesto Damiani 2002, Li Xiong 2002] where trust data is distributed across the P2P network in order to create confidence in the system. This trust data usually carries reputation information on participating peers. Aberer and Despotovic [Karl Aberer 2001b] present a reputation-based trust management system for P2P, it involves the peers running algorithms to determine if they should trust another peer to perform a transaction, hence trusting peers on the basis of their reputation in the system. This trust is computed on the basis of complaints logged against misbehaving peers. In the case of peers who do not consistently participate in the system, their reputation is lower and has to be verified. PeerTrust [Li Xiong 2002] is another reputation-based trust management system, which does not solely rely on complaints information to assess trust but computes it on the basis of various other metrics such as degree of satisfaction, number of interactions between peers and balance factor of trust which takes into account the potential of inaccurate feedback from participating peers. Other reputation-based systems [Ernesto Damiani 2002] allow users to judge the most trust worthy resources to use. But although these trust management schemes assist the P2P network user in determining whom to trust information from, none of these projects cater for electronic commerce in the P2P domain or assist the user in conducting trust worthy e-commerce transactions. Most other trust models rely on certification via a centralised authority to assign trust in electronic markets; this is not feasible in a decentralised environment such as P2P.

Yacine Atif's [Yacine Atif 2002] concepts of a Trust web and Trust Service Providers (TSP) proposes to establish confidence between participants to facilitate e-commerce transactions by utilising a set of intermediaries to build a trusted path from the consumer to the merchant until a TSP is found who is trusted by the merchant or until a pre-set constraint (i.e. the amount that may be paid to the set of TSPs is exhausted) is met. The trust path is established before the e-commerce transaction may be carried out. The algorithm used, aids the user to form the best possible trust path from the user to the merchant.

2.7 Web Services

Devices that are capable of providing various services form a part of distributed networking today. The Internet enables devices to have access to services which are not locally present via

the use of Web Services Architecture [Michael Champion 2002]. These services are self-contained applications that can perform simple functions or complicated processes. To allow interoperability the Web Services Architecture uses XML-based standards such as WSDL (Web Services Description Language) and SOAP.

Although web services are run in a centralised manner, i.e. they use central registries for lookup of services (UDDI), they are of relevance to us mainly because Peer Services also conform to the same behaviour where services are registered and can be looked up and used, however in this instance the system is not centralised because services generally tend to be replicated across a number of peers in the network.

2.7.1 Extensible Markup Language (XML)

XML was devised to make it easier to use SGML (Standard Generalized Markup Language) which is the international standard for defining descriptions of the structure of different types of electronic document. As SGML had too many options and was too complex to use, XML was created as a cut-down version of SGML. XML is one of the many projects run by the World Wide Web Consortium (W3C), which is supervised by the XML Working Group[25].

XML is a markup specification language used to design ways of describing information for use by programs or for transmission or storage. XML uses the concept of markup, which is that delimiter tags are used to delimit the start and end of a document or to delimit sections within the document like type declarations, processing instructions, character references, comments and many others. But it is also a meta-language as it allows users to design their own language to describe document structures. It uses concepts such as schemas to define documents with different structures so that they are more usable. XML can be used to exchange large amounts of data across the Web and between systems in a way which is understandable by different platforms, which assists in interoperability between platforms and systems. Many office applications [Rita E. Knox 2003], like Microsoft Office XP and Sun's StarOffice, allow users to save documents as XML to enable interoperability.

XML is described in some detail here because it has a bearing on work in the P2P field as well as on this body of work. The framework proposed by Arthorne et al in [Neal Arthorne 2003] describes P2P file-sharing communities in terms of structured XML documents, here each file-sharing community is modelled as a structured document containing metadata information about

[25] The W3C XML Website, W3C, Available at http://www.w3.org/XML/

the type of document that community shares. This allows the communities to be searched for in the same way a document would be searched for. Every shared document within a community also has additional metadata information.

The financial industry has identified the need for a unified standard of communication between financial institutions and hence announced the creation of a new XML standard [Nick Huber 2003] for payments. This proposal intends to create a unified XML based payments standard which would ensure that any payment method used would be 'interoperable' between different financial institutions; this would make payments made globally, viable and redeemable in any country via the financial institutions. This has a direct bearing on this research, as one of the main objectives is to make available the data necessary to enable the redemption of payments from financial institutions. This data is archived in the P2P network for the purposes of non-repudiation and to satisfy data archiving laws.

The Resource Description Framework (RDF)[Renato Iannella 1998] is an application of XML, which allows resources to be described in terms of metadata. If we treat a service as a resource then it is conceptually possible to define these services in terms of metadata as well. Defining a service in terms of metadata allows it to be located semantically.

2.8 Summary

This Chapter provides a concise definition of what P2P technologies mean in the context of this thesis and a survey of the current trends in P2P systems research covering various criteria such as naming, routing, lookup and discovery and the participation incentives being used including monetary and non-tangible compensation schemes. We have also observed that solutions to problems in this domain are many and derived from a variety of subject areas. Consequently, a detailed literature review of pertinent subject areas was also included.

Through the course of the literature review it was observed that apart from the problem of copyright infringement prevalent in P2P content-sharing networks there is also a trend of free-riding, which detracts from the benefits of P2P networks as an efficient method for content exchange. Other issues such as trust and confidence in the system were also uncovered. It became apparent that P2P technologies though filled with advantages for resource sharing and utilisation of redundant resources at the edge of the Internet are also fraught with challenges. If a compensatory technique is to be applied to overcome copyright infringement, the various parts of an e-commerce transaction need to be understood and explored. These led to the listing of requirements for payment instruments and transaction requirements. Based on these issues

pertinent requirements for the various components of a desirable system were unearthed which shall be discussed in the next Chapter.

Chapter 3

3 HIGH-LEVEL ARCHITECTURAL REQUIREMENTS AND MODELS

The previous Chapter described P2P technologies and the areas of research pertinent to this thesis. It concluded that using the power of P2P technologies can provide an excellent medium for content distribution and would empower personal publication at nominal cost to producers. However, this area is fraught with content theft and copyright violation. This Chapter will show how these technologies impact this research.

This Chapter defines the requirements for a system which facilitates equitable digital content exchange in the P2P domain. Further it introduces the concepts and models developed as a solution to fulfil these requirements and address the issues which are raised. These concepts will be used to construct a framework which will satisfy the objectives of this work. It is important to remind the reader that from this point on our focus is on content-sharing P2P networks.

3.1 System Requirements

As illustrated in Chapter 2, this research spans a number of areas. Accordingly it is necessary to state the requirements for each area as applicable to the overall objectives of the research.

The primary goal of this research is to address the issue of copyright violation and free-riding in current P2P content-sharing networks. We intend to accomplish this by providing a system of payment that would encourage fair system use as well as ensure that content owners are compensated for their work. To do this the primary requirements can be divided into three categories: Copyright, Transaction and Payment and P2P infrastructure requirements.

3.1.1 Copyright requirements

To uphold copyright and to encourage fair participation in the network the following requirements are defined:

- The content owner should receive compensation every time his/her content is downloaded.

There exist remuneration models which demand compensation to be proportional to the usage of content. Although this is not an accurate estimation of usage, I am working on the premise that a downloaded item shall be used at least once. As the payment to be made for this use is determined by the content owner, this conforms to established commercial practice where a purchased commodity is paid for once.

- In cases where a node other than the owner's node is used to propagate content, that node should also be compensated for its resource usage.

  One method to ensure content propagation and persistence in a P2P content-sharing system is by virtue of its popularity and subsequent replication [Kavitha Ranganathan 2002]. Following on from the scenario in Chapter 1 (§1.5), the content owned by Alice may be available on Bob's node. In this scenario, Bob should receive compensation for the use of his resources (disk storage, bandwidth) as well as gain by acting as a distributor for Alice.

The Cascading Payments model presented later in this thesis was developed to fulfil these requirements. This model is described in detail in this Chapter and its design is discussed in Chapter 4.

3.1.2 Transaction & payment requirements

These requirements focus on the financial transaction itself which should conform with fair practice.

- Buyers should be informed of the value of the content prior to purchase.

  By virtue of its nature a P2P content-sharing system gives the user access to the same content via multiple sources, informing the user of the price of the content prior to download gives the buyer the power to make an informed decision and encourages competition; consequently the system is fair to both buyer and seller.
- There should be guaranteed delivery of a service in exchange for the payment, in such a way that no single participant in the transaction is treated unfairly.
- Any monetary transaction needs to be recorded for the purposes of non-repudiation[26] and
- By law transaction records need to be maintained for a number of years, e.g. in UK this is seven years and ten years in France.

[26] The word *repudiation* means refusal to acknowledge a contract or debt. We will frequently use its antonym, non-repudiation, in discussions in this thesis.

3.1.3 P2P infrastructure requirements

To ensure the system remains decentralised and there is no centralised control; also ensuring that the solution is viable in any P2P content-sharing network, regardless of base technologies,

- No single party may act as a trusted third party to validate all the transactions.
- Payment schemes used, can not use online verification where a single online central authority is needed.
- The system should also ensure that payments are
  - Accurately distributed to the relevant peers in a dynamic P2P environment and
  - Ensure only those peers for whom the payment is meant may redeem the payment. This is a requirement because in a distributed environment such as P2P, proper system functioning needs to utilise the resources of the entire network. Consequently there will be situations where a payment may reside on other peers en route to the payee.

These requirements are considered infrastructure requirements because in the P2P system, it is the responsibility of the network to ensure requests are accurately handled and queries are accurately directed. Similarly, validation of a source is the infrastructure's responsibility.

In the pursuit of a solution to the problem domain some secondary requirements were realised. These include the requirements related to trust: trust in both the payment instruments and the types of payment instruments suited to the P2P domain. These requirements are not the main focus of this research; however they do have a bearing on the problem domain and hence are mentioned here.

3.1.4 Trust related requirements

Trust is an issue in any networked environment, where the various components of the network need to be trusted. For instance, in the P2P content exchange environment, there should be trust in the Content Exchange Application[27] the user intends to use to ensure that it will not damage the user's system. Trust is placed in the content being exchanged via this application which also implies a trust in the providers of the content. On the Internet this trust is hard to maintain due to the lack of relationships between the participants. Whereas in Groove [Roger Dingledine 2001] the establishment of trust is based on mutual knowledge of the participants, the content-sharing communities, as described in the scenario, are formed in an ad hoc fashion where users do not know each other (in the majority of cases) and hence may not trust each other.

[27] In this thesis the Content Exchange Application allows the user the ability to locate and purchase content for payment in the P2P domain

For the purpose of this research some trust issues are accepted, such as the content being exchanged via the application is not harmful and neither is the application used to share this content. Similarly, there is a degree of trust placed in the infrastructure itself. However, it cannot be taken for granted that two peers in the system will automatically trust each other and make a fair exchange of content for money. For this purpose a third party (who may be trusted by both parties) or a method for fair exchange is needed. On the other hand, a single third party is not suitable for P2P networks as it restricts their scalability. True P2P networks should be scalable and the use of a single third party in such an environment would make the network unstable by presenting a single point of failure. Hence, we propose the use of an overlay network of bank peers who will act as the trusted third party. This network of bank peers would allow two peers to exchange digital content for payment by providing transaction validation.

3.1.5 Trust in traditional payment instruments

In day-to-day monetary transactions in the real world, trust between participants is usually established by the physical presence of the participants and the established trust in the monetary instrument i.e. the payment instrument being used e.g. Cash, Credit Cards, Cheques or Bank Drafts. Here the risk in the transaction is removed because it is relegated to a trusted third party, like a bank or the government. Payment systems have been devised to imitate this for the digital world [Efraim Turban 1999] as well, where there are methodologies in place to process transactions paid for by credit cards, digital cheques, and money transfers.

3.1.6 Attributes of desirable payment instruments

In the context of the scenario presented in section 1.5 page 7 where users should be allowed to pay for content of varying values, there is a need for payment systems which deal with micropayments, like Payword [Ellis Chi 1997] or variations of Rivest's lottery scheme [Silvio Micali 2002], as well as macropayments like traditional credit cards [G. Winfield Treese 1999]. Hence, the system should be flexible to accommodate different types of payment instruments and be capable of recording the details of the various payment methodologies.

It is necessary to draw the readers' attention to the fact that the system in question is being developed to enable small producers and consumers to carry out e-commerce via P2P networks while protecting the producers' copyrights. The system is also being designed for maximum flexibility; hence it should accommodate both macro and micro payment schemes.

Bearing in mind the nature of the P2P environment, it is also necessary to stipulate the requirements of desirable payment scheme attributes. Primary among these is the need for payment schemes which are not credit based, because the environment is ad hoc and there is no guarantee that a peer who owes money to another peer would ever return after the transaction is complete. So it is necessary that applicable payment schemes are debit based.

The following sections shall describe this thesis' contributions to knowledge to fulfil the above requirements. These include the development of the Cascading Payments Model (CPM) (§3.2) and the Overlay Network of Bank Peers (§3.3) as well as the use of service integration (§3.4) to provide maximum flexibility in the system. These concepts are realised by the Cascading Payment Content Exchange (CasPaCE) Framework, discussed in section 3.5.

3.2 Cascading Payments Model (CPM)

One of the key requirements within this research is to fairly compensate users for the use of their intellectual property rights and to empower them to economically distribute their content and profit from it. Consequently, the concept of cascading payments was developed and is explained below. The policies, protocols and algorithms required to facilitate this model are also described in detail.

3.2.1 Cascading payments

There are many aspects to compensation for intellectual property rights such as monetary compensation and reputation based compensation which include archiving, persistence and many others (a detailed list is given in [L. Jean Camp 2002]). This work focuses on *monetary compensation* as a method for intellectual property rights compensation. The traditional method for monetary recompense to property owners is via a mechanism of royalty and commission, whereby the person who holds the property rights to a commodity (be it content, goods or ideas) is entitled to a share of the profits (a royalty) and the intermediary who acts as a distributor for said commodity makes a profit in the form of a commission.

Traditional systems for digital content distribution (Figure 3.1a) such as web portals sell e-books, music and video. This model is made up of a static distributor who takes responsibility for commodity sales and advertising. The content owner receives royalties and the distributor receives commission. To enable this, the distributor has to maintain dedicated servers, web portals and payment gateways. Our model makes it possible for every member of the system to be a potential distributor and distributes the responsibilities of server maintenance, content

advertising and content distribution across the network. This model is made possible in a present day P2P network (Figure 3.1b).

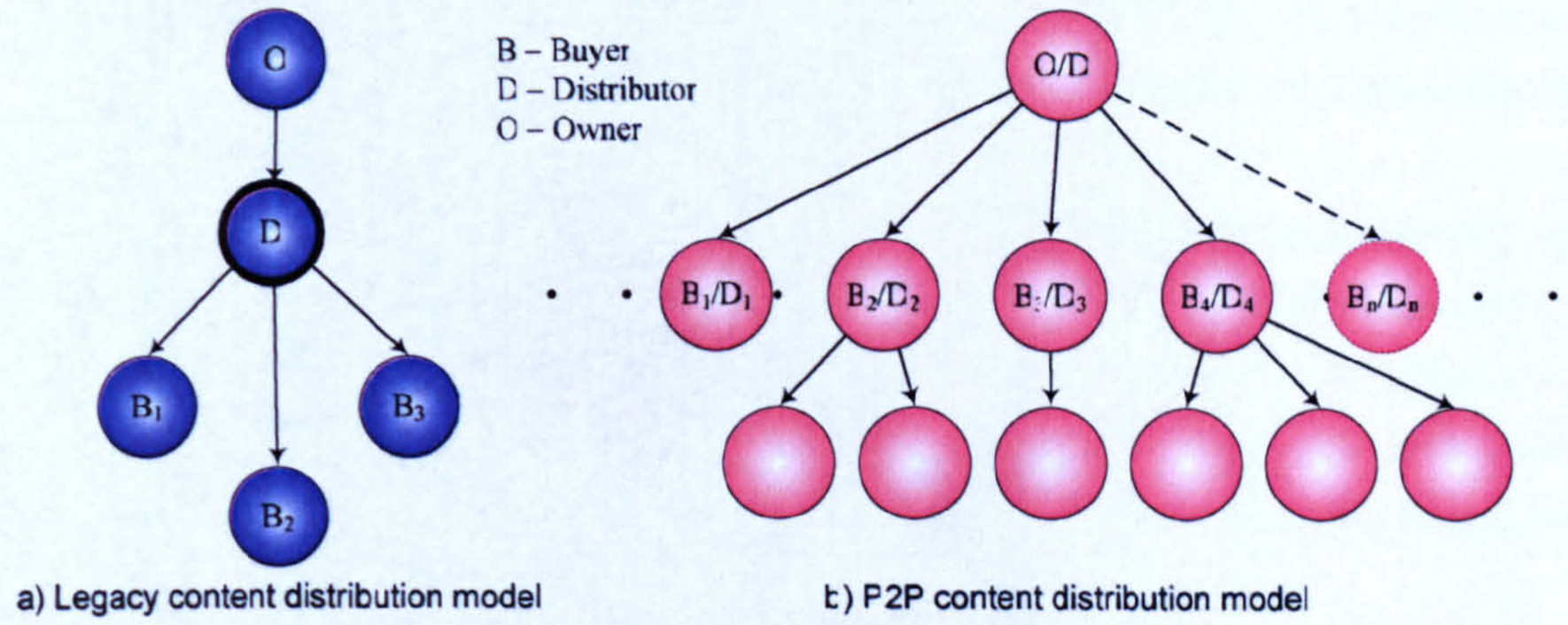


Figure 3.1: Legacy Distribution Vs P2P Distribution Model

We define *cascading payments* [Gurleen Arora 2003] as the process whereby payment flows towards the content owner through intermediary peers in a P2P network. Here every time the content is propagated from the seller to the buyer a proportion of the payment (the royalty) is sent to the content owner who holds the property rights to the content and the rest of the payment is given to the seller (who may be the intermediary) as his commission. Figure 3.2 demonstrates our concept, where A, B, C and D are participating peers in a P2P network. If A is the owner of content X, which B wishes to purchase, the payment flows to A; when this content X is discovered at B by node C who wishes to purchase it, a proportion of the payment should be delivered to A as the content owner and the rest to B as the content distributor.

In this scenario, peers A-D all provide resources to the network in the form of storage space and bandwidth. A gets compensated for his property rights and B and C get compensated as distributors for the use of their resources. Hence, satisfying the rules of economics and not violating copyright, while at the same time fully utilising the power of P2P technology.

For every transaction there is only one royalty and one commission paid. In instances where the owner is the seller, the commission is considered to be zero and the same model applies. This approach was adopted as opposed to the creation of a chain of distributors (a pyramid scheme) because reducing the number of distributors increases the profit margin for the owner as well as the distributor. It also removes the necessity to maintain information of every node in a never ending daisy-chain of distributor nodes.

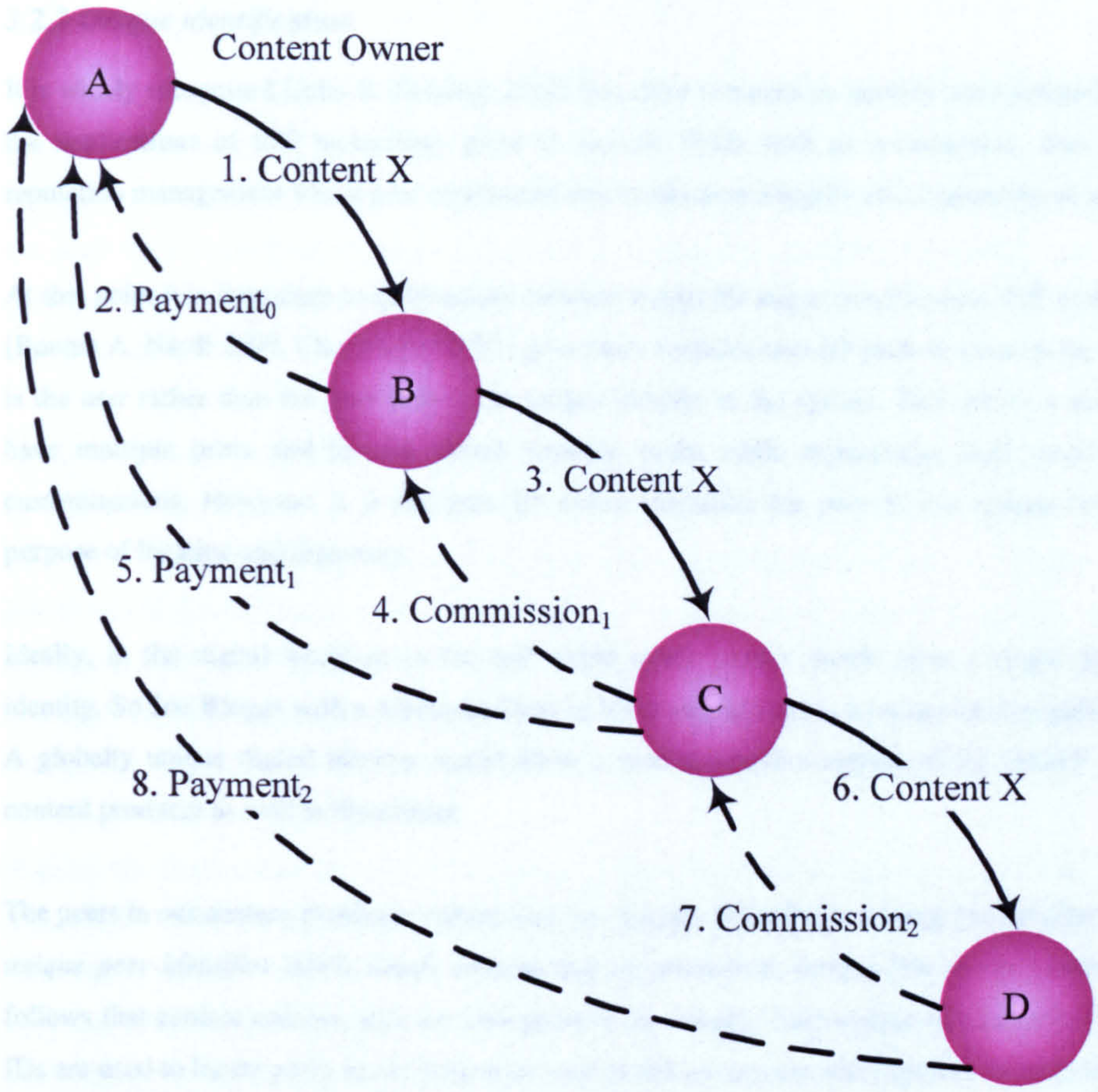


Figure 3.2: Cascading Payments Model

As any P2P environment is dynamic and peer presence in the system is transient, for our model to work it is necessary that:

1. The peer is uniquely identifiable across the network.
2. The content carries its owner's information with it wherever it is moved to in the network. This information includes the royalty and commission rules setup by the owner which could be in the form of a royalty to commission ratio or fixed value payment agreement or a complex rule policy which includes market state parameters.
3. The Buyer pays a single fee for the content, the system splits it up into royalty and commission (where applicable).
4. The royalty is delivered to the owner and the commission to the seller/distributor.

The following sections will describe the techniques used to facilitate this model.

3.2.2 Unique identification

It is widely recognised [John R. Douceur 2002] that there is a need to identify peers uniquely as the applications of P2P technology grow to include fields such as e-commerce, trust and reputation management where peer reputations and transaction integrity are of great importance.

At this point it is important to differentiate between a peer ID and a user ID; most P2P systems [Bonnie A. Nardi 2000, Clay Shirky 2001] give users a unique user ID such as a username so it is the user rather than the peer who is the unique identity in the system. This allows a user to have multiple peers and he can switch between peers while maintaining their respective customisations. However, it is the peer ID which identifies the peer in the system for the purpose of location and discovery.

Ideally, in the digital world as in the real world every person should have a single global identity. So Joe Bloggs with a particular Date of Birth and address is a unique person globally. A globally unique digital identity would allow a user to create a reputation for himself as a content producer as well as distributor.

The peers in our content exchange system may be uniquely identifiable through the medium of a *unique peer identifier* (UPI) which ensures that all peers have unique IDs in the system. It follows that content owners, who are also peers in the system, have unique IDs as well. These IDs are used to locate peers in our system as well as deliver content and payment to peers in the P2P content sharing system to enable payments to cascade.

The peer ID is made unique by including a date time stamp along with peer specific information at first connection to the network. However, to make this ID exclusive, every peer has a PKI key pair; by including the peer's PKI public key in the ID we can ensure that every peer who participates in the system can be uniquely identified. This unique peer identifier (UPI) is then used to identify peers, communicate with them and it also acts as a method for the distribution of their public key which is a component of their UPI.

Once a content owner has been uniquely identified it is possible to link his content to his identity exclusively and start building his reputation as a content producer as well as distributor.

3.2.3 Copyright protection

To uphold the notion of copyright protection, it is important to ensure that the content owner is always compensated. To enable our CPM in a completely decentralised system, it is necessary to ensure that the content owner's information is always moved with the content. This removes the need for a centralised directory to keep track of content copyright information. Also, the need for expensive procedures to cross check the validity of a copyright certificate from a single source is removed.

In our system, when the content is created the owner creates a *content descriptor* which contains its copyright information as well as the *royalty-commission policy*. This policy could be in the form of a royalty to commission ratio or fixed value payment agreement or a complex rule policy which includes market state parameters such as reducing the cost of the content when demand is low. The copyright information includes the content owner's identification i.e. his UPI and a royalty-commission parameter defined by him at content creation. These details may not be altered once they are committed. This information is then bound to the content itself.

If some content satisfies the criteria of a search within the system, it is the content descriptor which is extracted (in read-only form) from the content and sent as part of the *search results*. The content descriptor forms a part of the search results along with other information necessary for the routing and processing of a content purchase request (this information will be detailed later). However, the key to the system is the content descriptor and its successful propagation with the content that it describes.

Much research is being carried out in the field of digital watermarking [I. Cox 2000, Matt L. Miller 1998], which can used to embed information within digital content. At present this technique is most effectively used to watermark audio or video files that have particular patterns that may be used by the watermarking algorithm to embed the information in a non-destructive fashion.

We recognise that inserting a watermark in a file can be a non-trivial task, e.g. it can be as simple as writing the owner's name and address within the document and then attaching the hash of this document to the file to assist in detection of watermark modification. However, we believe this is not entirely efficient. Watermarking has not been efficiently developed for plain ASCII text files as yet, but it is possible to watermark formatted documents [Jack T. Brassil 1995]. Similarly, it is possible to watermark other types of digital content such as multimedia content (still images, moving images, sound and numeric data files) [Neil F. Johnson 1999].

These techniques can be used to embed owner, royalty and commission details within digital content that can be exchanged via our system. In this way it is possible to transmit owner information along with the content itself obviating the need for the content owner to maintain a constant online presence or the need for centralised servers to monitor content propagation and content management.

Steganography [Ross J. Anderson 1998, Scott Craver 1998] may be used at a later date to hide the existence of the watermark itself, however at present the use of a digital watermark is sufficient, whether we hide it from the intended recipient of the content is irrelevant as participants in the system are well aware of the fact that their IPR is being protected.

3.2.4 Payment separation

The Buyer has to generate payments which satisfy the conditions for the protection of the copyright. In the CPM, the buyer creates the royalty and the commission payments in accordance with the rules set out by the content owner. These rules are presented to the buyer in the *transaction negotiation phase* which precedes the transaction. The *transaction negotiation phase* would ensure that the buyer is made aware of the value of the content (Sale Price, where sale price = royalty + commission) prior to commencing the transaction. A typical *transaction negotiation phase* would involve a search for the desired content which would result in multiple responses; the subsequent selection of one of the instances of the content would constitute a *purchase request.*

We define Payment Separation as the process where the buyer generates two payments, the royalty and the commission, as dictated by the payment rules set up by the Seller(s). Both, the content owner and the content distributor are sellers in the system. The two payments are then encrypted with the public keys of the respective payees (owner and distributor) and are ready for dissemination. The public keys are exposed to the buyer through the UPI of each payee as described earlier in section 3.2.2 on page 55. Figure 3.3 illustrates the concept of payment separation, where the 'Seller Intermediary Peer' is the distributor. The payment generation process is dependent on the choice of payment scheme. The Payment Details, illustrated here, form a subset of the search results returned to the buyer in the transaction negotiation phase.

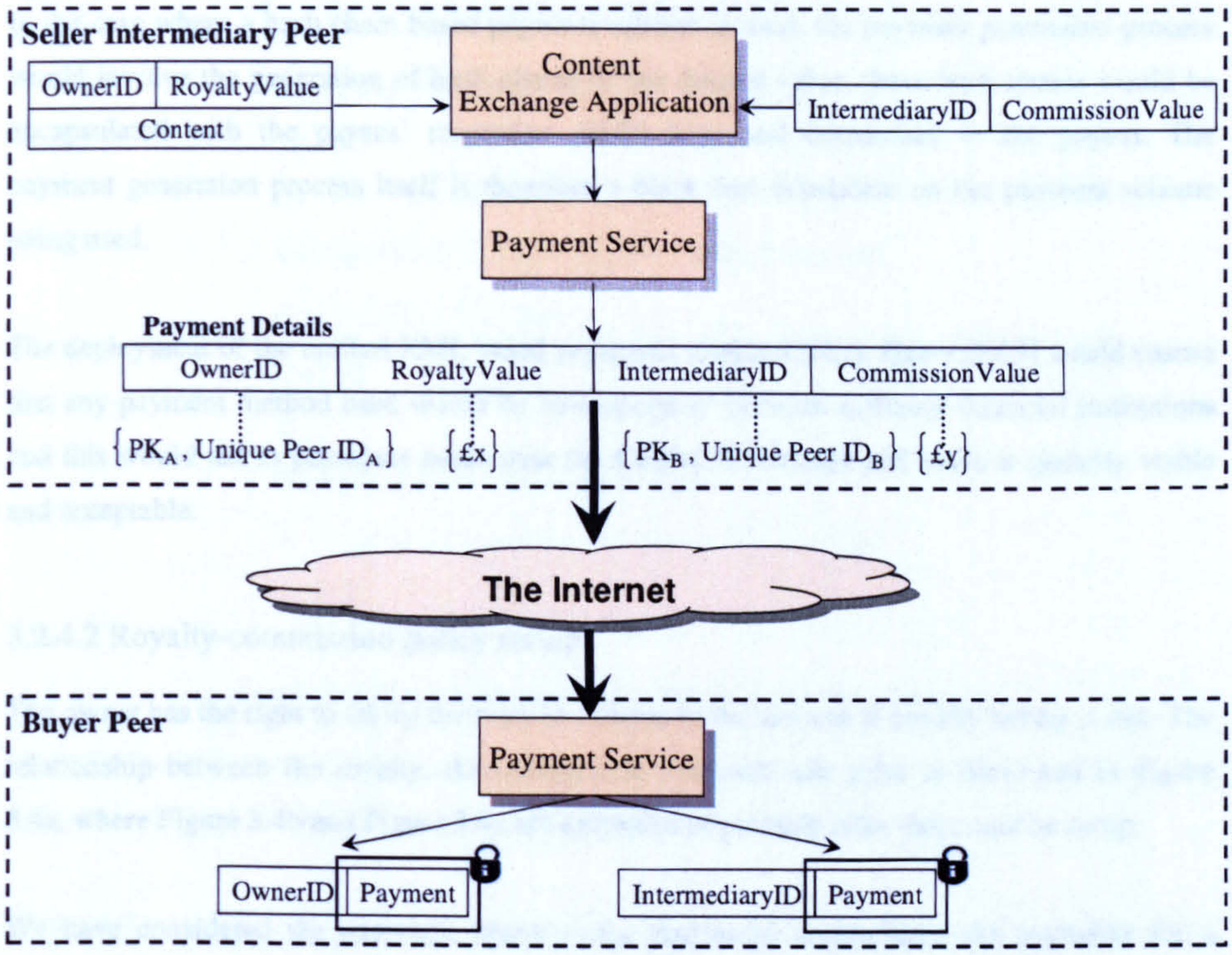


Figure 3.3: Payment separation

3.2.4.1 Payment generation

Applying traditional digital currency schemes to the P2P domain is challenging due to the security constraints of the payment schemes as well as the dynamic nature of P2P systems. Ideally e-coins would have been a desirable mode of payment in this scenario; however they are not suitable for the P2P domain due to the need for a constant broker presence for coin validation and checking for double-spending. In the course of this research it was determined that payment schemes which would be most suitable for this domain while retaining their security would have to be offline payment schemes as proposed by Rivest et. al. [Silvio Micali 2002, Ronald Rivest 1996]. Although these payment schemes are not considered as secure as the online verification schemes it is a trade-off we have considered. These schemes do not require the online presence of the broker during the payment transaction itself. PPay[Beverly Yang 2003] and PayCash[Jon M. Peha 2004] are payment schemes which have been designed specifically for the P2P domain. The PPay scheme, based on the concept of transferable coins, suits the needs of our environment as it is closely linked to Rivest et al's Payword style of micropayment, which we stated is most suited to the P2P domain (§3.1.6 pg 51). Supplemented by a non-repudiation technique the PPay scheme would work well within our scenario.

In the case where a hash chain based payment scheme is used, the payment generation process would involve the generation of hash chains of the desired value, these hash chains would be encapsulated with the payees' respective public keys and despatched to the payees. The payment generation process itself is therefore a black box dependent on the payment scheme being used.

The deployment of the unified XML based payments standard [Nick Huber 2003] would ensure that any payment method used would be 'interoperable' between different financial institutions and this would aid in payments made over the CasPaCE Network and make it globally viable and acceptable.

3.2.4.2 Royalty-commission policy set-up

The owner has the right to set up the rules in relation to the amount of royalty he/she is due. The relationship between the royalty, the commission and total sale price is illustrated in Figure 3.4a, where Figure 3.4b and Figure 3.4c are examples of possible rules that could be setup.

We have considered the approach where every distributor accumulates the payments for a particular item of content up to a limit before making a one off royalty payment to the owner. However, this approach was abandoned in favour of the current approach, primarily due to the possible security loops inherent in this approach. The accumulative approach would remove the need for micropayments in the P2P domain, but it would magnify the problems associated with the processing of macropayments in an ad hoc environment. For instance, any distributors would have to be registered with macropayment scheme providers such as MasterCard, Visa or a bank (to process cheques). It would also be necessary to monitor distributors and the values of the royalties being accumulated to ensure there were no major instances of fraud.

Our chosen approach is better suited simply because royalty payments, regardless of the value, are encrypted with the owner's key and sent to him. In the current approach it is possible that some royalty payments might be lost, however the possibility of frauds amounting to large values are slim as royalties are distributed by the system, encrypted and their distribution is logged which assists in non-repudiation and we also assume that the system is trust worthy (§3.1.4).

The buyer can act as a distributor and re-sell the purchased content. To do this the buyer can set up a commission value before the content is advertised for distribution these commission details are included in the content information sent to the searcher. The buyer does not have access to

a)

Sale Price(S)=Royalty(R)+Commission(C)

$S=R+C$

b)

Example Rule 1: Percentage of transaction value

$R=S\times(20\div100)$ then

$C=S-R$

c)

Example Rule 2: Fixed fee royalty

R = x where x is a constant such as 20 units then

$C=S-R$

Figure 3.4: Relation between sale price, royalty & commission

the watermarked content descriptor (owner details). In this way the original content is not tampered with and the integrity of the owner's details are maintained.

This mechanism allows the owner's royalty information to remain unchanged, however every subsequent distributor has the ability to set his own commission parameters, as long as the final payments conform to the owner's royalty policy there is no conflict. In a competitive market, a distributor who sets his commission value too high affects the overall price of the goods and hence could reduce his chances of a sale and hence profit. In a P2P community where content saturation is low the distributor would stand to make a generous profit.

3.2.5 Payment distribution

Once the payment has been separated into royalty and commission it has to be distributed to the respective peers, i.e. the owner and the distributor. In the CPM the distributor receives payment in real time, i.e. while the exchange is taking place because he is directly involved in the transaction, but the owner may or may not be online simultaneously. In that case we have to ensure that the royalty payment is accurately routed to the owner in such a way that only the owner may redeem the payment in the future and also ensure that no other intervening peer can hijack and use the payment. Two techniques have been devised to accomplish this in the CPM, *payment pushing* and *payment collection*. These techniques are mutually exclusive however they complement each other. Here the payment pushing pushes the payment closer to its destination and payment collection can simultaneously initiate a pull on these payments. These techniques rely on the underlying P2P routing protocol.

Location based P2P routing protocols use location metrics to determine the nearness of a node. In some instances this closeness is determined in the context of geographical location [J. Li 2000] and in others on the basis of logical proximity [Antony Rowstron 2001a, Ion Stoica 2001]. The payment pushing and collection techniques devised in this thesis exploit the location estimation property of these P2P routing protocols.

The payment pushing technique, works by constantly moving the payment closer to its destination. Here, the royalty payment is routed closer to the owner node, if the destination node is offline for any reason, the payment remains at the nearest node until the owner node is live again and ready to accept payment. In instances where a peer may go offline, systems such as PAST [Antony Rowstron 2001c] ensure that the content is redistributed to neighbouring nodes, similarly our payments get moved to a node closer to its destination. However, in this case a scenario may arise where if a payment bearing peer goes offline forever or is damaged for some reason and the payment is lost. To overcome this problem, we optimised our payment pushing algorithm by including a verification loop, where we make multiple copies of the encapsulated payment; these payments are then propagated via multiple paths towards the destination. The first payment to arrive at the destination sends back a received receipt along the same return path, if it exists otherwise it optimises the path based on the underlying routing protocol. The payment instrument ensures that it is not possible to redeem the same payment more than once.

The payment collection technique works in contrast to the payment pushing technique i.e. it pulls the payments, the content owner peer requests neighbouring peers for payments that may belong to him to be routed back to him. Here, the requests are broadcast to the neighbouring nodes until a payment is found, then the location of the payment is routed back to the owner node and the owner node forms a direct connection with the payment location and collects the payment. This is similar to the broadcast location and delivery techniques used in standard P2P systems. However, the system should be able to scale the number of hops to neighbouring peers to reduce unnecessary network traffic, we recognise that limiting the number of hops reduces a peer's search horizon in broadcast based lookup schemes, however our payment pushing technique will ensure that a payment will be constantly moved closer to the target negating the effect of the limited search horizon.

In both techniques, the payments can not be wrongly redeemed by intermediary nodes because they are encapsulated / encrypted with the recipient's public key in the payment separation process.

We have used a PKI in the CPM because it serves to secure the complete cascading payment process. It assists in upholding copyright:

- by uniquely identifying the peers,
- during payment separation and distribution and
- by securing the payments against wrongful redemption.

This section described how the CPM fulfils the requirements set out in section 3.1, by using techniques such as unique peer identification, copyright information propagation, payment separation and payment distribution. The following section will describe the techniques used to ensure that the payment transaction itself can not be repudiated and to ensure that there is no unfairness in the transaction.

3.3 Our Overlay Network of Bank Peers model

P2P networks form a logical layer over the Internet called an *overlay*, whereby the underlying physical connections between Internet nodes are not necessarily the actual structure of the P2P system; instead this connectivity aids the P2P network to form connections with widely distributed nodes across the world. Also, the routing mechanisms used by these P2P systems utilise the Internet as the transport medium but have their own routing protocols independent of or working over the Domain Name System (DNS). By utilising the Internet's base transport and communication protocols the current P2P networks assist in communication across platforms and devices of varying capabilities.

Super-nodes are nodes in a P2P network that provide extra functionality than the rest of the nodes in the network. In JXTA [Bernard Traversat 2003] they are defined as nodes that provide important infrastructure functionality to the network by acting as relays. KaZaA[28] is a Gnutella [Distributed Search Solutions 2001] style file-sharing P2P application that uses super-nodes to assist in the indexing of frequent requests so as to enable faster search times; this could be interpreted as the formation of an overlay network of super-nodes which function within the KaZaA network to provide added benefits to the entire system. My concept of an Overlay Network of Bank peers is inspired by these concepts of super-nodes and overlay networks. As

[28] Since KaZaA's inception, many techniques have been used by its developers to overcome the network overloading problems in the legacy Gnutella protocol. The latest version of KaZaA is available at http://www.kazaa.com/ and it has evolved beyond its file-sharing days into a multi-application P2P technology.

mentioned in Chapter 2 it is possible to use these overlays to provide extra functionality to the P2P system without changing the underlying layers.

The main functionality of the overlay network proposed in this thesis is to act as a joint 'Bank' for the content-exchange network. Here we need to distinguish between a content-sharing network and content-exchange network; content-sharing implies the content is shared for free whereas content-exchange implies there is something returned or exchanged and some value added to the process.

When handling payments, any system needs to address the issues of maintaining transaction records, ensuring transactions are atomic, ensuring transaction records support non-repudiation and the payment is valid (i.e. payment being used has not been previously spent, or is not an invalid method of payment). In a centralised environment all this functionality is managed by some central server, in the pure P2P environment there is a need to manage distributed transaction data. As the nodes in the overlay network would act as intermediaries they would also be in a position to log the transaction details, such as the identity of the participants and the content that was exchanged, along with the cost of the transaction. They would also assist in fair exchange of payment for content. The records would assist in non-repudiation of payment and content delivery; in case of disputes the records could be recalled to settle the dispute, hence, the term 'bank' peers and the Overlay Network of Bank Peers (Figure 3.5). For the purpose of this research our definition of a *Bank Peer* is a peer that acts as an intermediary for an e-commerce transaction by providing validation and non-repudiation services; hence a Bank Peer is a type of super-node as it provides additional functionality to a Normal Peer who can only buy and sell content.

Peers A-J in Figure 3.5 are all members of the P2P content exchange network, at present peers H, I and J act as 'banks' for the system, as the system state is dynamic and constantly changing it is possible that at a later stage peer F may become part of the overlay network to provide 'bank' functionality to a pair of peers who wish to exchange content. As mentioned earlier (§2.3), P2P overlay networks form a logical layer on top of the Internet, while still utilising the Internet's base transport protocols. In a similar fashion the peers in my Overlay Network of Bank Peers belong in the same plane as the normal peers but their functionality acts like a logical layer above the content exchange network.

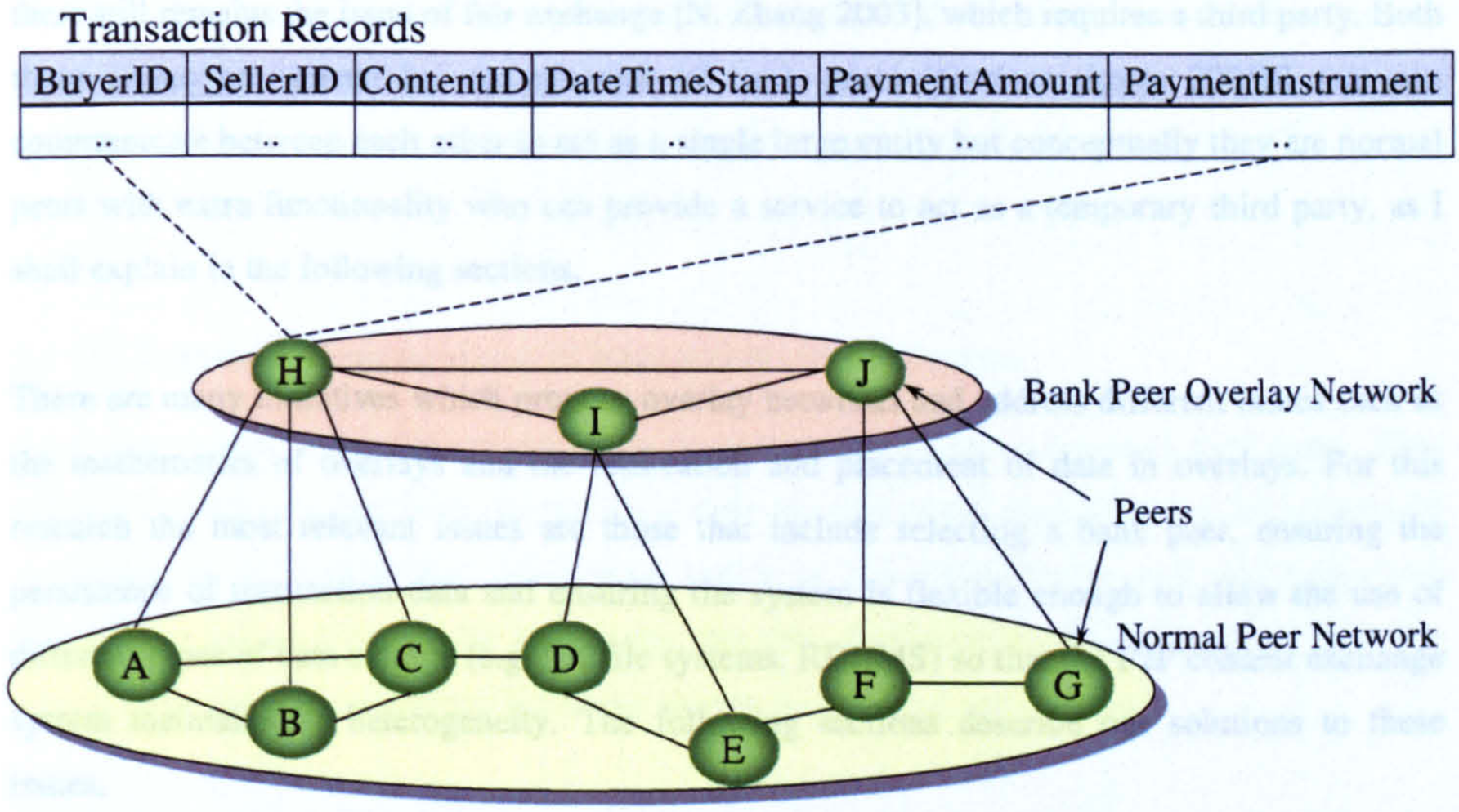


Figure 3.5: Overlay Network of Bank Peers

The transaction details (Figure 3.5) that have to be stored are BuyerID, SellerID, ContentID, DateTimeStamp, PaymentAmount, PaymentInstrument. Here the 'PaymentInstrument' refers to the payment method used, i.e. the name or identification of a payment scheme such as Digicash's Ecash [Daniel Minoli 1997] or hash chain based payment schemes like Payword [Ronald Rivest 1996] or [Hitesh Tewari 2003]. In order to provide confidence between the Buyer and Seller peer the sequence of events would be as follows:

- The Buyer Peer and Seller Peer select a common Bank Peer to act as an intermediary to facilitate the exchange.
- The common Bank Peer crosschecks for transaction records corresponding to the same transaction within its data store and within the Bank Peer Network for that Buyer and his payment.
- Once the transaction records have been checked for double spending, the Bank Peer authorises the transaction.
- The Buyer and Seller make the exchange; moving the content to the Buyer and the payment to the Seller. The transaction details are saved at the Bank Peer for non-repudiation.

One of the main challenges of this work is that any form of digital payment scheme requires some form of validation by a third party. In a P2P environment, specifying a single third party would make the system centralised and hence not pure P2P. However, there exist payment schemes that do not require online verification by a third party and other schemes that can be modified so that there is no need for online verification [Hitesh Tewari 2003]. At this point

there still remains the issue of fair exchange [N. Zhang 2003], which requires a third party. Both these issues are solved by our network of bank peers [Gurleen Arora 2005b] that can communicate between each other to act as a single large entity but conceptually they are normal peers with extra functionality who can provide a service to act as a temporary third party, as I shall explain in the following sections.

There are many initiatives which propose overlay networks and address different issues such as the mathematics of overlays and the replication and placement of data in overlays. For this research the most relevant issues are those that include selecting a bank peer, ensuring the persistence of transaction data and ensuring the system is flexible enough to allow the use of different types of data storage (e.g. flat file systems, RDBMS) so that the P2P content exchange system maintains its heterogeneity. The following sections describe our solutions to these issues.

3.3.1 The bank as a third party

By acting as a third party in a transaction the bank provides a transaction validation and authentication service. It can also be used as a mediator if there is a communication breakdown before a transaction has been successfully completed. In order to provide this functionality first a bank peer has to be selected and agreed upon. To provide third party functionality, the bank should be able to monitor and record the progress of the transaction as it proceeds. Another aspect of content exchange is the need to ensure that a transaction is fair, since the bank acts as a third party it can also assist in the fair exchange of content for payment. As bank peers are super-nodes in our system they can also maintain the transaction records for non-repudiation.

3.3.1.1 Common bank peer selection

A common bank peer has to be selected (*bank peer selection*) to assist in the content-payment exchange transaction; the role of this bank peer is to store transaction data. The selection of this bank service is not simple as both parties have to select and agree on the same bank from a collection of banks available and the bank in turn should be in a position to provide its services.

The simplest way to choose a bank peer would be for one party to choose a bank peer from among the bank peers it 'knows', however this could lead to collusion. Therefore, to avoid collusion a method is required which would make the selection more random. Knowledge of a peer may equate to knowledge of *neighbouring peers*. Every peer is aware of its immediate neighbours, termed the neighbouring peer. These are peers who can be reached by a single hop.

Based on the use of neighbouring peers the simplest form of selection would involve the two parties (buyer and seller) requesting the services of an advertised bank peer in their vicinities. A *random bank peer selection algorithm* has been developed (described in Chapter 4, §4.3.9 pg 1) to address this need.

3.3.1.2 Transaction management & transaction record creation

Bank peers assist in the content exchange transaction by acting as a trusted third party. They validate transactions and store transaction details for non-repudiation so they have to actively participate in the transaction. Apart from acting as a non-repudiation tool, the transaction records also act as proof of the transaction lifecycle. Consequently, the transaction record is created from the point where the bank peer service is requested and accepted.

The lifetime of the transaction document corresponds to the lifetime of the transaction it self. However, the lifetime of the transaction record, once committed, is dependent on its *expiration date*. All countries do not subscribe to the same set of data preservation laws, however if we take the longest period of data preservation required as the base case, it would satisfy the requirements for every country.

In any transaction process it is necessary that the system caters for loss of connections due to a variety of reasons. The two-phase commit is a technique used to maintain integrity in distributed databases. It ensures transactions in distributed databases are atomic ("all or nothing"). This is done by handling the transaction in two phases. First the databases prepare the transaction, confirming that it is possible to process it, and acquires locks on the relevant record(s). Once all the required databases confirm that the transaction is viable, the system instructs them all to commit it, i.e. make it permanent. If it is not possible to process it, the system instructs the databases to rollback (undo) the transaction.

In the proposed system described in this thesis the two-phase commit technique is used to assist in transaction management, however the transaction itself is not distributed across various peers i.e. at the time of the transaction only one data store is involved so the commit does not have to be replicated across the entire bank peer network. The bank peer has direct access to its data store but the transaction record is only committed once the transaction is complete, if the transaction fails midway for any reason the whole record is rolled back and the transaction is not said to be complete. In the case where the transaction has been completed past a critical stage the record is committed for future reference, this assists in non-repudiation.

3.3.1.3 Fair exchange

A transaction is said to be fair if both participants in the transaction benefit from the transaction. In the case of fair document exchange it means that if two parties P_a and P_b have to exchange two documents D_a and D_b respectively then there may not arise a situation where P_a may be in possession of D_b without P_b being in possession of D_a, if so then P_b should be able to retrieve D_a and vice versa.

The issue of fair exchange is raised to highlight the problem that when two peers exchange content for payment, it is imperative that no single party attains an unfair advantage over the other in the transaction and may be in a position to cheat the other either by receiving the content before sending the payment or vice versa.

Zhang et al's [N. Zhang 2003] technique for anonymous and fair document exchange concentrates on protecting the identities of the participants in the transaction while facilitating fair exchange, however in this thesis anonymity is not required. Therefore, we use the method of incomplete disclosure of secrets to ensure fair exchange in our research. Our fair exchange protocol will be described in detail in Chapter 4 as part of the content exchange transaction protocol.

3.3.2 Non-repudiation & data persistence

The other main role of banks is providing a non-repudiation service to the buyer and sellers. Non-repudiation of a transaction is possible through the storage and archiving of transaction data, which should also reflect the completion or cancellation of a transaction as well as what content was purchased, for how much and by whom. Anonymity of a transaction is not the focus of this research. In order to assist with transaction validation and maintain transaction records the banks have to be able to replicate the records, synchronise and maintain their transaction data stores.

3.3.2.1 Replication & synchronisation

In the proposed system, there is need for persistence of transaction data. This data is read-only once it is created and therefore needs to have instant tampering evidence. It should be accessible for the purposes of transaction validation or non-repudiation.

The technique used to ensure data persistence in distributed transaction processing systems is replication [Andrew S. Tanenbaum 2002], where data is regularly duplicated redundantly across

the system to ensure data consistency, longevity and improve system performance. Whereas in the distributed transaction processing systems it is done mainly to bring data closer to its users [LinkPro Technologies 2003], in P2P systems replication is a solution to increase data availability [Kavitha Ranganathan 2002], persistence and archiving [Brian F. Cooper 2002] as well.

We replicate the transaction data across the bank peer network to ensure the persistence of transaction data. As with the payment distribution (§3.2.5) approach, there are two processes for data replication. Data *synchronisation* is the process where bank peers request transaction data from neighbouring peers when they join the bank peer network, this is considered to be a pull operation. Data *replication* is the method where bank peers regularly send transaction data to neighbouring bank peers at regular intervals (determined by network activity); this is considered to be a push operation.

Data synchronisation ensures that bank peers work with the latest data available in the bank peer network; this reduces the need for constant network queries when the bank peer is participating in a transaction. The network queries are more localised and require fewer hops to reach the find the relevant record if it exists. Similarly, data replication also speeds up the lookup process by moving data closer to the bank peers while at the same time ensuring the persistence of transaction data; it also removes active data loss due to the disconnection (random or informed) of peers. Only bank peers participate in the replication and synchronisation procedures this is primarily why bank peers need to be rich in resources (explained in detail in §3.4).

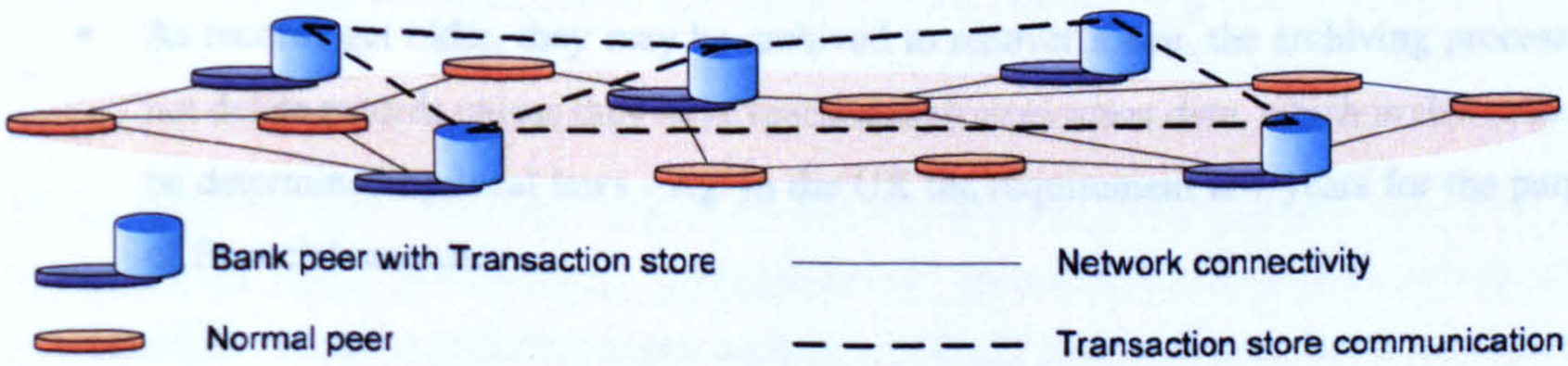


Figure 3.6: Replicated transaction database

As the amount of transaction data increases there is a need to speed up the lookup process, since the complete transaction record is only required if certain details match, different data indexing techniques can be applied to make lookup efficient. Indexing on the basis of transactionID alone is not sufficient in this case; the transaction should be indexed on the basis of three fields – the buyer ID, Seller ID and the content ID of the purchased content. These fields are unique in themselves since buyer and seller ids are UPIs and the content ID is generated from the content's hash value (described in §4.3)

This solution is viable mainly due to the scalability of P2P networks: the greater the number of nodes in the network the more conceptual storage space is being contributed. Consequently, there is the potential for transaction data persistence, as newer nodes join the network they can share the burden of the transaction records. Every transaction record has an expiration date which corresponds to the date past which the data is no longer required for preservation. At this point the transaction record can be dropped from the system to recover storage; similarly archiving [Brian F. Cooper 2002] the records after some time has elapsed (usually after the validity of the payment has expired) will also recover storage in the system while maintaining the persistence of data. Many techniques are proposed for replication of data in P2P networks [Brian F. Cooper 2002, Qin Lv 2002, Kavitha Ranganathan 2002, Hakim Weatherspoon 2002], Cooper and Garcia-Molinay's [Brian F. Cooper 2002] technique of data trading in blocks of space along with their optimisation policies to achieve maximum reliability, would be most suited to our scenario. Other research [Kavitha Ranganathan 2002, Hakim Weatherspoon 2002] is also being carried out investigating various replication techniques in P2P networks. Hence, to enable maximum flexibility our system has interfaces to the replication algorithms which can be implemented as per the situation and choices made by developers. As the object data formats have been standardised (Appendix C), consistency would be maintained in the system.

The pre-requisites to maintain the integrity of transaction data are:

- A bank peer can only create a new transaction record in its own data store.
- Once committed a transaction record may not be altered, even by its creator.
- In the replication process only copies of transaction records may be added to another data store.
- As records get older, they may be archived to recover space, the archiving process can not delete records unless they have reached their *expiration date*, which in this case may be determined by local laws - e.g. in the UK the requirement is 7 years for the purpose of financial records.

3.3.2.2 Data storage

The primary purpose of the overlay network of bank peers is to ensure that transaction data is persistent in the network for the purposes of non-repudiation as well as to conform to Data Protection Laws. Hence, the overlay network of bank peers would act as a distributed database for the entire system. However, we recognise that a range of devices could participate in the system. These devices can have different data storage formats such as flat file systems, relational databases, object relational databases, network and hierarchical databases.

In our approach, each peer has prior knowledge of how to communicate with its local data store. However, to ensure that all devices understand each other and send relevant transaction details information for storage, the exchanged data objects and their formats are dictated by our system. These data objects will be explained in detail in the Data Model in Chapter 4. By standardising the data objects being exchanged we give the system flexibility in the choice of data storage to use. The pre-requisite in this system is that all peers implement a prescribed interface to their respective data stores. This interface provides generic data manipulation and data definition functions such as retrieve, save, update and commit (Appendix C).

3.4 Services and the redundant service utilisation model

In the previous sections we discussed the various requirements of our system based on the main aims and objectives of this research. However to truly exploit the full potential of the P2P environment it is necessary to enable full utilisation of the massive resource redundancy which is a major feature of this environment. In this section we shall list the requirements to accomplish this and describe our model to maximise the use of these redundant resources.

3.4.1 Requirements to maintain a heterogeneous system

One of the major requirements of any ubiquitous system in the present climate of pervasive connectivity is that all the devices we own may be interconnected and provide us with the ability to perform our daily tasks with ease. These devices include our PCs, PDAs, mobile phones and portable media devices. All these devices are potentially capable of interacting with networks and acting as peers in a P2P network.

As peers in the P2P environment may have varying resources, we introduce the concept of *thin peers*, which are peers that are resource limited (for instance mobile phones and PDAs have limited battery power and memory). These peers may not necessarily be able to provide all the functionality required to perform content exchange in the P2P environment. So as not to exclude them from the network, they should be able to locate locally deficient functionality within the P2P network and utilise it to complete the required task. This is illustrated in Figure 3.7, where a PDA can locate content on a mobile phone and use the Bank Peer functionality of a PC to complete the transaction.

Conversely, a *fat peer* is a peer who is resource rich. We choose this terminology as it closely mirrors the concepts of thin and fat clients [Jason Nieh 2000, Thad Starner 2002], where

maximum processing and management responsibilities are moved away from a client to accommodate its lack of sufficient resources.

The use of web services [Steve Vinoski 2004] to enable interaction between heterogeneous systems has become a common phenomenon in recent times. Web services work on the premise that desired functionality, in the form of a unit of work, can be incorporated into a service which can be consumed to arrive at a desired state. Similarly, in the P2P domain more and more functionality is being encapsulated within Peer services to provide flexibility in the system. These services are hosted on peers and can be discovered and used to provide a peer with functionality it may lack thus utilising the massive redundancy inherent in P2P systems for the benefit of the entire P2P community.

In this thesis, *redundant service utilisation* (described later in Chapter 4) is our technique used to match services to application functionality. By using this technique it is possible to maintain the heterogeneity of our system since it allows thin clients to participate in the system and fully utilize the potential of the massive service redundancy. These services have predefined interfaces which allow them to be combined into different permutations combinations to perform a set functionality. Hence, as long as the devices which wish to participate in our system have local access to their core system services and can locate other redundant services in the network they can participate in the network and exchange content for payment.

3.4.2 Redundant service utilisation scenario

In a real world scenario, if a user wished to pay for some content and he only 'understood' Euro he could use a service which allowed him to convert Dollars into Pound Sterling and pay another user in the relevant currency. Thus allowing him to participate in a content exchange transaction. In a digital environment the same activity should be possible as well. P2P environments exhibit massive service redundancy, hence it makes sense to utilise these services to assist thin peers in fully participating within any system.

In our scenario (Figure 3.7), services [Groove Networks 2001] may be aggregated and utilised to allow thin peers to participate in the network. Here a thin peer such as a mobile phone (Content & Service Buyer), who is poor in resources, wishes to purchase some content from a Content Seller. It has access to Lookup, Delivery and Payment functionality. Once it has found the desired content (using the Lookup functionality) it can pay for the content, however in this case the payment value is in different currency to the one it understands. Consequently, it would require access to a Conversion facility or choose to abort the purchase transaction. Using the concept of service redundancy it could locate a Conversion facility within the network at a

Service Seller, who would process the currency conversion and return the appropriate response (Transaction 2), thus allowing our Buyer to continue with the purchase of his desired content (Transaction1). These purchases are supported by banks which act as a point for non-repudiation of transactions.

Conceptually speaking this method of service utilisation should allow peers to locate and utilise any service within a network regardless of its own resource capabilities. Hence if we apply this theory to another scenario where a peer who can search and pay for content may buy content from a thin peer who can sell content but not verify the payment locally. Then the thin peer would then locate the verification service within the network and use it to verify the payment prior to delivering the content. However, this raised some interesting issues for remote service utilisation, for instance, how do you trust a proxy to verify your payment and not defraud you. Consequently, we acknowledge the potential for a security risk in the service utilisation scenario.

A Service Oriented Architecture (SOA) is an architectural style whose goal is to achieve loose coupling among interacting software agents. A service is a unit of work done by a service provider to achieve desired end results for a service consumer (referred to as Functionality in the above scenario). Both provider and consumer are roles played by software agents on behalf of their owners. These services form part of a framework which will be defined in the following section

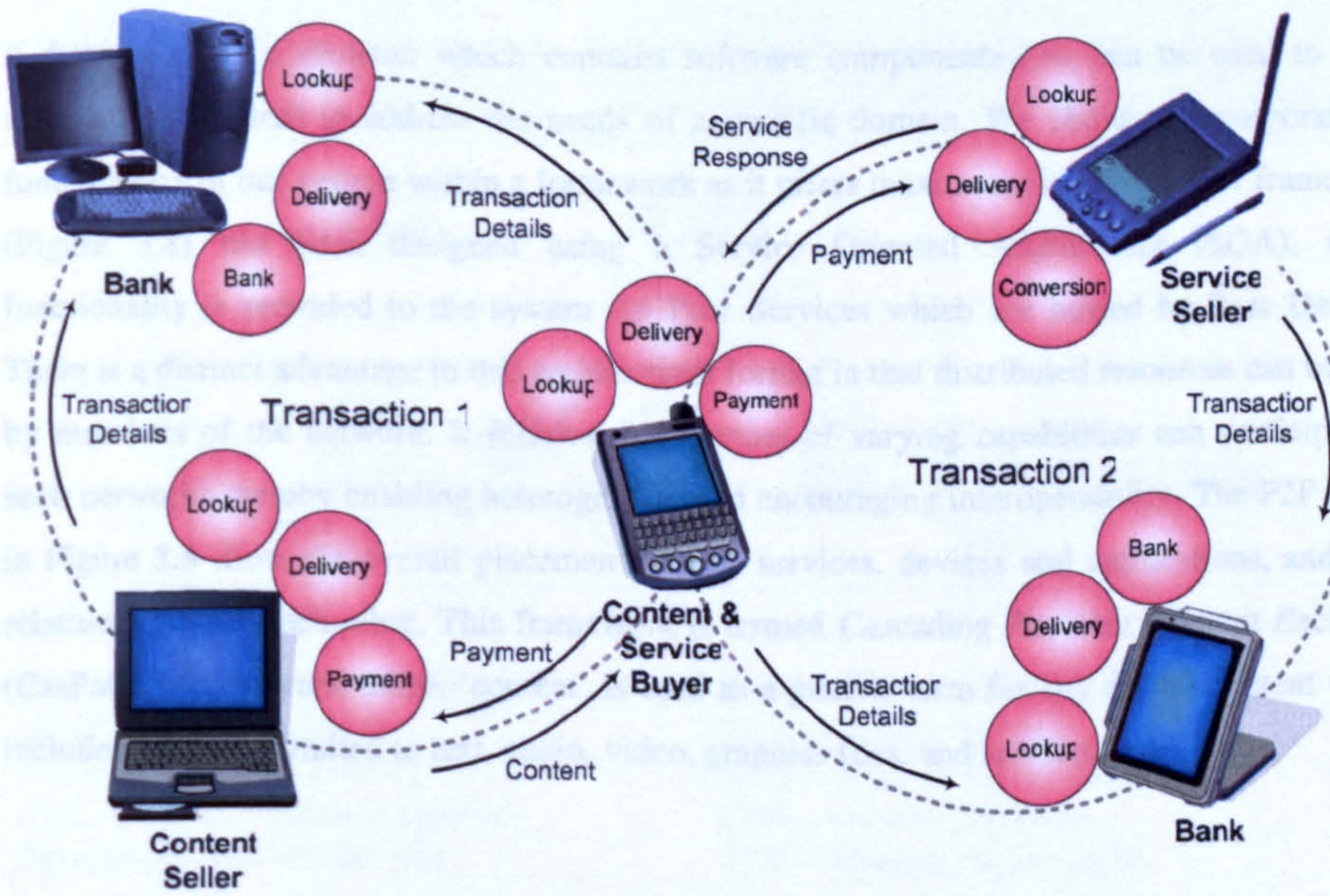


Figure 3.7: A service utilisation scenario

The scope of service definition is very vast, Jones [Steve Jones 2005] states that service specifications can fall under many categories such as performance, capacity, ownership, security, tolerance etc., we only define our services in terms of a service contract which incorporates the pre and post conditions and the invariants, the other categories are outside the scope of this research and may be considered in future work. One of the objectives of our research is to allow thin peers to participate in a P2P content exchange environment because even thin peers such as mobile phones and PDAs are capable of participating in P2P networks and storing digital content. There is a body of work being carried out in our research laboratory which specifically deals with the issues relating to service composition, this includes understanding the various categories of service specifications. We are using service composition as a technique to include thin clients.

The use of redundant services and service utilisation allows maximum flexibility in a dynamic P2P environment because it allows devices to find services which are most suitable to perform specific functionality.

3.5 Our new Framework

This section describes our proposal for a new framework to realise the concepts of the CPM, Bank Peer Overlay Network and service redundancy and utilisation to support equitable digital content exchange in the P2P environment.

A framework is a skeleton which contains software components that can be used to build applications tailored to address the needs of a specific domain. We chose to incorporate the functionality of our system within a framework as it offers maximum flexibility. Our framework (Figure 3.8) has been designed using a Service Oriented Architecture (SOA), where functionality is provided to the system via Peer Services which are hosted by Peer Devices. There is a distinct advantage in this architectural format in that distributed resources can be used by members of the network. It follows that devices of varying capabilities can participate in such networks thereby enabling heterogeneity and encouraging interoperability. The P2P layers in Figure 3.8 show the overall placement of P2P services, devices and applications, and their relationship with each other. This framework is termed *Cas*cading *Pa*yment *C*ontent *E*xchange (CasPaCE) Framework. Here, 'content' is used as a generic term for any digital content which includes but is not limited to text, audio, video, graphics files, and services.

3.5.1 The CasPaCE framework

The CasPaCE framework (Figure 3.8) has been developed to compensate producers and distributors of digital content in a P2P environment while ensuring atomicity and non-repudiation of transactions. It also ensures that the benefits of the P2P environment (§1.3 pg 4) are maintained. Our framework ensures the accurate distribution and provision of all data necessary for final redemption of payments. However, how the peer physically converts the electronic payment information into physical currency is currently outside the scope of our research.

This framework addresses the following requirements:

- Digital content producers are compensated for their work, protecting their copyright.
- Content distributors in the P2P network are compensated for ensuring content persistence and acting as distributors.
- There is fair exchange of digital content for payment ensuring non-repudiation and atomicity [J. D. Tygar 1996] of transactions.
- Heterogeneous peers can participate within the network.

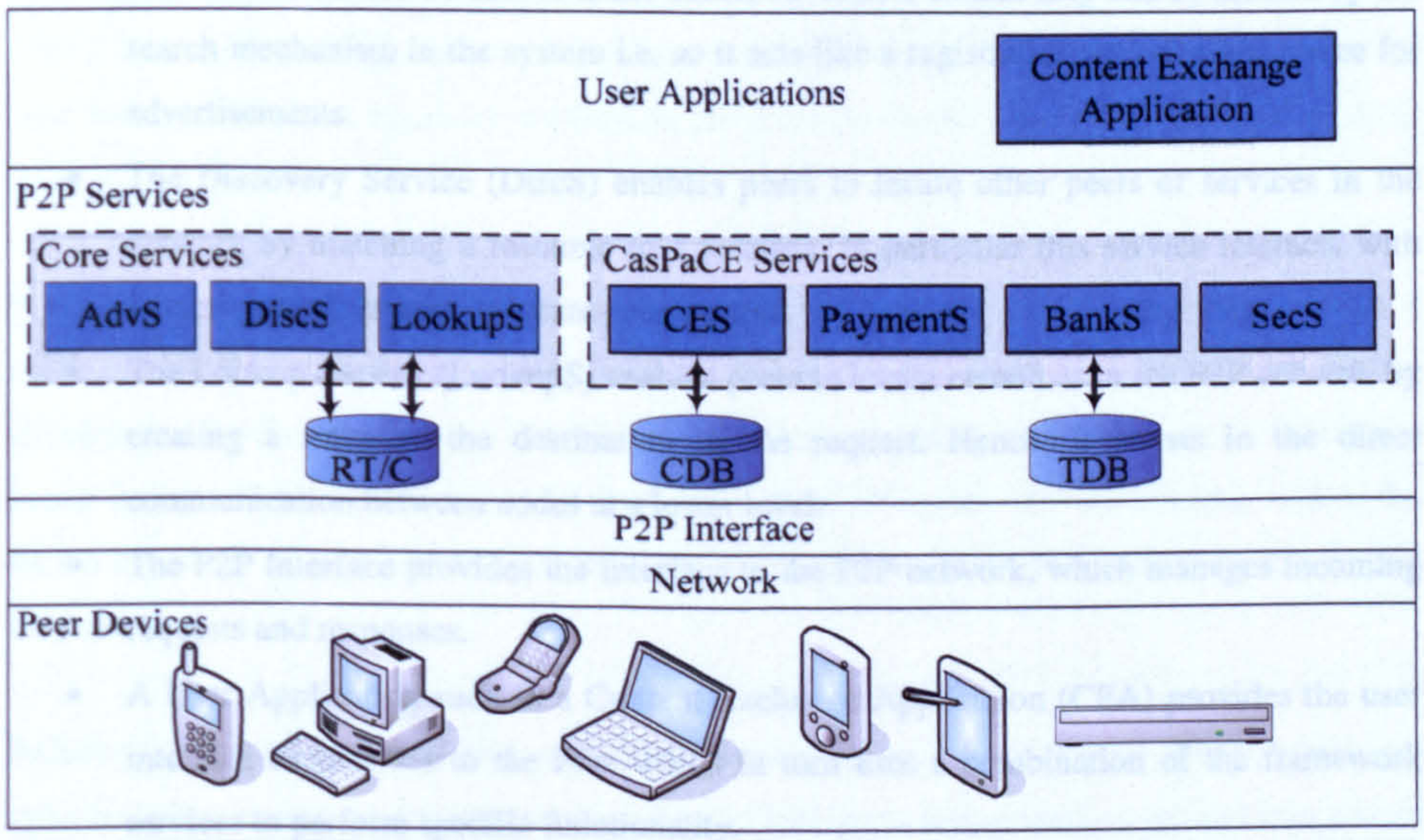


Figure 3.8: Cascading Payments Content Exchange Framework

The following section will give an overview of the various components of the CasPaCE framework.

3.5.2 The CasPaCE framework components

The requirements listed in the previous section (§3.5.1) are fulfilled by the various components of the CasPaCE framework. These components are:

- The Content Exchange Service (CES) manages the content storage and exchange. A Content Database (CDB) is used to store the content.
- The Payment Service (PaymentS) verifies payments (PayV) and handles payment separation (PayS) and distribution (PayD).
- The Security Service (SecS) handles encryption and PKI [Network Associates Inc. 2000] key management.
- The Bank Service (BankS) ensures fair exchange of content and that transactions are atomic and non-repudiated. It uses a Transaction Database (TDB) to store transaction data.
- The Advertisement Service (AdvS) is used to advertise available content and services. This service should be able to make content available so that they can be located by the search mechanism in the system i.e. so it acts like a registration or indexing service for advertisements.
- The Discovery Service (DiscS) enables peers to locate other peers or services in the network by matching a resource to a location. In particular this service interacts with advertisement services to locate peer entities.
- The Lookup Service (LookupS) enables peers to locate resources in the P2P network by creating a route to the destination of the request. Hence, it assists in the direct communication between nodes at a lower level.
- The P2P Interface provides the interface to the P2P network, which manages incoming requests and responses.
- A User Application such as a Content Exchange Application (CEA) provides the user interface capabilities to the Peer which in turn uses a combination of the framework services to perform specific functionality.

The required P2P services can be divided into two categories, overall Core Services required by all peers and the specific CasPaCE Services required as per the situation. Core services (essential to ensure a peer's participation in the P2P network) are provided by the Advertisement (AdvS), Discovery (DiscS) and Lookup (LookupS) Services. These services

work together to enable communication between peers, perform message relay and routing and node discovery.

The DiscS matches a query to a location and then uses the LookupS to find the location of the desired resource in order to generate a route to the peer where the resource is hosted. Based on the type of underlying P2P routing protocol being used (central server index, broadcast query or DHT) these activities could be superimposed or atomically separated. In the central server index and DHT, both discovery of the resource and lookup of the path to the host are combined as a single activity. However in the case of the broadcast discovery technique, finding the resource is performed first by broadcasting the query. This query is processed at every node and a local lookup is performed. If the node has a record of a similar request in its cache it returns the lookup path for the host otherwise the query is propagated forward. That is why these two services are placed in our framework as two separate entities but both accessing routing and cache (RT/C) tables.

The AdvS is used as the medium for informing the community of the presence of a resource. Since a pure P2P environment is decentralised in every respect, there is no central index where resources may be registered for discovery as is the case with Web Services[Steve Vinoski 2004]. This service allows the peer to generate an advertisement for its usable resources; in our case the digital content. These advertisements are published locally and may be located by other peers via the queries.

The CasPaCE Services we incorporate into the P2P services domain to enable paid content exchange are the Content Exchange Service (CES), Payment Service (PaymentS), Bank Service (BankS) and the Security Service (SecS). These services provide the ability to search for content based on keywords, content value and estimated time of arrival; pay for content; ensure the secure exchange of payment and content; and maintain transaction details for non-repudiation purposes.

Although some of these services are 'Peer Services'[Brendon J. Wilson 2002] which are specific to a peer and hence dependent on its availability in the network, most services should be 'Peer Group services' which are redundantly available to all the peers. In this thesis, each participating peer has the ability to Search for content, hence Lookup and Discovery are inherently present on all the peers as Peer Services. All other services may be considered as Peer Group Services as defined in [Brendon J. Wilson 2002], hence forth we shall refer to them simply as Peer Services. These Peer Services are discoverable on-demand and may be integrated to work in tandem [Groove Networks 2001]. The Content Exchange Service (CES)

provides the coordination between the services to enable Cascading Payment Content Exchange (CasPaCE) allowing the user to search for content and other services.

Different stores form a part of this framework to address the need for various types of data storage: content (CDB), transaction data (TDB), routing data and caching information (RT/C). The various components of the CasPaCE Framework and their functionality are described in detail in the following Chapter.

3.6 Summary

This Chapter discussed the high-level architectural requirements for enabling paid content exchange in a dynamic P2P environment thus protecting a content owner's copyright and acting as a participation incentive to prevent free-riding in P2P communities. It discussed the objectives of this work in terms of the various requirements of the overall system. These requirements were categorised into P2P system, copyright, transaction & payment and trust related requirements with particular reference to the application of e-commerce to the P2P domain.

The concepts and models required to create an equitable P2P content exchange system were described. These included the cascading payments model, Overlay network of bank peers, service redundancy and utilisation model and the use of a SOA to create a framework which promotes flexibility and interoperability. The Chapter also discussed the relevant terminology, to give the reader a better understanding of the main participants in this system thus setting the scene for the next Chapter. The CasPaCE framework was introduced which gives peers in a P2P system the ability to exchange digital content for payment in a secure fashion. To allow any Internet enabled digital device to participate in this system, our framework incorporates the ability of peers to locate services and integrate these services to perform the overall functionality of the system. It also caters for services which are present locally on the devices as well as remotely. We also discussed how and where the various areas of research discussed in Chapter 2 impact on this research.

In the previous sections we briefly described the roles of the various services in the CasPaCE framework. The following Chapter shall describe these services and their functionality in detail as part of the detailed design. It will also discuss the design decisions made and define the resultant protocols and policies which map to the concepts described earlier.

Chapter 4

4 CASPACE: A FRAMEWORK FOR CASCADING PAYMENTS AND CONTENT DELIVERY

The previous Chapter highlighted the requirements for the overall research and provided the reader with a high level specification for a framework that supports cascading payments. The main contributions to knowledge (the Cascading Payments Model, the Overlay Network of Bank Peers, Service redundancy & utilisation and the CasPaCE framework) were introduced along with the relevant terminology. This Chapter shall describe these components in detail and how they behave with respect to each other. The various design issues that have been addressed and the impact these have had on the overall design process will be discussed.

The layout of this Chapter is as follows. It will commence with an overview of design considerations and system modelling. This framework addresses certain needs, which are addressed by the CPM and the ONBP. To allow efficient functioning of the CPM and the ONBP within the P2P environment in such a way so as not to detract from the benefits of this technology we employ Service utilisation which relies on the replication or presence of services across the network for efficient functioning of the system. The CasPaCE framework incorporates these techniques into a service oriented architecture which allows for flexible deployment of the system while maintaining the heterogeneity of the P2P network. These services and their protocols are the focus of this Chapter.

4.1 System modelling

This section shall discuss the modelling methodology used to capture the requirements and system behaviour for our system. Important issues to consider when designing a framework are:

- That the framework is designed with maximum flexibility and extensibility in mind.
- Support for loose coupling to ensure any changes made to the framework do not have an adverse impact on other applications that are based on the framework.

All P2P environments are ad hoc in nature. The topology of the network is very dynamic. These networks are scalable and can utilise the dormant resources of other peers to accomplish complex tasks. To utilise this environment fully without detracting from the strengths of this

system we chose the use of a Service Oriented Architecture (SOA). In the strict P2P sense, a SOA would be slightly different in that the location and discovery procedures will not be centralised however the concepts of traditional SOAs would still apply. So services would have to be registered and discovered to be utilised. In the P2P sense, service registration would involve advertising the service to other peers in the system to keep the network truly decentralised as opposed to the traditional method of registering services at a central registry[Michael Champion 2002].

We have used the Unified Modelling Language (UML) to illustrate the detailed design for our system. Our design specification describes the system requirements in the form of a Use Case Model (Appendix A) which is mainly used to capture the functional requirements of the system in terms of what it has to do. The Behavioural Model (sequence diagrams which capture the system behaviour with passage of time and collaboration diagrams which capture component collaborations) captures the behaviour of the components and illustrates how they communicate with each other. The Data Model (class diagrams) describes the data which is used to support the entities in the system.

The system requirements were captured with the aid of the UML's use case model. This allowed us to implicitly define the roles of the various participants in the system within various use cases. These use cases can then be refined to define the precise functionality of each object within the system resulting in the compilation of the object class diagrams and subsequently the system data model (Appendix A).

In the functional requirement gathering process the very base functionality of a generic peer was identified as being able to:

- Join a P2P network
- Make a query
- Accept query
- Respond to a query and
- Leave the P2P network

This functionality of the peer is included within the peer's interface to the P2P network. In our framework these operations are encapsulated within the P2P Interface. The P2P Interface is a prerequisite to the functioning of this framework and can be considered a core requirement.

4.1.1 System actors

To identify the key roles of the objects in the system, such as a Buyer and a Seller, we used the UML. Since both these functionalities could be included within the same peer, the Normal peer of our concept can perform both roles. However, during the use case modelling procedure it became evident that the Bank peer role was completely different to the Normal peers except for the connection to the network and even at that stage bank peers need to locate and communicate with other bank peers in the network; hence the bank peers are a different class of objects. Similarly, it became evident that the owner's role in the system is more specialised than that of a normal seller because the owner has to be able to tag the content with his copyright information. Consequently, we were able to arrive at a hierarchy of roles as illustrated in Figure 4.1.

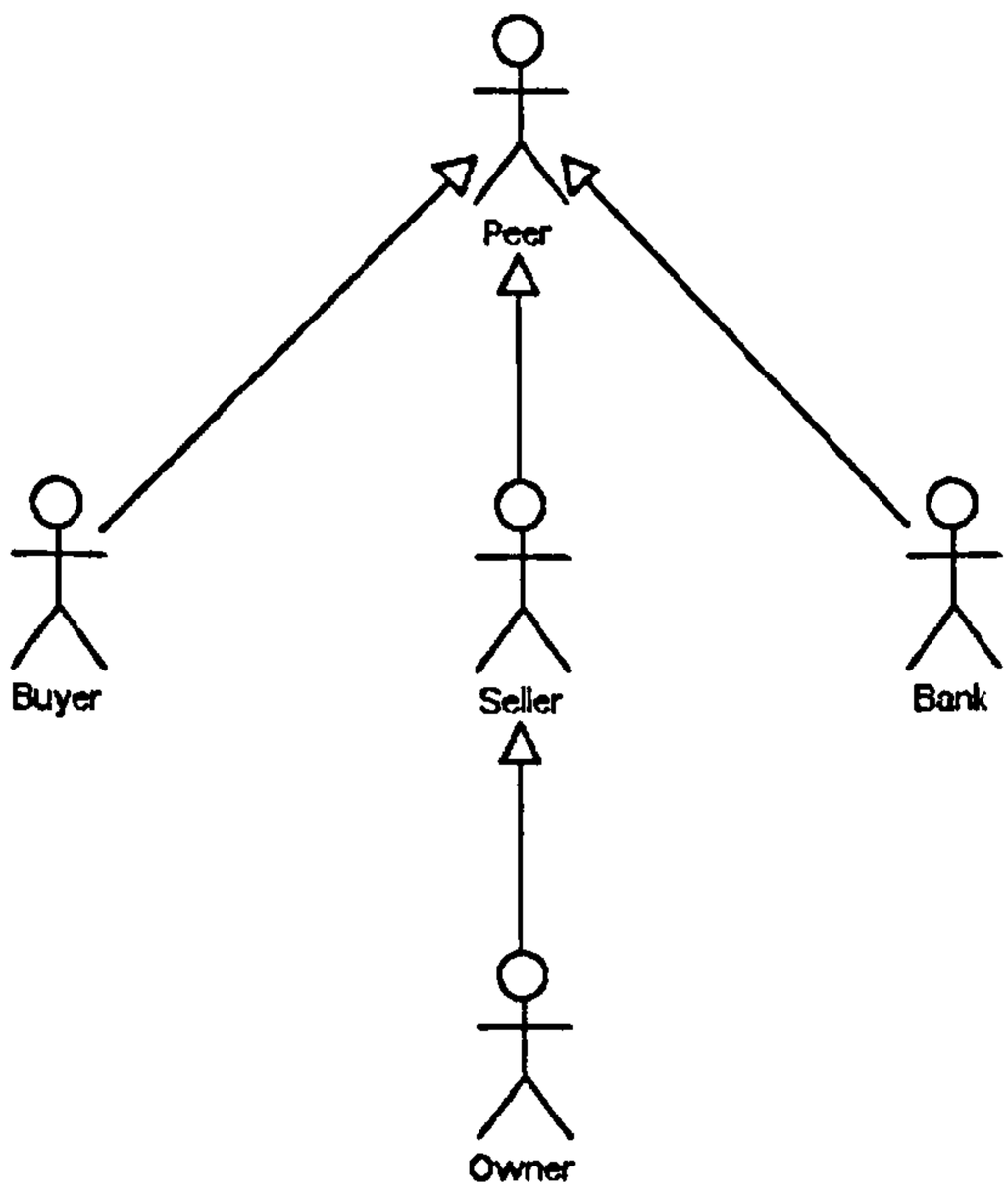


Figure 4.1: Use Case – P2P Payment System Actors

Each actor within the CasPaCE Framework must have some common attributes/properties as well as unique operations/functionality that it can perform. For instance every actor should have a unique identity, location address and a list of services that it can provide. Similarly, common functionality such as connecting and disconnecting from a network, being able to Search for various resources in the network are also common operations every actor must perform. Hence, we have represented the various roles as a hierarchy, where all actors inherit some roles from the *Peer* (Figure 4.2).

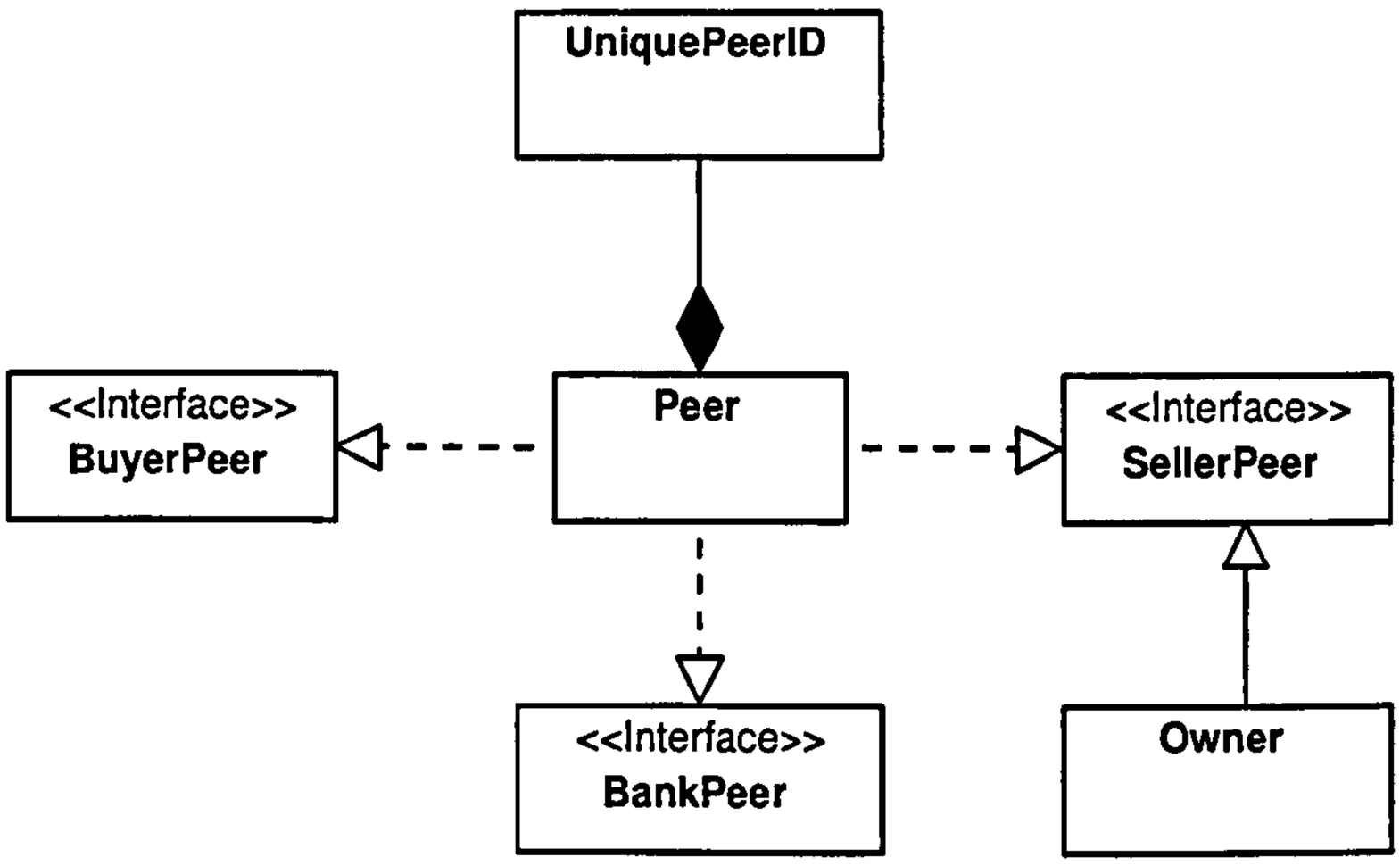


Figure 4.2: Class diagram – Peer roles in the cascpace framework

These roles of the Actors within the CasPaCE Framework are:

- *Peer* – the Peer actor is synonymous with the Normal Peer described in the requirements. It represents any device that is capable of participating in a P2P network. Hence, it has basic P2P functionality.

- *Buyer* – the Buyer actor is a specialised role of the Peer, therefore apart from basic P2P functionality it can Buy Content.

- *Seller* – the Seller actor is a specialised role of the Peer that can Sell Content as well as be a part of the P2P network.

- *Owner* – the Owner is a specialised role of the Seller. As Seller is a specialised role of the Peer, the Owner inherits the functionality of both the Peer and the Seller and may Sell Content. We separated the Owner as an explicit role from the Seller because the Owner is the only actor which can Create and Insert Content into the P2P Content Sharing Network. However, both the Seller and the Owner may Sell Content since an Owner is a Seller.

- *Bank* – the Bank Peer has the ability to perform various functions within the CasPaCE Framework apart from participating as a Normal Peer. The primary functionality is to act as a Third Party (TP), which involves participating in transactions between peers as well as maintaining the transaction data generated.

All the actors described above belong to our P2P payment system, which we refer to as the CasPaCE Framework. We would like to mention at this point that the CasPaCE Framework is designed to work in the P2P domain; hence this P2P Payment System can be a sub-system of any P2P network. The various roles in a standard P2P network are augmented by our system roles.

One of our framework's aims is to enable any peer in a P2P network to buy and sell content and get compensated. Therefore, making the CasPaCE framework adaptable to any P2P environment is important. Hence, the Normal Peer reflects the role of standard peers in P2P networks. These roles are augmented by other actors in our framework.

Some assumptions we make are that all Peers must have a P2P interface. The peers that are not resource rich may be referred to as thin peers (Mobile phones, PDA's). These peers are limited in their functionality due to a lack of resources such as excessive processing power, storage, battery life and bandwidth. Thin clients cannot act as Bank Peers.

The actors may undergo a transition (Figure 4.3) from one role to another dependent on the current system state as well as the User's input. For instance, when the User wishes to Search for Content, the Peer role is sufficient. However, if a User finds some Content he is interested in buying then as soon as he makes a *Purchase Request*, the Buyer role is activated. To be able to Buy Content the User must have access to the necessary functionality i.e. Service.

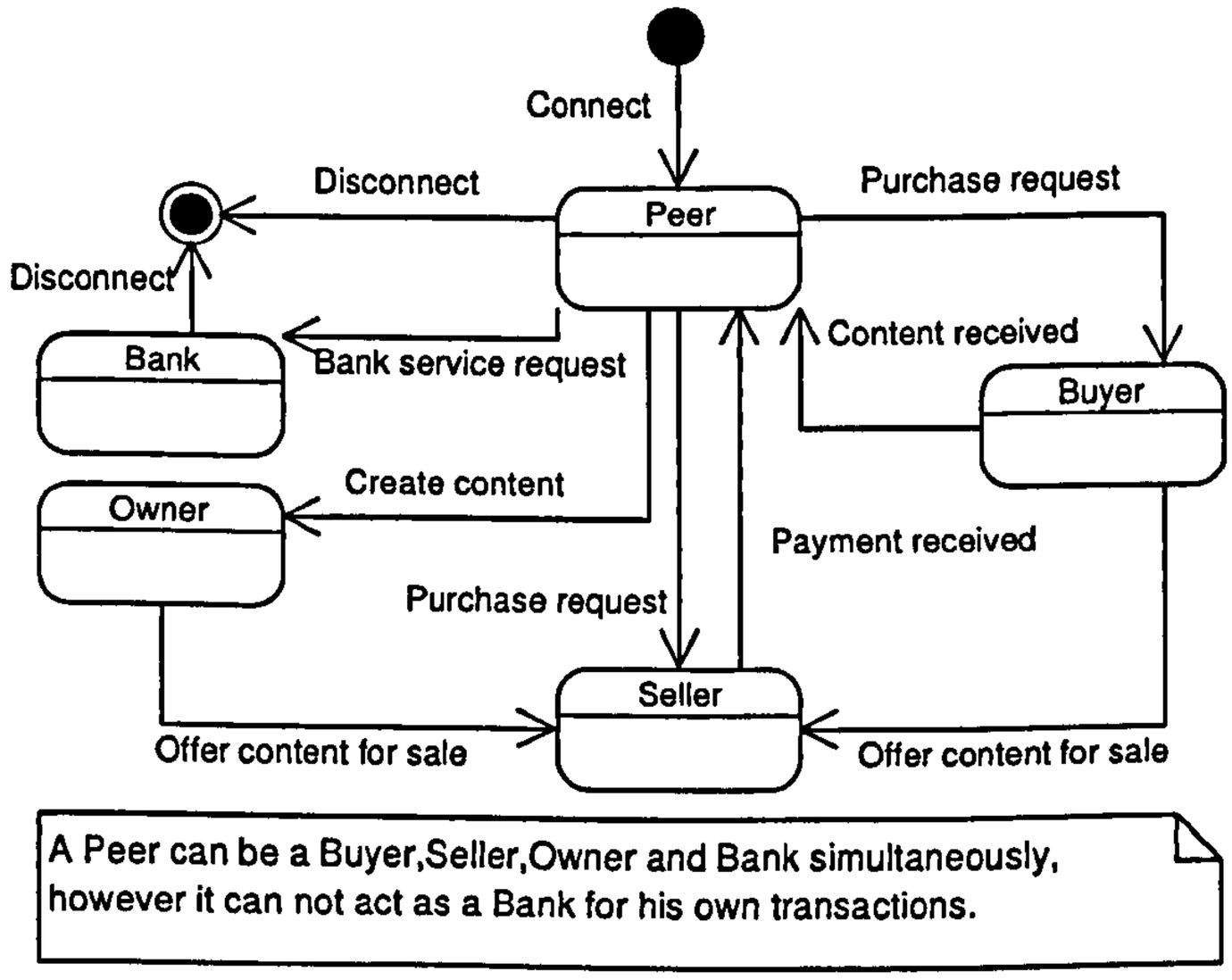


Figure 4.3: Peer roles state transition diagram

In the requirements gathering phase it was determined that the main responsibilities of a Bank Peer are:

- Joining the bank peer network,
- Accept Bank Service request,
- Participating in a content exchange,
- Manage a transaction,
- Manage transaction records.

Since all these responsibilities are exclusive to a bank, all this functionality can be encapsulated under the same set of modules which belong to the Bank Service. This service includes the functionality to create the overlay network of bank peers.

4.1.2 Our P2P payment system

Based on our requirements specified in Chapter 3, the major functionality required in our system can be broadly separated into the use cases shown in Figure 4.4. These are:

- Connecting to the CasPaCENet (content exchange network)
- Searching
- Selling content
- Buying content and
- Disconnecting from the network

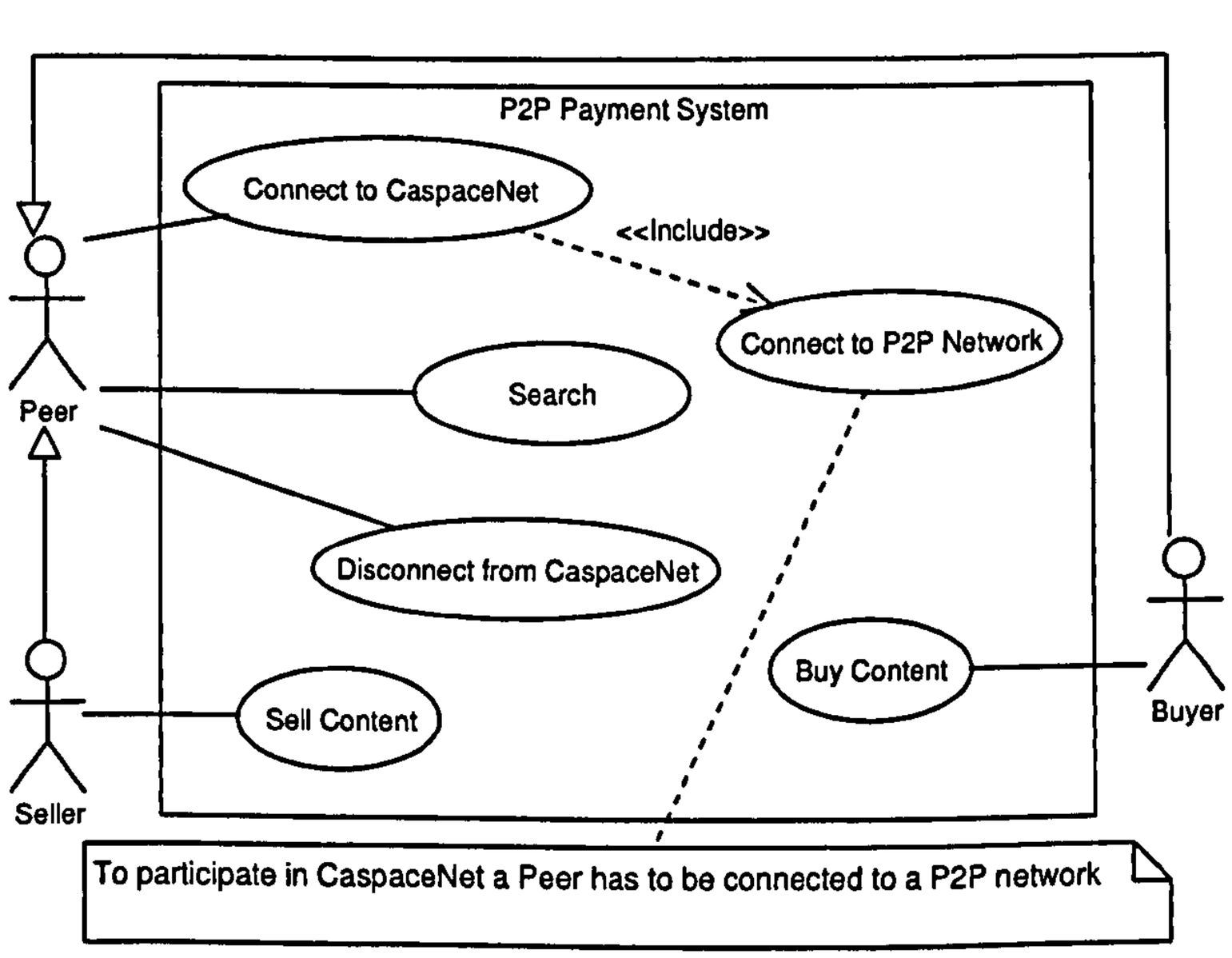


Figure 4.4: P2P payment system

'Search' is assigned to the Peer because, it will be specialised to allow searching for services as well as content and transaction records (Figure 4.5). Object oriented principles ensure that all actors in the system inherit this Use Case.

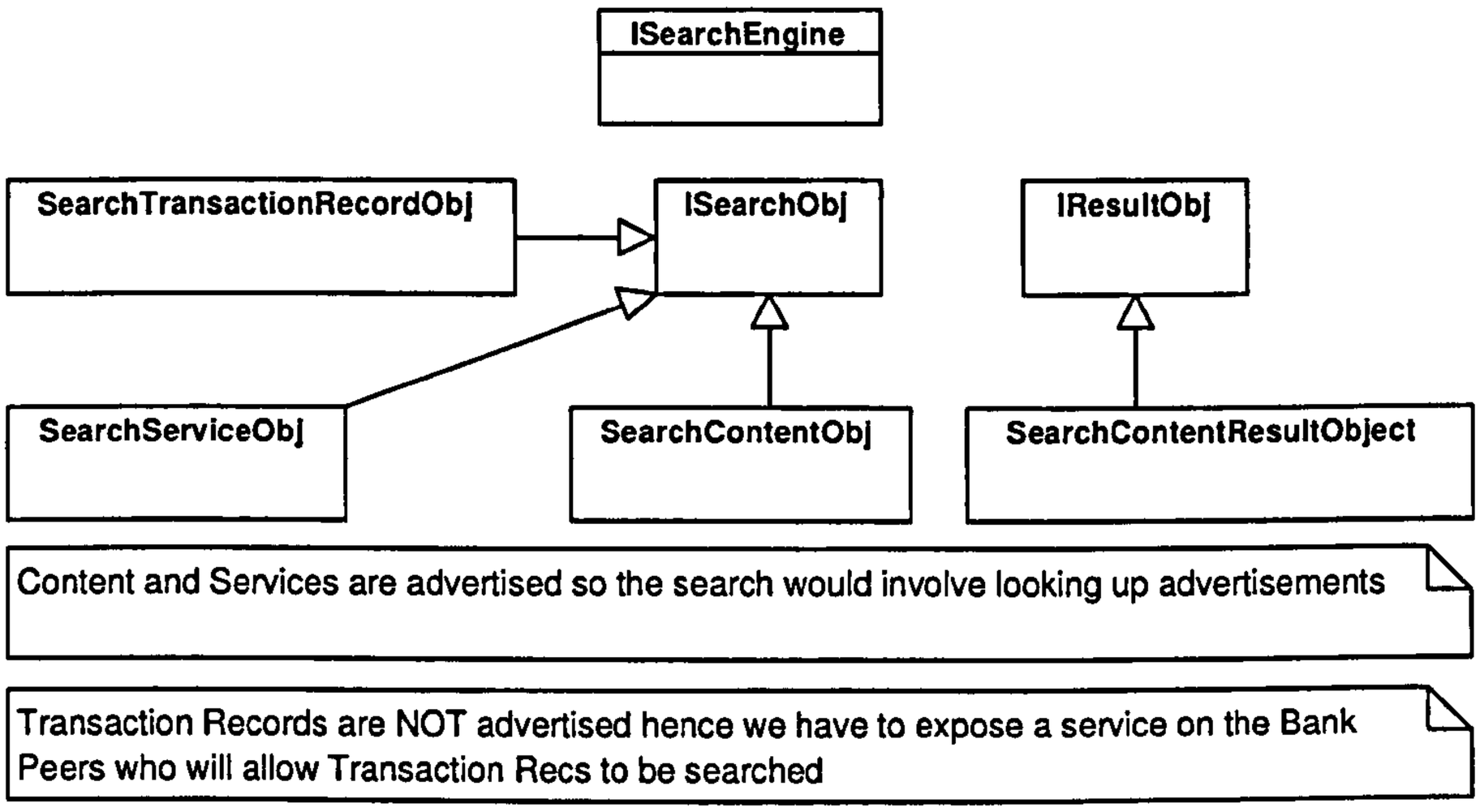


Figure 4.5: Class diagram – Search

In the following sub-sections we shall describe how the activities described above translate into objects, components and services.

4.2 CasPaCE services

The CasPaCE Services required to enable paid content exchange are the Payment Service (PaymentS), Bank Service (BankS), Security Service (SecS), the Content Exchange Service (CES), Lookup Service, Discovery Service and Advertisement Service. The description of these services is the focus of this section. All the services in the CasPaCE framework are subclasses of type Service which has a Service Descriptor. This service descriptor is used to locate and utilise the relevant services.

4.2.1 Content Exchange Service (CES)

The Content Exchange Service (CES) contains the interfaces for content management in the system. The CES allows the peer to manage content which it can locate and exchange for payment. It acts as a coordinator between the various services to perform content exchange. The

CES interfaces with the *Search Engine* and the *Content Store* to provide this functionality. This service is accessible through a Login facility which also links the users' UPI to their content.

4.2.1.1 Search Engine

P2P content-sharing networks allow users to locate content based on different criteria. Napster, KaZaA and other Gnutella variants allow users to search for content on general content information based on string-matching, where filenames and other content description information is searched. These initiatives are primarily used to share content such as music or video files, which are located on the basis of their filenames which users are intuitively aware of. Initiatives like PAST[Antony Rowstron 2001c] and CFS[Frank Dabek 2001b] do not provide keyword search utilities to users, they require users to be aware of the exact content name (derived via hash functions or based on keys) to locate it, as their routing protocols are based on key-value pairs. There are other initiatives that allow the location of content based on meta-data information[Sam Joseph 2002], semantics [Mallik Mahalingam 2003] and based on the indexing of the textual content within documents[Francisco Matias Cuenca-Acuna 2002].

In the context of our research, the search facility should enable the user to search for digital content based on name, but as the main purpose of this network is to provide the user with competitive rates for content and more information so he may make a more informed choice, the user should be allowed to search based on the cost of content as well as speed of delivery. Hence, information such as ContentName, ContentLocation, ETA, ConnectionType, Cost/Value, TransFees, TransFeesPayer, is available to the user as well as forms criterion for the search. Hence, a search for 'Java Docs' for 'less than £1' where 'Seller' pays the 'transFees' is a query entered into the application, the Search Engine formats this query into the relevant Search Object format (Appendix C) and returns the result set. For the location of the file itself the Search Engine communicates with the Lookup and Discovery Services, which return the location of the file along with its *content descriptor*, as search Result Objects. To allow maximum flexibility in the system, any type of Search Engine could be plugged in at a later stage as long as it conforms to our search interface as defined by our data model (Appendix C), for instance to receive more meaningful results a Semantic Search Service [Paul Fergus 2003] could be used. This would override our search service and feed results to our search engine.

In order to determine the Estimated Time of Arrival (ETA) of some content, the information would have to be calculated on-the-fly; the other components of the *SearchResultObject* are retrieved from the *Content Descriptor*. There are different ways [Jiangchuan Liu 2003, T. S. Eugene Ng 2002] to measure the distance to or proximity of nodes on the Internet, but the

simplest method is to measure the number of hops between the request originator and request server. This information is incremented with every hop when the reply to a request is being returned. This information can be used to calculate the distance between nodes in terms of hops and the time delay between the request and the reply can help determine the total time it would take which can be obtained by pinging the request server.

Although this might not give a highly accurate indication of the time taken, especially at peak network usage times, it would give a fairly accurate indication of distance in terms of delay associated. This information would have to be more accurate when people expect to pay for content and the speed or time taken for download may form an inherent deciding factor to make the decision. For instance some people would rather pay a slightly higher price for content that can be acquired faster. For the purpose of this research it is sufficient to provide an ETA using this method. To supplement the accuracy of this heuristic, future development may use other services [Jiangchuan Liu 2003] or algorithms to determine this value.

4.2.2 Payment Service (PaymentS)

The *Payment Service* is responsible for three protocols which deal with payment separation, verification and distribution. It also understands the different Payment instrument objects that may be used within the system. In order to manipulate these objects and provide the desired functionality, the Payment Service uses a *Payment Distributor* and *Payment Separator.* The *Payment Service* manager determines what type of message is passed to the service and forwards it to the relevant component (Separator or Distributor). When a Payment is received by the peer, it gets passed to the *Payment Service* which uses a *Payment Verifier* to verify the *Payment*; each payment instrument available in the system would have its own verification algorithm which is available to the *Payment Verifier.* The *Payment Distributor* uses different algorithms to distribute and collect payments, these algorithms are also included within the Payment Service package (Appendix C).

The *Payment Service* communicates to the peers (*Buyer* and *Seller*) through the CES. The Payment Separation (PayS) and Payment Distribution (PayD) Protocols, allow payments to cascade from buyer to owner every time the content is propagated. These protocols are supported by a combination of unique peer identification policies and mobilized copyright. The Random Bank peer selection algorithm enables random selection of bank peers.

4.2.3 Bank Service (BankS)

The Bank Service shown in UML in Figure 4.6 provides a peer with the ability to act as a decentralised trusted third party by allowing it to autonomously participate in the Bank Peer network, maintain transaction records at the time of the transaction and replicate these records across the Bank Peer network to ensure persistence and availability of the records. The Bank Service is the public interface to the service components (Overlay and Transaction Manager illustrated in Figure 4.6), it determines which messages are meant for which component and distributes them accordingly to the private services within the package.

The Bank Service is directly involved in initialising the process when a Bank Peer is requested for content exchange (described in §4.3.7 pg 104). The Transaction Manager then handles the transaction processing requests and communicates with the TDB to check for double-spending and transaction records of previous transactions as required.

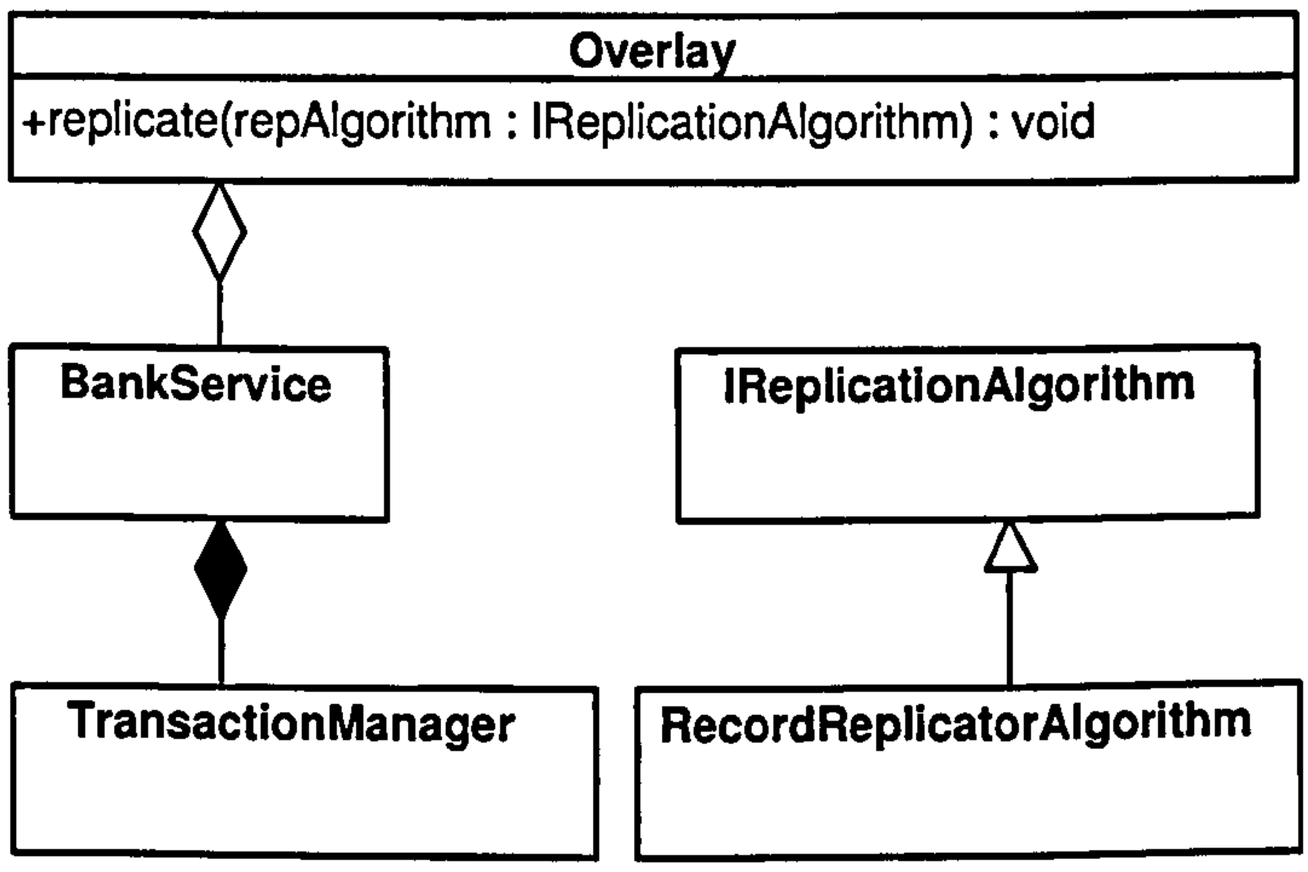


Figure 4.6: Bank Service Components

The Transaction Record Store synchronisation process triggers a record replication. Synchronisation is necessary for reconciliation of the changes made to data while the bank peer was offline. In our case, the transaction records themselves may not be altered once they are committed to the data store, but the state of the data stores would have been changed nonetheless as more new records would have been added to the system while a bank peer was offline. So when a bank peer comes back online after a period of absence (regardless of duration) it needs to bring its data store up to date and share the burden of the extra records added to ensure the efficient working of the network. This is handled by the Record Replicator algorithm in conjunction with the Overlay who decides when a synchronisation is required.

4.2.4 Security Service (SecS)

In section 3.1.3 we have referred to the need for secure distribution of payments and content in the system. Also, one of the fundamental requirements for the cascading payments model to work is the ability to attach the owner's credentials to his content. Hence, we require security services to provide unique identification for every peer in the network, so that his content may be uniquely tagged and his payment may be encapsulated for his use alone. In our framework we use standard security techniques such as encryption, decryption and hashing to accomplish these tasks. The Security Service is also involved in the creation of the UPI, Transaction ID and Content IDs by using their respective ID Factory.

We have developed the UPI naming scheme for network registration. The peer's identity should be unique so that there may be no ambiguity in locating the peer for communication. This identity also has to be unique across different states of the network so as to ensure payment delivery is made to the correct peer. Our solution to this is twofold. The use of a PKI key pair acts as the peer's identifier and forms part of the peer's identity assigned to it when it first connects to the network. This ID would include a hash of the Peer ID + Date Time stamp along with the PKI key pair's public half which the peer possesses. P2P initiatives [Bernard Traversat 2003] use some form of exclusive identification to identify its peers, since we also require this ID to act as the proof of ownership we have to include the peer's public key as part of the id to make it totally unique as well as enabling the buyer to use this key to encrypt the royalty payment for the owner. This UPI would also be the owner's id and hence get propagated with the content in its content descriptor. The Security Service is used to embed the peer's public key into his Unique Peer ID; it uses the PKI manager to retrieve the appropriate PKI Pair. The PKI manager is used to encrypt and decrypt payments as well as communications between peers.

4.2.5 Advertisement, Discovery& Lookup Services

Peer and service location and discovery can be accomplished by propagating messages and caching results. Using a system of caching and broadcasting it is possible to locate peers to connect to and to discover services provided by these peers (where service lists are cached). Caching reduces network traffic as more information may be gained from a single source who has earlier served a similar request. Limiting the number of hops of a broadcast message would also reduce network traffic. Once the peers and their services have been located the required services are invoked to complete the transaction. Some P2P initiatives [Bernard Traversat 2003] use the concept of advertising to advertise services and content, these advertisements are distributed across registry directories which are queried by peers who wish to locate any

services. We have decided to use a hybrid method to assist in location and discovery of peers and their services. This will include the use of advertising services as well as caching results locally to speed up the look up process.

4.3 CasPaCE protocols

The Cascading Payments Model allows a seller to sell content on the P2P network, while ensuring that he/she is compensated for every sale. In instances when the seller is the owner of the content this compensation is in the form of royalty payments and in the instances where the seller is acting as an intermediary or a distributor the compensation is in the form of a commission. To facilitate this in the P2P domain we identified the need for unique identification of the participants, copyright propagation with the content i.e. copyright mobilisation, payment separation and payment distribution. These protocols fall under the category of content exchange and security protocols which include registration, connection & unique peer identification, content creation & the copyright process and the fair content exchange transaction protocol.

The payment protocols used to generate separate payments for royalty and commission based on the information embedded in the Content Descriptors (copyright mobilisation) and to distribute these secured payments form a part of the Payment Service which communicates with the CES. The bank protocols are executed by the components within the Bank Service which delegates responsibility to the different components based on the incoming requests; these include the Bank peer activation & network initialisation, transaction management & transaction record creation, Replication & synchronisation protocols.

Other protocols which were designed to complement the afore mentioned protocols are the service utilisation protocols used to locate and utilise local and remote services, random bank peer selection protocol which is performed by the Peers using their Lookup and Discovery services and the generic Data Access Layer which assists in communicating with the local data stores. The designs of all these protocols are described in detail in the following sections.

4.3.1 Service redundancy & service utilisation protocols

In the previous Chapter we mentioned the ad hoc nature of P2P networks and how participants are heterogeneous, i.e. devices that may participate in a P2P network could vary from resource extensive to resource limited in capability. Bearing this in mind we realised the potential for aggregation of resources as well as the need to allow less resourceful peers to participate.

Our solution to this problem was exposing the framework 'services' provided by other peers. In the context of our compensation system this would primarily entail allowing peers to utilise remotely located services e.g. Payment Services. To enable this, a peer would have to be able to identify the services it requires, locate them and use them (Figure 4.7). We realise that using a service is a non-trivial exercise in itself and hence requires the use of Service Level Agreements (SLAs) as well as dealing with the issues of access control and security of the use of services on remote devices. These topics are vast in themselves and hence out of the scope of our research. However, for the purpose of our research we have well defined requirements for our services and do not need to be able to dynamically match services to our requirements. In our case we assume that the network is heterogeneous and has certain devices which are richer in resources and hence the system will have our Payment Service present redundantly. On the other hand in the future we intend to extend the capability of the framework to allow the use of other 'payment' services which may exist in the community.

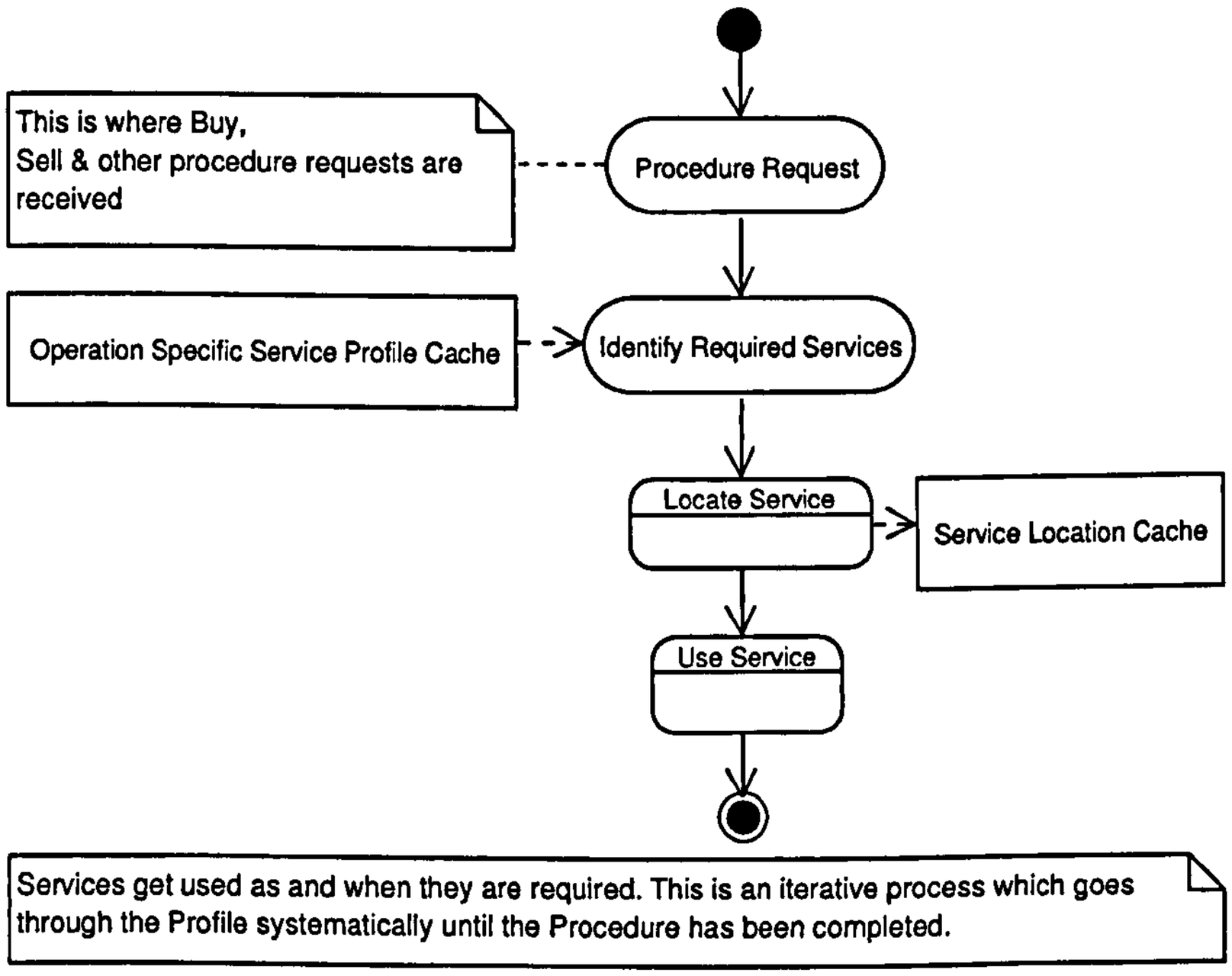


Figure 4.7: Service location and utilisation activity diagram

At initialisation, peers generate an Operation Specific Service Profile, which lists the services required for procedures such as Buy or Sell. Then the local repository is checked for matches, if a service is not present locally, it is discovered within the P2P network. The Service Location Cache is updated, to remove the need for expensive lookups during the sale/purchase procedure which would unnecessarily prolong the process later. A sample scenario would be thus. If the

peer needs to 'buy', the local host checks to ensure all services required to finish that procedure are available. It gathers information on the services and then proceeds with sending a *purchase request.* If any service is not available, it initiates Locate Service (Figure 4.8), which would gather the service attributes and make a service profile or use a pre-determined Buy Service Profile Cache. This profile is used to start the discovery process. Locate Service is an iterative process while the Service Profile Cache is not exhausted.

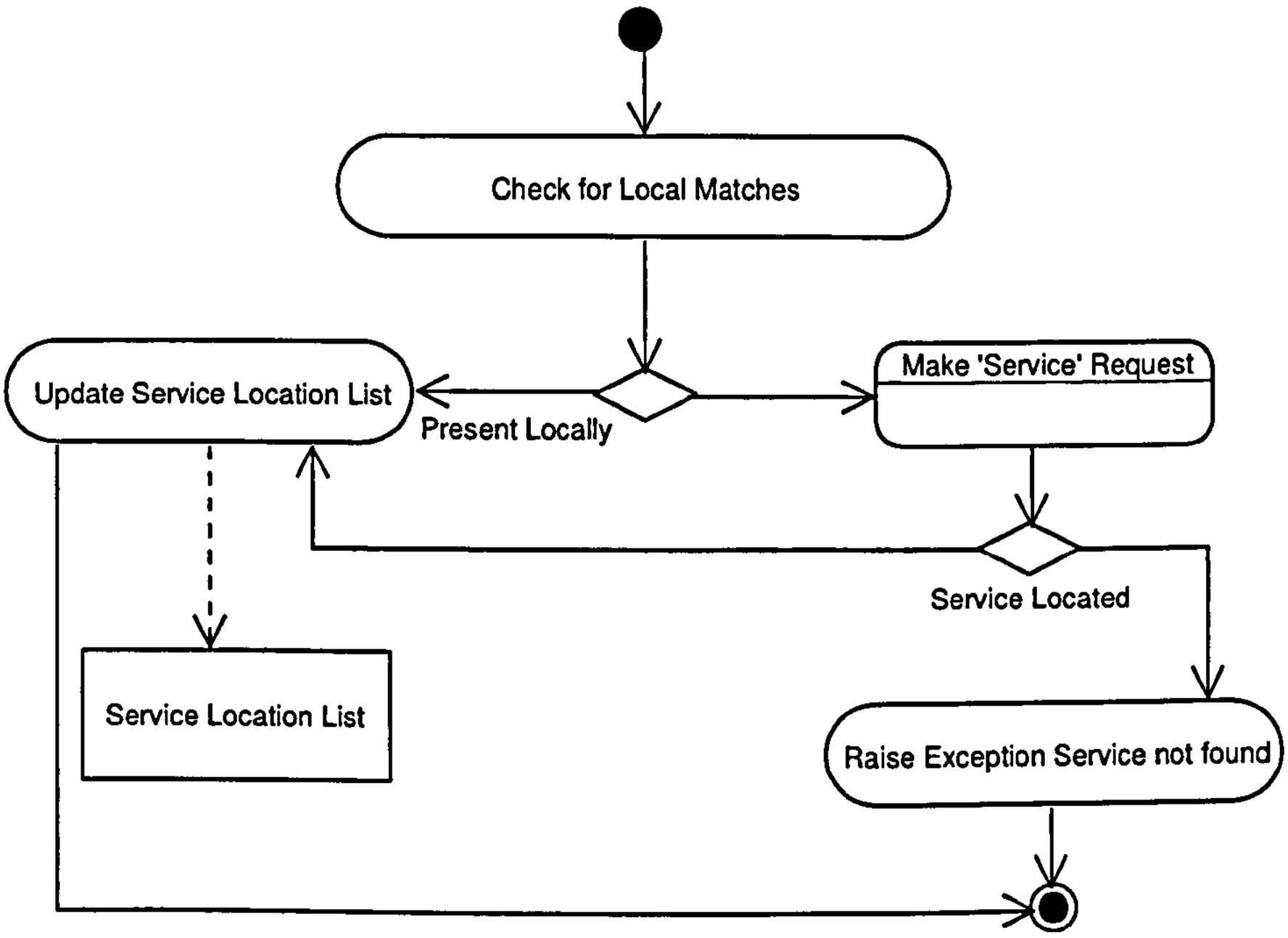


Figure 4.8: Locate Service

4.3.2 Registration, connection & unique peer identification

The peer's Unique Peer Id (UPI) is created when he connects to the network for the first time. On subsequent connections to the network this UPI is used to participate in the community. Connecting to the network (Figure 4.9) involves Login, which includes connecting to the CasPaCE network.

In the *registration protocol* each user/person creates a Person profile which includes details such as his name, date of birth, address and email address. This information is used to uniquely identify the user in the real world; similarly in the digital world this information is sufficient to uniquely identify the person with his real world details. The login details (Identity) of a user are

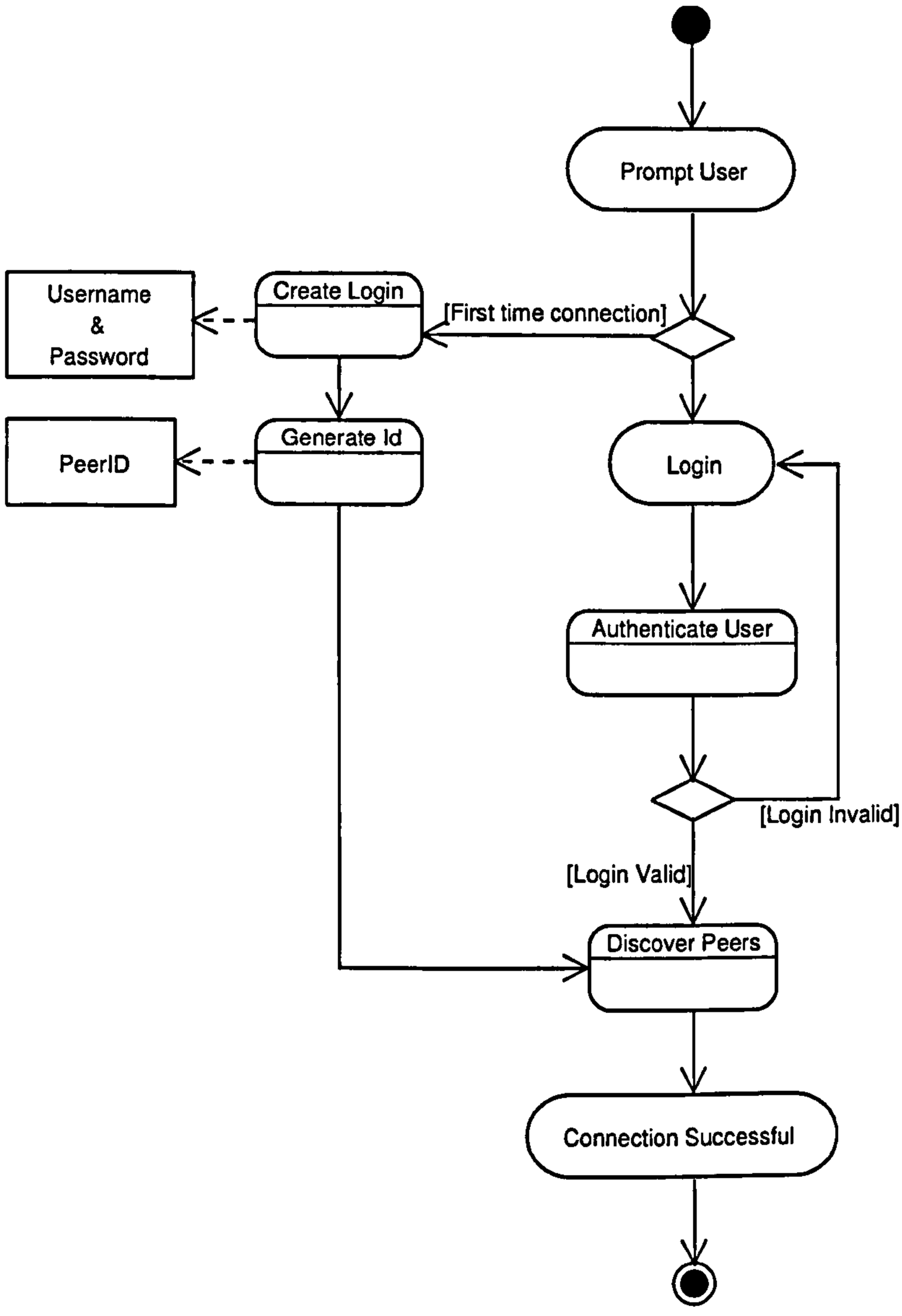


Figure 4.9: Connect to CasPaCENet

the username and password a user would create to login to his/her user account within his application. These are the minimum user level details to maintain a presence in the P2P network. The Peer and Unique Peer Id are system level details, which are hidden from the user. These details constitute the identity of the user within the P2P network and allow his peer to be discovered. Figure 4.10 shows the relationship between these objects in our system.

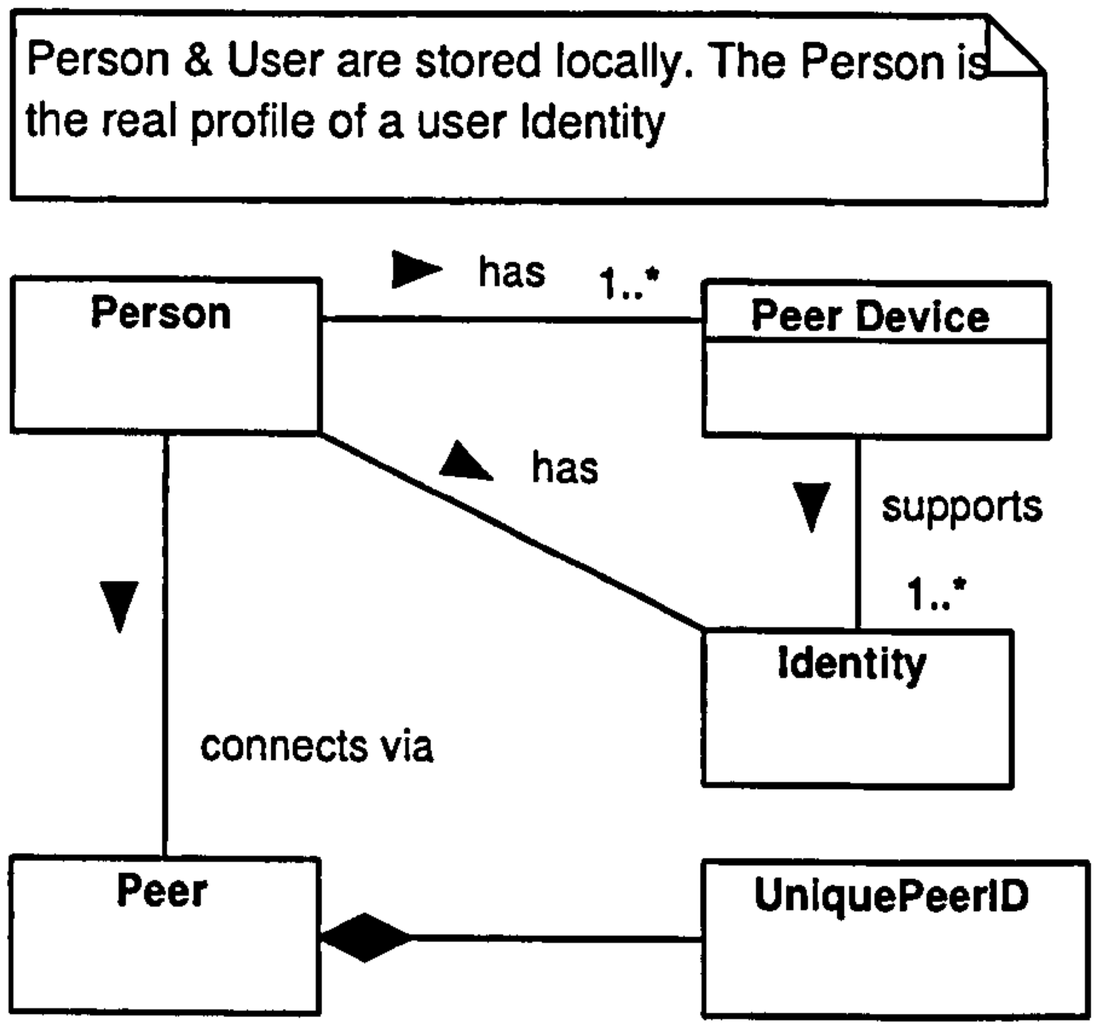


Figure 4.10: Class diagram – UPI & Person relationship

UPI and Identity are stored locally as secured objects. The username and password (Identity details) are hashed before they are stored on the local device. When a user attempts to log into the system, his username and password are hashed and compared to the stored value, this is user authentication. A successful login would imply that the details matched, since the stored values are hashed before storing no unauthorised login may succeed due to login details being hijacked. Also, storing the details locally removes the need for a centralised authentication store in the network. These secured login details are used to authenticate the user in the system before allowing connection to the network with that user's UPI. Once created, only the physical address details in the Person profile may be modified via the application interface. Login details are dynamic as the password may be altered regularly, in this case the hash of the old password would be deleted and replaced with the hash of the new password. In the interface password modification would be preceded by old password verification.

The Unique Peer Identifier in CasPaCE is comprised of:

- Peer ID (derived from the system)
- Date Time Stamp – includes the date (day, month, year) and time (hours, minutes, seconds)
- PKI Public Key – the public half of the asymmetric key pair.

The asymmetric key pair is managed by the Security Service which also assists in the UPI creation. Hashing algorithms are also used via the security service.

The use of the PKI public key has multiple benefits:

- It ensures that if any two peer IDs are generated simultaneously the UPI will be unique as no two parties would have the same asymmetric key (public key).
- It is embedded within the content's content descriptor and assists in securing the royalty and commission payments.
- Initially a system generated Peer Group ID was included in the UPI, however on reflection this was removed as it was deemed redundant because a Group ID would associate a peer for life with a particular community. Whereas our intention is to make a persistent identity for a peer who would in future have a reputation giving it a global presence.

In legacy P2P systems, the system derived Peer Id is usually the IP address of the node, which is unique for the duration of the connection or a hash of the location address (e.g. URL). Using our method of identification it is possible to give the peer a persistent identity for the duration of his existence in the network, not just per connection to the network.

One of the major requirements of any P2P network is the existence of a community without which a peer's existence is ineffective. Consequently, connection to the network also requires finding other members of the community. This is performed by the peer via its P2P interface which utilises the Lookup and Discovery services to locate other peers and updates its local routing table in the process (Appendix B, Figure B. 8: Discover peers).

If a DHT routing protocol is used then we perform similar operations however in this case Peer IDs are not looked up based on previous experience i.e. from the cache, but on the circular ID plane as in Chord.

4.3.3 Content creation & the copyright process

In the context of this system, content creation is the process of preparing the content for distribution via the CasPaCE network. To do this the owner inputs raw data which becomes the digital media content. The actual content generation process is performed outside of this system. However, to offer this content for cascading in the CasPaCE network, our stipulation is that it should propagate its copyright with it. One method of propagating this copyright information would be to watermark the content by implicitly embedding the copyright within the content.

In our case this watermark is a content descriptor (Figure 4.11) which should include the content's details such as its description, name and type. The user has to include the royalty

value he expects for his content within the content descriptor along with the commission policy. At present the estimation of the value of his content is performed intuitively by the user, however in future a system [Srinivasan Jagannathan 2002] to estimate the value of his content on the basis of its popularity as well as the buyers' ability to pay or the content's desirability could be included to assist the owner in establishing a price for his goods. But most importantly, the copyright is established by the inclusion of the owners' identity within the content, in our system this is the UPI.

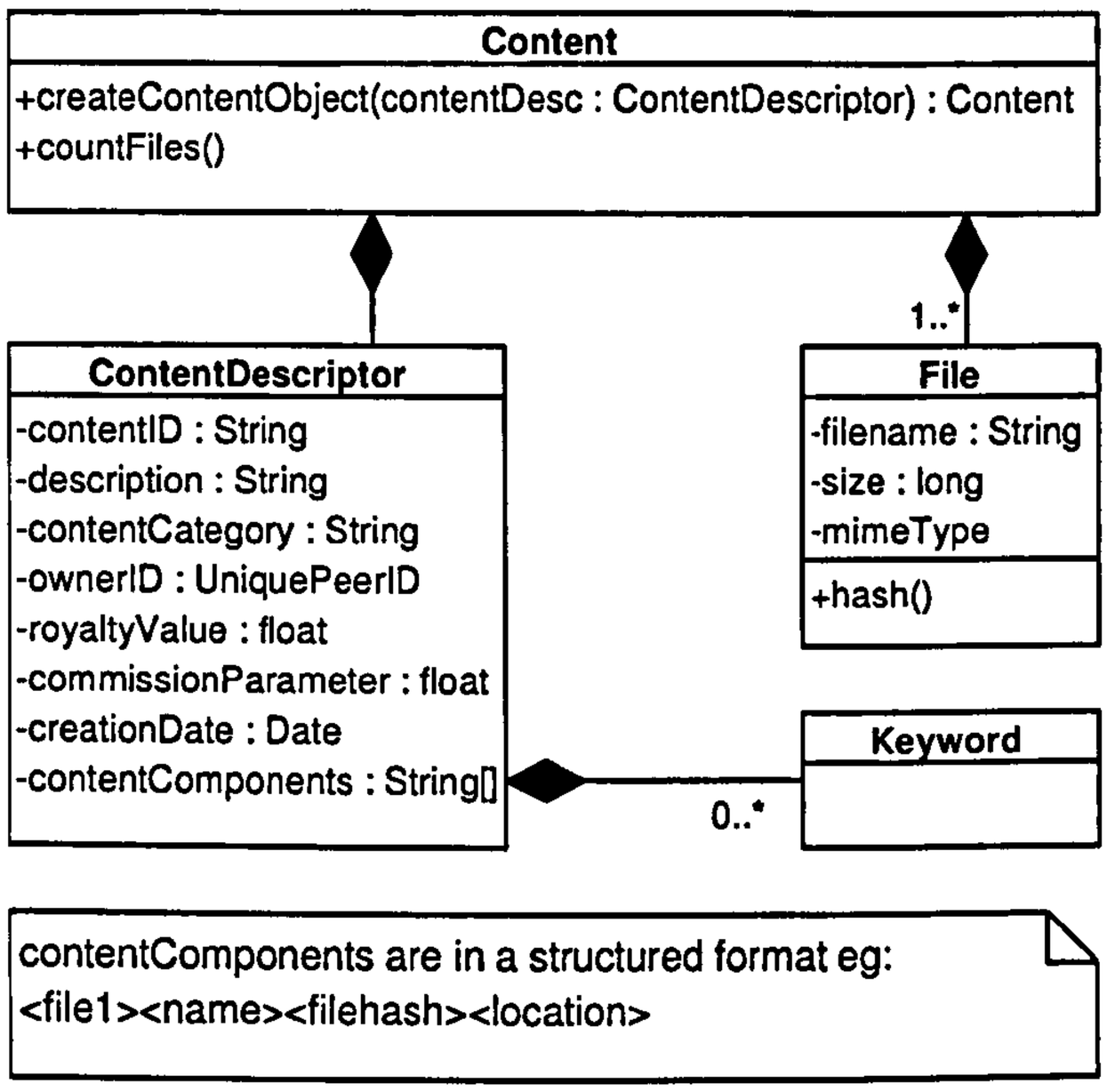


Figure 4.11: Class diagram – Relationship between content and content descriptor

Another distinction to bear in mind is that the content creation is only performed by the content *Owner*. This *Owner* can then sell his content by inserting it into the network. The *commissionParameter* is setup by the *Owner* at content insertion. The search results returned should reflect the commission rule included in the *commissionParameter*. However, the commission value is not necessarily sent by the *Seller* but determined by the rule setup i.e. there may be instances where the *Owner* has setup a rule for the commission in which case the *Seller* is not able to setup his own commission value for that content.

To uniquely identify this content, it is given a content ID which is a hash of the content file itself. This method allows us to uniquely identify each content. If the file were modified it would result in a different hash value. The added benefit of doing this is that if the owner decides to modify his original content and offer it for sale, then two different files with different

content IDs will exist. Since the content descriptor is watermarked into the content, this method also removes the need for the content owner to have to remove older versions of his files from the content exchange network which is a non-trivial exercise in a P2P network, although systems exist where it is done. Initially, we had decided to allow the content owner to modify the commission parameters once the content is well known in the P2P community so as to allow for dynamic market states. However we realised that this was not possible as the owner would have to have access to every copy of his propagated content and also access control to it so he could modify the copyright information. Although it is possible to remove older versions of a file from a P2P community it is a nontrivial task to modify those files. Consequently, the *commissionParameter* and *transactionFeePayee* attributes are incorporated into the content descriptor and may not be altered once the content is watermarked. Who pays the bank the transaction fee is also determined by the content *Owner*. However, an *Owner* can create a new version of the same content with different parameters and both versions will co-exist in the system with different *creationDate* which includes the time and is precise to seconds.

A design decision was made to separate a seller's payment information from the owner's primarily because we want to maintain the integrity of the content descriptor so it should only be inserted by the owner. Hence the seller's information is sent to the buyer on-the-fly at the time that search results are returned to the buyer.

Once the marked content has been created, it is ready to be offered for sale (Figure 4.12). This can be performed in a variety of ways, either placing the content in a shared store, such as a shared folder as done in BitTorrent [M. Izal 2004] and KaZaA [Nathaniel S. Good 2003] or by passing a hash or unique reference ID [David Hausheer 2003] which gives access to that content. The main intent of this process is to provide access control to the content. Our method to accomplish this is by advertising the content for sale and the advertisement would contain the URI of the content's location. This URI is included within the Content Advertisement which is propagated as part of the search results. The search results are returned to the peer who performs a search for the content.

The use case 'Offer the content for sale' (Figure 4.12) differs from 'Make a sale' because it is the actual selling process where money is exchanged whereas offering the content for sale is the same as inserting it into the network or advertising it. Making a sale will be discussed in the payment protocol section.

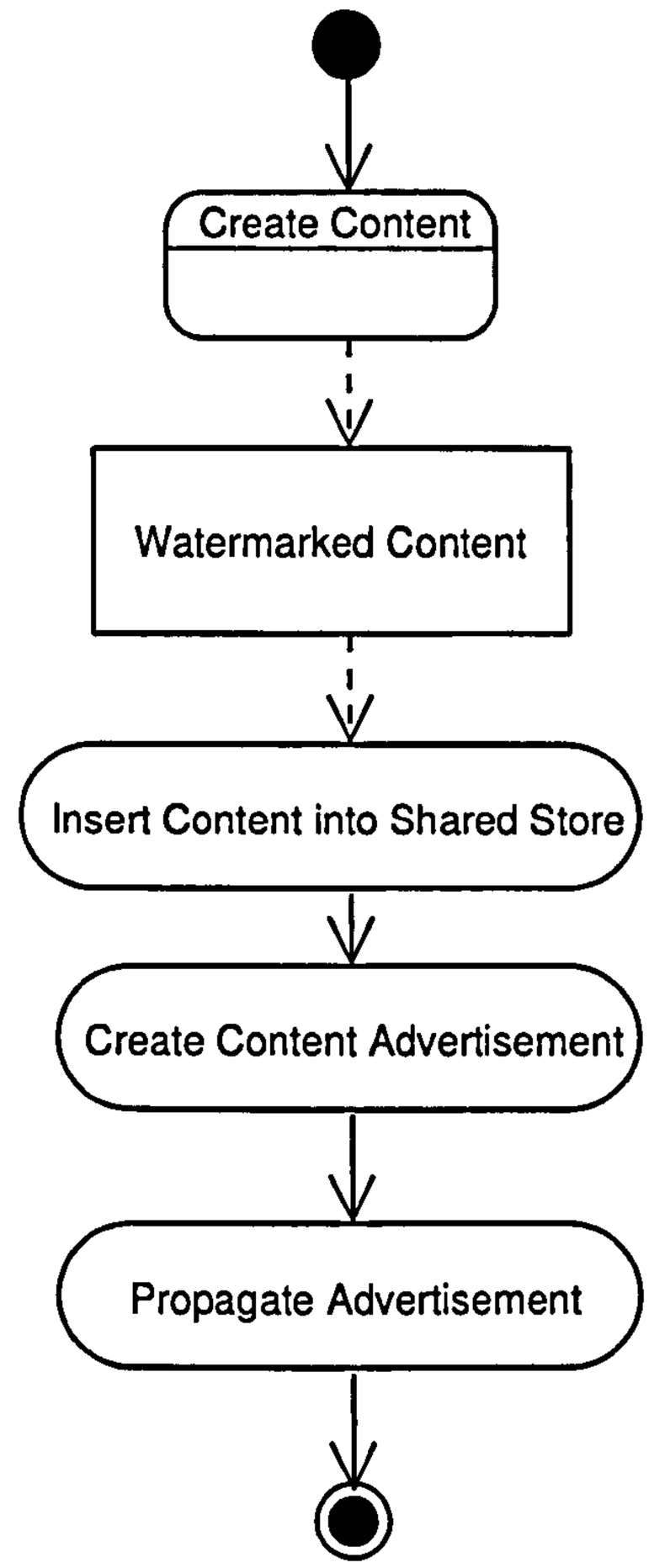
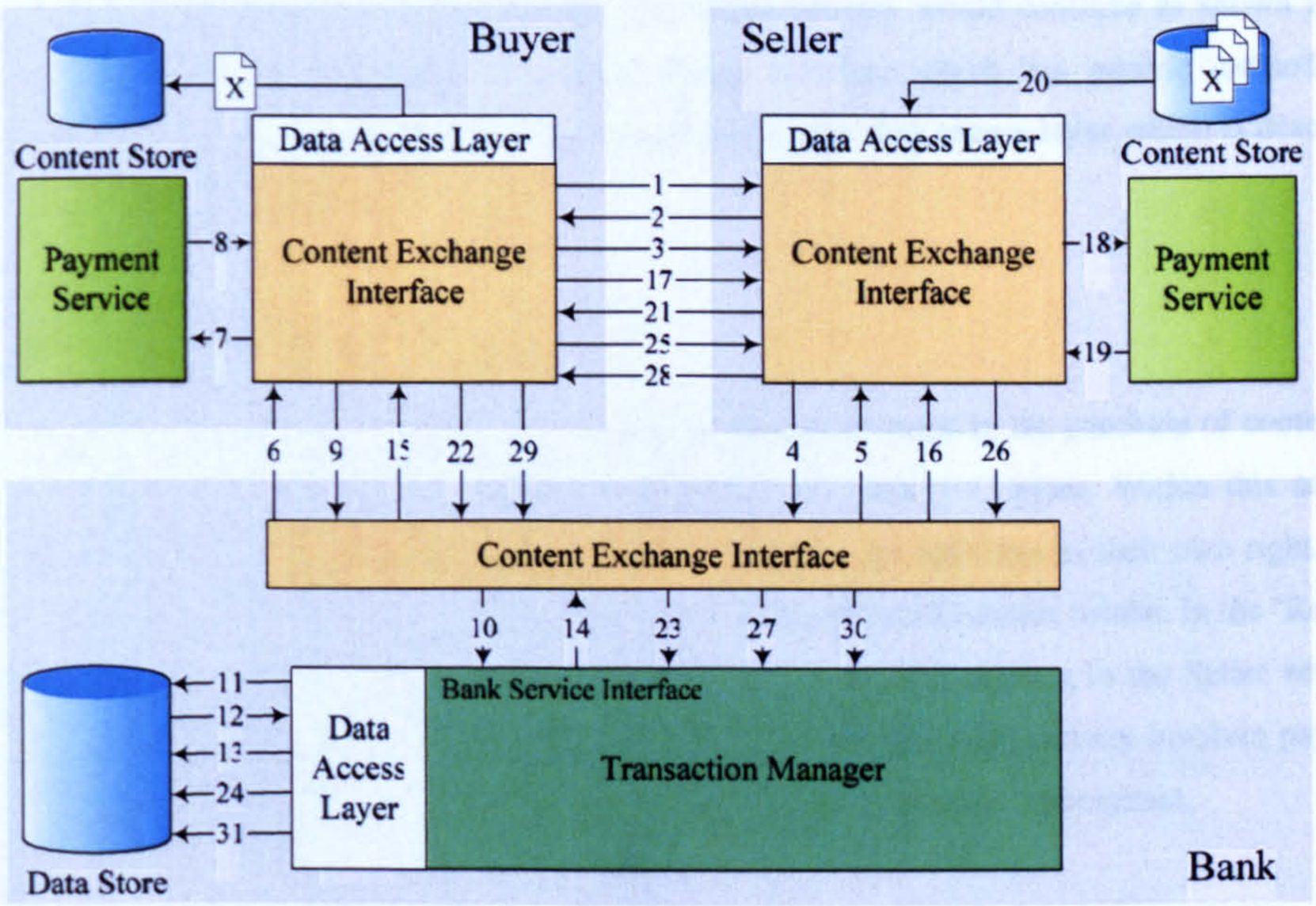


Figure 4.12: Offer content for sale

4.3.4 The fair content exchange transaction protocol

Figure 4.13 shows the sequence of events between the buyer, seller and bank peer to show the exchange of content for payment. It gives a detailed illustration of the data that is passed between the various peers and their internal interactions as well. We illustrate the encryption and PKI required ensuring the fair exchange of content for payment where it is not possible for the buyer to receive the content without disclosing the payment to the seller. The encryption and key management are not shown in this sequence of events; the security service is used to perform this functionality. We have adapted Zhang et al's [N. Zhang 2003] technique to our scenario to facilitate the fair exchange. It also illustrates how the various services work together to facilitate the paid content exchange. This diagram demonstrates the scenario where the services are present locally. The ability to allow location of remote services is not modelled in



Key	
X	Content to be exchanged for Payment
PC	Commission (Payment)
PR	Royalty (Payment)
PK_O	Owners Public Key
PK_S	Seller's Public Key
PK_T	Bank's Public Key (third party)
$cert_X$	certificate which certifies that content X is genuine
$PK_X(X)$	cipher text of content X encrypted with its Key PKX
Pkr_x	item for the computation of key PK_X
Hd_x	hash value of the ciphertext of content X
$desc_X$	description of content X
$sign_{CA}$	signature of Certifying Authority
r_c and r_x	Random numbers picked to compute keys
co_s	Seller's commitment

Seq. No.	Event
1	Search X
2	Result ($cert_X$, Purchase Details) $cert_X = \{desc_X, hd_x, f(Kx), sign_{CA}\}$ Purchase Details = {contentID, ownerID, RoyaltyVal, SellerId, CommissionVal}
3	Request Purchase
4	Bank Service Request
5,6	Ok – Bank peer sends its public key (PK_T) to both buyer and seller
7	Payment Details
8	Payment (separated into commission & royalty)
9, 10	Transaction Details
11	Check Record
12	Result
13	Add Record
14,15,16	Transaction Start Logged – return the TransactionID to both parties
17, 18	Payment - $\{PK_C(PK_S (PC)), pkr_c\}$
19	Payment Verified
20	Content retrieved from store
21	$\{PK_X(PK_B (X)), pkr_x , co_S\}$ where co_S is based on pkr_c and r_x
22, 23	Buyer sends r_x recovery details to Bank $\{r_c , f(r_x), co_S\}$ to log as part of the transaction & Content Received
24	Update Record
25	Buyer sends r_c to Seller
26, 27	Payment Received sending r_x
28	Seller sends r_x to Buyer
29, 30, 31	Buyer confirms successful completion of transaction.

Figure 4.13: Fair content exchange sequence in CasPaCE

this sequence of events, however as described in the redundant service utilisation section (§4.3.1 pg 89) the local node possesses a Service Profile which contains the location of services not present locally. In that case, a process of remote location of services would be initiated if the required service is not present locally. Once the service has been located it will be utilised to

provide the desired functionality and the rest of the process would continue as shown in the diagram.Each peer implements a generic storage interface which has generic methods for retrieve, save, update and commit. This is captured by the data access layer which is described in detail later (§4.3.10).

4.3.5 Payment protocols

The payment protocols are used by the Payment Service to assist in the purchase of content. In our framework this is the 'Buy Content' (Figure 4.14) activity diagram. Within this activity 'Find Content', 'Request Purchase' and 'Make Payment' are activities in their own right. Find content is a specialised form of 'Search' and returns the required content results. In the 'Request purchase' the Buyer generates his connected bank peer set and sends it to the Seller with the reference to the required content. The 'Make Payment' (Figure 4.15) activity involves payment separation; this involves generation of royalty and commission where appropriate.

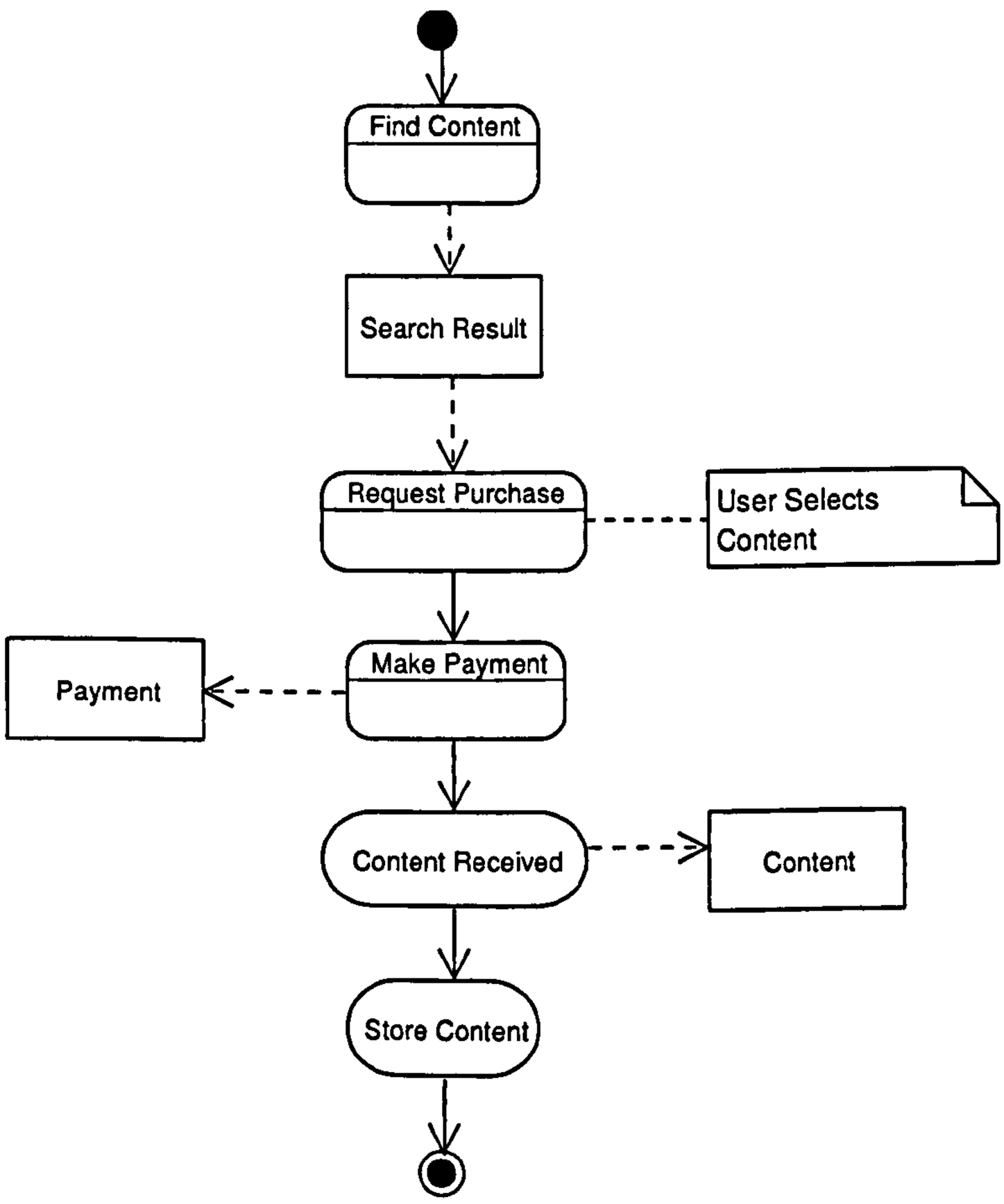


Figure 4.14: Buy Content

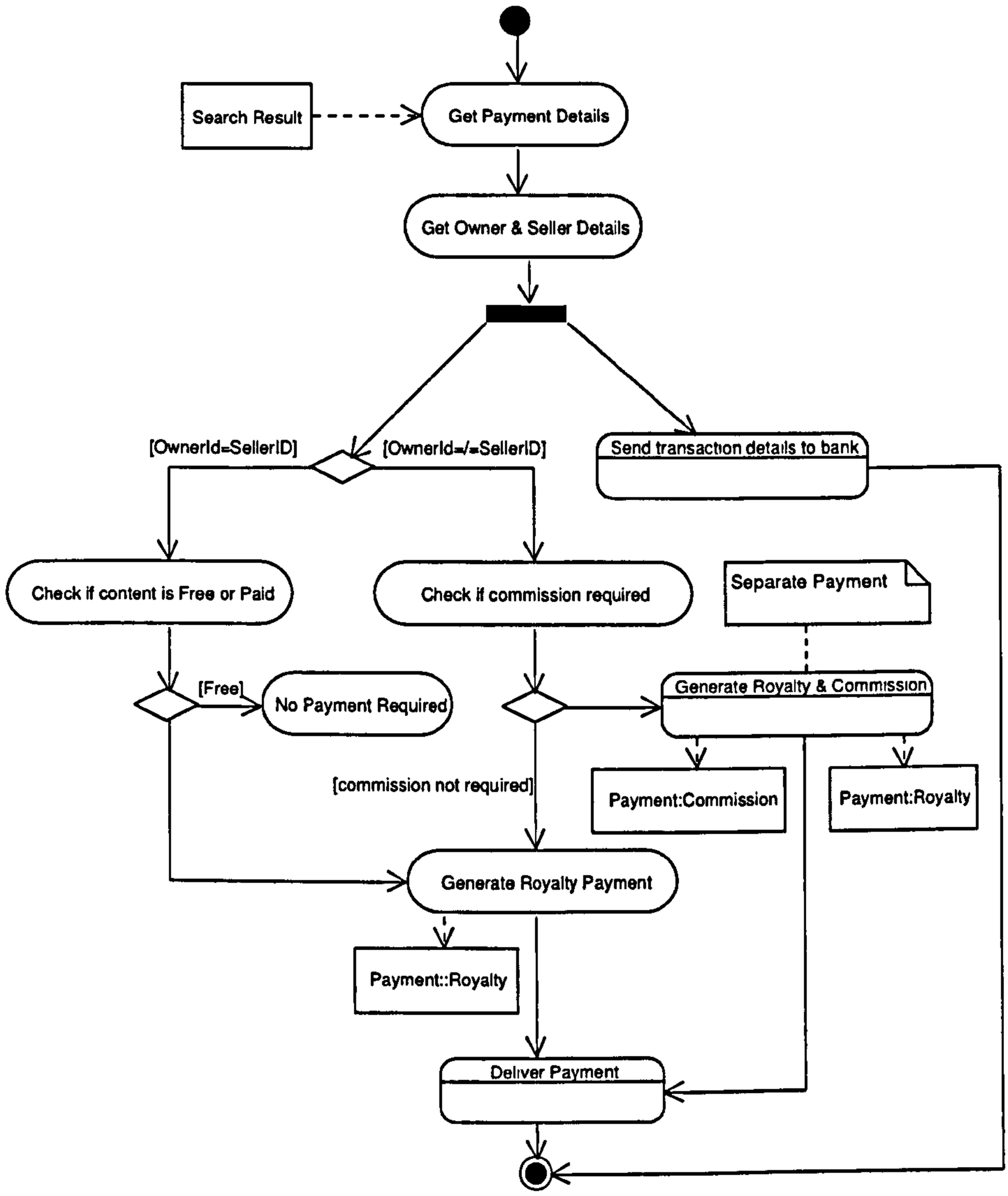


Figure 4.15: Make Payment activity diagram

The Payment Separation (PayS) protocol relies on the embedded identity of the content owner within the digital content. At the time of content exchange, this owner identity has to be retrieved by the system and, in conjunction with the royalty value (also embedded), it is used to create the payment for the owner and the intermediary seller (where applicable). The Payment Separator extracts the owner specific information from the returned SearchResults to assist in payment separation as illustrated in Figure 3.3 in section 3.2.4 page 57.

The Buyer's Payment Service uses its Payment Separator to separate the payment into royalty and commission and encapsulate each payment within the respective peers' identities (the

Unique Peer IDs contain each peer's public key). Once the payment has been separated into royalty and commission it has to be given to the respective peers, the owner and the seller/distributor. In our solution the seller receives payment in real time i.e. while the exchange is taking place, but the owner may or may not be online simultaneously. In that case we have to ensure that the royalty payment is accurately routed to the owner in such a way that only the owner may redeem the payment and so no other intervening peer can hijack and use the payment. To accomplish this we devise the Payment Distribution protocol which include the Payment pushing and Payment collection algorithms which work with each other.

Initially, we designed the PayD protocol to distribute the royalty and the commission individually to both parties; however when fair exchange became an issue to be addressed, the PayD protocol was modified. The Payment Distributor (PD) is used to route the royalty payment back to the Owner of the content and the 'commission' is routed to the Seller. In the fair exchange both payments are encrypted with their respective public keys and sent to the Seller. When the Seller has received both encrypted payments its PD routes the royalty to the owner. The location of the owner peer is determined by the PD via the Discovery Service. This location information assists the PD in determining whether the royalty may be delivered directly to the peer, if he is online, or if it has to be passed to a peer closer in proximity to the owner. This process is termed Payment Pushing (Figure 4.16). Another aspect of payment pushing has to ensure that a payment is not lost due to node failure, i.e. ensuring the integrity of the payments while they reside on remote nodes. To overcome this problem a fail safe has been built into the system, whereby multiple copies of the encrypted payment object are made by the PD and distributed via different routes to the Owner. The first instance of the encrypted royalty which is received is redeemed by the Owner; the other copies are deleted upon detection as they are non-redeemable. The only stipulation in this instance is that the payment instrument being used should be non-double redeemable. The other issue is of having copies of a payment on multiple nodes simultaneously. Since the payment is encrypted the wrong node can not redeem it, so a copy of the payment at the wrong node will not be a problem.

The Payment Collection algorithm is initialised by the Owner's PD when he comes online. This involves a request being sent to its neighbouring peers for undelivered royalty payments[Gurleen Arora 2005a]. The inclusion of transaction records within the system ensure that the owner has the option to collect payments directly from the Seller as he has access to Seller details through his transaction records stored on the bank peers.

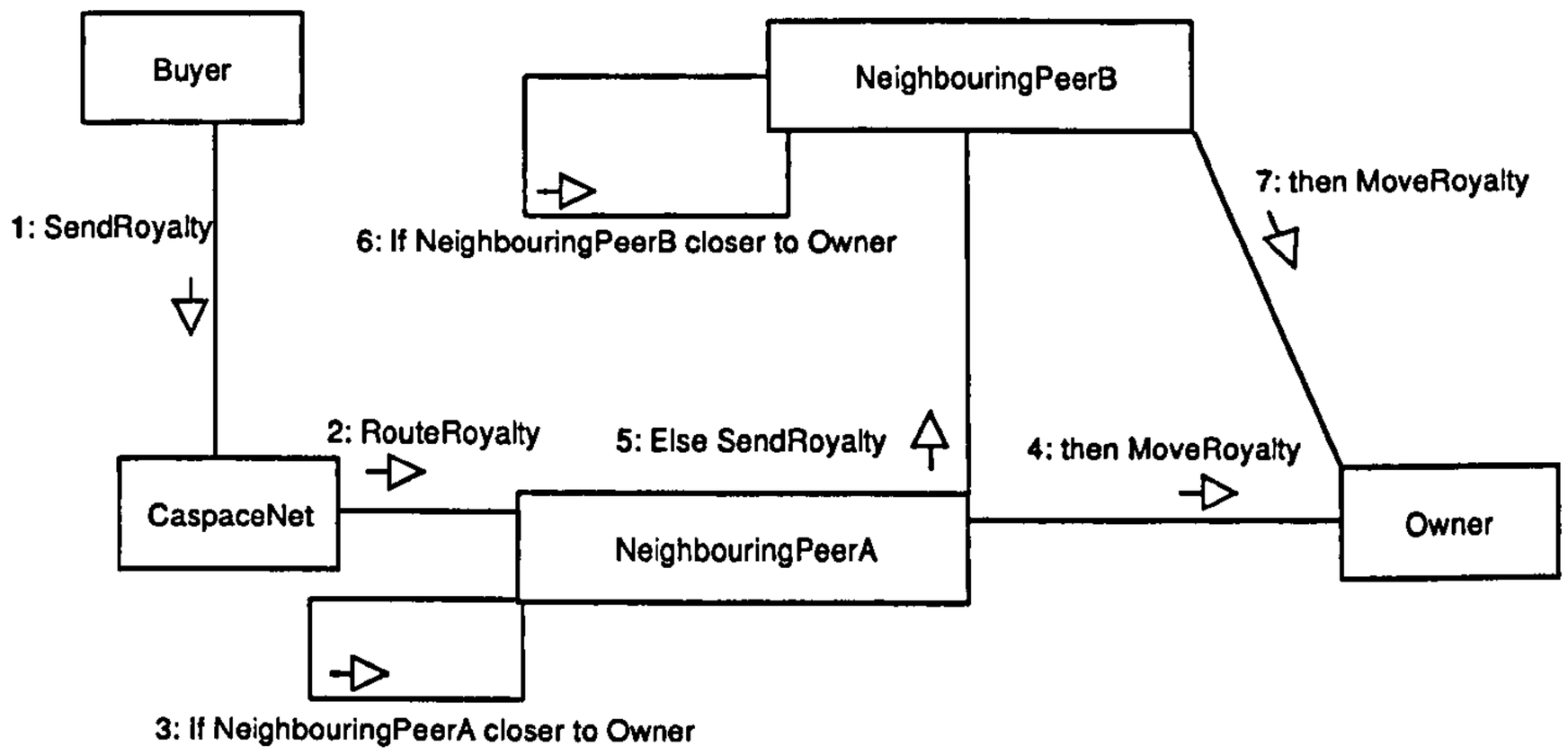


Figure 4.16: Payment pushing

The Payment Verification (PayV) protocol is dependent on the type of payment instrument being used. For instance in hash chain based payment schemes like Payword [Ronald Rivest 1996], the payment verification involves running the verification algorithm (which is delivered with the payment details) on the Payword to verify that they belong to the correct chain. Other similarly structured payment schemes, which do not rely on online verification [Hitesh Tewari 2003] would also be verified in the same way where the verification algorithm is part of the payment itself and is run by the payee on receipt of the payment. So payment verification is done on the fly and locally.

Alternatively if a payment instrument was used which the verifier did not understand, the verifier would have to locate another verifier service within the network which could perform this functionality. In real life terms, if a user was receiving payment in Dollars but only understood Yen then the payment receiver (payee) would have to find someone who would be able to convert dollars to yen or vice versa.

4.3.6 Bank peer activation & network initialisation

There needs to be major restrictions on ordinary peers becoming bank peers. Firstly these are physical constraints which do not allow peer devices with limited physical resources to become bank peers. Hence only peers with sufficient memory and storage capabilities and no restriction on battery life can become part of the bank peer network. Thus our thin peers can not participate in the bank peer network.

Initially we believed every peer, who had the resources to function as a bank peer, would be able to decide when to act as a bank peer. This would also work in favour of further extending the project to include the ability to exchange services for payment. We also considered paying Bank Peers for their services. However, on further reflection and research it seems unfeasible to allow this functionality on the sole criteria of 'availability of required resources'. Any malicious node could come online to offer its services as a 'bank'. Also some nodes might not wish to participate in any other role except as a bank peer, which would lead to system degradation due to the power-law effect [Lada A. Adamic 2001]. Hence, we have to implement a basic requirement metric whereby only peers that fulfil that condition may provide a bank service and get promoted from normal peers to bank peers. Issues of trust become of great importance in this situation.

Since bank peers are charged with acting as trusted third parties in a transaction, not just any peer can become a bank peer. For elevation to bank peer status, we stipulate that the peer registers with a certificate authority and also combine this with a requirement to participate in the network for a fixed amount of time which would generate a reputation for the peer. There are P2P reputation management systems[Karl Aberer 2001b, Fabrizio Cornelli 2002, Sepandar D. Kamvar 2003] which assign reputation scores to peers based on behaviour. However, they do not give peers reputation on the basis of the type of service provided, just on the valuation of other peers views. PeerTrust's [Li Xiong 2003] method of calculating trust is more relevant to our system. In future work a reputation generation mechanism such as PeerTrust will allow us to assign concrete reputation scores to peers which will give them the right to become a Bank peer. Similar to our overlay network of bank peers which provide assistance in content exchange, the PeerTrust system functions as an overlay. At present we use the certificate as a form of validation.

If there is no bank in the system, it is essential that a bank service be initialised. It is common practice in P2P systems to have boot strapping mechanisms such as seed startup servers which contain locations of other peers or services [M. Izal 2004, Qin Lv 2002]. For the purpose of our network we use a peer with certification to start the bank service. This certificate is purchased from a traditional certificate authority (CA) and validates the identity of the Bank Peer. Once certified, this peer can advertise its Bank Service.

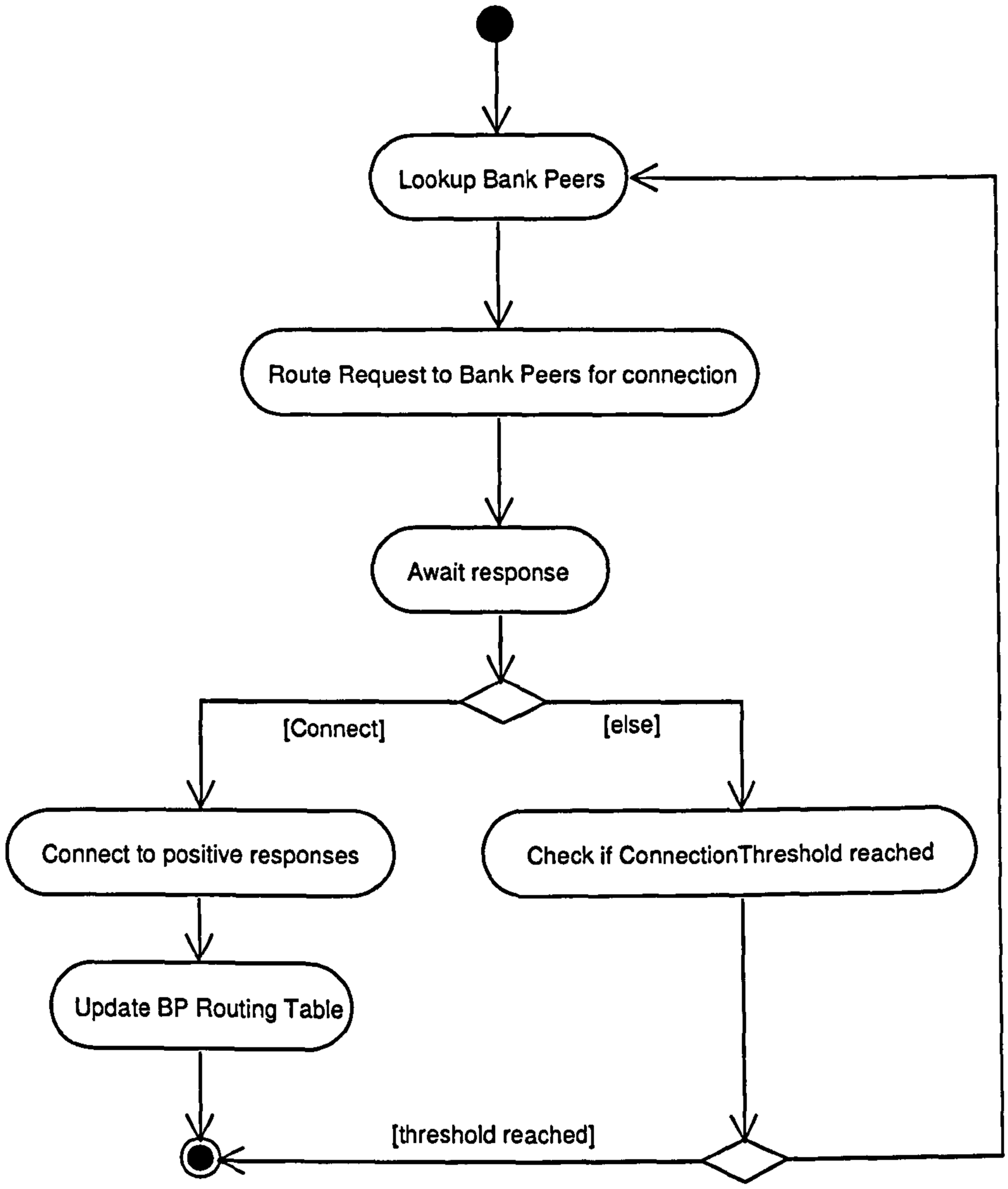


Figure 4.17: Connect to bank peer network

4.3.7 Transaction management & transaction record creation

It was deemed important to this research to mirror the transaction record creation process to coincide with the progress of the transaction itself. Therefore record creation coincides with

transaction initiation and committing a record is equivalent to transaction completion or transaction cancellation (in cases of communications breakdown). If a transaction is terminated for any reason before completion (including loss of the bank peer) the transaction record should be available for resuming the transaction or to prove it was not completed, especially if this occurs at a point after payment was made. This activity is managed by the *Transaction Manager* in conjunction with the local *Transaction Record Store*. This record store is accessed via a generic Data Access Layer described later. This activity is modelled with the fair exchange protocol in Figure 4.13 on page 97.

4.3.8 Replication & synchronisation

There are various techniques of replication in computer science research as well as in P2P systems[M. Izal 2004, Qin Lv 2002, Kavitha Ranganathan 2002, Tyron Stading 2002], however in our case we do not need to maintain a complete duplicate of the entire *Transaction Record Store* at each location but instead we need to ensure that multiple redundant copies of every record are placed evenly with all Bank nodes. During a transaction, the complete transaction record is maintained on a single bank peer.

The synchronisation (Figure 4.18) is performed by the bank peer when it first comes online to update its *Transaction Record Store*. This activity is performed by the Overlay manager when it first joins the BP network. The overlay manager transmits a *synchronise* request to its network members which triggers the Record Replicators on the member nodes. The *replication* process refers to the actual duplication of records and despatch of the duplicates to the *synchronise* requestor. Synchronisation is not intended to duplicate a single bank's complete transaction store on all other bank peers, instead it is intended to push out copies of the records onto other bank peers to speed up lookup (at the transaction validation stage), enhance system performance and ensure record persistence in the system. The other instance when a *replicate* is performed is when a long time has elapsed and the bank wishes to replicate its records as described above. Finally, if a BP is going offline it also performs a *synchronise_push* where it replicates and transmits its records since the last synchronisation and these records are added to the *Transaction Record Stores* on the BP member nodes for persistence.

Records are also replicated at regular intervals to enhance overall system performance. Two methods were considered to determine the frequency of replication, first option was 'after a predetermined time has elapsed' and the second option was once a 'set number of transactions have been completed'. Calculating the frequency of replication based on time elapsed is a non-trivial task as it has to take into consideration parameters such as network size, frequency of

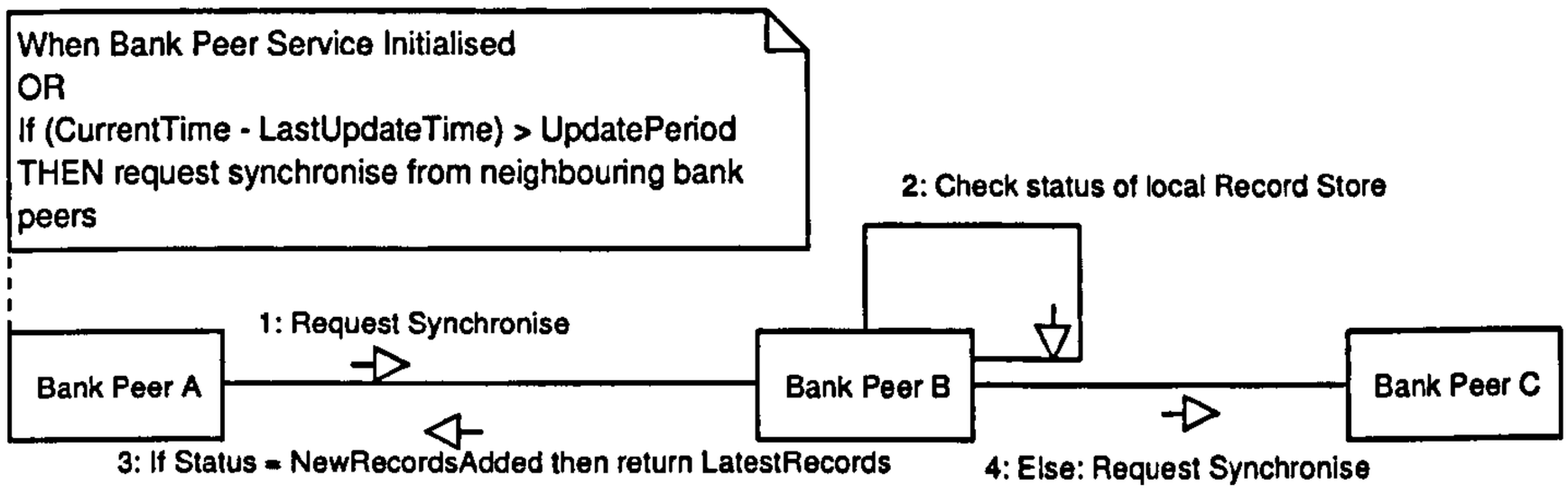


Figure 4.18: Synchronise in bank peer overlay

transactions, average activity of the bank peer within the system and many others. Consequently we choose to trigger local replication on the basis of number of transactions completed since last replicate and when the bank peer goes offline. Including date time stamps into the Transaction record at record creation and at transaction completion allows the bank peer to identify new completed transaction records since its last replicate. These replicated records are sent to all active bank peers based on a replica placement policy determined by the use of Ranganathan et al's [Kavitha Ranganathan 2002] dynamic model-driven replication technique. This technique addresses our need for increasing data availability which encourages faster lookup times and also assists in data persistence. This activity is managed by the *Record Replicator* in the *Bank Service*. Other replication techniques may be used within our framework via the *IReplicationAlgorithm* interface (refer to Appendix C).

4.3.9 Random bank peer selection algorithm

In order to participate in a content exchange transaction, both the buyer and seller need to select a common bank peer. In our approach the buyer and seller exchange the information they have on the bank peers they 'know'[29]. Based on the underlying routing protocol being applied, this could be cached routing information which identifies bank peers in the peers' neighbourhood or it could be derived from a separate routing table dedicated to listing the location of bank peers. From this information a set of common bank peers are selected by performing an intersection (Figure 4.19) on the exchanged sets and one bank peer is randomly picked. The common bank peer is contacted and bank service is requested. This process is termed *random bank peer selection*. If the intersected set is empty, both parties 'discover' bank peers to update their tables and then proceed as before.

[29] Here 'knowing' a bank peer refers to the state where peers are aware of other peers in the system due to a previous transaction with them or because their location is cached via previous queries.

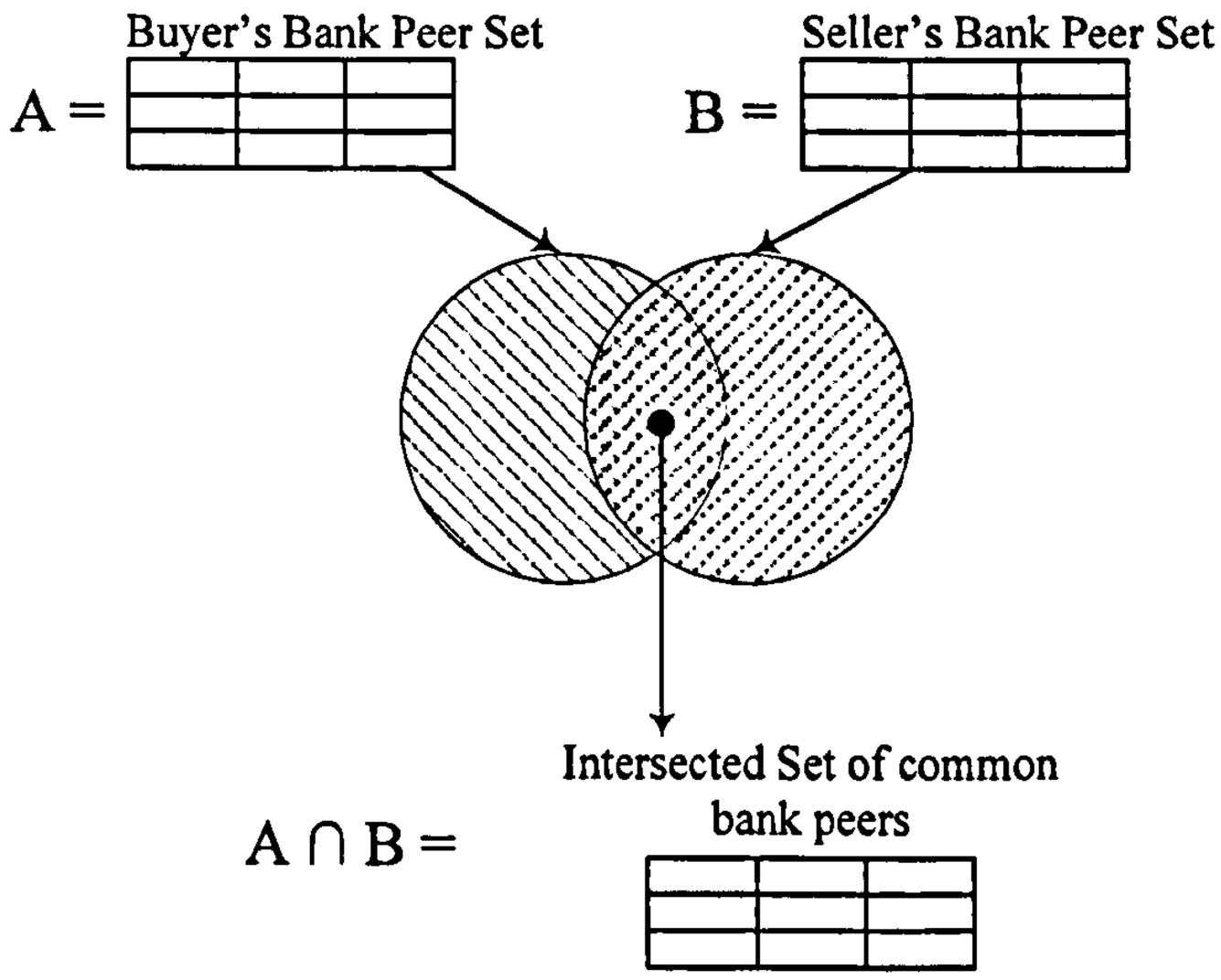


Figure 4.19: Selection of set of common bank peers

The Seller initiates the random bank peer selection protocol, when he receives the *purchase request* from the buyer. This *purchase request* includes a set of connected bank peer IDs which the buyer has logical connections to. The Seller's Lookup and Discovery services assist in the location of its connected bank peer set. These sets are matched and common bank peer IDs selected. If no common IDs exist the seller performs a discovery of bank peers to update its set and then performs an intersect. Once a common set of IDs is extracted, the seller peer runs a random function on the intersected set and selects a Bank Peer. This Bank Peer ID is sent to the Buyer so he may poll the Bank for his service and the Seller polls the Bank on his own as well through a *bank service request*. Once both parties have successfully contacted the selected Bank and received confirmation of its service availability the transaction may begin.

4.3.10 Data storage

Due to the heterogeneity of the P2P environment, we recognise devices (e.g. PDAs, mobile phones and iPods) will have access to different data stores. Hence they have to know and understand their local storage. The data objects that are exchanged between peers should seamlessly be exchanged regardless of the type of data storage technique being used by the participating peers. Our framework provides a set format for all its data objects (Appendix C); this is the format in which the data is exchanged between the peers. Internally the peers implement an interface (Figure 4.20) to their respective storage to convert the data into the format compatible with its local store.

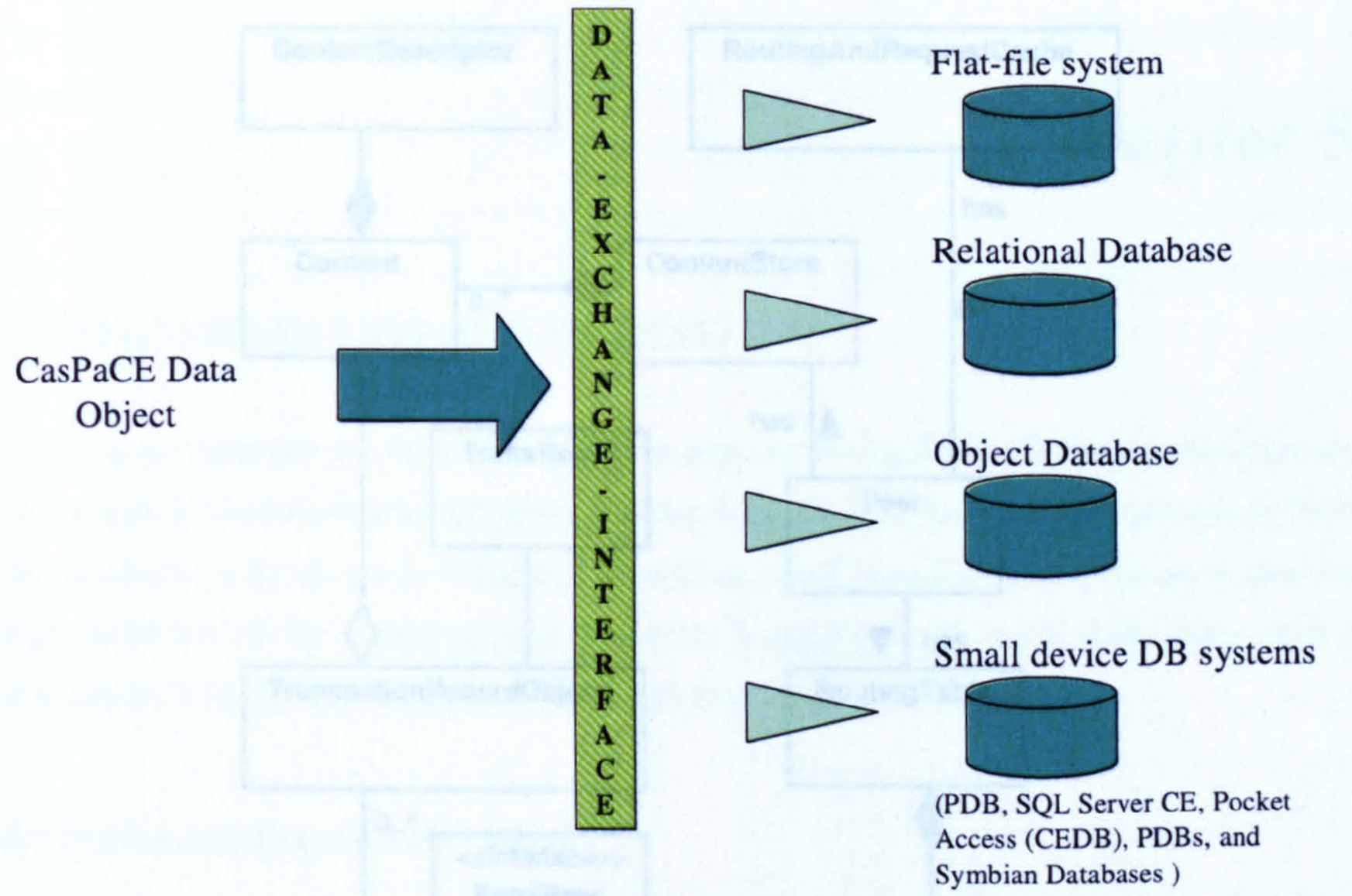


Figure 4.20: Data access middleware

The different storage requirements in our framework are met by three main stores: the *Content Store* (Content Database – CDB), *Transaction Record Store* (Transaction Database – TDB) and the *Routing Tables* and *Request Caches* (RT/C). Each store interface includes methods to retrieve objects from its respective store, however the *Transaction Record Store* has additional methods which allow record manipulation such as create, update and commit. Figure 4.21 illustrates the relationship between the various data objects and their data stores within the CasPaCE framework.

Appendices A-D contain the complete system design notes illustrating the CasPaCE services and their components in detail.

4.4 Summary

The understanding of P2P technologies gained from the literature review (Chapter 2) assisted us to derive the requirements (Chapter 3) for a system to enable paid digital content exchange in the dynamic P2P environment and devise a framework for cascading payments content exchange. Chapter 3 discussed the high-level architectural requirements and model design for a framework for cascading payments and content delivery – the CasPaCE framework.

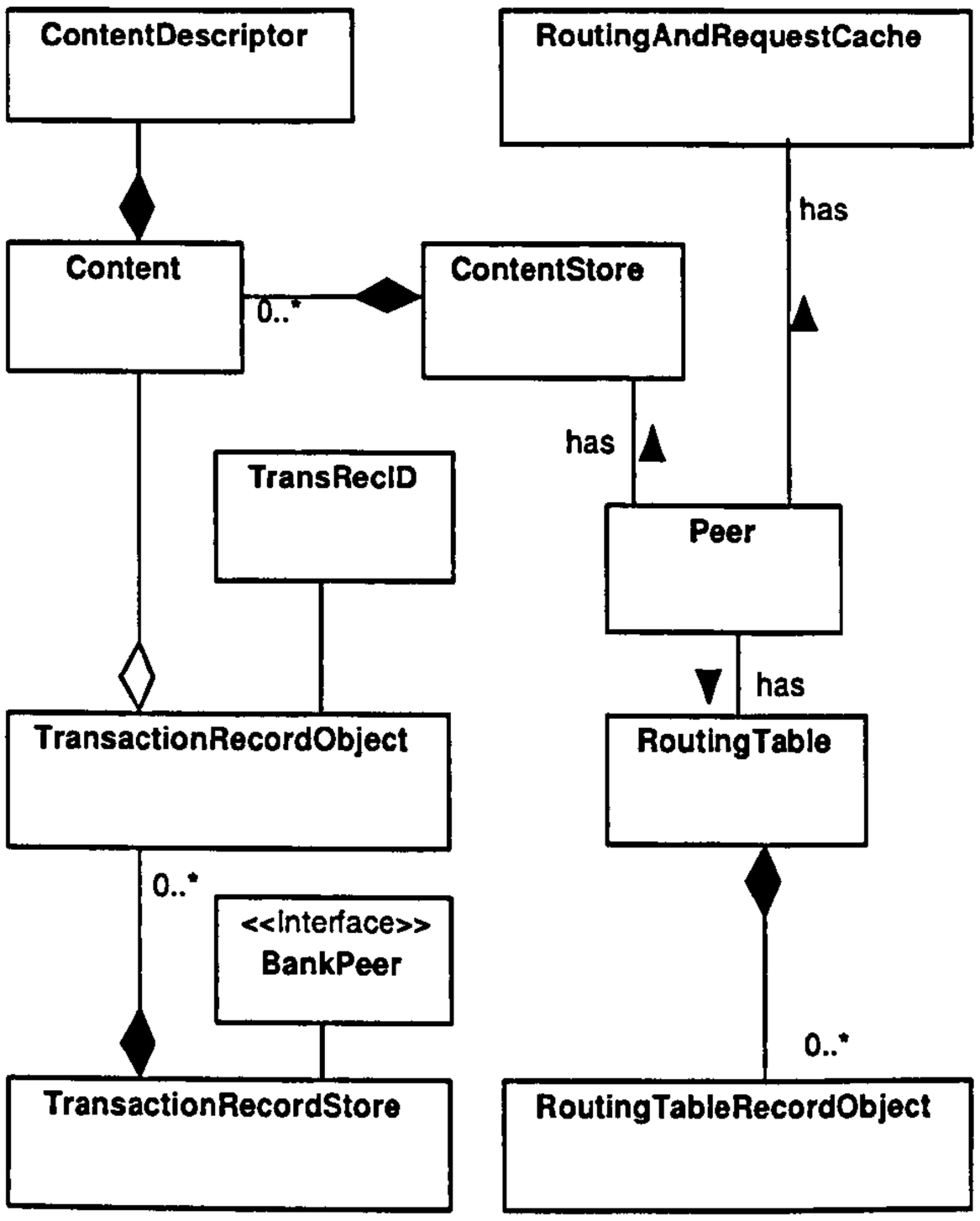


Figure 4.21: Class diagram - Data and the different data stores

This framework gives peers in a P2P system the ability to exchange digital content for payment in a secure fashion, assisted by an overlay network of peers acting as transaction authenticator, providing the ability to validate payments and act as trust enabler. The framework also ensures the accurate distribution of payments to the appropriate peers.

This Chapter discussed the various sub-components of the CasPaCE framework, i.e. its services and protocols, in greater detail. The services and their protocols were designed based on the functional requirements gathered via the use of the UML. This allowed us to separate the functionality of the framework into component services. The design decisions taken to overcome the technical issues of this problem domain were discussed here. These design decisions assisted in the formulation of the UML data and behavioural model which captures the system design and is illustrated in detail in Appendices A-D.

In the next Chapter we shall present the implementation and testing of a prototype of this framework, which shall enable us to evaluate the framework in Chapter 6.

Chapter 5

5 IMPLEMENTATION AND TESTING

This Chapter describes the implementation of a prototype of the CasPaCE framework as per the design laid out in the previous Chapters. Chapter 4 detailed the design decisions made to fulfil the requirements as set out in Chapter 3. This Chapter will describe the various tools used for implementation of the prototype and the author's experience of using them along with a description of the prototype implementation and the testing undertaken.

5.1 Implementation considerations

There are a number of initiatives that provide protocols for P2P communications such as Gnutella [Distributed Search Solutions 2001], Pastry [Antony Rowstron 2001a] and Chord [Ion Stoica 2001], these protocols essentially provide lookup and discovery functionality for P2P networks. Since our system is designed to address the need for paid content delivery in the P2P domain we needed more than the basic P2P routing protocols for our implementation. The JXTA[30] framework is ideal for our scenario because it allows developers to extend its functionality. This allows us to implement our prototype without having to deal with the low level implementation of connection, routing and discovery protocols as well as providing us with the base services required to implement content exchange in a P2P network. JXTA handles the management of peers and their connections in the system. Since it is primarily developed within an open source community, all JXTA APIs are readily available for modifications. On the other hand the use of such a young technology has its disadvantages in the teething stages where it is more difficult to find material help as opposed to the easy availability of assistance in more established technologies. For instance, a common problem encountered was that the release of newer JXTA libraries resulted in deprecation errors as well as having to re-write code to conform with the newer class libraries to get maximum benefit. Although the new library releases fixed previous problems they invariably caused more issues with our implementation.

---

[30] JXTA, Project JXTA: Java™ Programmer's Guide, Sun Microsystems, Inc., Available at http://www.jxta.org/docs/jxtaprogguide_final.pdf

Our system is implemented using JXTA's P2P protocols suite, as opposed to the use of Microsoft's .NET framework, primarily because JXTA is written to be platform independent and being open source we have access to its source code which made it easier to modify and extend to suit our needs. Finally, JXTA was also considered an appropriate choice because its fundamental principles are based on the P2P paradigm and hence closely fulfil our needs for decentralised P2P networking.

Although, we are using JXTA's P2P protocols suite, in particular the Resolver Service, for the implementation of routing and discovery in our system prototype, our framework does not exclude the use of other P2P routing protocols thereby demonstrating its flexibility. So in essence any of the other routing protocols such as Pastry, Gnutella or Chord could be used for the purposes of routing and discovery implementation within this system.

5.2 Implementation

The system implementation was carried out using the Java SDK 1.4.2 within the Netbeans IDE version 3.6 development environment. JXTA API version 2.3 was used to implement the P2P interface. The following sections shall describe how the various components of our framework were realised to create a Paid Content Exchange Prototype application using a combination of these tools and technologies.

5.2.1 JXTA basics and our framework

The main objective of this research is to provide a mechanism to overcome copyright violation in the P2P environment by compensating owners for their intellectual property rights (IPR) and providing participation incentives for intermediary participants in the content value chain. As mentioned in (Chapter 3 pg 33) the framework relies on the use of core Peer services like Discovery and Lookup to handle communication between peers. These services use routing tables and caches to store information to assist with communication and routing. On the other hand services such as the Bank Service, Payment Service, Security Service and Content Exchange Service (CES) can be available in the P2P system to be located and used by all peers.

The essence of our design uses the Lookup and Discovery services to locate content and purchase it. The method for inserting content into the system is via advertisements where once the content has been created and its copyright embedded the content is stored on the local peer or remote peer (dependent on the system) and then is made available to be picked up in search queries. Using JXTA to implement this prototype allowed us to use the *Advertisement* entity to

register the content and facilitate the search, where every element in the system is located via its corresponding advertisement. In our implementation approach we used a combination of entity Advertisements and their publication techniques to realise our prototype. These advertisements had to be extended to suit our needs because in many cases the JXTA API was too restrictive, for instance JXTA's standard advertisement suite did not include information specific to our needs such as copyright information for cascading payments. Similarly, peer specific information such as a peer's ID was accessible through the *Peer Advertisement*. However because our Peer ID is more specialised and is one of the main building blocks of our system we had to embed the CasPaCE Peer ID explicitly in other entities apart from the *Peer Advertisement* to conserve resources for Peer Advertisement lookups as well as to facilitate easier access to peer information such as its Public Key.

Peer services such as the Content Exchange Service, Bank Service, Payment Service and Security Service were implemented using Java. These services exist in the same plane as the Lookup and Discovery services, i.e. the Peer Services plane, however they are specifically required within our framework to realise our research objectives. These services provide the essential building blocks for the implemented prototype.

The following sub-sections will describe general implementation details of these services as well as how they fit into the overall prototype implementation for the ElConE case study (which will be the subject of the next Chapter) and assist in the evaluation of the complete system.

5.2.2 Discovery, lookup and advertising services in CasPaCE

To implement our discovery and lookup services we extended JXTA's *Discovery Service* and *Resolver Service*. In the JXTA API the *Discovery Service* is used to publish advertisements as well as to discover published advertisements (Figure 5.1). There are advertisements for each type of entity in the JXTA protocols suite. However, these advertisements do not meet our requirements and had to be extended for our framework implementation. In our framework the CasPaCE Discovery Service performs the routing functionality for the system and the Lookup service performs the actual mapping between resource and its address which has similar objectives to the *Resolver Service*.

For our prototype we had to extend the advertisements to create new advertisement classes. For instance the CasPaCE enabled content has to have its copyright and compensation information advertised before it can be purchased so the contentDescriptorAdvertisment was written. This advertisement type contains the *CasPaCE content descriptor* which contains the content owner

```
public void discoverBanks(DiscoveryService discovery){
while (true) {
   try {
     en = discovery.getLocalAdvertisements(DiscoveryService.ADV
                                          , "Name"
                                          ,"CasPaCESPEC:BANK");
                if ((en != null) && en.hasMoreElements()) {
                 break;
                }
                discovery.getRemoteAdvertisements(null
                                          , DiscoveryService.ADV
                                          , "Name"
                                          , "CasPaCESPEC:BANK",5, null);
                try { Thread.sleep(5000);
                } catch (Exception e) {}
            }
            catch (IOException e) {}
            System.out.print(".");
        }
}
```

Figure 5.1: Discovery of a service based on its advertisement

ID (including public key), seller ID (including public key), royalty, commission and other content specific information such as the content's unique ID (md5 hash), its size and filename as specified by our design. Figure 5.2 illustrates how the Owner's Peer ID (UPI) is added to the contentDescriptorAdvertisement to create an XML structured document with a nested OwnerPID document. The same method is used to add the additional content descriptor information to the content's advertisement. Figure 5.3 illustrates the resultant advertisement for content 'DIY.doc', written in the XML. Similarly, other entities in the system were advertised for discovery.

In JXTA, the methodology used for forming connections between nodes is via the use of Pipes, which have endpoints to connect the two nodes. The advertisements contain references to the node's pipe advertisement which includes the address of the endpoints. By extracting and binding to these pipe parameters a peer can connect to another peer. This is one method used to exchange information with other peers. We only use this method where direct communication between two peers is required in a transaction and when services are being used remotely. For general management of the system in our prototype we used the broadcast approach, where we discover the peers and then send a general message to them all. If a peer is configured to handle that type of query it processes it and responds to the origin else the query is passed on by the

```
StructuredDocument docMetaOPID =
      StructuredDocumentFactory.newStructuredDocument(new
      MimeMediaType("text/xml"),"metadata",readinOwnerID);
Element elSchemeOPID =(Element)docMetaOPID.createElement("scheme",
                                    null);
docMetaOPID.appendChild(elSchemeOPID);
elSchemeOPID.appendChild((Element)docMetaOPID.createElement(
                                    "name"
                                    ,"OwnerPID"));
elSchemeOPID.appendChild((Element)docMetaOPID.createElement(
                                    "locate on"));
elSchemeOPID.appendChild((Element)docMetaOPID.createElement(
                                    "content-type"));
ContentMetadata[] mdata = new ContentMetadata[6];
mdata[0] = metaOPID;
```

Figure 5.2: Adding content metadata to describe content using XML

Lookup service to the next peer in its routing table. This methodology is mainly used for locating content in the network. The *search query handler* then looks in its local content store for content matching the desired query and sends a response back to the query propagator.

The main problem encountered with Discovery was that there is no method in the standard API which gives direct access to the local cache to check whether it is empty or not. Similarly there is no method to directly access an advertisement found remotely, so the implementation has to fool the system by reiterating the *getLocalAdvertsiments()* and *getRemoteAdvertisments()* methods in such a way that if a desired advertisement is found remotely it has to look in the local cache again to pick it up by calling the *getLocalAdvertisment()* method.

Three methodologies were used to implement the communication between the various peer nodes. We utilised the JXTA *Pipe Service* interface which allows for the creation of dedicated communication channels between the peers. The Bank Service which has to regularly listen for incoming service requests and receive sensitive information from other Banks uses the Pipe interface which other peers can bind to and send information to. All Bank peers implement their own pipe advertisements with bank service specific information to bind to that pipe when communication with that bank is required. Since we have a Random Bank Peer Selection Algorithm in place we specify the bank the transaction participants can bind to but they all use the same bank Pipe and connect and disconnect from it as the situation requires.

```
<?xml version="1.0"?>
<!DOCTYPE jxta:ContentAdvertisement>
<jxta:ContentAdvertisement>
      <name>  DIY.doc  </name>
      <cid>    md5:dd498c832fe14ca88caedaf0f918112c  </cid>
      <length>   419  </length>
<metadata>
     urn:jxta:uuid-
5961626164616261 4A787461503250 33DF7AB751D2FB4F8E85DAEC5288D8941003
     <scheme>
          <name> OwnerPID   </name>
          <location/>
          <content-type/>
     </scheme>
     </metadata>
     <metadata>
   urn:jxta:uuid-
5961626164616261 4A787461503250 33EEFDE6C8BF044AF5A055B49151D89F7803
     <scheme>
          <name>      SellerPID     </name>
          <location/>
          <content-type/>
     </scheme>
     </metadata>
     <metadata>
          20
     <scheme>
          <name>      Royalty     </name>
          <location/>
          <content-type/>
     </scheme>
   </metadata>
   <metadata>
          5
     <scheme>
          <name>      Commission  </name>
          <location/>
          <content-type/>
     </scheme>
  </metadata>
    <metadata>
    <metadata>
¬í   •sr   -org.bouncycastle.jce.provider.JCERSAPublicKey%"j[úl„•    •L
•modulust    •Ljava/math/BigInteger;L    publicExponentq    ~    •xpsr
•java.math.BigIntegerŒüŸ©;û•• •I •bitCountI    bitLengthI
```

```
•firstNonzeroByteNumI lowestSetBitI •signum[    magnitudet          •[Bxr
•java.lang.Number†¬••"à‹•   xpÿÿÿÿÿÿÿÿÿÿþÿÿÿþ     •ur •[B¬ó•ø••Tà•   xp
@%!®õI®•aq^h      $@˜»•ÒKµ¶M4•¦¦ÁÌŠ?>x¨üàhEIQoŸSe½"±Ó½8 h¨%p‰m'"[—
Á•xsq ~ ÿÿÿÿÿÿÿÿÿÿþÿÿÿþ   •uq ~ •    •• •x
      <scheme>
            <name>OwnerPubKey </name>
            <location/>
            <content-type/>
      </scheme>
   </metadata>
   <metadata>
      <scheme>
            <name>    SellerPubKey      </name>
            <location/>
            <content-type/>
      </scheme>
      </metadata>
```

Figure 5.3: Content Advertisement with content descriptor

The other methodology used is the *Resolver Service* directly instead of using the higher level and more abstracted *Discovery Service* interface. The benefit of using the *Resolver service* interface in certain scenarios is that it allows us to embed our data payload directly into a message and route that message directly to the peers without the need to use repetitive code to access and manipulate the message at each stage of its transit. It also gives us access to the underlying query routing protocols. By extending the *Resolver Service* to create our Lookup Service we were able to register peer specific handlers which enabled direct communications between peers. However in other scenarios it is easier to extend the JXTA *Discovery Service* to abstract away unnecessary lower level information from our prototype development as well as to gain access to the advertisement lookup features. When implementing our payment pushing protocol the low level detailing of the *Resolver Service* allowed us better access to routing information.

5.2.3 Content exchange, payment, bank and security services

This section discusses the implementation of the underlying components that make up the content exchange, payment, bank and security services and the issues addressed in their implementations.

Our Content Exchange Service (CES) extends the CMS libraries to use their base XML handling libraries, however since the base libraries are restrictive in their implementation, the

XML handlers had to be extended to grant us direct access to the Content advertisements. By doing this we were able to create Content Advertisements which included the CasPaCE *content descriptor.*

5.2.3.1 Unique peer identification

JXTA has a concept of unique resource ids (URI) where every entity in the system is allocated a unique id. Our UPI (Unique Peer ID) is implemented by creating a Peer ID object which includes the URI for the peer ID, the peer's public key and a date time stamp. In order to insert the public key into the UPI, the public key has to be serialised and inserted as an element in a StructuredTextDocument. This UPI is created when the peer first registers with the system and persists for the duration of the Peer's lifetime in the network. This persistence is guaranteed in our prototype by storing the UPI in a local directory. One of the challenges encountered in this implementation was to embed the Public Key into the Peer ID to create our UPI, ensuring that the integrity of the Public Key was maintained so that it could be extracted and used in the subsequent processes. Hash values of the username and password are stored on the local device as part of the Login object, to ensure they can be modified. This interface is accessible via the CasPaCE prototype application.

5.2.3.2 Identity embedding

The content descriptor for the content acts as its watermark, this is written using XML. The use of XML has a twofold benefit. Firstly, the XML object can be exchanged via the JXTA platform since JXTA has legacy support for XML as the fundamental advertisements and messages exchanged use XML. New XML data structures can be easily embedded within JXTA messages by using the StructuredTextDocument class which allows the embedding of elements with attribute and values. Secondly, XML is widely accepted as a standard for interoperability hence XML structured documents can be used to exchange information between heterogeneous systems which conforms to our requirement for interoperability in the P2P domain.

Once the content and its contentDescriptor have been created, it has to be made available for purchase by insertion into the network. In our prototype, the content is placed locally however it is advertised both locally as well as remotely so that other peers can discover it. For this the *contentDescriptor* (Figure 5.4) was advertised by using the Discovery Service's *publish()* and *remotePublish()* methods.

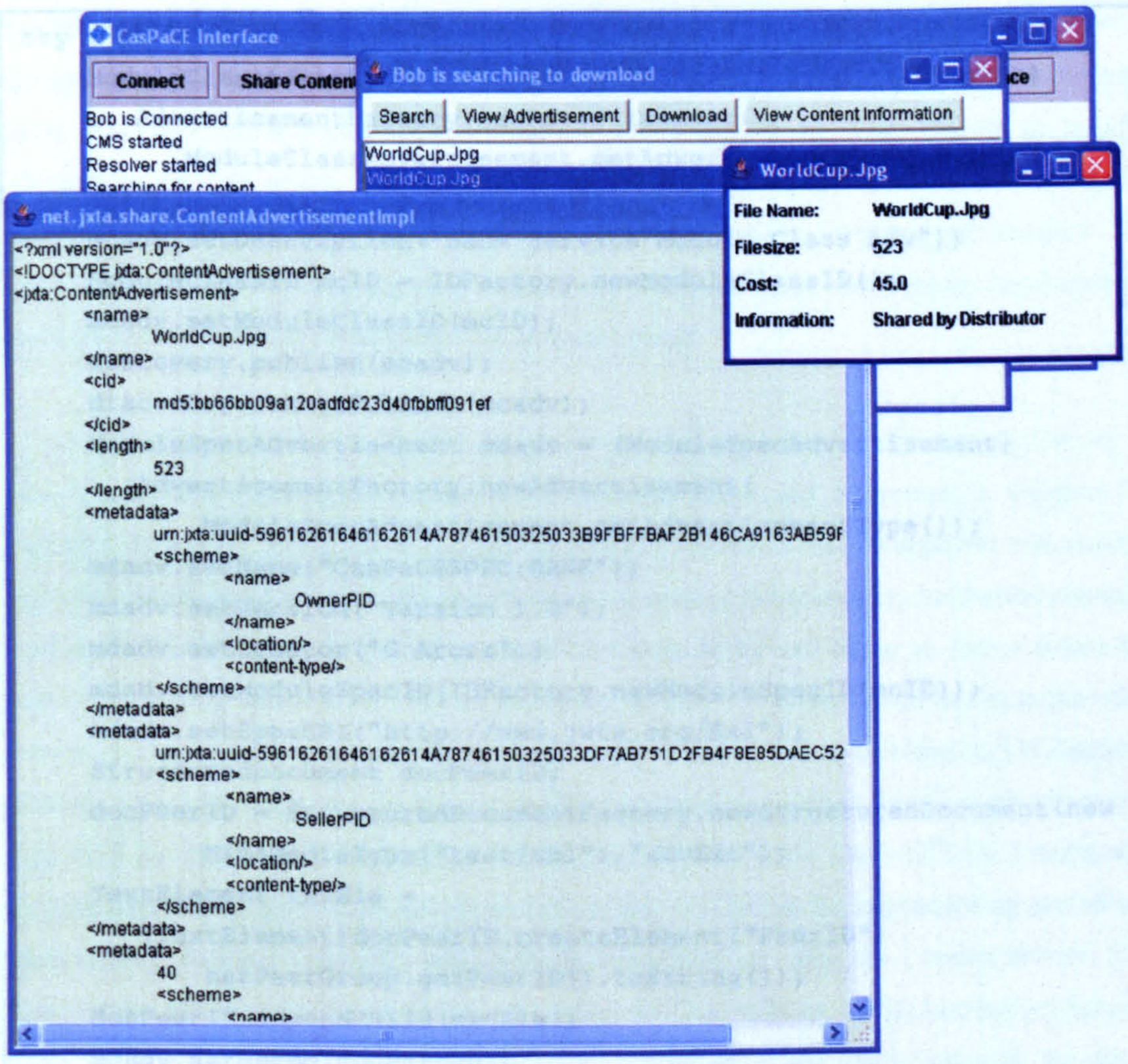


Figure 5.4: Content Descriptor information

To advertise an entity we first create a structured document with MimeMediaType as XML, and then cast the structured document into a Module Specification Advertisement (Figure 5.5). This advertisement is published using the *publish()* method for local publishing and *remotePublish()* for publishing at other nodes within the network. The same method is used to advertise other entities within the system such as content, peers and other peer services (e.g. Bank Service).

The *contentDescriptor* advertisement has a pipe advertisement, which is how the buyer connects to the seller to buy the content. However, this method was later discarded because it meant that the *contentDescriptorAdvertisements* had to be created every time the peer rebooted, so the pipe advertisement was sent once the buyer peer made a *PurchaseRequest* ensuring the latest version of the pipe was available to the buyer for binding.

```
try {
    ModuleClassAdvertisement mcadv = (ModuleClassAdvertisement)
       AdvertisementFactory.newAdvertisement(
          ModuleClassAdvertisement.getAdvertisementType());
    mcadv.setName("CasPaCEMOD:BANK");
    mcadv.setDescription("Bank Service Module Class Adv");
    ModuleClassID mcID = IDFactory.newModuleClassID();
    mcadv.setModuleClassID(mcID);
    discovery.publish(mcadv);
    discovery.remotePublish(mcadv);
    ModuleSpecAdvertisement mdadv = (ModuleSpecAdvertisement)
       AdvertisementFactory.newAdvertisement(
          ModuleSpecAdvertisement.getAdvertisementType());
    mdadv.setName("CasPaCESPEC:BANK");
    mdadv.setVersion("Version 1.0");
    mdadv.setCreator("G Arora");
    mdadv.setModuleSpecID(IDFactory.newModuleSpecID(mcID));
    mdadv.setSpecURI("http://www.jxta.org/Ex1");
    StructuredDocument docPeerID;
    docPeerID = StructuredDocumentFactory.newStructuredDocument(new
          MimeMediaType("text/xml"),"advExt");
    TextElement txtEle =
       (TextElement)docPeerID.createElement("PeerID",
          netPeerGroup.getPeerID().toString());
    docPeerID.appendChild(txtEle);
    mdadv.setParam(docPeerID);
    } catch (Exception e) {
    System.out.println("failed to read/parse pipe advertisement");
    e.printStackTrace();
    System.exit(-1);
    }
   discovery.publish(mdadv);
   discovery.remotePublish(mdadv);
```

Figure 5.5: Creating a Service Advertisement

5.2.3.3 Payment separation and distribution

Initially the payment separation technique was prototyped through the use of text files which represented the payments. Here the commission and royalty values were extracted from the XML *contentDescriptor* along with the owner and distributor UPIs. These amounts were written into the royalty and commission payment files along with payee and payer specific information. These files acted as a token form of payment instrument. Standard Java file input and output streams were used to create files to represent royalty and commission, which were encrypted and decrypted locally without any errors. These files were encrypted using the RSA asymmetric

key generator, which includes the Bouncy Castle RSA key provider as part of the JCE. One of the main implementation requirements was to serialise objects before transferring them between peers. Although JXTA's messaging techniques allow the use of XML objects to transfer information between nodes, and XML allows embedding of binary data. When transferring encrypted payments in this way we encountered errors, mainly involving serialisation. We solved this problem by using payment objects which did not cause serialisation/deserialisation conflicts as they had their serialisable objects well defined.

In our ElConE case study we decided to compensate the owner and distributor via the use of these tokens, which are signed receipts. As mentioned earlier, our framework is designed for maximum flexibility, so the use of this type of payment instrument can be replaced with another payment instrument like E-cash. For the exchange of routing information to implement *payment pushing*, Gnutella's protocol probably would have been better and easier to use to access the underlying routing tables as it is not as abstracted as JXTA's. However, in this case the other network management functionality would have required more effort to implement. The Gnutella protocol primarily relies on *push()* and *pull()* functions to perform routing and discovery supported by caches to store routing information. On the other hand JXTA's routing and discovery is abstracted from the developer and enables ease of use for implementing specialised functionality. *Payment collection* was easier to implement by using the Lookup service. The Payment distributor in the Payment Service queries the CasPaCENet, via its Lookup service, for advertisements advertising royalty payments for the payee. If the Payee value in the query message matches the payee field of the payment advertisement, the owner creates a socket with the delivery location and 'collects' the royalty.

5.2.4 Bank Peer Overlay network implementation

The bank peer overlay network was developed using Java. Since the concept of this network is virtual, bank peers are peers who run a bank service. The exclusive communication between the bank peers creates the 'overlay' network. Peers running the bank peer service are members of a peer group which is a subset of the default NetPeerGroup. Messaging between these peers is conducted using a combination of the *Resolver Service*, *Discovery Service* and the *Pipe* interfaces (as described on pp. 112) however bank specific messages are only sent to members of the bank peer group using the *Resolver Service*.

5.2.4.1 Transaction management

As both buyer and seller send the bank request to the bank, this ensures that if there are multiple transactions between the same two peers there is no conflict in the transaction record and also ensures that both peers agreed to the transaction and are online at present. When the first request is received the bank starts querying the Bank Peer network while awaiting the second request. At the design stage the bank only started the BP network query once the Seller's request was received, however at testing it was discovered that dependent on network conditions this could hang a transaction indefinitely and act as a bottleneck, so in the implementation either party's Bank Request initiates the Bank Service's Transaction Manager. The Buyer and Seller are sent the unique transaction ID which is the medium of access to the transaction record and all its subsequent updates for the duration of that transaction.

5.2.4.2 Random Bank Peer Selection

We developed an algorithm using Java, to randomly select a bank peer from the bank peer network. When a buyer authorises a purchase, his local peer generates a PurchaseRequest which includes information about the content to be purchased, the peer's details and a set of bank peer ID's known to the buyer.

The first step in this sequence is the bank peer selection algorithm initialisation. The Buyer peer performs a *discoverBanks()* operation, generating a set of Bank peer ID's it knows called the *conBPSet.* This knowledge set is based on a combination of locally cached bank peer advertisements as well as discovered bank services. It starts the lookup for bank peers in its local cache, if no bank peers are found; it looks for bank peer advertisements remotely. Once a desired number of advertisements are found (for the implementation of our prototype we have limited the lookup to five lookups at a time) the peer performs an *extractBPConnectedSet()* operation to extract the bank peer ID's from the returned results and creates a set of bank peer ID's. This conBPSet is serialised and sent to the seller peer as part of the *purchase request* along with the attributes of the content to be purchased.

Only the set of bank peer ID's is sent between the buyer and seller. This was an implementation decision taken to reduce the size of messages between peers. Once the bank peer has been randomly selected, it is contacted individually by both parties. Once the connected sets have been exchanged the Seller performs the intersection on the two sets and random BP selection. The resultant selected Bank Peer ID is returned to the Buyer while the Seller contacts the selected Bank peer.

In the initial implementation the selected bank peer's advertisement was discovered based on its randomly selected peer ID. Then the bank peer's pipe advertisement was extracted from its peer advertisement and used to contact the bank peer. In this case a direct connection was initialised with the bank peer using its pipe. However, during unit testing there were problems when this implementation was tested over a busy network. The problem was being caused either when attempts to establish the connection failed after a set number of attempts finished, or they occurred when a connection was established successfully but the messages being received were older messages from other simultaneous transactions on the same peer. Although the use of the bank peer ID's proved to be beneficial in reducing the number of failures, it still required the use of bi directional pipes.

The use of BiDipipes removed the negative results due to older messages being picked up by the bank peer. However, it resulted in an extra overhead on the bank peer, which now had to also ensure it was connected to the buyer and seller. Secondly, the delay in an active transaction was increased if the peers missed the preset limit of connection attempts. To overcome this problem the resolver service was implemented for bank peers as well and messaging between bank peers and sellers or buyers was done using the resolver service with specific peer IDs specifying the destination of the messages.

5.3 The CasPaCE prototype

This section introduces the configuration and user interface of a prototype of the CasPaCE framework which allowed us to test the functionality of the framework within real world scenarios for content exchange.

5.3.1 Prototype configuration

Currently the prototype allows for location of digital content based on keywords which are substrings of the content name and a value for the content. Other information such as connection speed, estimated time of arrival (ETA) and peer specific information is also available however the user interface to access this information has not been developed. Multiple instances of the prototype can be executed on the same device.

5.3.2 The user interface

The CasPaCENet application User Interface, the CasPaCE Interface (Figure 5.6), is a Java Swing application. The Java Swing package was chosen over the standard AWT package

because it promotes the development of a more aesthetic and pleasing look and feel for the user interface. It is also more intuitive for users because it is comparable to the look and feel of the platform it is being run on, thus promoting user acceptability. The interface has a basic design which allows the user to input a search 'keyword' and returns a list of search hits with the purchase cost. The user can then select a document for download which will initiate the PayS protocol followed by the PayD protocol. The user is informed when the transaction has been completed successfully. The content owner and the intermediary are informed when their payments (royalty and commission respectively) are received.

The 'Search' algorithm searches on the basis of keywords in the *content descriptor's* filename, cost and owner fields in a Content advertisement. The search object is passed to the Peer, who uses the CES to locate content in the network.

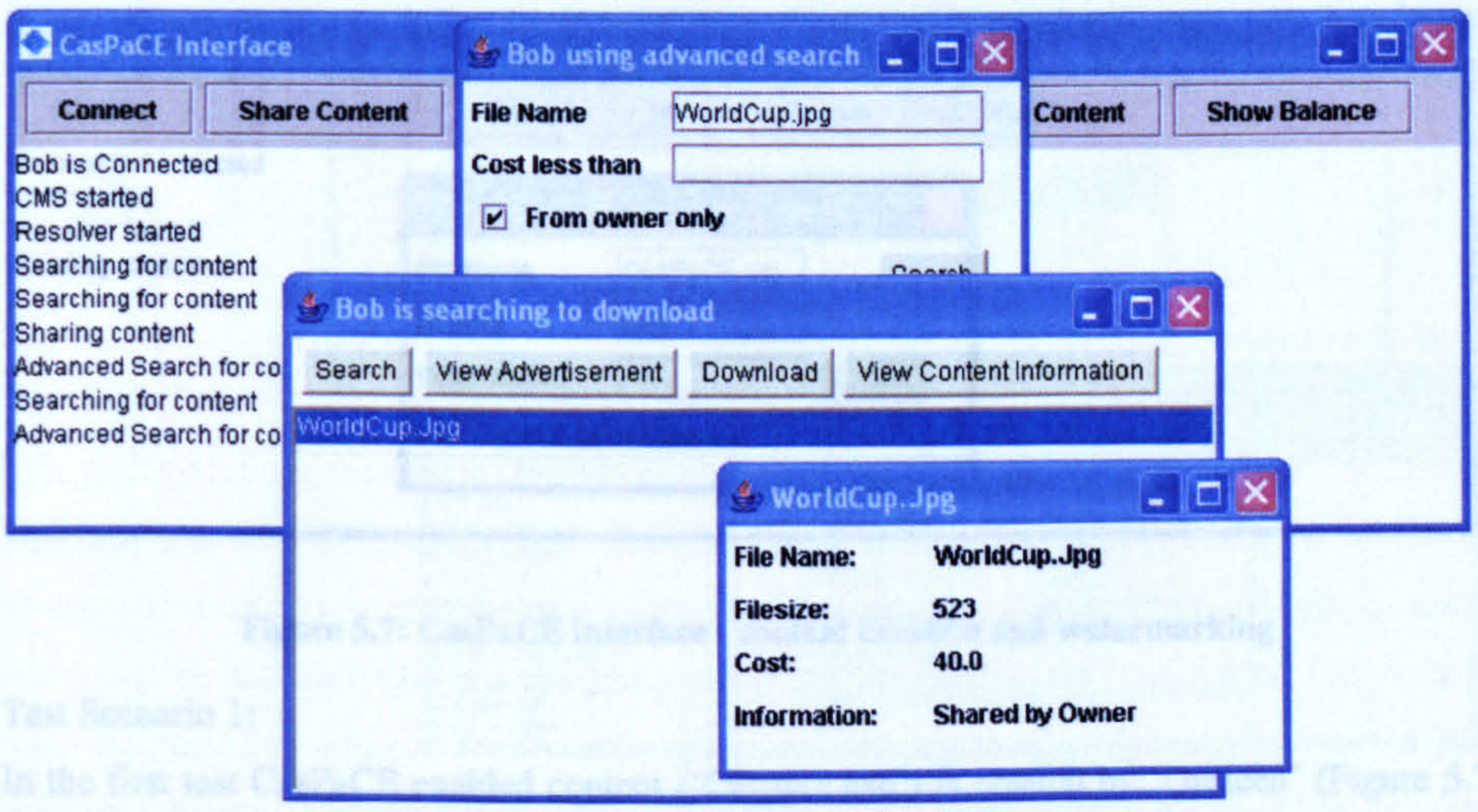


Figure 5.6: Search in CasPaCENet GUI

The interaction with the bank peers is transparent to the user and hence is not reflected in the user interface. Although for testing purposes a simple message display format was used to inform the author of successful process completions. Data on the working of the prototype was also collected in peer logs generated by using the Java Logger API.

5.4 System and integration testing

The system was implemented in a bottom up manner where system components were implemented and tested in parallel. Once all the underlying components had been implemented integration testing was performed where the various components were linked to the relevant

GUI and the sequence of events was tested to make room for network delays as well as testing for system performance issues. Tests were conducted on local machines running multiple peers as well as by creating a network with peers distributed on multiple devices across a network composed of over 100 PCs.

Two test scenarios were developed to assess the CasPaCE framework. The first scenario was designed to test the working of the cascading payments model. The second scenario demonstrates the working of the CPM in conjunction with the bank peer network where a bank peer is randomly selected and assists in the transaction by acting as a third party and providing non-repudiation for the transaction. In order to perform these tests a number of normal peers (Alice, Bob, Joe etc.) were launched along with a set of bank peers. These peers constituted the CasPaCENet prototype test bed.

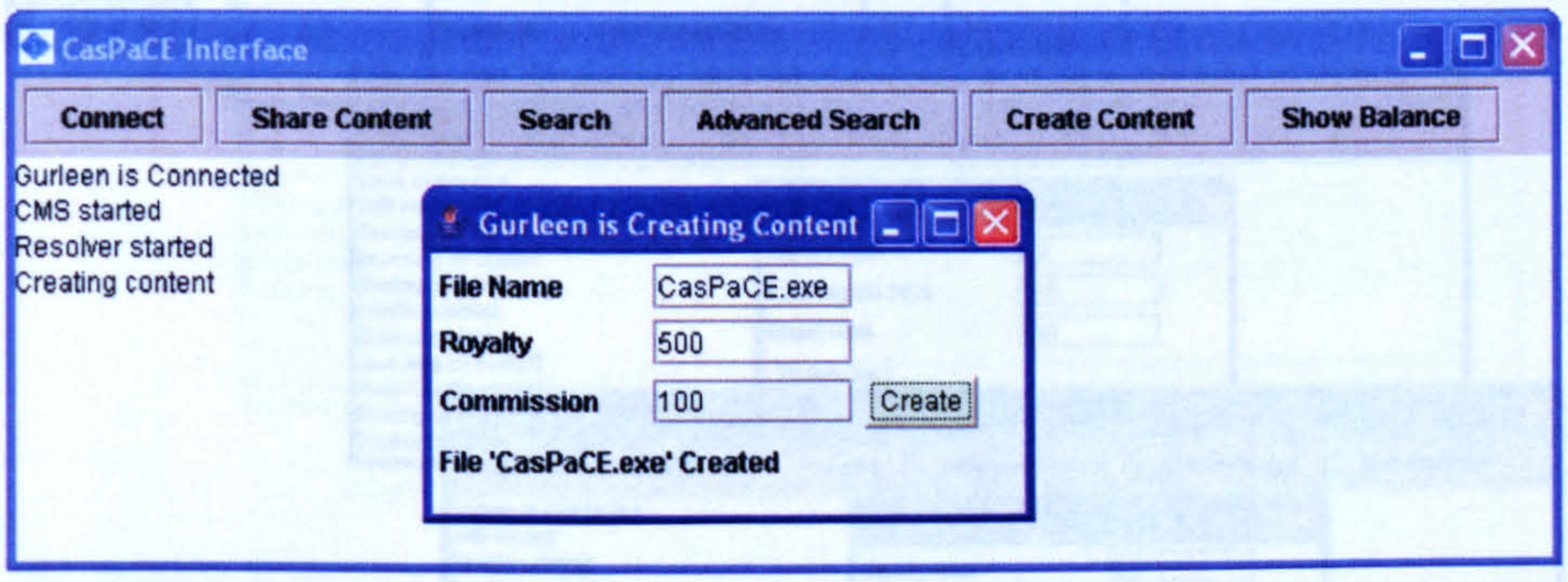


Figure 5.7: CasPaCE interface - content creation and watermarking

Test Scenario 1:

In the first test CasPaCE enabled content ('Caspace.exe') is created by 'Gurleen' (Figure 5.7), containing the copyright information where total cost is 600 units, and then inserted into the CasPaCENet. In this test scenario the commission parameter is inserted by the creator to demonstrate the functionality in the framework where commission policy can be setup by the owner. This content is discovered and purchased by 'Joe' for 500 units (Figure 5.8). 'Joe' reinserts into the CasPaCENet with relevant distributor information. On a subsequent search by 'Bob' for the same content ('Caspace.exe') multiple results are returned (Figure 5.8). When 'Bob' purchases this content from its distributor 'Joe', 'Gurleen' receives compensation for her IPR (500 units) and 'Joe' receives a commission of 100 units (Figure 5.9). Thus demonstrating that the payments cascade.

This test demonstrates that CasPaCE enabled content can be created and inserted into the CasPaCENet. This content is watermarked with the content descriptor which contains the

copyright information. This activity can be performed seamlessly; however we have demonstrated it explicitly in our prototype GUI (Figure 5.7). In a real world scenario this functionality would be part of off the shelf content creation products like Word processors, video generators, and application deployment procedures. Figure 5.8 shows that 'Bob' has results from multiple sources including the owner and distributors of 'Caspace.exe'. 'Bob' can become a distributor by sharing 'Caspace.exe' and both its owner (Gurleen) and intermediary seller (Joe) have been compensated for their roles in the content value chain (Figure 5.9).

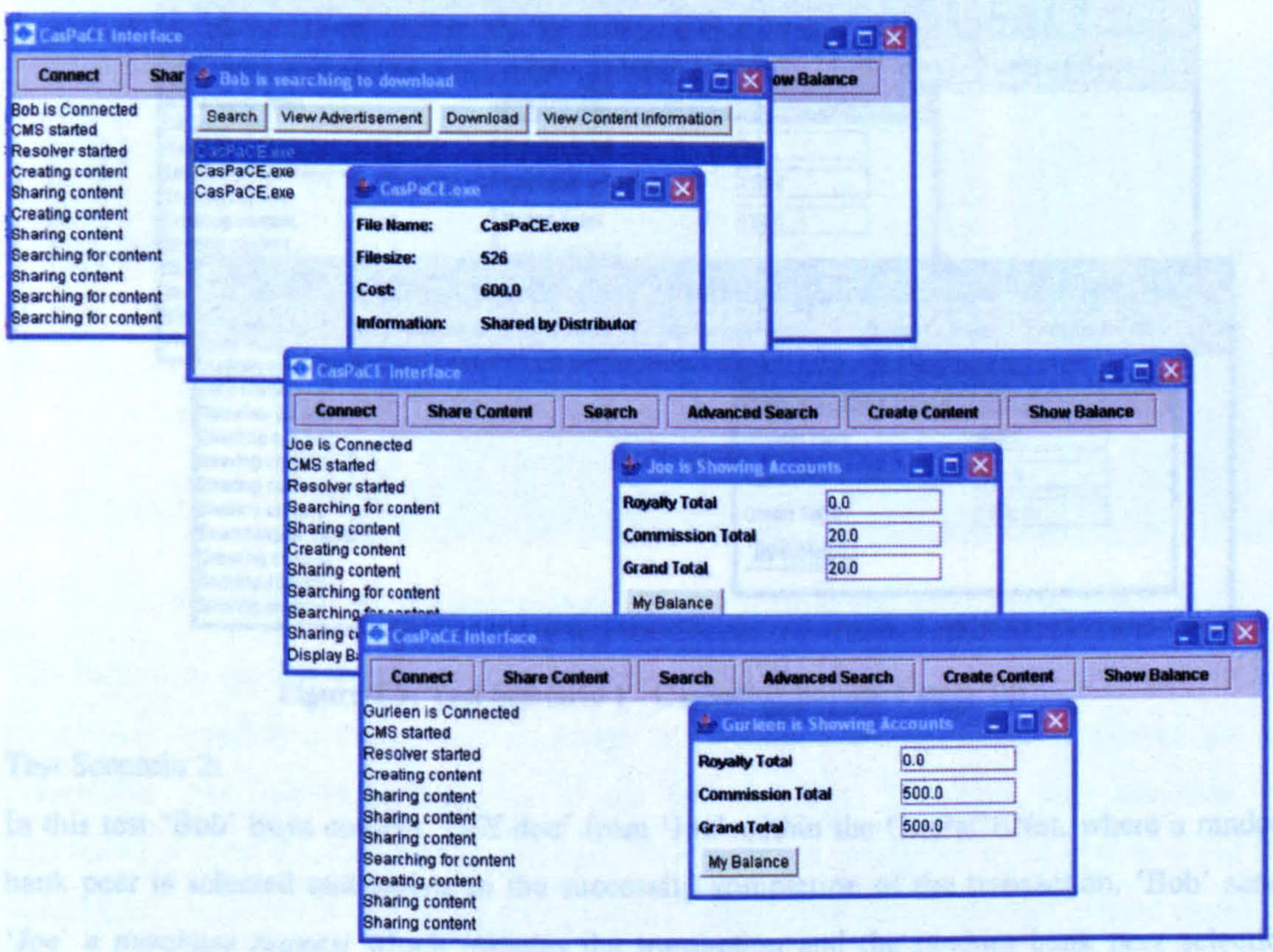


Figure 5.8: Test Scenario 1 - Cascading payment stage (i)

In our prototype the insertion of the content into the system is equivalent to it being advertised within the P2P network. Other implementation alternatives for content insertion could include content information broadcasts or content registration at nodes with nomenclature proximity[Ian Clarke 2001, Antony Rowstron 2001b]. This test proved the successful creation and insertion of CasPaCE enabled content. This content is discoverable along with its copyright information intact. Finally, this test demonstrated that payments can cascade from buyer to intermediary seller to owner successfully over a decentralised P2P network.

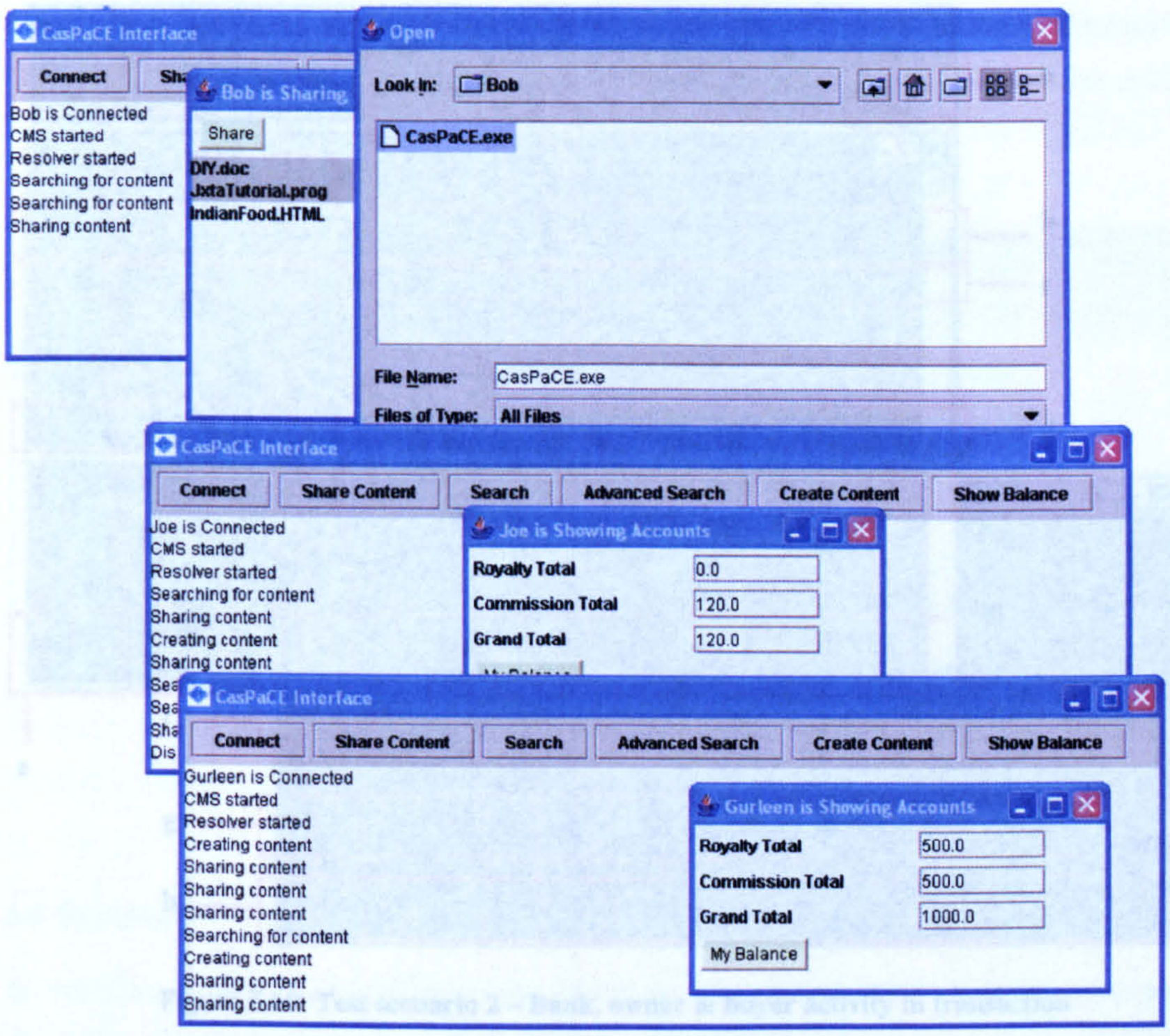


Figure 5.9: Test Scenario 1 - Cascading payment stage (ii)

Test Scenario 2:

In this test 'Bob' buys content 'DIY.doc' from 'Joe' within the CasPaCENet, where a random bank peer is selected and assists in the successful completion of the transaction. 'Bob' sends 'Joe' a *purchase request* which initiates the transaction and the random bank peer selection algorithm. 'Joe' randomly selects the 'Bank' (Figure 5.10 – a, f) from the set of connected bank peers. Both 'Bob' and 'Joe' contact the same 'Bank' (Figure 5.10- b, g) which manages the transaction via its Transaction Manager. The Transaction Manager ensures the fair exchange of content for payment by tracking and recording the transaction (Figure 5.11) until the payment and the content have been received at both ends (Figure 5.10 – d, h). Figure 5.10a illustrates the Bank Peer ID.

Figure 5.11 illustrates the persistence of the transaction record for this transaction, where transaction id is '25' (Figure 5.10c) and the same buyer and owner ids exist (Figure 5.10e). The persistence of the record is useful for non-repudiation of the transaction. This test demonstrates that a common bank peer can be randomly selected to assist with a content purchase transaction between two peers.

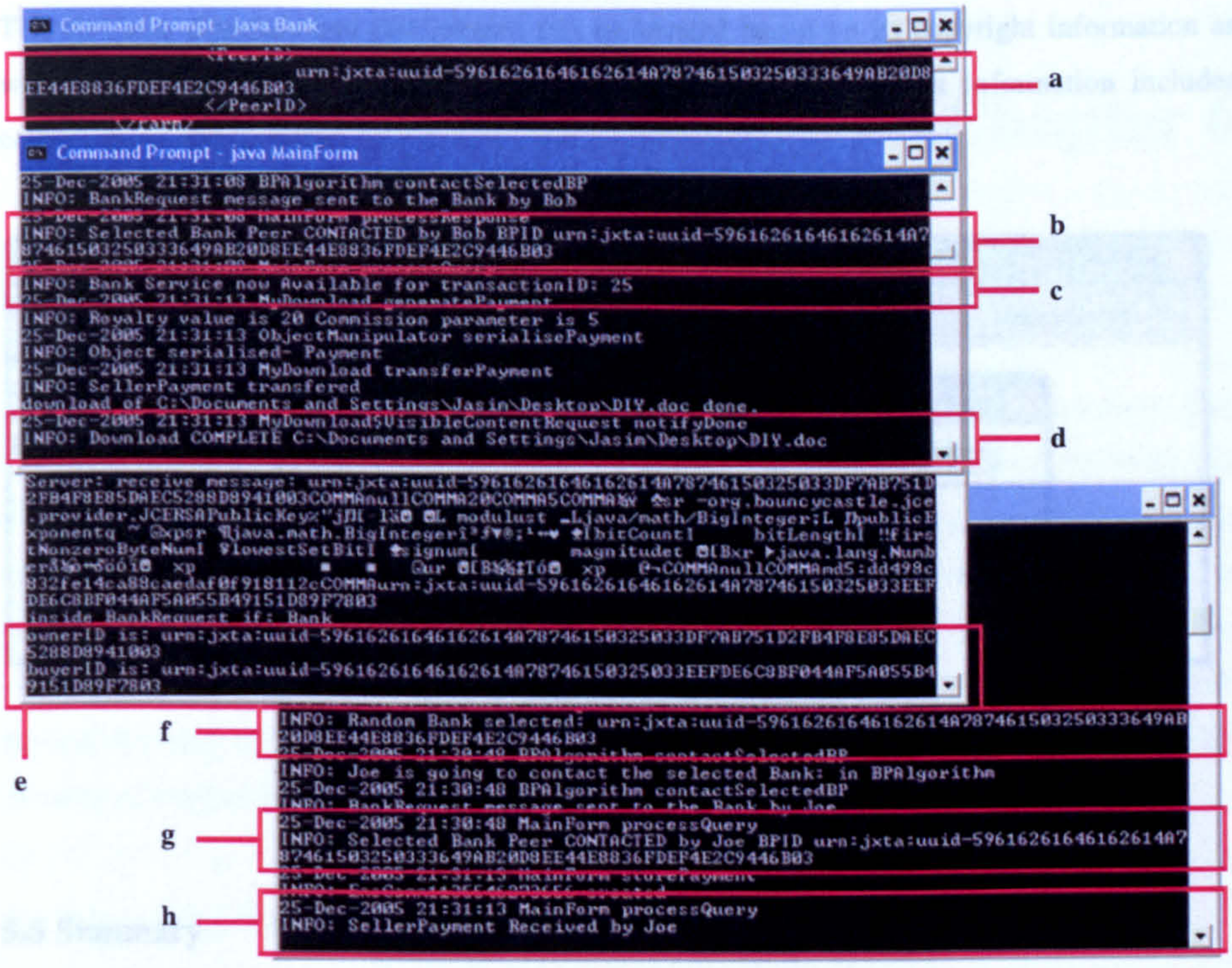


Figure 5.10: Test scenario 2 – Bank, owner & buyer activity in transaction

This bank peer records the progress of the transaction, if any party is made unavailable before transaction completion the content exchange is terminated and payments generated are not redeemable thus upholding requisites of the fair exchange. The bank peer selection and interaction process is transparent to the user. This is how it should be in the case of a real life transaction where one of the requirements for our system is to allow seamless content exchange between peers with minimum interaction with the user. The user is informed when payment is received and his accounts are updated.

tranID	bankID	contentID	ownerID	ir	buyerID	royalty	commissi
20	EE44E8836FDEF4E2C9446B03	md5:b84c12	E7C5AFAC041247BCAE8A8E3FD204048303	n	E7F78A96342B48FEB4D093054BA923DA03	753	951
21	EE44E8836FDEF4E2C9446B03	md5:3b9e53	A2DDB789F5E24F3A8C60F383E61E290A03	n	E7C5AFAC041247BCAE8A8E3FD204048303	45	20
22	EE44E8836FDEF4E2C9446B03	md5:7851efb	E7C5AFAC041247BCAE8A8E3FD204048303	n	E7F78A96342B48FEB4D093054BA923DA03	20	10
23	EE44E8836FDEF4E2C9446B03	md5:27b334	E7C5AFAC041247BCAE8A8E3FD204048303	n	E7F78A96342B48FEB4D093054BA923DA03	34	10
24	EE44E8836FDEF4E2C9446B03	md5:27b334	E7C5AFAC041247BCAE8A8E3FD204048303	u	A2DDB789F5E24F3A8C60F383E61E290A03	34	10
25	EE44E8836FDEF4E2C9446B03	md5:dd498c	DF7AB751D2FB4F8E85DAEC5288D8941003	n	EEFDE6C8BF044AF5A055B49151D89F7803	20	5
26	EE44E8836FDEF4E2C9446B03	md5:dd498c	DF7AB751D2FB4F8E85DAEC5288D8941003	u	EEFDE6C8BF044AF5A055B49151D89F7803	20	5

Figure 5.11: Test scenario 2 – Bank peer transaction record store

Tests were conducted to ensure that transaction integrity was not compromised when the same set of events occurred simultaneously. For instance, if a single bank peer was involved in multiple transactions the integrity of the other transactions was not compromised.

This test also demonstrates that content can be located based on its copyright information as well as substrings of the content name (Figure 5.12). The copyright information includes content value and its name.

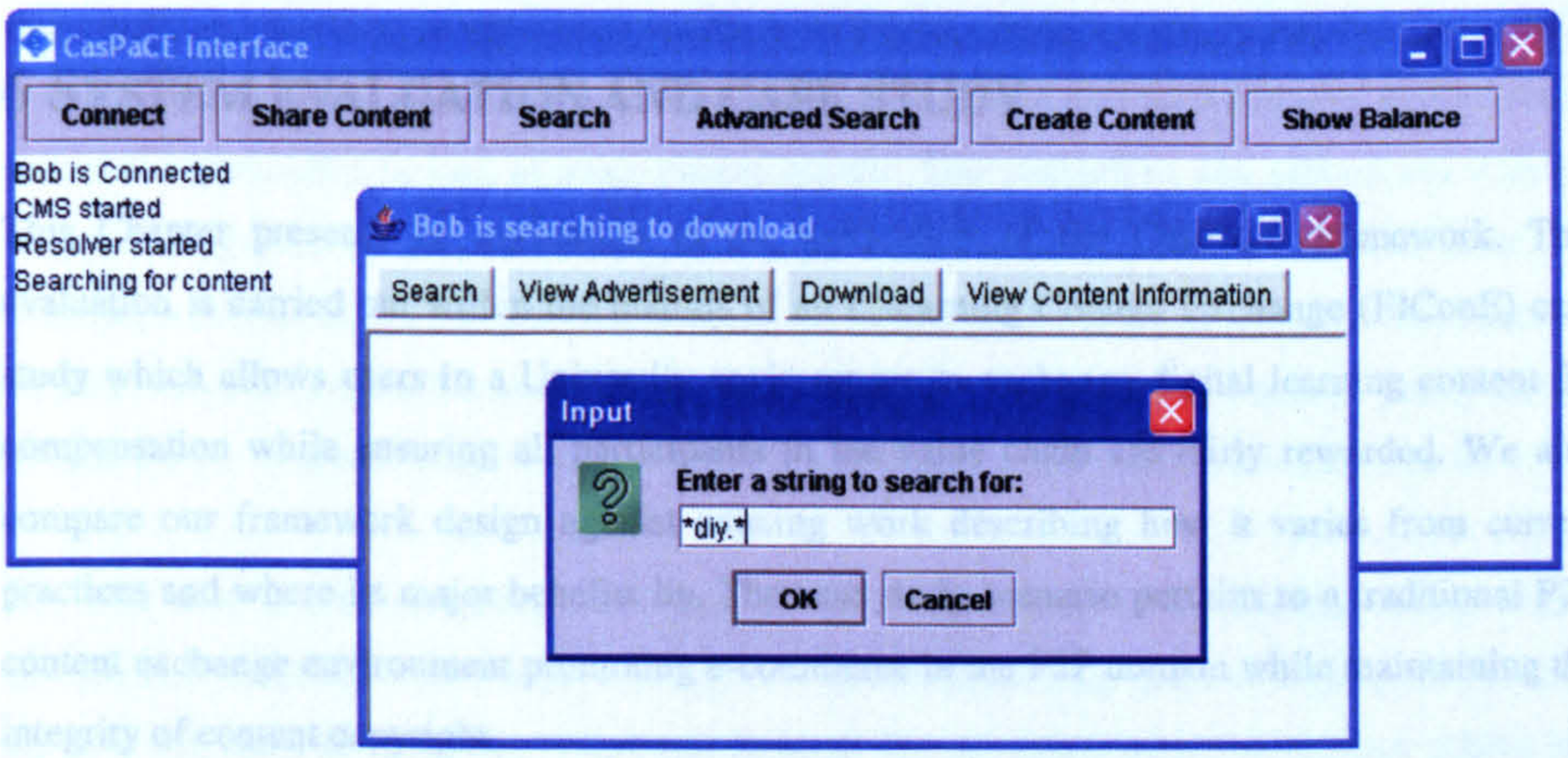


Figure 5.12: Test scenario 2 – search (substring)

5.5 Summary

In this Chapter we described the implementation of the CasPaCE framework prototype to demonstrate the working of our framework in a content exchange scenario. We used advances in P2P technology, which is devoid of any centralisation and the Java object-oriented programming language to implement our prototype. In our prototype we used JXTA's message propagation techniques to implement a P2P interface to illustrate our cascading payments model in a decentralised P2P environment. The prototype implementation gave a strong proof-of-concept of the framework design. This prototype helped us in evaluating our framework design which is explained in detail in the following Chapter.

We created a prototype test environment – the CasPaCENet, to assess our test scenarios. These test scenarios were described which illustrated our system and integration testing for the overall working of the prototype. These test scenarios incorporated various smaller tests to prove our concepts and working of the protocols we had designed in earlier Chapters. They also allowed us to extract data results pertaining to the performance of the system. The analysis of these results is presented in the next Chapter.

Chapter 6

6 SYSTEM EVALUATION AND CASE STUDY

This Chapter presents an evaluation of an application of the CasPaCE framework. This evaluation is carried out within the bounds of an E-learning Content Exchange (ElConE) case study which allows users in a University environment to exchange digital learning content for compensation while ensuring all participants in the value chain are fairly rewarded. We also compare our framework design against existing work describing how it varies from current practices and where its major benefits lie. The case study scenario pertains to a traditional P2P content exchange environment promoting e-commerce in the P2P domain while maintaining the integrity of content copyright.

6.1 Case study: E-learning Content Exchange (ElConE)

This section presents the ElConE case study which demonstrates the flexibility of our framework. This case study allowed us to test and evaluate our system against the requirements set out in the requirements analysis stage.

6.1.1 Scenario

Within this case study, the system can be utilised to exchange digital learning material for payment within a large UK University. Students have access to the content via machines that run the prototype application and may pay for material on a one off basis; they may also use the system to distribute their own content. As there are no physical banks present in the system, the notion of money is credit based, whereby each student has an account with the University that is debited every time they download information and credited when other students download information from them. Students top up their accounts with tokens when required via the University infrastructure. A fully operational system in this scenario would allow the University to move teaching material from a client-server infrastructure to a P2P network enabling efficient use of resources.

In this case study, learning material (content) was placed across a number of machines that had a P2P interface. These machines constitute the physical peers in the network and together with

the Content Exchange Application (CEA) form the environment referred to as the CasPaCE Network (CasPaCENet). In this case study the ElConE interface is the CEA which has access to content location, sale and purchase services. Peers within the CasPaCENet can search for content such as market research reports, computing guides, lecture notes, reading material, past exam papers and other learning material such as audio visual learning aids. This material is contributed by members of the University and ideally can be exchanged between students from different departments as well as intra-departmentally. Our evaluation was carried out within a large teaching laboratory in the School of Computing and Mathematical Sciences at Liverpool John Moores University, using a combination of wireless and wired machines. In this instance, all peers run the Content Exchange Service (CES) and the Payment Service, each peer joins the default JXTA 'NetPeerGroup' and has the ability to locate content, sell it and purchase it.

The first time a user connects to the CasPaCENet he is assigned a username along with a Unique Peer ID (UPI) linked to his PKI key pair. These objects are generated using the Security Service and the P2P interface which are retained for all future peer interactions within the CasPaCENet. The user has to be logged in (Figure 6.1) to the ElConE application to access the network and the application functionalities such as prepare, share and purchase content.

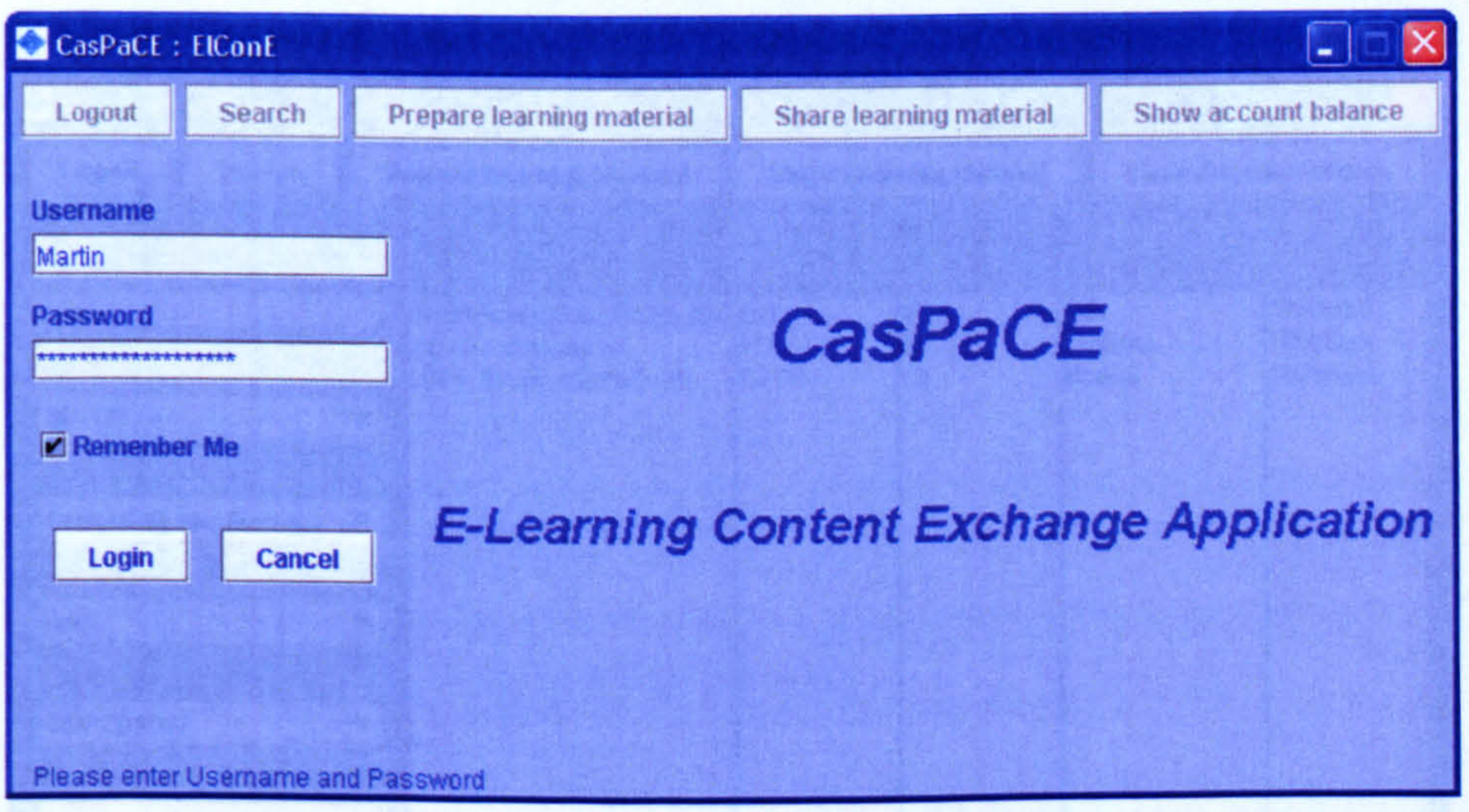


Figure 6.1: ElConE GUI logon screen

In our case study, to purchase content, a user would go through the following sequence of events:

1. Search for content
2. Review results

3. Select desired content
4. Request purchase
5. Pay for content – this step is transparent to the user and is authorised at step 4
6. Verify payment – performed at the seller end of the transaction
7. Receive content
8. Store content
9. Use content, return to step 1

Based on this sequence of events, the ElConE application's user interface (Figure 6.2) is used to interact with the network. A search is initiated by inputting keywords into the search text field (Figure 6.2). The search results can also be filtered on the basis of value of content and by University department, program of study, module ID or module name. The user can also choose to purchase content from a particular source of the content as this information is available to the application. Within ElConE the CES uses the Lookup, Discovery and Advertising services to search for content and retrieve content information such as content name, size, value, source and connection speed. This information can be utilised by the user to choose the most suitable content to purchase. Once the purchase is authorised by clicking the "Download" button, the rest of the events occur automatically without any user interaction.

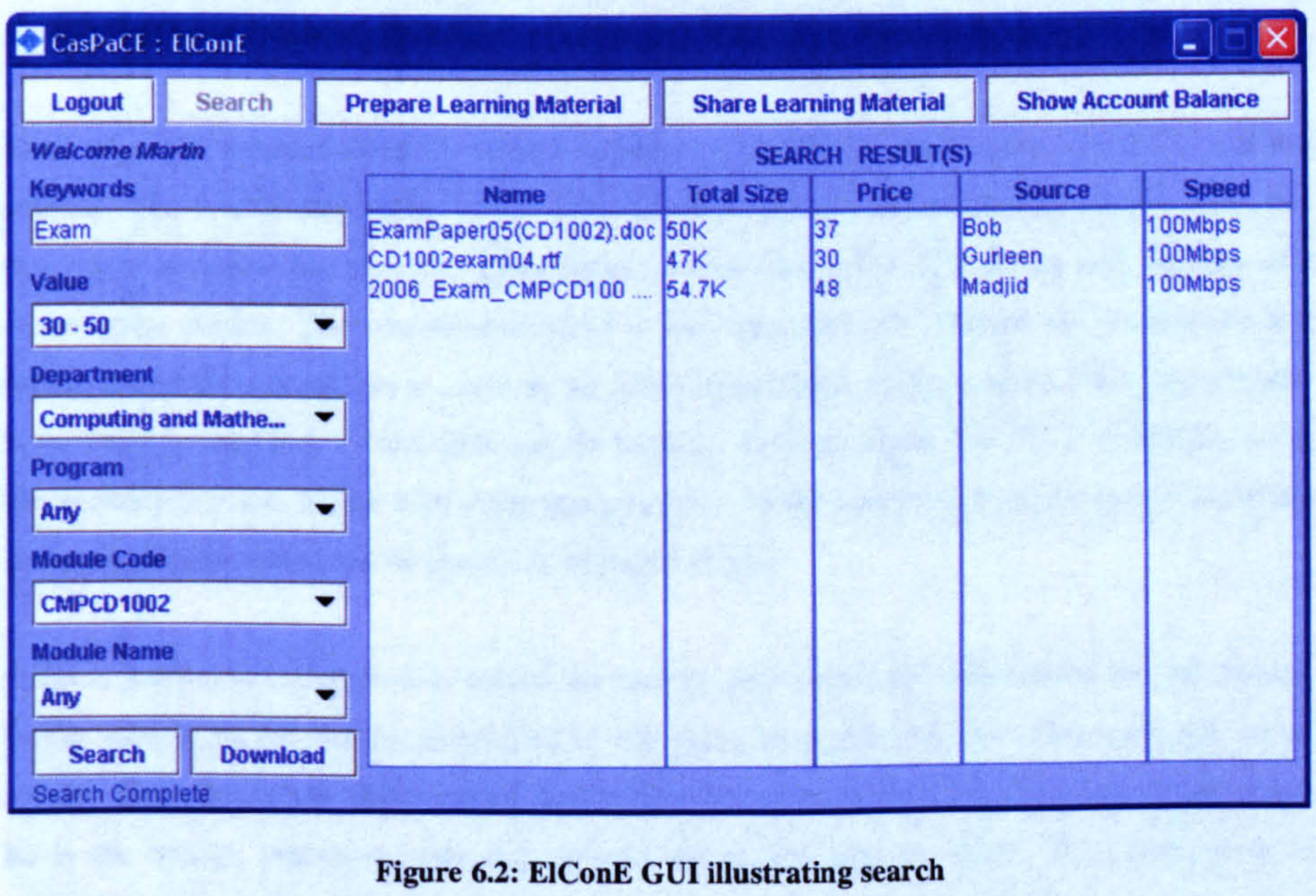


Figure 6.2: ElConE GUI illustrating search

The 'Purchase Request' triggers a set of events in the background which include a purchase request being sent to the Seller, this involves the use of the CES, Lookup and Discovery

services. Once the purchase request is accepted at the Seller's end an appropriate Bank service is selected and the purchase transaction is completed.

The ability to purchase content requires the Payment Separator and Payment Distributor components, the ability to sell content requires the Payment Verifier component, and these components form a part of the Payment Service, which is implemented using Java. Payment verification is carried out when a payment is received by the Seller. In this case study the verification criterion is based on the returned payment object containing desired information in a set format. The underlying payment related protocols are transparent to the user. However, the user is informed of the progress of the transaction via messages in the user interface. Each payment is represented by a payment object that is encrypted at payment separation and then propagated to the destination peer (owner/seller) via the payment distributor, which uses the underlying P2P interface to serialise and send the payment object. Here the PayD protocol returns these payment objects to the Owner and Seller.

In this case study, to create CasPaCE enabled content; a user would go through the following sequence of events:

1. Select desired content
2. Input copyright information
3. Embed copyright information

Each peer can create CasPaCE enabled content or distribute other peers' CasPaCE enabled content. The content descriptor which contains the copyright information is created here; this descriptor includes file specific information, owner and seller IDs along with royalty and commission values. The content descriptor is later utilised in the framework to advertise and locate this content as well as to assist in the sale and purchase of this content. This functionality is provided to the CEA via the CES and the Security Service, where the UPI is embedded using the Security Service. In this case study this process is called content preparation and content has to be prepared before it can be shared or offered for sale.

Content preparation (Figure 6.4) allows the user to input copyright information for the content. In this case study the content is selected by the user, the application then allows the user to set royalty and commission values where applicable. The user can only input royalty details where he is the owner, otherwise only the commission details may be input. This information is embedded within the content at the click of the 'Create' button. Different watermarking techniques may be applied here to embed the content descriptor.

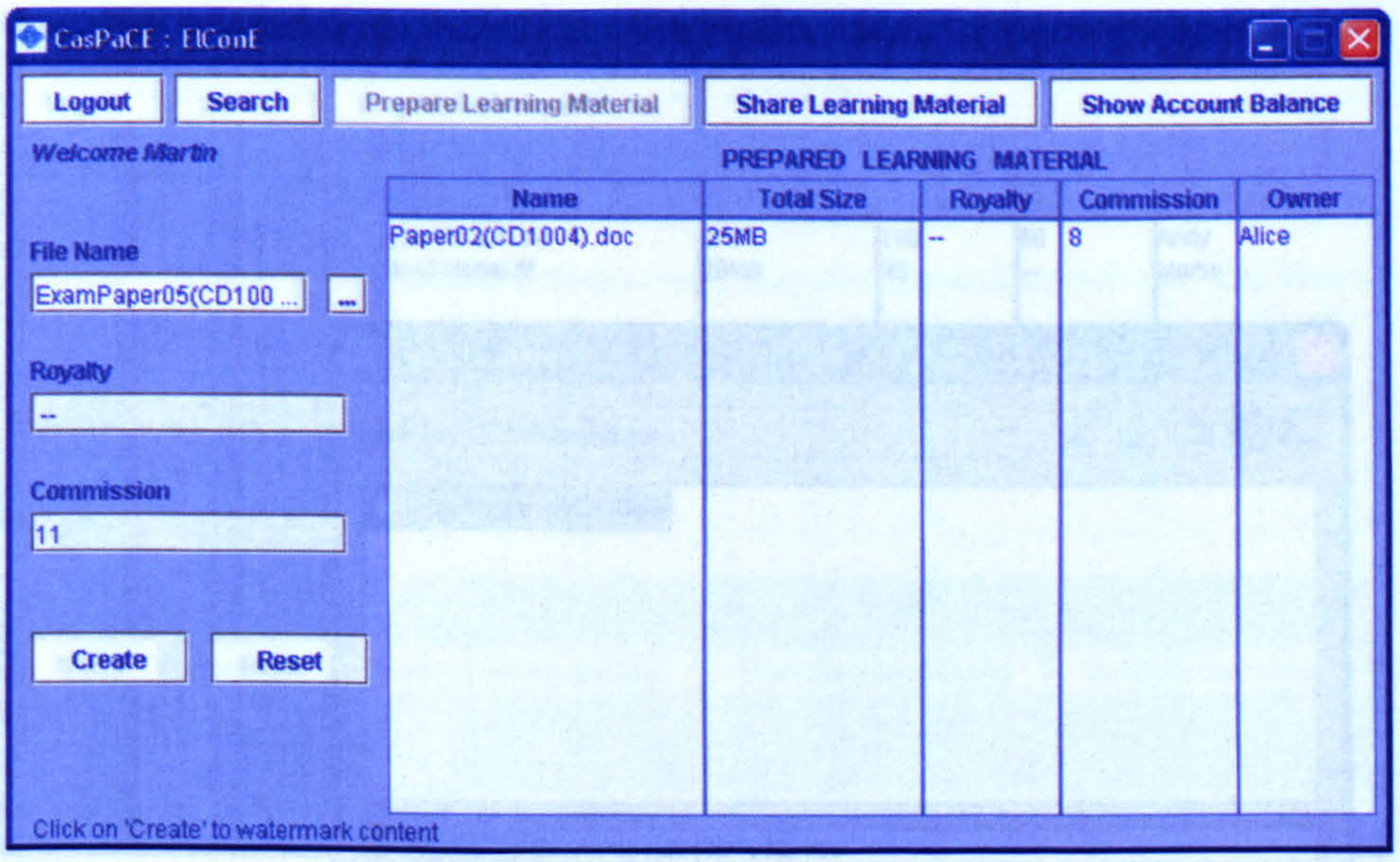


Figure 6.3: ElConE content preparation

In this case study, to offer the content for sale; a user would go through the following sequence of events:

1. Select the CasPaCE enabled (prepared) content
2. Share the content

The ElConE application uses the CES and the P2P interface to share (Figure 6.4) the CasPaCE enabled (prepared) content. The CES in turn uses the Advertising Service to offer the content for sale within the network. The interface tells the user basic information about the content he is sharing such as the content size and its copyright information including the content royalty, commission and ownership. In this way the user can sell his own content to earn royalty as well as other users' content for commission, enabling him to act as an owner as well as distributor in the system. Hence, students and teachers can create and distribute learning material such as programming guides, study tips and techniques, practical training videos etc. The prepared content gets shared in the system once the user selects it and clicks the 'Share' button effectively advertising it in the network.

Finally, we use this case study to illustrate (Figure 6.5) the effective accumulation of monies from the exchange of content. The payments are simulated in the form of payment objects, stating the value of the payment in units, for users to view. The payments received by the peer are accumulated and logged against the content they were meant for. Clicking the 'Update Balance' button shows the accumulated royalty, commission and total receipts. The interface

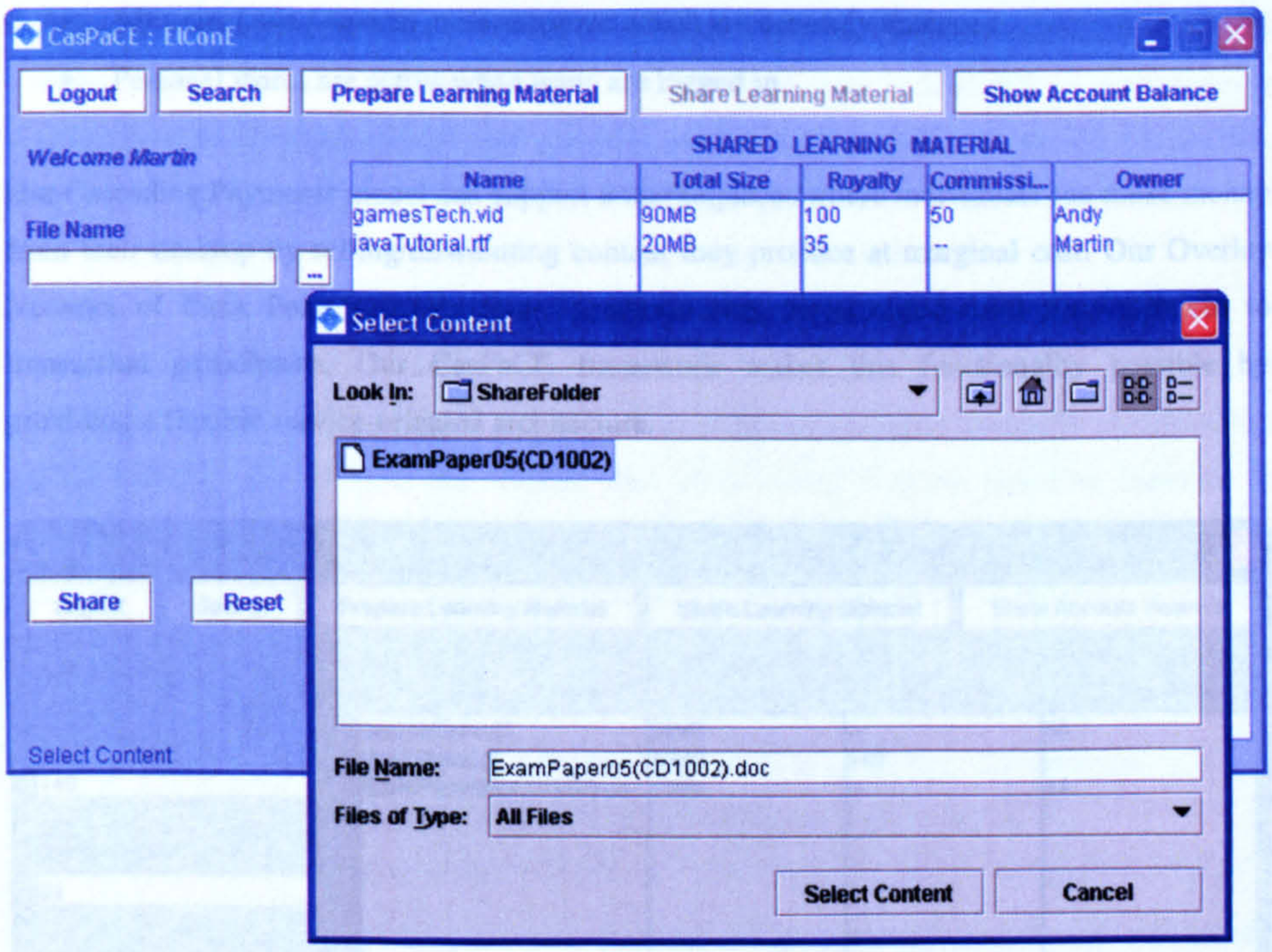


Figure 6.4: ElConE content sharing

also shows the gains per content, where owner's content registers only royalty gains and distributed content shows commission gains.

Bank peers in the CasPaCENet also have access to the P2P interface to enable interaction with the network. The Transaction Manager within the Bank Service assists in the content exchange. Similarly the Replication and Overlay managers assist in the replication and synchronisation of the transaction data store which is spread across the bank peer network. The interaction with the bank peers is transparent to the ElConE user.

Some salient features of the prototype environment developed for this case study are identified below:

- Payment verification is carried out locally at the student node, when some content is selected for purchase; the payment is made in the form of an IOU token.
- Dedicated bank peer nodes are running in the student network. All nodes are registered with the University. A number of bank peer services are hosted by the University to boot-strap the system.
- There is no distinction between staff and student accounts. The same method of exchange also applies to staff accounts.

- All users have personal store accounts which are centrally managed.
- Personal stores are active when users are logged in.

Our Cascading Payments model can support e-marketplaces where individuals can make money from their desktop by selling/distributing content they produce at marginal cost. Our Overlay Network of Bank Peers provides non-repudiation and decentralised third party services to transaction participants. Our CasPaCE framework makes this functionality possible by providing a flexible service-oriented architecture.

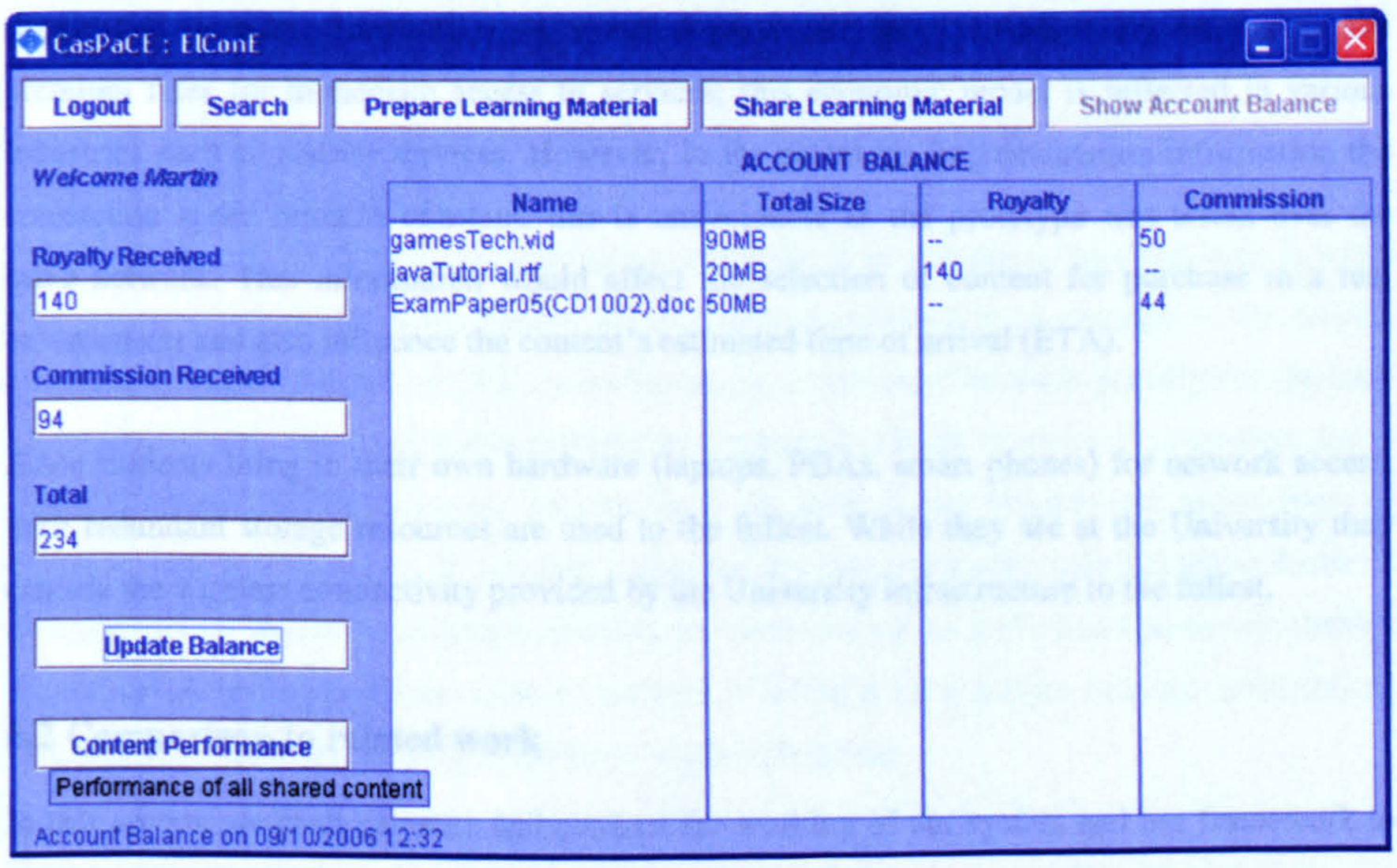


Figure 6.5: ElConE GUI showing accounts

6.1.2 Anomalies in the study

Firstly, it was stated in the requirements for a payment instrument that a credit based payment scheme is not feasible in the P2P domain primarily because of the ad hoc nature of the environment and the lack of knowledge of other peers in the system. However, in this case study a credit based payment instrument was used. We can justify this with the nature of the environment of the case study; because all students have accounts with the University it is logical to manage their balances centrally as well. This does not violate any step in our cascading payments model.

Students are allowed to freely move within the University i.e. they can access their accounts from across the university, so their account management is controlled centrally. This reflects real world practice where banking functions are managed and maintained at a centralised point.

However, this is against one of the main motivational factors of this research as stated in Chapter 1. On the other hand, testing of the prototype was carried out in a completely decentralised environment, where user accounts were credited and debited locally. In this case study the University only acts as a bank, providing a background service to attach monetary value to a payment instrument. The actual transactions are decentralised.

The development of this framework was partially motivated by the need to create e-marketplaces which would promote the sale and distribution of digital content in a competitive market based on the value of the content. This value would fluctuate based on variables in supply and demand. Similarly, it is a well known fact that consumers are willing to pay premium rates for immediate access to services; this economic model is reflected in various industries such as courier services. However, in the prototype implementation information the connection speed remains constant, this is unavoidable as the prototype was tested over the same network. This information would affect the selection of content for purchase in a real environment and also influence the content's estimated time of arrival (ETA).

Since students bring in their own hardware (laptops, PDAs, smart phones) for network access, their redundant storage resources are used to the fullest. While they are at the University they can use the wireless connectivity provided by the University infrastructure to the fullest.

6.2 Comparison to related work

In this section we shall compare and contrast the working of our system and our framework to other initiatives which address similar issues.

6.2.1 Critique

We are working on the assumption that the content descriptor is inseparable from the content, we are aware that the system is also vulnerable to the type of attacks where the content descriptor may be ripped from the content and hence copyright may be violated, but we work on the assumption that people will not break a system that has the potential to benefit them if they were to work within its perimeters. User's can only gain if they participate in the system with the owner details attached. If a user were to provide content for free it would be a loss to him as well as to the owner. Only in instances where the owner provided the content for free would it be of no consequence to the distributor if there were a charge or not – here there is a possibility of the distributor preventing propagation of free content where he won't benefit from the paid content propagation. The provision of reward by acting as a propagator of content in the P2P

network acts as a participation incentive [S. M. Lui 2002] as well as a method of dissuasion from committing fraud.

6.2.2 Our framework as a solution to prevent free-riding

As mentioned in Chapter 2 section 2.4.3, free-riding has a multitude of side effects on the performance of P2P networks. This practice is primarily prevalent in 'open' P2P content sharing networks, primarily due to the lack of incentives to encourage equal participation. However, efforts have been made to reduce the free-riding phenomenon in active P2P communities by the introduction of mechanisms which prevent or limit downloads where the user does not have a history of contribution [Nathaniel S. Good 2003, M. Izal 2004]. Or users are given 'time-shares' for downloading in exchange for contributions. Similarly, the concept of P2P 'currency' [Marc Waldman 2001] was introduced which provided a similar mechanism albeit without actual monetary gain, the 'currency' was used as a token rather than translated into hard cash. In other P2P communities where CPU cycles are the shared commodity or storage is the shared commodity, equal amounts of CPU cycles/storage are exchanged between participants [Antony Rowstron 2001b]. None of these systems encourage members to share in exchange for a financial incentive. The monetary gain in some instances is not tangible but in the form of discount coupons. However, if you pay a user for the use of his resources the greed factor is reversed and he would be willing to provide resources for use as well. Our framework enables the use of electronic payment schemes because it is based on a service oriented architecture, which has the flexibility to plug in different payment schemes.

In our system, the content producer is always compensated – thus protecting his copyright and preventing free-riding due to fear of legal prosecution. The buyer pays for the goods once, removing the need for pay-per-view or licensing for use models of payment which would prove expensive for 'small' goods (e.g. recipes) – this reflects the practice of purchase of goods which includes ownership but not copyright. The buyer becomes the 'distributor' and receives a commission for his services, in the case of P2P - contribution of storage space, advertising the goods, bandwidth and contribution of secondary resources such as electricity thus preventing free-riding due to resource conservation.

Horne et al [Bill Horne 2001] state that the use of economic incentives forms better motivation than tamper resistance for users to keep the content within a subscription community. In our system the use of cascading payments acts as a participation incentive in the system and ensures content persistence. This benefits the entire content exchange community. In contrast, the PAST [Antony Rowstron 2001c] storage initiative uses smartcards to manage and allocate storage to

users based on the amount of storage they contribute. Not only is this cumbersome because specialised hardware is needed but it is also limiting for system users whose usage is varied. The only users who can participate would have to contribute the same amount of storage which negates the reason for using a P2P storage utility. Our framework provides financial incentives as a motivation against breaking the copyright on the content, and all successful propagations of the content compensate the distributor as well as the owner.

6.2.3 Our framework as a solution to prevent copyright infringement

The use of cascading payments also ensures that content owners receive compensation for their work, apart from acting as a participation incentive to overcome free-riding our model also overcomes the issue of copyright infringement. Grimm and Nutzel's [Rüdiger Grimm 2002] Potato system uses a different approach to preventing copyright infringement via P2P music sharing networks. They rely on the voluntary contribution of music owners to their network which allows users access to music either for free or purchase. If a user buys the music he automatically becomes a licensed re-distributor and receives a 'commission' from the music author/publisher. On the other hand using the music for free does not entitle the user to any payments. This model works in a reverse order to our Cascading payment model, particularly as the buyers pay the distributor a commission and not the owners. In traditional commodity exchange it is always the owner who receives his dues; our CPM follows this business rule thus being more intuitive for users to use.

An added benefit of the cascading payment model is that small producers and consumers can distribute their content over any P2P network without the need to establish relationships with traditional publishing concerns. In Napster, PeerImpact and other P2P media portals content owners can only be compensated via their record labels which forces content producers to be bound to a particular label to receive their royalties. The choice of recording label determines the mode of media distribution and non-entertainment material can not be sold this way as there is not sufficient market interest in it. In our system any content producer can use the redundant resources available at his desktop to push his content into a global content exchange network. This is demonstrated within our case study where students can share content for payment by inserting appropriately created content into the CasPaCENet.

6.2.4 Our framework and trust issues

The Overlay network of bank peers facilitates the transition from a centralised to decentralised infrastructure which is better suited for the P2P environment promoting its scalability. The

overlay network of bank peers acts as a collective database to store transaction data, which ensures non-repudiation. By participating as a third party in a transaction, the bank peer provides an element of trust which facilitates the fair exchange in a monetary transaction. This improves the users' trust in the overall system [Bomil Suh 2002].

In our system the Overlay Network of Bank peers acts as a collective 'bank' for the entire P2P content exchange community, as opposed to the creation of a trusted path between the consumer and merchant as proposed by Atif [Yacine Atif 2002]. The basic aim of both systems is the same, i.e. to create confidence between the buyer and seller for the purpose of conducting e-commerce, in our system in effect only one 'bank' peer is directly involved in the transaction as an intermediary thus freeing up resources at other bank peers. This peer provides the validation for the transaction by acting as an impartial intermediary and ensures non-repudiation for content delivery as well as payment by maintaining transaction records. These transaction records are replicated across the overlay network of 'bank' peers to ensure persistence of the transaction data, also the system is self-organising whereby the lack of the presence of any bank peers would lead to peers automatically performing the bank peer functionality by updating their records and trying to contact other bank peers in their vicinity. On the other hand, use of Atif's technique requires prior knowledge of the amount of monetary compensation for all nodes in the trusted path and complex negotiations with these nodes, this can be a time consuming technique and wasteful in our scenario. The random selection of a bank peer from the pool of active bank peers discourages collusion hence strengthening trust.

In comparison with PeerTrust [Li Xiong 2002] and Aberer and Despotovic's [Karl Aberer 2001b] reputation-based P2P trust management systems, our framework assumes a basic degree of trust in the entire system whereby all peers may be equally trust worthy. Additionally, the bank service can only be provided by a certified peer (via a certification authority). Consequently any randomly selected bank peer should be able to provide the functionality of a third party, since the peer is not gaining from being a bank there is no incentive to cheat. However, we can include compensation for bank service provision (which is possible within our framework) in the future.

6.2.5 Comparison against other systems which provide generic payment mechanisms

Peer Impact[31] is a P2P network which allows users to legally purchase multimedia content online. The objectives of this initiative closely mirror those of our research however their

[31] http://www.peerimpact.com/

approach and the resultant outcome are different. Similar to our system, PeerImpact provides content distribution in conjunction with compensation for content distributors ('paid redistributors'). However, it has distinct differences as well; the compensation to content owners is indirect via a trusted third party (e.g. record labels in the case of music). This trusted third party acts on behalf of the content owner as an agent. Redistributors are compensated 'in kind' where the monetary compensation is bound to the network. Payment processing and monitoring is handled by a third party who is centralised and controlled by another company [Robert X. Cringely 2006]. Their system is not ad hoc; our framework allows content exchange across any P2P network and works towards interoperability. The system performs differently to our research primarily in its lack of complete decentralisation. The PeerImpact servers are used for indexing and payment processing[32 33]. Peer Impact falls under the same architecture model as Napster and uses Microsoft DRM. The Peppercoin [Ronald L. Rivest 2004] payment scheme, used by PeerImpact, is based on aggregation of micropayments, which are then processed as macropayments hence reducing the processing cost per transaction.

Gerke and Stiller's [Jan Gerke 2005] P2P service architecture, from the MMAPPS project, describes a service-oriented architecture with similar objectives and ideology as the CasPaCE framework. They provide a generic framework for combining P2P services, same as the service integration within CasPaCE. However, SOPPS [Jan Gerke 2003] concentrates heavily on the negotiation of SLAs which are agreed upon at the time of service negotiation. The system is meant to be a generic platform to plug in different P2P services and assist with service compositions. The services described within the CasPaCE framework can easily be plugged into the SOPPS architecture.

6.2.6 Comparison to DRM and Licensing systems

The CasPaCE framework addresses the same issues as current DRM technologies, however it is more powerful because it mobilises the copyright with the content, disassociating it from a centralised copyright management system. This has a two fold advantage; firstly this enables the use of P2P systems for content distribution reducing the cost of content sale and distribution and

[32] Inferred from background research via PeerImpact's mission statement and company related material such as press releases.

[33] Note – no technical papers published for Peer Impact, Wurld Media seem to be gearing towards a payment for any digital content strategy. The tag line is 'distributed delivery systems'. Their LXsystems (http://www.lxsystems.com) subsidiary uses Grid computing to benefit the commercial marketplace for the purpose of multimedia content marketing and distribution. Data distribution is decentralised but control of content is completely centralised.

secondly it conforms to consumer expectations [Deirdre K. Mulligan 2003] with reference to copyright and personal use of content. In section 3.1, we stated that one of our requirements to uphold copyright was that the owner should be compensated every time his content is downloaded i.e. a one off payment made when the content is purchased the first time, this can be equated to a flat-rate charging scheme. Another traditional model for digital commodity sales is that of usage based charging which includes licensing schemes [Russell Perry 2002] and most DRM tools. The usage based charging model requires constant monitoring of the content, every time the content has to be used both the user and content owner would have to be online to get the content license validated. In the case where the content owner was a single entity the P2P network's scalability would be affected. On the other hand our approach removes the need for the content owner to be online constantly.

Our framework allows the distribution of royalties to the owners without the intervention of a centralised server as is common in most Digital Rights Management (DRM) [Peter Biddle 2002, L. Jean Camp 2002] technologies. There is no need for licensing systems or the purchase of unique licenses to gain access to content. Once the content has been purchased access rights are automatically associated with it, this reflects the practice of purchase of goods which includes ownership but not copyright and is consistent with user expectations[Deirdre K. Mulligan 2003]. This was proved in our prototype where no single centralised authority was used to determine copyright. Buyers only communicated with the seller directly for the purchase of content once it was located.

However, we do realise that in a real e-marketplace it is important to provide means to support and not exclude other business models. Since our framework is designed to be flexible, services could be implemented to include usage based charging. By extending our system to include a usage meter in the form of tokens the cascading payments model would still apply. However, in that case the payment separation and distribution would get triggered every time the content is opened for use instead of on payment at download and the user would either have to be online to distribute payments or store the payments for distribution. This is a method for extending the framework and is part of future work since it raises some interesting questions especially with respect to the distribution of commissions. The tokens in question could be extensions to the content descriptor which would allow content manipulation based on rights assigned by the content owner.

6.2.7 Analysis of the use of unique identification in the CPM

A well understood problem with digital communities is that of uniquely identifying their members [Judith S. Donath 1998, John R. Douceur 2002, Sherry Turkle 1995]. Consequently, we recognise that the same user can have multiple identities within our system. However, each content and its payments are associated with a single user ID. In the long run a user's reputation will be associated with a particular ID, the private key linked to that ID will be used to redeem the payments for that particular content. So it is in the best interest of the user to manage his IDs well to ensure that he does not lose potential earnings from any of his content being advertised under a particular ID.

To ensure secure transmission of payments we use public key encryption and secure transport protocols (discussed in detail in Chapter 4). It is recognised that there are many aspects to security, in this section we rationalise the use of asymmetric (public key) encryption techniques for unique identification of peers and how our system overcomes basic issues such as identity theft and impersonation.

A malicious node could try to impersonate another peer in two ways; by generating an identical ID or by generating an identical ID by extracting the target's public key:

1. By generating an identical ID

   It is rendered impossible by virtue of the use of the date time stamp and asymmetric key pair. To use this method of subterfuge the impostor's ID would have to be generated at the exact same time as the target ID and the malicious node would have to be in possession of the target's Public Key. This is not possible because the public key is not available to anyone apart from the owner at the first instance of peer ID creation as it has not yet been published.
2. By generating an identical ID by extracting the stolen ID's public key

   Once the target's public key is freely available in the network, this method of replication of the target's ID is rendered difficult by the inclusion of the date time stamp when the peer first registers to participate in the network.

In case identity theft occurs, the primary reasons for impersonation would be for direct financial benefit or for indirect benefit by ruining a target's reputation:

1. Direct benefit would be by unfairly redeeming money meant for the target

   It is not possible to gain monetarily by stealing the ID, because the impostor can not redeem payments which are encrypted with the target's Public Key as he will not have access to the Private Key to decrypt the payments.

2. Indirect benefit would be by tampering with the peer's reputation as a reliable content provider

   The impostor can only provide content claiming to be from the target. This would not benefit the malicious node, except by ruining the reputation of the stolen ID holder. Maintaining the reputation of the content owner is out of the scope of our work, which is why we have assumed that the system and content being exchanged via the system are trust worthy (§3.1.4). Flooding the system with bad files is not viable as demonstrated by live P2P file-sharing communities, where files are replicated by virtue of their popularity. Similarly in our system content will be replicated if other peers initiate a purchase, or choose to distribute it further. However, our framework is designed for flexibility and interoperability, consequently reputation management algorithms such as EigenTrust [Sepandar D. Kamvar 2003] or Xiong and Liu's trust model[Li Xiong 2003] can be plugged in.

It is recognised that PKI encryption is better suited for smaller payloads and symmetric encryption is recommended for large payloads. We believe that the use of PKI to encrypt payments in our system is feasible as the size of payments (payload) in digital payment schemes is small. However, encrypting/decrypting content in this way can be slow for large content payloads. In the case of encryption, the content is encrypted a priori; however decryption times could cause a bottleneck in the transaction. In the content exchange protocol (§4.3.4), the delivery of the seed (i.e. the key generation item - r_x) is sufficient to complete the transaction, so the content is not being decrypted live, hence the transaction should not be stalled.

6.3 Performance evaluation

This section presents the system's performance evaluation with reference to issues such as the load on a peer while participating in content exchange transactions as well as in bank peer activity and query propagation times. Since these issues are relevant to this environment particularly as the environment is heterogeneous and has to accommodate thin peers as well as normal peers. This performance analysis is performed on a single transaction between a buyer, seller and bank peer.

6.3.1 Theoretical performance evaluation for a transaction involving a bank peer

Taking into consideration a single transaction from start to finish in the network, the scope of the transaction is from the initial search for content X to completion of the transaction resulting in the content being paid for and downloaded. To evaluate the performance of a transaction

involving a bank peer we performed our calculations on the data exchanged in the fair content exchange sequence described in Chapter 4 Figure 4.13. To calculate the search (step 1 in Figure 4.13) query propagation time, we calculate the number of hops required to reach the desired number of peers. In a network where each peer's connectivity is 10, it would take 6 hops to reach a million peers. To determine the query propagation times[34] we use the formula $(QuerySize \div UploadSpeed) \times NumberOfHops$. We calculate the best case scenario by assuming all nodes are on broadband connections and the worst case scenario by assuming all peers are on dialup connections, Table 6.1, shows the mean query propagation times for 100 – 1000000 nodes.

The data being exchanged between the three participants in a transaction is in the range of 2 - 4.5KB per message; this excludes the size of the actual content and the number of results being returned. This range is based on a certificate size of 700-2000 bytes, message headers of 2000 bytes and other data of maximum 50 bytes. Based on these values, the message size at step 1 in Figure 4.13 would be 2KB.

Table 6.1: Query propagation times across dialup and broadband networks

Nodes	Hops	Query Propagation Time (s)		
		All nodes on Dialup	All nodes on Broadband	Mean time
100	2	0.119	0.031	0.075
1000	3	0.178	0.046	0.112
1000000	6	0.357	0.093	0.225

Similarly each result in step 2 would be between 2.5KB-4.5KB in size and control messages would be 2KB in size. The size of the downloadable content is variable; however the larger the content the longer it would take to download onto smaller devices and the more space it would require, hence thin peers would have to work within those limitations. The amount of data returned (step 2 in Figure 4.13) is dependent on the number of results returned which in turn depends on the size and scope of the P2P network. In a typical CasPaCE network 10% Bank Peers would be required to assist 90% of the peers. For instance, in a network of 100 nodes, there would be 10 Bank Peers, 1 Buyer and 89 Sellers. If we take the upper limit of the result size (4.5KB) and all selling peers returned a positive response to a query for Content X, then the

[34] these calculations assume there are no queuing delays within the network

Buyer would receive 0.4MB of results data. To accommodate the limitations of the thin peers, the search service has the ability to scale down the number of incoming results by limiting the search query's (step 1 in Figure 4.13) number of hops. Devices such as PCs or laptops with more resources can process considerably larger amounts of data and are not bound by these constraints.

Once a purchase has been authorised (step 3) the majority of messages passed between the peers are control messages which are of negligible size (approx. 2KB). Other inter-peer messages do not exceed 4.5KB. Hence, the major load on the system in a typical purchase life cycle would be in the initial two steps when content is being located, all subsequent communications would involve negligible amounts of data being exchanged directly between the 3 parties involved. A transaction is initiated once the Buyer authorises a purchase, at step 3 in Figure 4.13. Hence, in a single transaction, step 3 onwards, the Buyer, Seller and Bank respectively send 15KB, 10.5KB + y and 8KB in total, where y is the size of Content X being purchased. Any other information objects used are internal to the peers and would not have header information which would reduce their sizes further. Also, as the amounts of data being encrypted/decrypted are negligible in size and speed of encryption/decryption is a function of processor power versus size of payload; encryption/decryption is almost instantaneous. Transaction completion time between the 3 parties is calculated as $(0.997 + x)$ seconds, when all three parties are on dial up connections and $(0.261 + x)$ seconds, when all three parties are on broadband connections. Here x is a variable value which represents the sum of the processing times of the participating devices and varies dependent on the devices' specification.

The Bank Peers can participate in considerably more transactions as they exchange very small amounts of data of negligible size with the Buyer and Seller during a transaction and are not limited by the lack of resources as thin peers can never act as bank peers. Also, the only processing overhead on a bank peer in transaction participation is in maintaining the session with the buyer and seller and querying the transaction database for record validation. The major processing load on a Bank Peer is during transaction record replication and synchronisation.

6.3.2 Actual bank peer records size and performance evaluation

The amount of storage and memory required is a function of the number of transactions allowed on a peer. If we restrict the number of transactions to 3 at a time then it takes 20 seconds for each individual transaction to finish and the bank peer is available to participate in other transactions. Three simultaneous transactions conducted using the same bank peer (ran multiple normal nodes and only one bank peer) in a test bed of over 500 networked machines with

connection speeds of 100 Mbps resulted in an overall transaction completion time of 21 seconds.

A single transaction over a network of over 500 machines took 20 seconds, therefore it takes (20 + x) seconds, where x is the delay caused by variance in network traffic and the device configurations (e.g. processor speed, memory and network connection speeds). The tests were performed with seven identical computers. Each computer was equipped with a 3.0 GHz Intel Pentium 4 processor, 1 GB of RAM, and a 100 Mbit Ethernet adapter running the Microsoft Windows XP operating system.

We determined the size of a transaction record store by using a flat file to store a set of records (transID, bankID, contentID, ownerID, intSellerID, buyerID, royalty, commission, requestReceived, paymentReceived, contentReceived, transComplete, dateTimeStamp, paymentInstrument). The size of a standard text file containing 100 records is 32 KB based on this very nominal size it was determined that the cost of storage in the system to any one peer device acting as a bank peer is nominal for the maintenance of bank peer records as 10^6 records would only need 312.5 MB storage space. Performing the same test using a relational database (Apache Derby[35]) resulted in a database size of 1.71 MB for 100 records and 512 MB for 10^6 records. Although there is a large variance in size when using the two different data storage techniques, the use of a RDBMS enhances system performance by allowing quicker lookups due to indexing facilities. Thin peers can not act as bank peers as stated earlier, however Fat peers can accommodate these resource requirements easily because the specifications of standard computers are increasing due to the reduction in hardware costs.

6.4 Framework application areas

A direct result of the flexibility of our framework is its applicability to various application domains. This section describes some of these application areas, which may be broadly categorised into content distribution and service utilisation.

6.4.1 Content exchange applications

Within the content exchange application domain our framework can provide new models for the exchange, distribution and sharing of content. Our system can be used to compensate content

[35] http://db.apache.org/derby/

producers and distributors as well as aiding the development of new business models while providing new participation incentives in these application areas.

6.4.1.1 Digital libraries

Digital libraries provide a service to education and research institutions. The success of the digital library service model relies on it providing latest journal articles, conference proceedings, and magazine and newspapers articles at reasonable costs to subscribing institutions. If libraries can embrace P2P technologies into their own services, they will possibly develop new service models, or improve existing ones [Ying Dong 2002]. By using our system they can overcome the common issues of copyright violation while providing the aggregated resources of contributing institutions to the whole community.

One benefit of the use of the P2P model in this problem domain would be as a method for preservation or archiving of the digital content. For example, a group of digital libraries would cooperate with each other to provide preservation by storing copies of each other's digital materials. In this scenario, each library acts as an autonomous peer in a distributed, heterogeneous collection replication mechanism. Such a community would not require a central controller to manage the replication of data; instead, each peer would communicate with other peers to replicate its own collections. The result of this inter-working between libraries would be a global community in which every library's collections are protected, as well as an overall efficient and fault-tolerant digital library. Within this scenario the use of the cascading payments model would act as a method for revenue generation, where the overlay network of bank peers would be another autonomous system maintaining transaction records. The royalties would get channelled to the content authors and the commissions would be paid to sister libraries in the community where the content was accessed from. A marketplace would be created where accessing content from different global locations would be priced differently.

6.4.1.2 Gaming

In recent years there has been a constant increase in revenue from games sales, with the advent of mobile gaming this is predicted to increase further. The exchange of games over small P2P communities such as commonly formed between friends or colleagues would be an ideal application area for our system. The games authors would be compensated for their IPR while the community of friends/ colleagues would benefit from acting as games distributors. This would lead to an increase in revenue for games authors especially where they are networked games which would encourage sharing further.

For traditional single player games, our system can be used for game distribution by game producers, where gaming enthusiasts can benefit from acting as game distributors and both parties are appropriately compensated.

In the networked gaming scenario, where gamers have to pay to play games in a client server environment, using P2P technologies allows the setup of small gaming communities while taking away the burden from the gaming companies' resources. The additional cost of setting up gaming servers is also removed. In the P2P community the gamer who initiates the game acts as the game host and pays the producer for his IPR, other gamers pay the host for acting as an intermediary. For subsequent plays, the winner gets to choose who would run the game (act as game host) and his play would be free while other members pay to play, factors such as the winner node's capability may influence this choice. The game author is always compensated for his IPR.

6.4.1.3 Small scale publishing

Our system enables small publishers to generate incomes by utilising the redundant resources on their desktop. It pulls away from the need to maintain web portals to host and distribute content. It provides a platform for publishing and distributing inexpensive digital content such as DIY guides, recipes, programming guides, hints and tips and generating income from it which would pay for itself (the publishing). Not limiting itself to static content, when taking the view point that services are content, the framework allows for the paid utilisation of services written by programmers and hosted via their P2P network connections. These services may be discovered and utilised while compensating the service author and/or publisher fairly. The ElConE case study discussed earlier is an example of small scale publishing as well as the use of P2P for e-learning.

6.4.1.4 Content distribution e-marketplaces

Our framework can enable the development of new business models for traditional media firms. Already conglomerates such as Time Warner, AOL, Apple and others are seeing the financial benefits from the commercial potential of digital content sales through web portals. Companies such as Napster have moved into paid digital content sales while using the power of P2P technologies; however they still rely on the centralised P2P model for payment management and use a subscription model for their heavy users.

These companies can set up peer nodes to push out content into the P2P network which will remove the burden of maintaining expensive servers and push the effort of payment processing on to the participating buyers. Similarly, banks and payment processing firms can set up bank peers or payment service providers and get compensated for the use of their services.

Currently the favoured model uses DRM techniques to limit the access time to downloaded media content. In our framework this content can be further propagated by the buyer while ensuring each member of the value chain is compensated and there are mechanisms for non-repudiation.

Another area which can benefit from P2P content distribution is that of market research publications. In the current atmosphere of cutting edge market research, it is becoming increasingly common to require reports on market research for every aspect of day-to-day life; from commodity sales to generating user demographics profiling. Institutions which carry out this research are not necessarily always funded by industry and need new models for revenue generation. Using a P2P topological network to push out these reports is an inexpensive method for revenue generation while cutting down operational expenditures.

6.4.2 Service utilisation applications

The class of P2P applications which fall under the service utilisation category allow service to be used locally by direct download or remotely through service invocation. Within this application domain our framework can provide added benefit by enabling compensation for service utilisation as well acting as a suitable participation incentive.

6.4.2.1 Paid service composition and utilisation

Our system can be applied to any type of digital content from e-books to multimedia content to digital services such as online currency converters or recipe measurement converters.

In a typical scenario a person wishing to read an article in English which is originally published in French can discover a 'translation service' written by a small provider and utilise it. Our system would ensure that the service author is compensated for his IPR and compensate any intermediary for hosting the service if necessary.

An extension to this work is in the area of Paid Service Utilisation and service composition. A strand of research within our research laboratory focuses on the dynamic composition of

services [Paul Fergus 2005] to provide efficient utilisation of networked appliances. Within this body of work a scenario of device functionality utilisation is one where a less capable device, such as a smart phone, is used to dynamically discover better services on more capable devices in its vicinity and utilise them. For example in a train compartment, a smart phone user wishes to listen to his mp4 encoded audio file on his phone which is not capable of decoding the mp4 format file for play back. There exist other users in the compartment with portable devices such as laptops, PDA, palmtop etc. which may possess services capable of decoding the mp4 format to alternative formats. The smart phone is capable of playing mp3 files. Hence, a service is required to convert mp4 to mp3. This service could be provided by one device or multiple devices by aggregating resources. Since these services belong to different user(s) there is need for compensation for service utilisation. The services should be paid for before they are utilised.

We acknowledge that there are many complex research issues to address, particularly with respect to issues such as service interactions. Services may operate well when used in isolation or within small compositions; however problems will occur when trying to inter-work a large number of services at the same time. In particular how do you reconcile a payment for some content i.e. service which has only been utilised in part due to a conflict of usage with a different configuration? This is a matter for future research in this area.

One of the key requirements of service usage in this scenario would be the immediate provision of services. Hence a service which is immediately usable may be more expensive than similar services which have delayed availability. An automated service selection algorithm would be required to make this decision for the user or the user can be provided with an interface which gives him this information and awaits his selection. This latter method is equivalent to our current approach. Hence, in this scenario this method is used to accept selection of a 'service' which was considered equivalent to authorisation of a transaction.

An extension to the services enabled by this work is already being carried out by other researchers in our research laboratory. The CasPaCE services are being integrated into the Home Appliances Integration Unit (HAIU) [Madjid Merabti 2005, Anirach Mingkhwan 2004] which deals with integrating networked appliances with different network interfaces (802.11a, b or g, IrDA, Bluetooth) together through a single unit the HAIU. In this scenario our framework provides the ability for these appliances to share services and resources while ensuring the service or device owners and intermediaries are compensated.

6.4.2.2 Networked appliances

In the present day there is a trend towards networking users' home appliances to enhance consumer experiences and facilitating different life styles. The large number of digital devices present in the common household today may be networked to create Personal Area Networks (PAN); P2P networks can be created in this setting where every device would act as a peer node in the environment.

Another scenario for service utilisation in this environment is one where each device's functionality may be decomposed into individual services; these services can then be recomposed in different combinations to create new virtual appliances within the PAN.

An example situation would be where a peer device capable of DVD playback is loaded with a movie DVD; a user at a remote location discovers this movie and wishes to watch it. By using our system, the viewer pays a royalty to the movie production studio and a commission to the DVD player owner for the use of his resources through the P2P infrastructure thus compensating all parties in the content value chain.

An enhancement to the above scenario could be that the viewer wishes to get subtitles for said movie in a different language to that available. Using the P2P infrastructure, he may locate the appropriate subtitle service and pay for it directly/indirectly while integrating the services to create a virtual DVD player most suited to his life style at his location.

6.4.2.3 Paid P2P streaming (radio and television)

In this era of pod-casting and digital radio it has become common for users to access on-demand entertainment. Research is being conducted where P2P networks are used to efficiently stream multimedia content based on receiver driven demand [Jin Li 2002] as well as server controlled streaming [Venkata N. Padmanabhan 2002]. With these technologies it is possible to move into the field of P2P radio or television broadcasting, where our framework can be used to provide an alternate model to the traditional TV licensing scheme or even the traditional method of revenue generation based on advertising. Instead of a one off payment to the TV licensing board, users pay for a 'TV license' on a pay-per-view basis where the program makers are compensated for copyrights and intermediary broadcasters (peer nodes) are compensated for their contribution to the value chain.

There are many benefits to the use of P2P networks for multimedia streaming. Issues such as jitter and skipping are better handled by P2P systems by providing resource redundancy and farming of resources. In our scenario instead of paying an annual subscription fee for viewing TV programs, the user only pays for what he watches making the system fairer for content producers who get appropriately compensated for good quality and popularity of their programs. This can lead to advertisement free broadcasting ensuring content is produced for its merit instead of commercial value e.g. documentaries. Consumers who prefer to watch advertisement free broadcasting would pay for this service, as has been displayed by the popularity of paid email service providers such as MSN.com and USA.NET. Our system provides the ability to directly compensate content producers and intermediary broadcasters to facilitate paid P2P broadcasting.

6.5 Summary

This Chapter demonstrated how the CasPaCE framework's prototype addresses the requirements listed in Chapter 3, to provide equitable digital content exchange while using a decentralised P2P environment. The framework was evaluated against the requirements listed in Chapter 3 and against other initiatives which address similar concerns through the ElConE case study. This case study captured the working of the various services within a user based environment where e-learning content can successfully be exchanged for 'payment' while maintaining the intellectual property rights of content owners and compensating distributors in the content value chain.

We proved that in the case of exchange of goods for money we ensure that transactions are atomic and non-repudiated. A performance analysis was provided to illustrate the resource consumption of a typical transaction. The performance evaluation demonstrated that the actual transaction from start to completion is not resource intensive on the participants or the overall network. It also demonstrated that bank peers can easily handle multiple transactions without having a detrimental effect on the performance of the transaction or itself. The resource (storage, bandwidth and computation) demands on the actual bank peer are negligible in view of the criteria for selection of bank peer devices. We also provided a comparison against other initiatives in the same field. Finally, we demonstrated how our framework can be applied to alternative application areas which include integration with work within our research laboratory.

Chapter 7

7 CONCLUSIONS AND FURTHER WORK

This thesis describes the results of research into the problems associated with content sharing P2P networks especially with respect to the problems of content theft and free-riding. As a result of this research, further issues relevant to the field of e-commerce within the P2P domain were unearthed. We have highlighted the problems associated with P2P technologies, their classification and peripheral technologies which are used in current P2P systems in the literature survey. Consequently the problem domain was defined and the research outcomes defined. We then described a novel solution, its analysis, design, implementation and evaluation in the preceding chapters.

This Chapter presents the conclusions for this body of work. Firstly, it presents a summary of each Chapter in this thesis. This is followed by suggestions for a number of future enhancements to the framework. Finally, concluding remarks which include a quick review of the contributions to knowledge and overall conclusions are presented.

7.1 Thesis summary

Chapter 1 introduced the main theme of this thesis along with the motivation behind this research. It highlighted the need for a mechanism for compensation in the P2P domain to discourage content theft while stating the advantages of using P2P as an economic model for content distribution. We described our vision of a producer-centric content dissemination model which does not rely on middlemen to distribute the producers' content but rather uses the redundant resources available at their fingertips to financially benefit themselves. The Chapter also gave the reader a flavour of the research methodology that was used to conduct this work while highlighting the main aims and objectives of this research, thus, setting the scene for the remainder of the thesis and hence influencing its structure.

Chapter 2 acquainted the reader with the main research domain, P2P technology. It discussed the different definitions of P2P in the modern day, describing its various classifications and applications and the underlying techniques used in modern P2P systems. The issues relating to digital content theft and copyright infringement were explained. As part of the e-commerce

literature review in this Chapter, it was concluded that majority of digital payment schemes required the presence of a third party to complete monetary transactions. Consequently it was concluded that the P2P system infrastructure would have to be extended to accommodate modern day digital payment schemes. It was deduced that in order to use financial participation incentives to discourage copyright infringement, content theft and free-riding in P2P environments, we would have to satisfy transactional requirements such as non-repudiation. It was also observed that this research is an amalgamation of various subject domains; copyright protection, economic models, payment mechanisms, e-commerce, security techniques, trust issues, service-oriented architectures, databases and others. These domains were introduced and discussed in this Chapter.

Chapters 3 and 4 presented the design of a new framework to support equitable digital content exchange in a P2P environment, while retaining the advantages of a decentralised ad hoc heterogeneous environment with low cost of entry and a natural model for resource scaling with increasing community size. Chapter 3 detailed the requirements for this system and introduced the various models (cascading payments, overlay networks, 'bank' peers, redundant service utilisation and the service oriented Cascading Payment Content Exchange (CasPaCE) framework) developed to fulfil these requirements. It also presented a high level design for these models which included their functional and non-functional requirements. Chapter 4 presented the detailed design for a framework and its components in terms of the various services and protocols developed, including the design decisions made and design techniques utilised. These protocols included the unique identification of peers and copyright mobilisation for copyright protection. The payment separation, distribution and verification protocols belonging to the Payment Service were discussed. A suite of Bank Service protocols: random bank peer selection, transaction management, transaction record replication and synchronisation, were also described. Other protocols and services designed to complement the CasPaCE services were also discussed in this Chapter. The combination of these services and functionality allows any Internet enabled digital device to participate in a paid content exchange transaction within the P2P domain by utilising our framework.

Chapter 5 discussed the implementation and testing of our framework. It discussed the issues that arose during the implementation of the framework components as per the requirements and design laid out in Chapters 3 and 4. We used advances in P2P technology, which is devoid of any centralisation and the Java object-oriented programming language to implement our prototype. This Chapter presented a set of test scenarios which were used to test the prototype within the CasPaCENet environment setup in our own laboratory. These scenarios assisted in

the system integration testing as well as extraction of performance results. This prototype demonstrated the successful working of our framework and acted as a strong proof-of-concept for our solution.

An evaluation of the system was presented in Chapter 6 which compares and contrasts it to other relevant work in this field. The evaluation was carried out in a case study which applies to a typical user centric application of the P2P environment i.e. content exchange within a large University. This case study focused on the exchange of content for compensation within an e-learning scenario, where users interacted with each other over a P2P network to share different learning materials. A performance evaluation of the system was also presented to demonstrate the typical resource usage in a content exchange transaction as well as the resource consumption on a bank peer.

Finally, this Chapter presents the concluding remarks and possible enhancements for this research. It highlights the contributions to knowledge presented by this body of work.

7.2 Contributions to knowledge

The contributions of this research work are various and distributed throughout this thesis. The principal contributions of this research are:

1. A definition of a P2P network in the contemporary environment (Chapter 1/2).
2. A survey of the classifications of P2P technology (Chapter 2).
3. A Cascading Payment Model (CPM) for the fair distribution of royalty and commission in exchange for content has been devised. This model mirrors real life economic models [Gurleen Arora 2003] (discussed in Chapter 3) and ensures that the content owner is always recognised as the intellectual property right owner, while the distributor is compensated for his contribution to the transaction. This includes:
    a. The use of a user centric approach for intellectual property rights owner identification [Gurleen Arora 2005a]
    b. Making the intellectual property rights owner identity mobile as opposed to the use of a centralised point of management of IPR [Gurleen Arora 2005a].
4. Formulation of the payment separation and distribution techniques [Gurleen Arora 2005a] to facilitate the CPM.
5. The use of an Overlay Network of bank peers as a collective Bank [Gurleen Arora 2005b, Gurleen Arora 2006] (discussed in Chapter 3) for the system ensures that the P2P network can function in a truly P2P (discussed in Chapter 2) manner while still following the accepted rules of commerce.

6. A Cascading Payment Content Exchange (CasPaCE) framework that supports the CPM and the Overlay network of Bank Peers for the P2P domain.
7. The use of a service-oriented architecture to design the CasPaCE framework which enabled the creation of an open and flexible framework. This allows inter-operability and paves the way for the application of the CPM to other digital commodities such as services and functionality as discussed in Chapter 6.
8. A UML architectural model describing the various components of the framework.

In conclusion, the findings of this thesis are as follows:

- P2P networks are here to stay and P2P is a viable model for content distribution. They allow for the participation of heterogeneous devices which are becoming more 'intelligent' and inexpensive with the advances in semiconductor technologies and broadband communications.
- P2P networks are flawed with free-riding and copyright violation which inhibits the efficient working of the P2P network. Leaving them open to attacks and detracting from their cost and performance benefits.
- To overcome these flaws, participation incentives are required which also discourage copyright infringement. Current participation incentives restrict the efficient use of the P2P network and do not provide financial incentives to its user communities. They are used as a mechanism to prevent free-riding and hence do not discourage copyright infringement. Traditional technologies for the protection of copyright such as DRM can not be applied to the P2P domain without detracting from its true potential: i.e. aggregated resource usage as well as a cost effective method for content distribution. These technologies rely on centralised management of copyright and do not empower the content producers.
- Cascading Payments is a solution to benefit all participants and to overcome these flaws. It ensures every member of the content value chain is compensated by using the traditional royalty/commission model of compensation. The financial benefit for sharing copyrighted material acts as a participation incentive which discourages free-riding. The use of this decentralised method to prevent copyright infringement also empowers content producers by giving them control over their content.
- Monetary transactions have to be trusted and atomic so neither participant may deny the successful completion of a transaction. Consequently it is necessary to store transaction data and ensure its persistence for the duration legally required. To enable this functionality within the P2P environment without detracting from the benefits of a pure P2P system it is necessary to apply distributed storage of data. The bank peers in our

system act as both data stores to maintain the persistence of transaction data as well as trusted third parties to facilitate fair exchange.

- The CasPaCE Services Framework enables the implementation of the CPM and ensures the atomicity of transactions. While allowing heterogeneous devices to participate in economic transactions within a P2P environment. It encourages interoperability and flexibility because of its service-oriented nature.
- This Framework can be used in different scenarios (Chapter 6), such as:
  - Content exchange applications
    - Support for e-learning
    - Digital libraries
    - Gaming
    - Small scale publishing
    - Content distribution e-marketplaces
  - Service utilisation applications
    - Paid service composition and utilisation
    - Networked appliances
    - Paid P2P streaming (radio and television)

7.3 Further work

This section suggests further work for the enhancement of this framework which can lead to further research in this field.

7.3.1 Guaranteeing the quality of content being exchanged

Currently there exists no system of preview in the system. Due to the ad hoc nature of the environment, buyers in the system would require some guarantees with regards to the content they purchase. At present our model supports free content distribution as highlighted in section 4.3.5, however there is no mechanism for content preview even within this scenario.

Research initiatives exist [Fabrizio Cornelli 2002, Ernesto Damiani 2002] which rely on reputation management techniques to qualify nodes as providers of good quality resources. However, these systems rely on polling techniques to generate reputation scores. These scores are given once the resource has been accessed. In future work we would look at extending our system to allow users to preview the content prior to purchasing it. For audio/visual media, this could be done by streaming a sample of the content prior to payment authorisation, once the buyer is satisfied with the quality of the content, payment may be authorised. In order to ensure

that the content being exchanged and the sampler are one and the same a content hash is already included within our framework design. In the case of textual content such as ebooks or smaller articles, abstracts or extracts from the material could be sent on-the-fly prior to payment authorisation. However, the preview of digital content such as processes may require a different form of quality verification such as the use of certificates from authorised certificate authorities. This task in itself is non-trivial and raises challenges for future research. In particular how to guarantee that the previewed material is a part of the encrypted content which shall be delivered. How do you ensure 'services' do what they advertise with regards to ambiguity in service descriptions?

7.3.2 Multiple PKI key pairs

During the course of our research we came across an interesting problem. The P2P environment is ad hoc and most users probably won't encounter the same peer more than once. However, as per our vision the system will develop into a full fledged system for economic content delivery. In that case it will become customary for peers to build up a reputation as sources of good content; hence they will either be discovered intentionally or may be discovered due to their high reputation scores. In that case situations will arise where a buyer and seller have participated in a prior transaction and the buyer is in possession of the seller's public key and this may violate the fair exchange protocol. To overcome this problem, we would need to maintain an individual set of public key pairs per content created or investigate the possibility of using layered PKI. This would not affect our present model for content exchange; however the peers' security management capabilities would have to be extended so it can maintain multiple PKI key pairs. New issues regarding the management of content specific keys arise here, since a user has the potential to be the provider of millions of items for sale novel solutions would be required to deal with the re-use of security keys to reduce the effort of security management. We envisage these extensions to be reflected in the Security Service of our framework, which has been designed for this type of extensibility.

7.3.3 Developing new business models

Most of this research has been focused on the idea for compensation for digital goods while effectively utilising the strengths of the P2P environment. In this scenario we concentrated on the traditional model for content sale, which is based on transfer of right for resale, where the copyright remains with the content producer but the buyer has the right to resell the commodity as long as the original copyright is intact. However we don't charge the buyer for every time he uses the commodity – pay-per-view/use. Our system currently models the pay-once-use-forever

model. In the future we will look into usage based charging, although the essential CPM would remain unchanged, we would have to extend the system to monitor content usage and accounting. One way to accomplish this is through the use of tokens, where the Content Descriptor would contain a predefined number of tokens, which can be set by the content owner. These tokens can be used in combination with the payment separation technique to generate payments every time the content is used locally. However, this raises interesting issues of how to keep the tokens intact once the content has been purchased or distributed for free (i.e. strengthen the watermarking technique). Research could potentially look into methods for reducing the number of tokens as they are used or linking token presence to resource usage. This would require new participation incentives and business models to discourage cheating within the system.

The fair exchange protocol would have to be refined to be more efficient for larger number of transactions originating from the same source. This could possibly be made more efficient through the use of new payment schemes which do not require constant verification for transactions involving the same parties, as well as looking into the implementation of new payment schemes.

7.3.4 Optimum number of services

Another problem we encountered while implementing our framework was deciding on the appropriate number of bank peers which will be required so as to allow the system to function efficiently.

The number of bank peers required will be directly related to the number of transactions in the system. The number of transactions that can be handled by a single bank peer is dependant on its physical capacity in terms of processing power as well as the type of network connectivity it possesses, such as dialup or broadband. These are just some of the parameters which influence the number of transactions a bank peer can participate in at any one time. In our evaluation (§6.3.2 pg 145) we have illustrated that the resource demand on a bank peer for a transaction is negligible, hence it can participate in multiple transactions. Apart from these factors there are also the issues of the amount of local activity on a bank peer aside from its duties as a bank peer, particularly since the bank peers service might be only one of many services running on the peer device. The estimation of the number of services required is a non-trivial problem and would require further research because it is a function of the type of underlying network topology being used in combination with the routing algorithms being used. Other issues which could effect the determination of optimum number of services within the network would also be

the users' desire to act as bank peers or even whether they have the necessary capability to act as one. A possible solution to this could be a higher overlay network that could monitor the system in real-time and identify opportunities for new bank peer creation/activation. This is a new area of research within the P2P field.

7.3.5 Payment schemes requiring online verification

We take into consideration the fact that our framework could be used in areas requiring the exchange of content for large value payments (macropayments) whereby the issue of concrete trust in the system and its entities shall become of paramount importance. At that stage a reputation-based trust management system such as that proposed by Aberer and Despotovic [Karl Aberer 2001b] or PeerTrust [Li Xiong 2002] could be incorporated to add an extra dimension of trust; where the user can make an informed choice on which peer to choose to act as an intermediary. Damiani et al's [Ernesto Damiani 2002] reputation based selection approach can be used to determine whether the other participant (peer) is a source of good quality content. In our present scenario we use payment schemes that do not require online verification. While some payment schemes could be modified to ensure they do not require online verification of payments [John Kelsey 1996, Hitesh Tewari 2003] we do need to adapt the system for payment schemes that do require online verification. A solution to this problem would be Blaze et al's [Matt Blaze 2001] purpose generated certificates as micropayments, however they too rely on an underlying trust management system. Applying these techniques should not affect our base model since it allows extensions in the verification algorithms to adapt to different payment techniques; however the impact of new payment techniques would have to be considered in the future.

We would also need to investigate the impact of the potential for a very low number of available verifiers to exist (giving the illusion of a single party effect and exhibiting all its disadvantages). This can cause a bottleneck in a system which is scalable to millions of nodes and transactions. Basic criteria for the optimum number of services would have to be defined to ensure this does not happen. Guaranteed resource location and new payment schemes which allow verification of payments without unlocking them are issues which would have to be further investigated with reference to this area.

7.3.6 Third party compensation

In our research we focused on compensating content producers for their content and providing an incentive to content distributors to prevent free-riding as well as discourage copyright

violation. However we recognise that for peers to act as bank peers there need to be incentives for them to provide this extra functionality. Although our framework data model has in place the information required to compensate banks for their services the issue of placing atomicity in that transaction has not been covered and will have to be dealt with in future research. The requirements placed on payment schemes and transactions need a third party to support fair exchange. The implications of financially compensating banks (third party) for their functionality are vast and varied. These include questions such as: Who validates the payments meant for the validator? Do we have a chain of banks validating each others payments? This can lead to a daisy chain effect of banks validating banks.

7.4 Final remarks

The novelty of our research lies in the use of cascading payments which compensate content owners every time their content is propagated and also benefit the intermediaries who assist in the propagation and persistence of the content. Our model is based on a royalty and commission basis. This acts as a participation incentive and a deterrent against free-riding [Eytan Adar 2000] which is prevalent in current content sharing P2P systems. Furthermore our research implements a mechanism to cascade payments. This model requires the use of globally persistent identification for peers in a P2P environment along with a method for dispersing the identity with the content. The PayS and the PayD protocols enable cascading payments to be delivered back to the owner and intermediary in fair content exchange transactions.

Due to the potential of scalability in existing P2P systems, it is not possible for the entire P2P network to efficiently function using a single centralised trusted third party. To overcome this problem another novelty of our research lies in the use of an overlay network of peers to act as a bank for the P2P content exchange system, devoid of any centralisation. The implementation of this model requires the use of specialised peers with extra resources; hence a metric for bank peer activation was produced to accommodate the heterogeneity of the P2P environment.

The cascading payments model and the overlay network of bank peers addressed the issues relating to copyright protection and non-repudiation in a commodity exchange in the P2P domain. However, we recognised that diversity is a key feature of any P2P environment and that it is important not to exclude any device from participating in the network. Consequently, we developed a service oriented framework to facilitate our models and which accommodates the characteristics of the P2P environment without detracting from any of its advantages. Furthermore the use of a P2P services oriented framework promotes flexibility and extensibility for the future enhancements to our model. It also encourages the use of other initiatives in

conjunction with our work to improve the performance of our system which would benefit the P2P community as a whole.

Another outcome of this research was a better understanding of the pros and cons of the use of open source projects. For instance due to the immaturity of the JXTA protocols suite at the commencement of this research it was difficult to setup a development environment which remained consistent through the course of this work. Among the many problems encountered when setting up the environment a major one was that, newer versions of the JXTA API and the Java API displayed conflicts. The Netbeans IDE was used as the development interface, this exhibited problems in integration with the JXTA and Java libraries, however towards the end of the implementation lifecycle these problems had been overcome. One reason for this could be the package of the Java SDK with Netbeans as a single install, another could be the fact that the Netbeans IDE has evolved through many versions as well and is more stable now. In retrospect it was felt that open source content though exhibiting many positive benefits for system developers and researchers does have a steeper learning curve due to insufficient documentation and regular unstructured changes. On the other hand using the Apache Derby RDBMS proved easier because it was well documented and comparatively simple to utilise.

P2P technology provides an excellent vehicle for the distribution of digital content at low cost to the content producer and distributor obviating the need for publishing middlemen. However, P2P content sharing networks are flawed with free-riding and copyright infringement. We believe that providing financial participation incentives to users encourages sharing of content and discourages content theft, if users gain access to good quality content at competitive prices which they can use as and when they want they are less likely to wish to break the system which makes this possible. Media sales companies are already moving to the P2P model of content distribution and changing the way they conduct sales and developing new business models for content sale, they are recognising the cost benefits of transferring the production cost of hard media to the user who is willing to absorb this cost if the content is competitively priced.

REFERENCES

[Karl Aberer 2001a] Aberer, K. and Hauswirth, M., "Peer-to-peer information systems: concepts and models, state-of-the-art, and future systems," In Proceedings of *8th European software engineering conference held jointly with 9th ACM SIGSOFT international symposium on Foundations of software engineering*, pp.326 - 327, Vienna, Austria, ACM Press New York, NY, USA, (10-14 September 2001).

[Karl Aberer 2001b] Aberer, K. and Despotovic, Z., "Managing Trust in a Peer-2-Peer Information System," In Proceedings of *Tenth International Conference on Information and Knowledge Management*, pp.310 - 317, Atlanta, Georgia, USA, ACM Press New York, NY, USA, (5-10 November 2001).

[Lada A. Adamic 2001] Adamic, L. A., Lukose, R. M., Puniyani, A. R. and Huberman, B. A., "Search in Power-Law Networks," *Physical Review E*, vol.64 (4), http://arxiv.org/abs/cs/0103016, 2001.

[Eytan Adar 2000] Adar, E. and Huberman, B. A., "Free Riding on Gnutella," *First Monday*, vol.5 (10), pp.Online, Available at http://www.firstmonday.dk/issues/issue5_10/adar/, 2000.

[Sameer Ajmani 2002] Ajmani, S., Clarke, D., Moh, C. and Richman, S., "ConChord: Cooperative SDSI Certificate Storage and Name Resolution," In Proceedings of *1st International Workshop on Peer-to-Peer Systems (IPTPS'02)*, pp.141 - 154, MIT, Cambridge, MA, USA, Springer-Verlag LNCS 2429, (7-8 March 2002).

[Ross J. Anderson 1998] Anderson, R. J. and Petitcolas, F. A. P., "On the Limits of Steganography," *IEEE Journal On Selected Areas In Communications*, vol.16 (4), pp.474-481, 1998.

[Stephanos Androutsellis-Theotokis 2004] Androutsellis-Theotokis, S. and Spinellis, D., "A Survey of Peer-to-Peer Content Distribution Technologies," *ACM Computing Surveys (CSUR)*, vol.36 (4), pp.335-371, 2004.

[Gurleen Arora 2003] Arora, G., Hanneghan, M. and Merabti, M., "CasPaCE: A framework for cascading payments in peer-to-peer digital content exchange," In Proceedings of *4th Annual PostGraduate Networking Conference (PGNet 2003)*, pp.110 -116, Liverpool, UK, (16-17 June 2003).

[Gurleen Arora 2005a] Arora, G., Hanneghan, M. and Merabti, M., "Payment Separation and Distribution for Cascading Payments in P2P Networks," In Proceedings of *6th Annual PostGraduate Symposium on the Convergence of Telecommunications, Networking & Broadcasting (PGNet2005)*, pp.148-154, Liverpool, UK, (27-28 June 2005).

[Gurleen Arora 2005b] Arora, G., Hanneghan, M. and Merabti, M., "P2P Commercial Digital Content Exchange," *Elsevier's Journal on Electronic Commerce Research and Applications*, vol.4 (3), pp.250-263, 2005b.

[Gurleen Arora 2006] Arora, G., Hanneghan, M. and Merabti, M., "P2P Overlay Network to Support E-commerce," In Proceedings of *7th Annual PostGraduate Symposium on the Convergence of Telecommunications, Networking & Broadcasting (PGNet2006)*, pp.55-61, Liverpool, UK, (26-27 June 2006).

[Neal Arthorne 2003] Arthorne, N., Esfandiari, B. and Mukherjee, A., "U-P2P: A Peer-to-Peer Framework for Universal Resource Sharing and Discovery," In Proceedings of *USENIX 2003 Annual Technical Conference - FREENIX track*, pp.29-38, San Antonio, Texas, USA, (9-14 June 2003).

[Yacine Atif 2002] Atif, Y., "Building Trust in E-Commerce," *IEEE Internet Computing*, vol.6 (1), pp.18-24, (Jan-Feb 2002) 2002.

[Hari Balakrishnan 2003] Balakrishnan, H., Kaashoek, M. F., Karger, D., Morris, R. and Stoica, I., "Looking up data in P2P systems," *Communications of the ACM*, vol.46 (2), pp.43-48, 2003.

[Cecily Barnes 2001] Barnes, C., "Intel locks up security code for P2P," Online magazine article, CNET News.com, http://news.com.com/2100-1001-252275.html?legacy=cnet, (8 February 2001).

[BBC News 2005] BBC News, "Software legend joins Microsoft.," BBC, Available at http://news.bbc.co.uk/1/hi/business/4340313.stm, (Accessed: 11 March, 2005).

[Giampaolo Bella 2002] Bella, G., Paulson, L. C. and Massacci, F., "The verification of an industrial payment protocol: the SET purchase phase," In Proceedings of *9th ACM conference on Computer and communications security*, pp.12-20, Washington, DC, USA, ACM Press, (18-22 Nov 2002).

[Peter Biddle 2002] Biddle, P., England, P., Peinado, M. and Willman, B., "The Darknet and the Future of Content Distribution," In Proceedings of *the 2002 ACM Workshop on Digital Rights Management*, pp.155-176, Washington DC, USA, Springer, (18 November 2002).

[Matt Blaze 2001] Blaze, M., Ioannidis, J. and Keromytis, A. D., "Offline Micropayments without Trusted Hardware," In Proceedings of *5th International Conference Financial Cryptography 2001*, pp.21-39, Grand Cayman, Springer Verlag, (19-22 February 2001).

[John Borland 2003] Borland, J., "Stalemate on digital content?," CNET News.com, Available at http://news.com.com/2100-1025-5103601.html, (Accessed: 6 November 2003).

[Sherif Botros 2001] Botros, S. and Waterhouse, S., "Search in JXTA and Other Distributed Networks," In Proceedings of *First International Conference on Peer-to-Peer*

Computing (P2P2001), pp.30-35, Lingköping, Sweden, IEEE Computer Society, (27 - 29 August 2001).

[Jack T. Brassil 1995] Brassil, J. T., Steven Low, Maxemchuk, N. F. and O'Gorman, L., "Electronic Marking and Identification Techniques to Discourage Document Copying," *IEEE Journal on Selected Areas in Communications*, vol.13 (8), pp.1495-1504, 1995.

[Beth Snyder Bulik 2004] Bulik, B. S., "The iPod Economy," *Advertising Age*, vol.75 (42), pp.1-37, (18 October 2004) 2004.

[L. Jean Camp 2002] Camp, L. J., "DRM: doesn't really mean digital copyright management," In Proceedings of *the 9th ACM conference on Computer and communications security*, pp.78 - 87, Washington, DC, USA, ACM Press New York, NY, USA, (18-22 November 2002).

[Miguel Castro 2002] Castro, M., Druschel, P., Kermarrec, A.-M. and Rowstron, A., "SCRIBE: A large-scale and decentralized application-level multicast infrastructure," *IEEE Journal on Selected Areas in Communications: Special issue on Network Support for Multicast Communications*, vol.20 (8), pp.1489 -1499, 2002.

[Michael Champion 2002] Champion, M., Ferris, C., Newcomer, E. and Orchard, D., "Web Services Architecture W3C Working Draft 14 November 2002," W3C, Available at http://www.w3.org/TR/2002/WD-ws-arch-20021114/, (Accessed: November 2002).

[Ellis Chi 1997] Chi, E., "Evaluation of Micropayment Schemes," Technical report, HP Labs, Bristol, HPL-97-14, http://www.hpl.hp.com/techreports/97/HPL-97-14.html, (20 January 1997).

[Ian Clarke 2001] Clarke, I., Sandberg, O., Wiley, B. and Hong, T. W., "Freenet: A Distributed Anonymous Information Storage and Retrieval System," *Designing Privacy Enhancing Technologies: International Workshop on Design Issues in Anonymity and Unobservability*, vol.LNCS 2009, 2001.

[Ian Clarke 2002] Clarke, I., Miller, S. G., Hong, T. W., Sandberg, O. and Wiley, B., "Protecting Free Expression Online with Freenet," *IEEE Internet Computing*, vol.6 (1), pp.40 - 49, http://computer.org/, 2002.

[Bram Cohen 2003] Cohen, B., "Incentives Build Robustness in BitTorrent," In Proceedings of *Workshop on Economics of Peer-to-Peer Systems*, Berkeley, CA, (5-6 June 2003).

[Brian F. Cooper 2002] Cooper, B. F. and Garcia-Molinay, H., "Peer-to-peer data trading to preserve information," *ACM Transactions on Information Systems (TOIS)*, vol.20 (2), pp.133 - 170, 2002.

[Fabrizio Cornelli 2002] Cornelli, F., Damiani, E., Vimercati, S. D. C. d., Paraboschi, S. and Samarati, P., "Implementing a Reputation-Aware Gnutella Servent," In Proceedings of *NETWORKING 2002 Workshops: Web Engineering and Peer-to-Peer Computing*, pp.321-334, Pisa, Italy, (19-24 May 2002).

[Brad Cox 2002] Cox, B., "The MyBank Distributed Digital Rights Management System," Byte.com, Available at https://www.sdmediagroup.com/byte/download.htm, (Accessed: 11 March 2002).

[I. Cox 2000] Cox, I., Miller, M. and Bloom, J., "Watermarking applications and their properties," In Proceedings of *IEEE International Conference on Information Technology: Coding and Computing 2000*, pp.6-10, Las Vegas, USA, (27-29 March 2000).

[Scott Craver 1998] Craver, S., Memon, N., Yeo, B.-L. and Yeung, M. M., "Resolving rightful ownerships with invisible watermarking techniques: Limitations, attacks and implications," *IEEE Transactions on Selected Areas of Communications*, vol.16 (4), pp.573–586, 1998.

[Robert X. Cringely 2006] Cringely, R. X., "Peering into the Future: Why P2P Is the Future of Media Distribution Even If ISPs Have Yet to Figure That Out," PBS Online, Available at http://www.pbs.org/cringely/pulpit/pulpit20060302.html, (Accessed: 2 March 2006).

[Francisco Matias Cuenca-Acuna 2002] Cuenca-Acuna, F. M. and Nguyen, T. D., "Text-Based Content Search and Retrieval in Ad-hoc P2P Communities," In Proceedings of *International Workshop on Peer-to-Peer Computing (co-located with Networking 2002)*, pp.220-234, Pisa, Italy, Springer-Verlag, (19-24 May 2002).

[Kevin Curran 2002] Curran, K., "Peer-to-Peer Networking Collaboration Within Education," *Journal of Educational Multimedia and Hypermedia (JEMH)*, vol.11 (1), pp.21-30, http://www.aace.org/dl/index.cfm/fuseaction/ViewPaper/id/9131/toc/yes, 2002.

[Tom Curran 2001] Curran, T., "E-P2P: the new middleware?," *Middle Spectra*, vol.15 (February), 2001.

[Frank Dabek 2001a] Dabek, F., Brunskill, E., Kaashoek, M. F., Karger, D., Morris, R., Stoica, I. and Balakrishnan, H., "Building peer-to-peer systems with Chord, a distributed location service," In Proceedings of *8th IEEE Workshop on Hot Topics in Operating Systems (HotOS-VIII)*, pp.71–76, Elmau/Oberbayern, Germany, (20-21 May 2001).

[Frank Dabek 2001b] Dabek, F., Kaashoek, M. F., Karger, D., Morris, R. and Stoica, I., "Wide-area cooperative storage with CFS," In Proceedings of *18th ACM Symposium on Operating Systems Principles (SOSP'01)*, pp.202-215, Banff, Canada, (21-24 October 2001).

[Ernesto Damiani 2002] Damiani, E., De Capitani di Vimercati, S., Paraboschi, S., Samarati, P. and Violante, F., "A reputation-based approach for choosing reliable resources in peer-to-peer networks," In Proceedings of *9th ACM conference on Computer and communications security*, pp.207 - 216, Washington, DC, USA, ACM Press (18-22 November 2002).

[Roger Dingledine 2001] Dingledine, R., Freedman, M. J. and Molnar, D., "Chapter 16: Accountability," "Peer-to-Peer: Harnessing the Benefits of a Disruptive Technology." Edited by Andy Oram (March 2001 Edition). USA, O'Reilly & Associates.

[Distributed Search Solutions 2001] Distributed Search Solutions (DSS), "The Gnutella Protocol Specification v0.4 (Document Revision 1.2)," Clip2 Distributed Search Solutions (DSS), Available at http://www9.limewire.com/developer/gnutella_protocol_0.4.pdf, (Accessed: June 2001).

[Judith S. Donath 1998] Donath, J. S., "Identity and deception in the virtual community," "Communities in Cyberspace." Edited by Peter Kollock and Smith, Marc. London, Routledge.

[Ying Dong 2002] Dong, Y., Li, M., Chen, M. and Zheng, S., "Research on intellectual property right problems of peer-to-peer networks," *Electronic Journal*, vol.20 (2), pp.143-150, 2002.

[John R. Douceur 2002] Douceur, J. R., "The Sybil Attack," In Proceedings of *1st International Workshop on Peer-to-Peer Systems (IPTPS'02)*, MIT, Cambridge, MA, USA, Springer-Verlag LNCS 2429, (7-8 March 2002).

[Paola Dubini 2005] Dubini, P. and Raviola, E., "Emerging Business Models in Content Industries," In Proceedings of *8th International Conference on Arts & Cultural Management (AIMAC) 2005*, HEC Montréal, Montreal, Canada, (3-6 July 2005).

[Carol Hovenga Fancher 1997] Fancher, C. H., "In your pocket: smartcards," *IEEE Spectrum*, vol.34 (2), pp.47-53, 1997.

[Paul Fergus 2003] Fergus, P., Mingkhwan, A., Merabti, M. and Hanneghan, M., "DiSUS: Mobile Ad Hoc Network Unstructured Services," In Proceedings of *IFIP-TC6 8th International Conference Personal Wireless Communication (PWC2003)*, pp.484-491, Venice, Italy, (23-25 September 2003).

[Paul Fergus 2005] Fergus, P., Merabti, M., Hanneghan, M. B., Taleb-Bendiab, A. and Minghwan, A., "A Semantic Framework for Self-Adaptive Networked Appliances," In Proceedings of *(CCNC'05) IEEE Consumer Communications & Networking Conference*, pp.229-234, Las Vegas, Nevada, USA, IEEE Computer Society, (3-6 January 2005).

[Gary Gentile 2006] Gentile, G., "Warner Bros. to distribute movies, TV shows via BitTorrent," *USA Today*, http://www.usatoday.com/tech/products/services/2006-05-09-warner-bros-p2p_x.htm?ord=14, (9 May 2006) 2006.

[Jan Gerke 2003] Gerke, J., Hausheer, D., Mischke, J. and Stiller, B., "An Architecture for a Service Oriented Peer-to-Peer System (SOPPS)," *Praxis der Informations-und*

Kommunikationsverarbeitung PIK, vol.2/03 (April), pp.90-95, Available at http://www.mmapps.org/papers/sopps.pdf, 2003.

[Jan Gerke 2005] Gerke, J. and Stiller, B., "A Service-Oriented Peer-to-Peer Middleware," In Proceedings of *Trade conference - Communication in distributed systems 2005 [Fachtagung Kommunikation in Verteilten Systemen 2005] (KiVS 05)*, Technical University of Kaiserslautern, Kaiserslautern, Germany, (March 2005).

[Richard Gold 2001] Gold, R. and Mascolo, C., "Use of Context-Awareness in Mobile Peer-to-Peer Networks," In Proceedings of *the 8th IEEE Workshop on Future Trends of Distributed Computing Systems (FTDCS'2001)*. Bologna, Italy, (31 October - 2 November 2001).

[Nathaniel S. Good 2003] Good, N. S. and Krekelberg, A., "Usability and privacy: a study of KaZaA P2P file-sharing," In Proceedings of *SIGCHI conference on Human factors in computing systems*, pp.137-144, Ft. Lauderdale, Florida, USA, ACM Press, (5-10 April 2003).

[Tyrone Grandison 2000] Grandison, T. and Sloman, M., "A survey of trust in Internet applications," *IEEE Communications Surveys*, vol.Fourth Quarter, pp.2-16, 2000.

[David Greer 2004] Greer, D., "E-micropayments sweat the small stuff," *Computer* (August), pp.19-22, 2004.

[Rüdiger Grimm 2002] Grimm, R. and Nützel, J., "Peer-to-Peer music-sharing with profit but without copy protection," In Proceedings of *Second International Conference on WEB Delivering of Music (WEDELMUSIC'02)*, Darmstadt, Germany, (09 - 11 December 2002).

[Groove Networks 2001] Groove Networks, "Web Services and Peer Services," http://www.groove.net/pdf/web-peer_services.pdf (June 2001).

[Nicholas J. A. Harvey 2003] Harvey, N. J. A., Jones, M. B., Saroiu, S., Theimer, M. and Wolman, A., "SkipNet: A Scalable Overlay Network with Practical Locality Properties," In Proceedings of *Fourth USENIX Symposium on Internet Technologies and Systems (USITS '03)*, Seattle, WA. USA, (26-28 March 2003).

[David Hausheer 2003] Hausheer, D. and Stiller, B., "Design of a distributed P2P-based content management middleware," In Proceedings of *29th Conference on EUROMICRO 2003*, pp.173- 180, Belek, Turkey, IEEE, (1-6 September 2003).

[Steven Hazel 2002] Hazel, S. and Wiley, B., "Achord: A Variant of the Chord Lookup Service for Use in Censorship Resistant Peer-to-Peer," In Proceedings of *1st International Workshop on Peer-to-Peer Systems (IPTPS'02)*, MIT, Cambridge, MA, USA, Springer-Verlag, (7-8 March 2002).

[Bill Horne 2001] Horne, B., Pinkas, B. and Sander, T., "Escrow services and incentives in peer-to-peer networks," In Proceedings of *3rd ACM conference on Electronic Commerce*, pp.85 - 94, Tampa, Florida, USA, ACM, (14-17 October 2001).

[Hung-Chang Hsiao 2003] Hsiao, H. C. and King, C. T., "A Tree Model for Structured Peer-to-Peer Protocols," In Proceedings of *3rd IEEE/ACM International Symposium on Cluster Computing and the Grid (CCGRID'03)*, pp.336-343, Tokyo, Japan, IEEE Computer Society, (12 - 15 May 2003).

[Nick Huber 2003] Huber, N., "Financial industry unites to create XML payments standard to cut processing costs," *Computer Weekly*(December), pp.16, http://www.computerweekly.com, (2 December 2003) 2003.

[Renato Iannella 1998] Iannella, R., "An Idiot's Guide to the Resource Description Framework," *The New Review of Information Networking*, vol.4, (3 September 1998) 1998.

[IETF 2002] IETF, "Session Initiation Protocol (SIP), RFC 3261," Available at http://www.ietf.org/rfc/rfc3261.txt, (Accessed: Jan 2003).

[M. Izal 2004] Izal, M., Urvoy-Keller, G., Biersack, E. W., Felber, P. A., Hamra, A. A. and Garcés-Erice, L., "Dissecting BitTorrent: Five Months in a Torrent's Lifetime," In Proceedings of *5th International Workshop Passive and Active Network Measurement (PAM 2004)*, pp.1 - 11, Antibes Juan-les-Pins, France, Springer-Verlag GmbH, (19-20 April 2004).

[Srinivasan Jagannathan 2002] Jagannathan, S., Nayak, J., Almeroth, K. and Hofmann, M., "A Model for Discovering Customer Value for E-Content;," In Proceedings of *8th ACM SIGKDD international conference on Knowledge discovery and data mining*, pp.532 - 537, Edmonton, Alberta, Canada, ACM Press New York, NY, USA, (23-26 July 2002).

[Neil F. Johnson 1999] Johnson, N. F., Duric, Z. and Jajodia, S., "A Role for Digital Watermarking in Electronic Commerce," *ACM Computer Surveys*, 1999.

[Steve Jones 2005] Jones, S., "Toward an Acceptable Definition of Service," *IEEE Software*, vol.22 (3), pp.87-93, (May 2005) 2005.

[Sam Joseph 2002] Joseph, S., "NeuroGrid: Semantically Routing Queries in Peer-to-Peer Networks," In Proceedings of *NETWORKING 2002 Workshops, Web Engineering and Peer-to-Peer Computing*, pp.202-214, Pisa, Italy, (19-24 May 2002).

[Sepandar D. Kamvar 2003] Kamvar, S. D., Schlosser, M. T. and Garcia-Molina, H., "The EigenTrust Algorithm for Reputation Management in P2P Networks," In Proceedings of *12th international conference on World Wide Web (WWW2003)*, pp.640-651, Budapest, Hungary, ACM, (20-24 May 2003).

[Krishna Kant 2002] Kant, K., Iyer, R. and Tewari, V., "A Framework for Classifying Peer-to-Peer Technologies," In Proceedings of *2nd International Workshop on Global and Peer-to-Peer Computing on Large Scale Distributed Systems in IEEE International*

Symposium on Cluster Computing and the Grid (CCGrid'2002), pp.368-375, Berlin-Brandenburg Academy of Sciences and Humanities Berlin, Germany, (May 21 - 24, 2002).

[John Kelsey 1996] Kelsey, J. and Schneier, B., "A Peer-to-Peer Software Metering System," In Proceedings of *The Second USENIX Workshop on Electronic Commerce*, pp.279 - 286, Oakland, California, (18-21 November 1996).

[Rita E. Knox 2003] Knox, R. E. and Silver, M. A., "Vendors take two paths to XML-enabled office suites," Research Report, Gartner Research, Available at http://techrepublic.com.com/5100-6296-5074929.html, (Accessed: 2 October 2003).

[Eric Korpela 2001] Korpela, E., Werthimer, D., Anderson, D., Cobb, J. and Lebofsky, M., "SETI@home-massively distributed computing for SETI," *Computing in Science & Engineering*, vol.3 (1), pp.78 - 83, Available at http://ieeexplore.ieee.org/search/wrapper.jsp?arnumber=895191, 2001.

[John Kubiatowicz 2000] Kubiatowicz, J., Bindel, D., Chen, Y., Czerwinski, S., Eaton, P., Geels, D., Gummadi, R., Rhea, S., Weatherspoon, H., Weimer, W., Wells, C. and Zhao, B., "OceanStore: An architecture for global-scale persistent storage," In Proceedings of *9th international Conference on Architectural Support for Programming Languages and Operating Systems (ASPLOS 2000)*, pp.190–201, Cambridge, MA, (12-15 November 2000).

[Claudia Leopold 2001] Leopold, C., "Parallel and Distributed Computing: a survey of models, paradigms and approaches," Wiley, New York, USA, 0471358312.

[J. Li 2000] Li, J., Jannotti, J., De Couto, D., Karger, D. and Morris, R., "A scalable location service for geographic ad hoc routing," In Proceedings of *6th ACM International Conference on Mobile Computing and Networking*, pp.120–130, Boston, Massachusetts, (6-11 August 2000).

[Jin Li 2002] Li, J., "PeerStreaming: A Practical Receiver-Driven Peer-to-Peer Media Streaming System," In Proceedings of *9th International Conference on Parallel and Distributed Systems (ICPADS2002)*, pp.157-162, National Central University, Taiwan, (17-20 December 2002).

[LinkPro Technologies 2003] LinkPro Technologies, LinkPro Technologies, "Web Site Mirroring From A Staging Server To Multiple Web Servers," White Paper, http://www.linkpro.com/pdf/lpwp_Web%20Site%20Replication.pdf.

[Jiangchuan Liu 2003] Liu, J., Zhang, X., Li, B., Zhang, Q. and Zhu, W., "Distributed distance measurement for large-scale networks," *Computer Networks: The International Journal of Computer and Telecommunications Networking*, vol.41 (2), pp.177 - 192, 2003.

[S. M. Lui 2002] Lui, S. M., Lang, K. and Kwok, S., "Participation Incentive Mechanisms in Peer-to-Peer Subscription Systems," In Proceedings of *35th Annual Hawaii*

International Conference on System Sciences (HICSS'02), pp.3925-3931, Big Island, Hawaii, IEEE Computer Society, (7-10 January 2002).

[Qin Lv 2002] Lv, Q., Cao, P., Cohen, E., Li, K. and Shenker, S., "Search and replication in unstructured peer-to-peer networks," In Proceedings of *16th international conference on Supercomputing*, pp.84-95, New York, USA, ACM Press, (22-26 June 2002).

[Mallik Mahalingam 2003] Mahalingam, M., Tang, C. and Xu, Z., "Towards a Semantic, Deep Archival File System," In Proceedings of *9th International Workshop on Future Trends of Distributed Computing Systems (FTDCS 2003)*, pp.115-121, San Juan, Puerto Rico, (28-30 May 2003).

[Dahlia Malkhi 2002] Malkhi, D., Naor, M. and Ratajczak, D., "Viceroy: A Scalable and Dynamic Emulation of the Butterfly," In Proceedings of *21st annual ACM symposium on Principles of Distributed Computing (PODC)*, pp.183 - 192, Monterey, California, ACM Press New York, NY, USA, (July 2002).

[Gary Marshall 2006] Marshall, G., "The fight against DRM," *PCPlus*(249), pp.83-87, http://www.pcplus.co.uk/home, (December 2006) 2006.

[Petar Maymounkov 2002] Maymounkov, P. and Mazieres, D., "Kademlia: A Peer-to-peer Information System Based on the XOR Metric," In Proceedings of *the 1st International Workshop on Peer-to-Peer Systems (IPTPS'02)*, pp.53 - 65, MIT, (7-8 March 2002).

[Declan McCullagh 2000] McCullagh, D., "Get Your Music Mojo Working," Online Newspaper, Wired News, Available at http://www.wired.com/news/technology/0,1282,37892,00.html, (Accessed: 29 July 2000).

[Madjid Merabti 2005] Merabti, M., Abuelma'atti, O. and Fergus, P., "Networked Appliances and Home Networking," In Proceedings of *1st International Workshop on the Ubiquitous Home*, Kyoto University, Japan, (March 2005).

[Silvio Micali 2002] Micali, S. and Rivest, R. L., "Micropayments Revisited," *Cryptography Track at the RSA Conference*, vol. LNCS 2271, pp.149-163, (2002) 2002.

[Matt L. Miller 1998] Miller, M. L., Cox, I. J. and Bloom, J. A., "Watermarking in the real world: An application to DVD," In Proceedings of *Multimedia and Security-Workshop at ACM Multimedia'98 (GMD Report 41)*, pp.71-76, Bristol, U.K., (12-16 September 1998).

[Anirach Mingkhwan 2004] Mingkhwan, A., Fergus, P., Abuelma'atti, O. and Merabti, M., "Implicit Functionality: Dynamic Services Composition for Home Networked Appliances," In Proceedings of *IEEE International Conference on Communications (ICC'2004)*, pp.43-47, Paris, France, IEEE Computer Society, (20-24 June 2004).

[Daniel Minoli 1997] Minoli, D. and Minoli, E., "Chapter 4: Electronic Cash and Electronic Payment Schemes," "Web Commerce Technology Handbook (McGraw-Hill Series on

Computer Communication)." Edited by S. Elliot. New York, McGraw-Hill Osborne Media: pp. 145-175.

[Mirabilis Ltd. 1996] Mirabilis Ltd., "E.T. Surf Home: Mirabilis Ltd. Provides New Solutions for Peer-to-Peer Internet Communications," Tel Aviv, Available at http://company.icq.com/info/press/press_release2.html, (Accessed: 2002).

[David Molnar 2000] Molnar, D., "The SETI@home Problem " *ACM Crossroads: The Student Journal of the Association for Computing Machinery* (September), http://www.acm.org/crossroads/columns/onpatrol/september2000.html, 2000.

[Deirdre K. Mulligan 2003] Mulligan, D. K., Han, J. and Burstein, A. J., "How DRM-Based Content Delivery Systems Disrupt Expectations of "Personal Use"," In Proceedings of *3rd ACM Workshop On Digital Rights Management*, pp.77-89, Washington, DC, USA, ACM Press, (27 October 2003).

[Napster 2003] Napster, "The Napster Homepage," Available at www.napster.com, (Accessed: 2003).

[Bonnie A. Nardi 2000] Nardi, B. A., Whittaker, S. and Bradner, E., "Interaction and Outeraction: Instant Messaging In Action," *Computer Supported Cooperative Work*, pp.79-88, 2000.

[Wolfgang Nejdl 2002] Nejdl, W., Wolf, B., Qu, C., Decker, S., Sintek, M., Naeve, A., Nilsson, M., Palmer, M. and Risch, T., "EDUTELLA: A P2P Networking Infrastructure Based on RDF," In Proceedings of *the 11th International World Wide Web Conference (WWW2002)*, pp.604-615, Honolulu, Hawaii, USA, ACM Press, (7-11 May 2002).

[Network Associates Inc. 2000] Network Associates Inc., "An Introduction to Cryptography," Guide, Network Associates, Inc. 3965 Freedom Circle, Santa Clara, CA 95054, Available at ftp://ftp.pgpi.org/pub/pgp/7.0/docs/english/IntroToCrypto.pdf, (Accessed: January 2003).

[T. S. Eugene Ng 2002] Ng, T. S. E. and Zhang, H., "Predicting Internet Network Distance with Coordinates-Based Approaches," In Proceedings of *21st Annual Joint Conference of the IEEE Computer and Communications Societies(INFOCOM'02)*, New York, NY, (23-27 June 2002).

[Jason Nieh 2000] Nieh, J., Yang, S. J. and Novik, N., "A Comparison of Thin-Client Computing Architectures," Columbia University, Technical Report CUCS-022-00, (November 2000).

[Venkata N. Padmanabhan 2002] Padmanabhan, V. N., Wang, H. J., Chou, P. A. and Sripanidkulchai, K., "Distributing streaming media content using cooperative networking," In Proceedings of *12th international workshop on Network and operating systems support for digital audio and video*, Miami, Florida, USA, (12-14 May 2002).

[Vijay Pande 2000] Pande, V., "folding@home Distributed Computing," Stanford University, Available at http://folding.stanford.edu/, (Accessed: 2002).

[Parabon Computation 2000] Parabon Computation, "Compute Against Cancer Project," Available at http://www.parabon.com/cac.jsp; http://www.computeagainstcancer.org/, (Accessed: 2000).

[Jon M. Peha 2004] Peha, J. M. and Khamitov, I. M., "PayCash: a secure efficient internet payment system," *Electronic Commerce Research and Applications*, vol.3 (4), pp.381-388, (Winter 2004) 2004.

[Russell Perry 2002] Perry, R. and Williamson, M. M., "A Licensing and Payment System for Distribution of Digital Content," Technical Report, Information Infrastructure Laboratory, HP Laboratories Bristol, Bristol, HPL-2002-167, http://www.hpl.hp.com/techreports/2002/HPL-2002-167.pdf, (12 June 2002).

[Johan Pouwelse 2004] Pouwelse, J., "The BitTorrent P2P file-sharing system," Available at http://www.theregister.co.uk/2004/12/18/bittorrent_measurements_analysis/, (Accessed: 18 December 2004).

[Lee Rainie 2005] Rainie, L., Madden, M. and Pew Internet & American Life Project, "Podcasting catches on," Pew Research Center, Washington, Available at http://www.pewinternet.org/pdfs/PIP_podcasting.pdf, (April 2005).

[Kavitha Ranganathan 2002] Ranganathan, K., Iamnitchi, A. and Foster, I., "Improving Data Availability through Dynamic Model-Driven Replication in Large Peer-to-Peer Communities," In Proceedings of *Global and Peer-to-Peer Computing on Large Scale Distributed Systems Workshop*, pp.376-381, Berlin, Germany, (21 - 24 May 2002).

[Sylvia Ratnasamy 2001] Ratnasamy, S., Karp, R., Francis, P., Handley, M. and Shenker, S., "A Scalable Content-Addressable Network," In Proceedings of *ACM Special Interest Group on Data Communications (ACM Sigcomm2001)*, pp.161-172, San Diego, California, USA, (27-31 August 2001).

[Matei Ripeanu 2002a] Ripeanu, M., Foster, I. and Iamnitchi, A., "Mapping the Gnutella Network: Properties of Large-Scale Peer-to-Peer Systems and Implications for System Design," *IEEE Internet Computing Journal special issue on peer-to-peer networking*, vol.6 (1), 2002a.

[Matei Ripeanu 2002b] Ripeanu, M., Iamnitchi, A. and Foster, I., "Mapping the Gnutella Network," *IEEE Internet Computing*, vol.6 (1), pp.50 - 57, http://computer.org/, 2002b.

[Ronald Rivest 1996] Rivest, R. and Shamir, A., "PayWord and MicroMint: Two Simple Micropayment Schemes," In Proceedings of *1996 International Workshop on Security Protocols*, pp.69-87, Cambridge, United Kingdom, Springer 1997, (10-12 April 1996).

[Ronald L. Rivest 2004] Rivest, R. L., "Peppercoin Micropayments," In Proceedings of *Financial Cryptography '04*, pp.2-8, Springer, (9 February 2004).

[Lawrence G. Roberts 2000] Roberts, L. G., "Beyond Moore's law: Internet growth trends," *Computer*, vol.33 (1), pp.117-119, 2000.

[Antony Rowstron 2001a] Rowstron, A. and Druschel, P., "Pastry: Scalable, distributed object location and routing for large-scale peer-to-peer systems," In Proceedings of *18th IFIP/ACM International Conference on Distributed Systems Platforms(Middleware 2001)*, pp.329-350, Heidelberg, Germany, (12-16 November 2001).

[Antony Rowstron 2001b] Rowstron, A. and Druschel, P., "PAST: A large-scale, persistent peer-to-peer storage utility," In Proceedings of *HotOS VIII*, pp.75-80, Schoss Elmau, Germany, (20-23 May 2001).

[Antony Rowstron 2001c] Rowstron, A. and Druschel, P., "Storage management and caching in PAST, a large-scale, persistent peer-to-peer storage utility," In Proceedings of *18th ACM Symposium on Operating Systems Principles (SOSP'01)*, pp.188-201, Lake Louise, Alberta, Canada, (21- 24 October 2001).

[William T. Rupp 2004] Rupp, W. T. and Smith, A. D., "Exploring the impacts of P2P networks on the entertainment industry," *Information Management & Computer Security*, vol.12 (1), pp.102-116, 2004.

[Peter Saint-Andre 2002] Saint-Andre, P., "Jabber protocol approved for use as an IETF instant messaging and presence technology," *Jabber Journal* (3), http://www.jabber.org/journal/, 2002.

[Peter Saint-Andre 2004] Saint-Andre, P., "Jabber's Extensible Messaging and Presence Protocol (XMPP) instant messaging and presence protocol - IETF RFC 3920, Request for Comments, http://www.ietf.org/rfc/rfc3920.txt, (October 2004).

[Clare Saliba 2000] Saliba, C., "Napster, Bertelsmann To Develop Paid Service," Online News, NewsFactor Network, Available at http://www.newsfactor.com/perl/story/4703.html, (Accessed: 1 November 2000).

[R.R. Schaller 1997] Schaller, R. R., "Moore's law: past, present and future," *IEEE Spectrum*, vol.34 (6), pp.52-59, 1997.

[Rüdiger Schollmeier 2002] Schollmeier, R. and Schollmeier, G., "Why Peer-to-Peer (P2P) Does Scale: An Analysis of P2P Traffic Patterns," In Proceedings of *IEEE International Conference on Peer-to-Peer Computing (P2P2002)*, pp.112 -119, Linköping, Sweden, IEEE, (5-7 September 2002).

[Keri Schreiner 2001] Schreiner, K., "Distributed projects tackle protein mystery," *IEEE Computing in Science & Engineering*, vol.3 (1), pp.13-16, 2001.

[Mark Shelstad 2005] Shelstad, M., "Content matters: analysis of a website redesign," *OCLC Systems & Services*, vol.21 (3), pp.209 - 225, (Sep 2005) 2005.

[Clay Shirky 2001] Shirky, C., "Chapter 2: Listening to Napster," "Peer-to-Peer: Harnessing the Benefits of a Disruptive Technology." Edited by Andy Oram. USA, O'Reilly & Associates.

[Curt Simmons 2002] Simmons, C. and Rofail, A., "Microsoft .NET Platform and Technologies," Pearson Education, Prentice Hall Inc., Upper Saddle River, NJ, USA, 1st, 0130341789, (12 January 2002).

[Tyron Stading 2002] Stading, T., Maniatis, P. and Baker, M., "Peer-to-Peer Caching Schemes to Address Flash Crowds," In Proceedings of *1st International Workshop on Peer-to-Peer Systems (IPTPS'02)*, pp.203 - 213, MIT, (7-8 March 2002).

[Felix Stalder 1999] Stalder, F. and Clement, A., "Exploring Policy Issues of Electronic Cash: The Mondex Case," *Canadian Journal Of Communication*, vol.24 (2), pp.261-288, http://www.cjc-online.ca, 1999.

[William Stallings 2006] Stallings, W., "Cryptography and network security : principles and practices," Pearson Prentice Hall, Upper Saddle River, N.J., 4th, 0131873164 (2006).

[Thad Starner 2002] Starner, T., "Thick clients for personal wireless devices," *Computer* (January), pp.133-135, 2002.

[Ion Stoica 2001] Stoica, I., Morris, R., Karger, D., Kaashoek, M. F. and Balakrishnan, H., "Chord: A Scalable Peer-to-peer Lookup Service for Internet Applications," In Proceedings of *ACM Special Interest Group on Data Communications (ACM Sigcomm2001)*, pp.149-160, San Diego, California, USA, (27-31 August 2001).

[Damien Stolarz 2001] Stolarz, D., "Peer to peer streaming media delivery," In Proceedings of *the First International Conference on Peer-to-Peer Computing (P2P2001)*, pp.48-52, Lingköping, Sweden, IEEE Computer Society, (27 - 29 August 2001).

[Lee S. Strickland 2003] Strickland, L. S., "Copyright's Digital Dilemma Today: Fair Use or Unfair Constraints?," *Bulletin of the American Society for Information Science and Technology*, vol.30 (1), pp.7-11, 2003.

[Bomil Suh 2002] Suh, B. and Han, I., "Effect of trust on customer acceptance of Internet banking," *Electronic Commerce Research and Applications Journal*, vol.1 (3-4), pp.247-263, 2002.

[Andrew S. Tanenbaum 2002] Tanenbaum, A. S. and Steen, M. v., "Distributed systems : principles and paradigms " Prentice Hall Pearson Education International, International edition, 0131217860.

[Hitesh Tewari 2003] Tewari, H. and O'Mahony, D., "Multiparty Micropayments for Ad Hoc Networks," In Proceedings of *IEEE Wireless Communications and Networking Conference WCNC03*, pp.2033-2040, New Orleans, Louisiana, USA, (16-20 March 2003).

[Stephen Thomas 2000] Thomas, S., "SSL and TLS Essentials: Securing the Web," Publisher: John Wiley & Sons Inc, New York, NY, USA, 0471383546, (2000).

[Peter Thorsteinson 2003a] Thorsteinson, P. and Ganesh, G., "Chapter 3: Symmetric Cryptography," ".NET Security and Cryptography." Edited by Karen Gettman (1st Edition). NJ, USA, Prentice Hall PTR: 496pgs.

[Peter Thorsteinson 2003b] Thorsteinson, P. and Ganesh, G., "Chapter 4: Asymmetric Cryptography," ".NET Security and Cryptography." Edited by Karen Gettman (1st Edition). NJ, USA, Prentice Hall PTR: 496pgs.

[Bernard Traversat 2003] Traversat, B., Arora, A., Abdelaziz, M., Duigou, M., Haywood, C., Hugly, J. C., Pouyoul, E. and Yeager, B., "Project JXTA 2.0 Super-Peer Virtual Network," White Paper, Sun Microsystems, Inc., http://www.jxta.org/project/www/docs/JXTA2.0protocols1.pdf, (25 May 2003).

[G. Winfield Treese 1999] Treese, G. W. and Stewart, L. C., "Chapter 14: Payment Systems," "Designing Systems for Internet Commerce." Edited by Karen Gettman. Reading, Massachusetts, Addison Wesley Longman Inc.

[World Trade Organization 2003] TRIPS (trade-related aspects of intellectual property rights) Council, "TRIPS Agreement: What are 'intellectual property rights'?," World Trade Organization, Geneva, Available at http://www.wto.org/english/tratop_e/trips_e/intel1_e.htm, (Accessed: 2003).

[Efraim Turban 1999] Turban, E., Lee, J., King, D. and Chung, H. M., "Chapter 8: Electronic Payment Systems," "Electronic Commerce: A Managerial Perspective." Edited by Efraim Turban. New Jersey, Prentice Hall.

[Sherry Turkle 1995] Turkle, S., "Life on the Screen: Identity in the Age of the Internet," Simon & Schuster.

[J. D. Tygar 1996] Tygar, J. D., "Atomicity in Electronic Commerce," In Proceedings of *15th Annual ACM Symposium on Principles of Distributed Computing*, pp.8-26, Philadelphia, Pennsylvania, United States, ACM Press New York, NY, USA, (23-26 May 1996).

[Jon Udell 2001] Udell, J., Asthagiri, N. and Tuvell, W., "Chapter 18: Security," "Peer-to-Peer: Harnessing the Benefits of a Disruptive Technology." Edited by Andy Oram. USA, O'Reilly & Associates.

[Florian Unterkircher 2002] Unterkircher, F. and Welzl, M., "Reliable Unicast Streaming with Multiple Peers in IP Networks," In Proceedings of *NETWORKING 2002 Workshops: Web Engineering and Peer-to-Peer Computing*, pp.252-259, Pisa, Italy, Springer-Verlag, (19-24 May 2002).

[Steve Vinoski 2004] Vinoski, S., "Web services notifications," *IEEE Internet Computing*, vol.8 (2), pp.86- 90, 2004.

[Marc Waldman 2000] Waldman, M., Rubin, A. D. and Cranor, L. F., "Publius: A robust, tamper-evident, censorship-resistant web publishing system," In Proceedings of *the 9th USENIX Security Symposium*, pp.59-72, Denver, Colorado, USA, (14-17 August 2000).

[Marc Waldman 2001] Waldman, M., Cranor, L. F. and Rubin, A., "Chapter 15: Trust," "Peer-to-Peer: Harnessing the Benefits of a Disruptive Technology." Edited by Andy Oram. USA, O'Reilly & Associates.

[C. Wang 2002] Wang, C., Carzaniga, A., Evans, D. and Wolf, A., "Security Issues and Requirements for Internet-Scale Publish-Subscribe Systems," In Proceedings of *35th Annual Hawaii International Conference on System Sciences (HICSS'02)*, pp.303, Big Island, Hawaii, IEEE Computer Society, (7-10 January 2002).

[Hakim Weatherspoon 2002] Weatherspoon, H. and Kubiatowicz, J. D., "Erasure Coding vs. Replication: A Quantitative Comparison," In Proceedings of *1st International Workshop on Peer-to-Peer Systems (IPTPS'02)*, MIT, (7-8 March 2002).

[Howard Wen 2002] Wen, H., "Internet Radio the P2P Way," Available at http://www.openp2p.com/pub/a/p2p/2002/09/24/p2pradio.html, (Accessed: 24 September 2002).

[Brendon J. Wilson 2002] Wilson, B. J., "JXTA," Newriders, Indianapolis, 0-73571-234-4.

[Li Xiong 2002] Xiong, L. and Liu, L., "Building Trust in Decentralized Peer-to-Peer Electronic Communities," In Proceedings of *International Conference on Electronic Commerce Research (ICECR-5)*, Montreal, Canada, (23-27 October 2002).

[Li Xiong 2003] Xiong, L. and Liu, L., "A Reputation-Based Trust Model for Peer-to-Peer eCommerce Communities," In Proceedings of *IEEE Conference on E-Commerce (CEC'03)*, Newport Beach, USA, (24-27 June 2003).

[Beverly Yang 2003] Yang, B. and Garcia-Molina, H., "PPay: Micropayments for Peer-to-Peer Systems," In Proceedings of *10th ACM conference on Computer and communication security*, pp.300-310, Washington D.C., USA, (27-30 October 2003).

[Nezer J. Zaidenberg 2002] Zaidenberg, N. J. and Malka, L., "Providing High Quality Streaming While Leveraging Bandwidth Usage," White Paper, vTrails Ltd., Tel Aviv, http://www.vtrails.com, (May 2002).

[N. Zhang 2003] Zhang, N., Shi, Q. and Merabti, M., "An efficient protocol for anonymous and fair document exchange," *Computer Networks*, vol.41 (1), pp.19–28, 2003.

[Ben Y. Zhao 2001] Zhao, B. Y., Kubiatowicz, J. and Joseph, A. D., "Tapestry: An Infrastructure for Fault-tolerant Wide-area Location and Routing," Technical Report, Computer Science Division, University of California, Berkeley, UCB/CSD-01-1141, (April 2001).

[Ben Y. Zhao 2002] Zhao, B. Y., Yitao Duan, Huang, L., Anthony D. Joseph and Kubiatowicz, J. D., "Brocade: Landmark Routing on Overlay Networks," In Proceedings of *the 1st*

International Workshop on Peer-to-Peer Systems (IPTPS'02), pp.34 - 44, MIT, (7-8 March 2002).

APPENDICES

Appendix A. Use case model

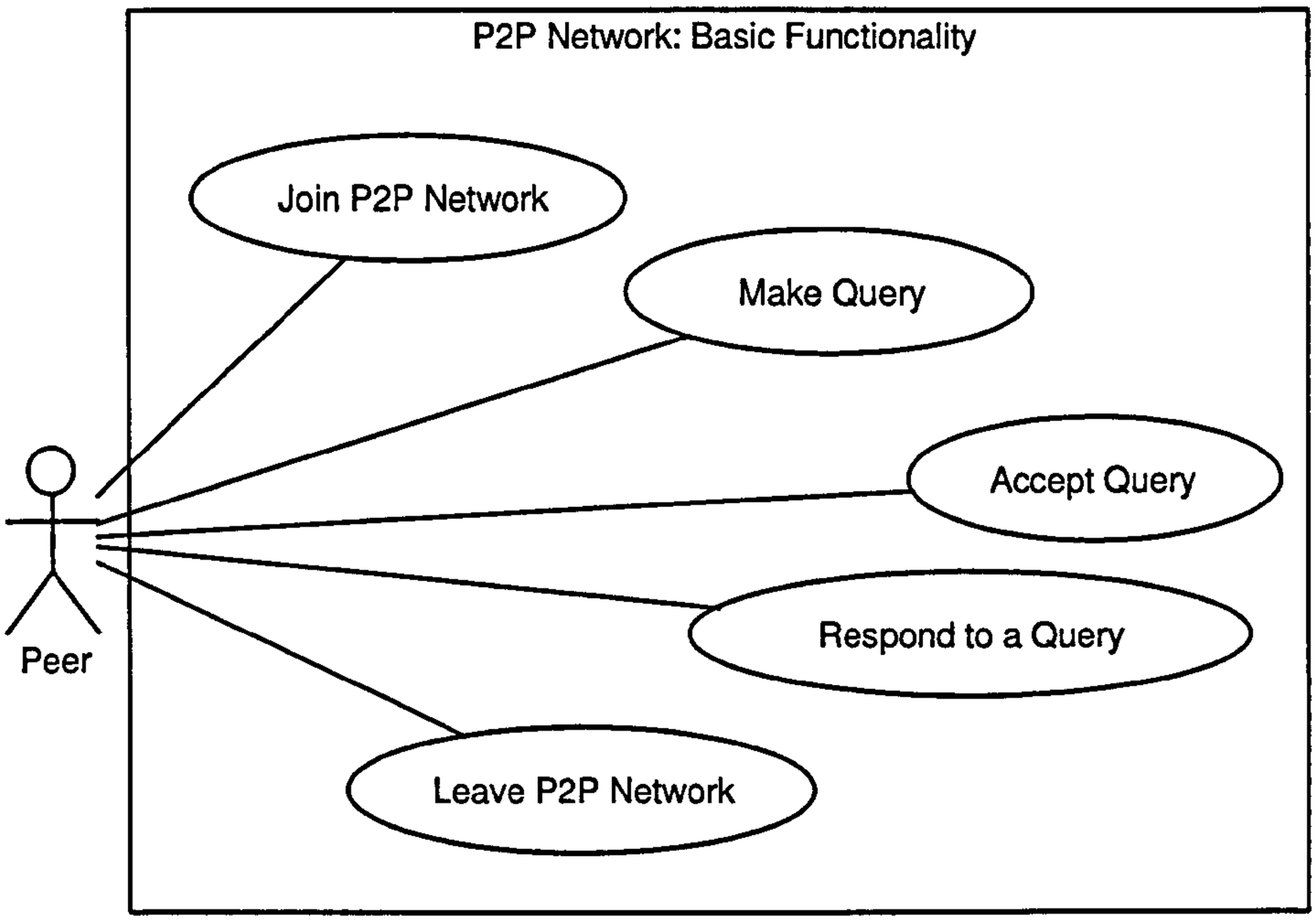


Figure A. 1: P2P system basic functionality

Description:

This Use Case illustrates the typical functionality of a normal peer within a P2P system. Our framework resides within a P2P network; hence it is a sub-system of the P2P Network.

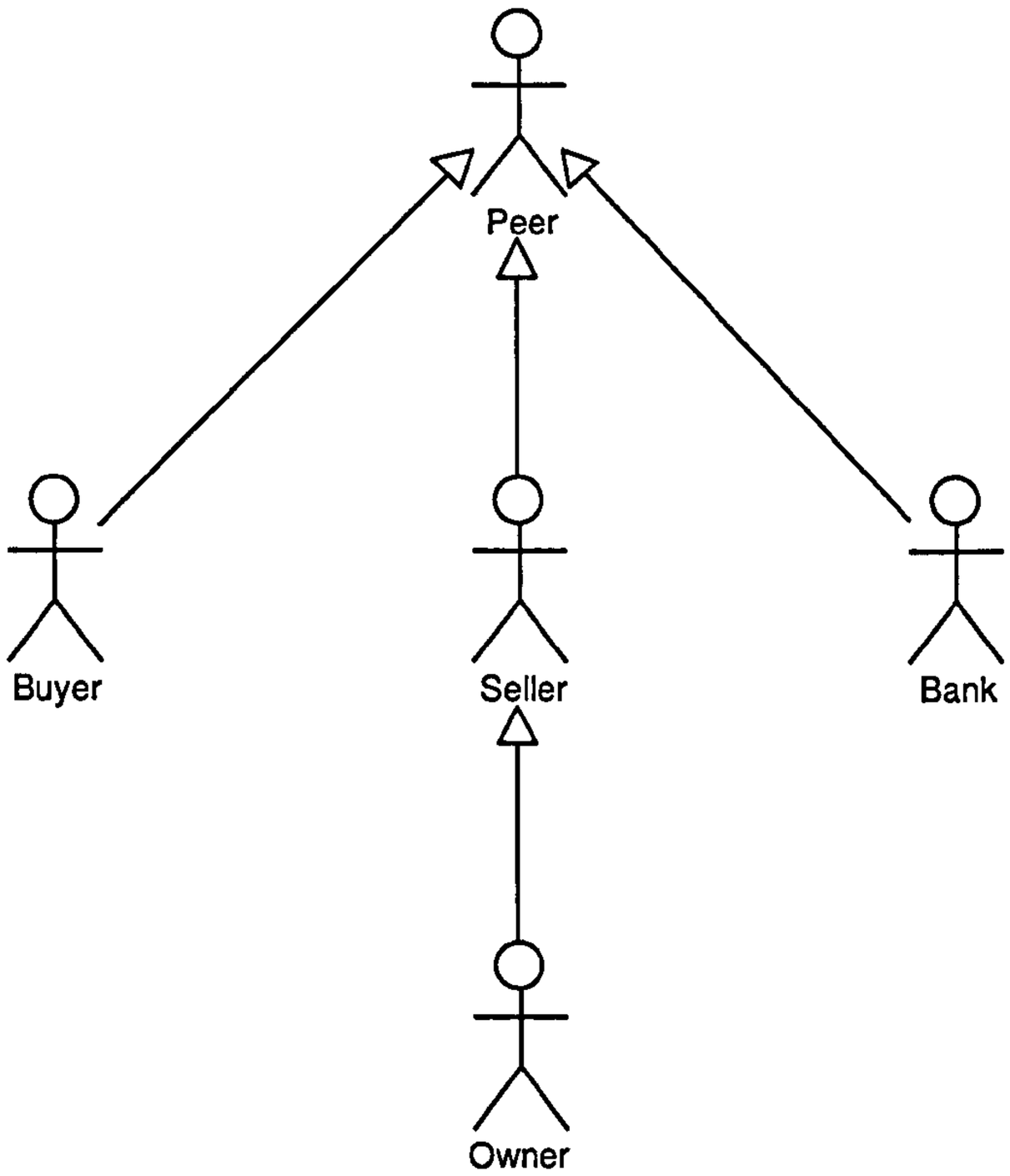


Figure A. 2: P2P payment system actors

Description:

This Use Case illustrates the different peers and their hierarchy within this framework.

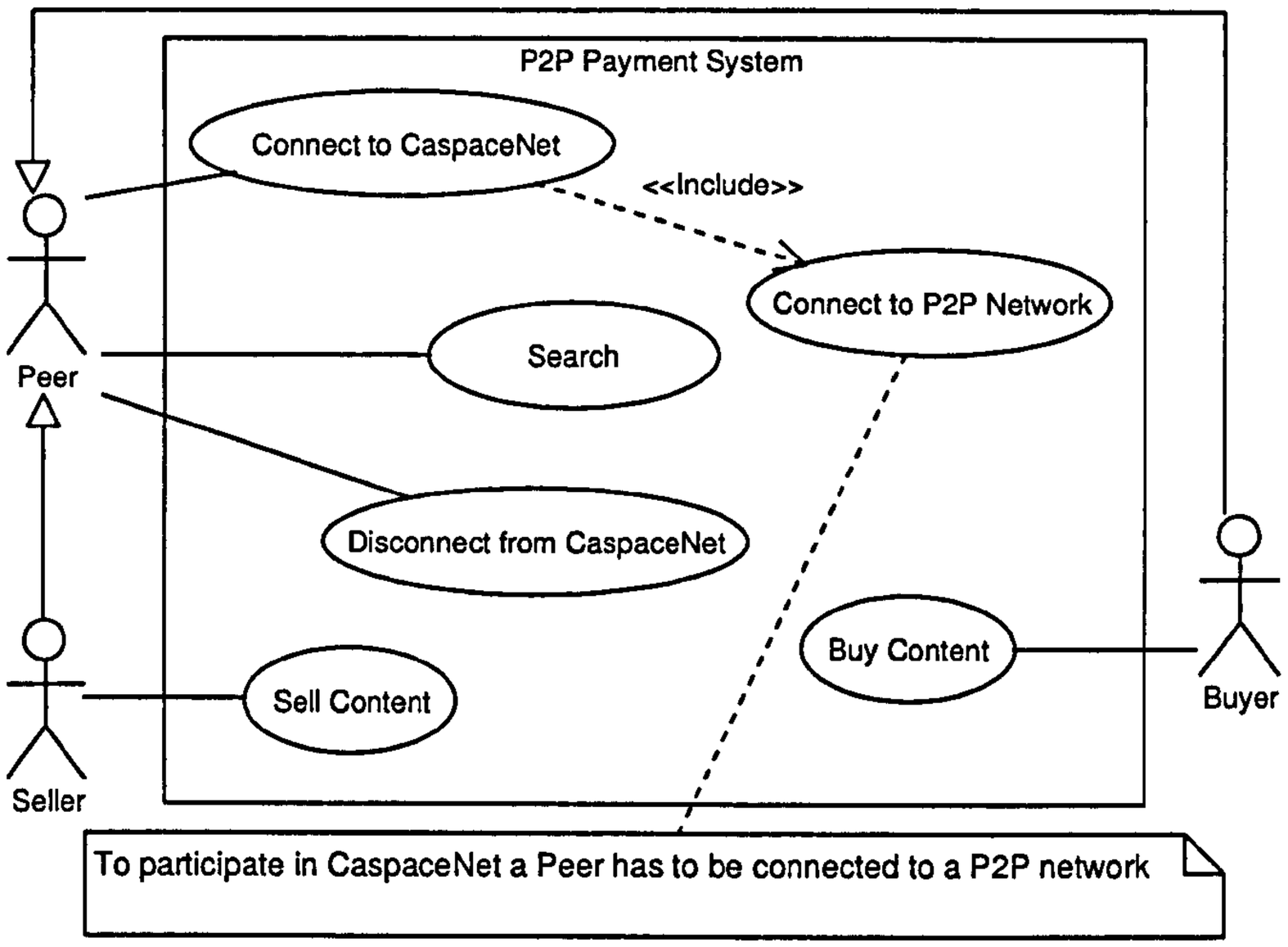


Figure A. 3: P2P payment system

Description:

This Use Case gives a high-level overview of the P2P payment system in this framework.

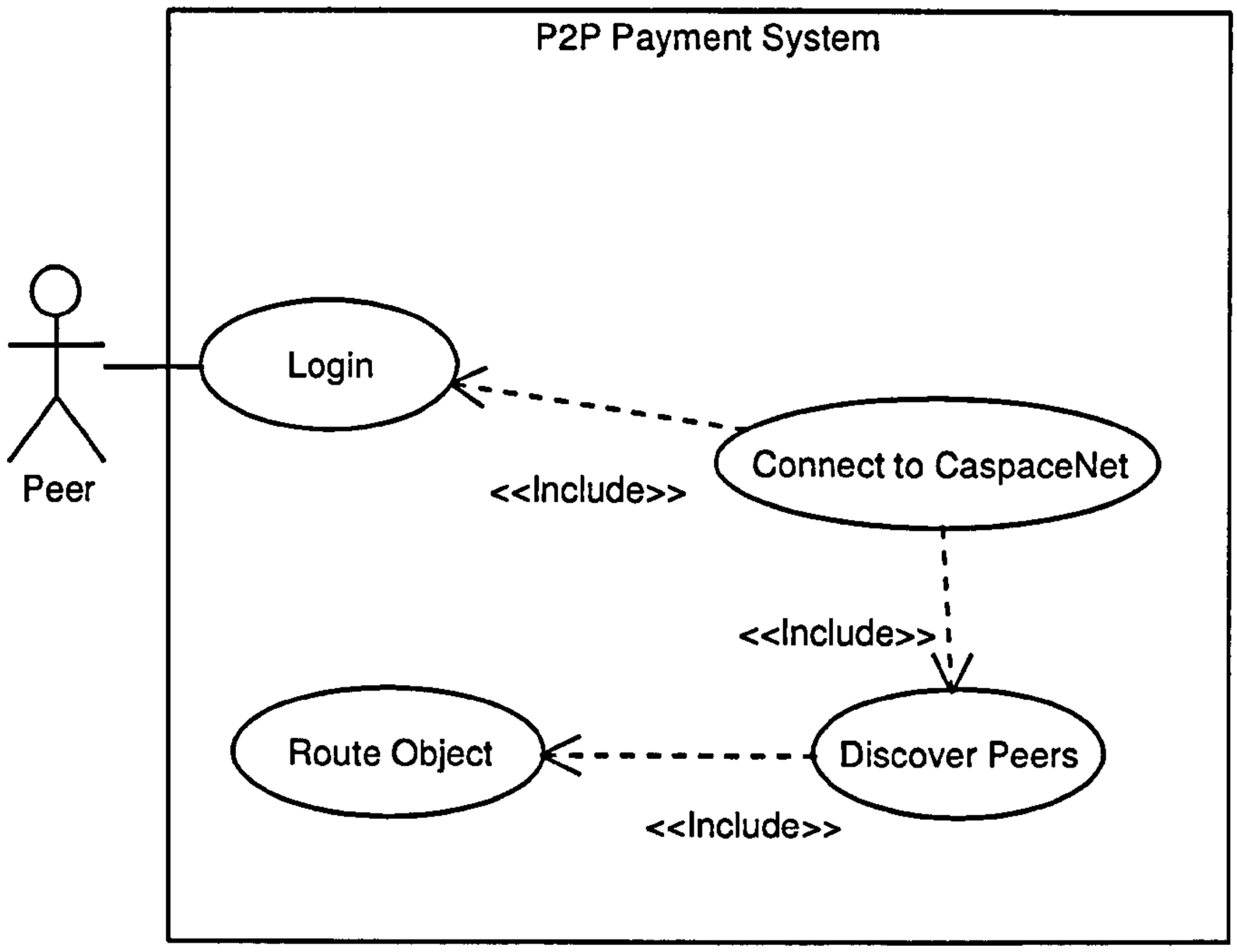


Figure A. 4: Connect to P2P network

Description:

This Use Case illustrates a scenario when a peer comes online.

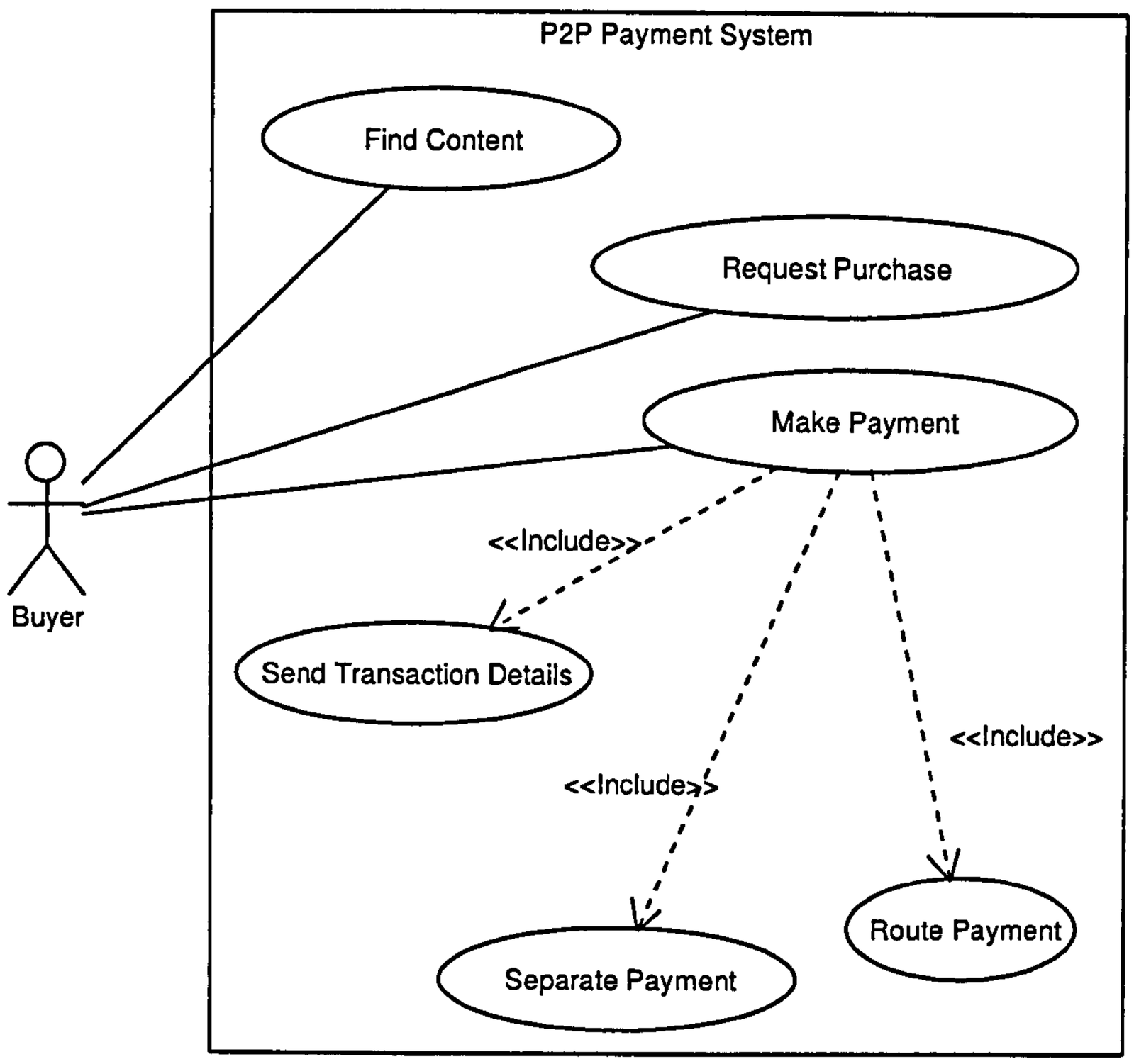


Figure A. 5: Buy content

Description:

This Use Case illustrates a typical scenario when a peer wishes to buy content. The peer in this scenario is a Buyer.

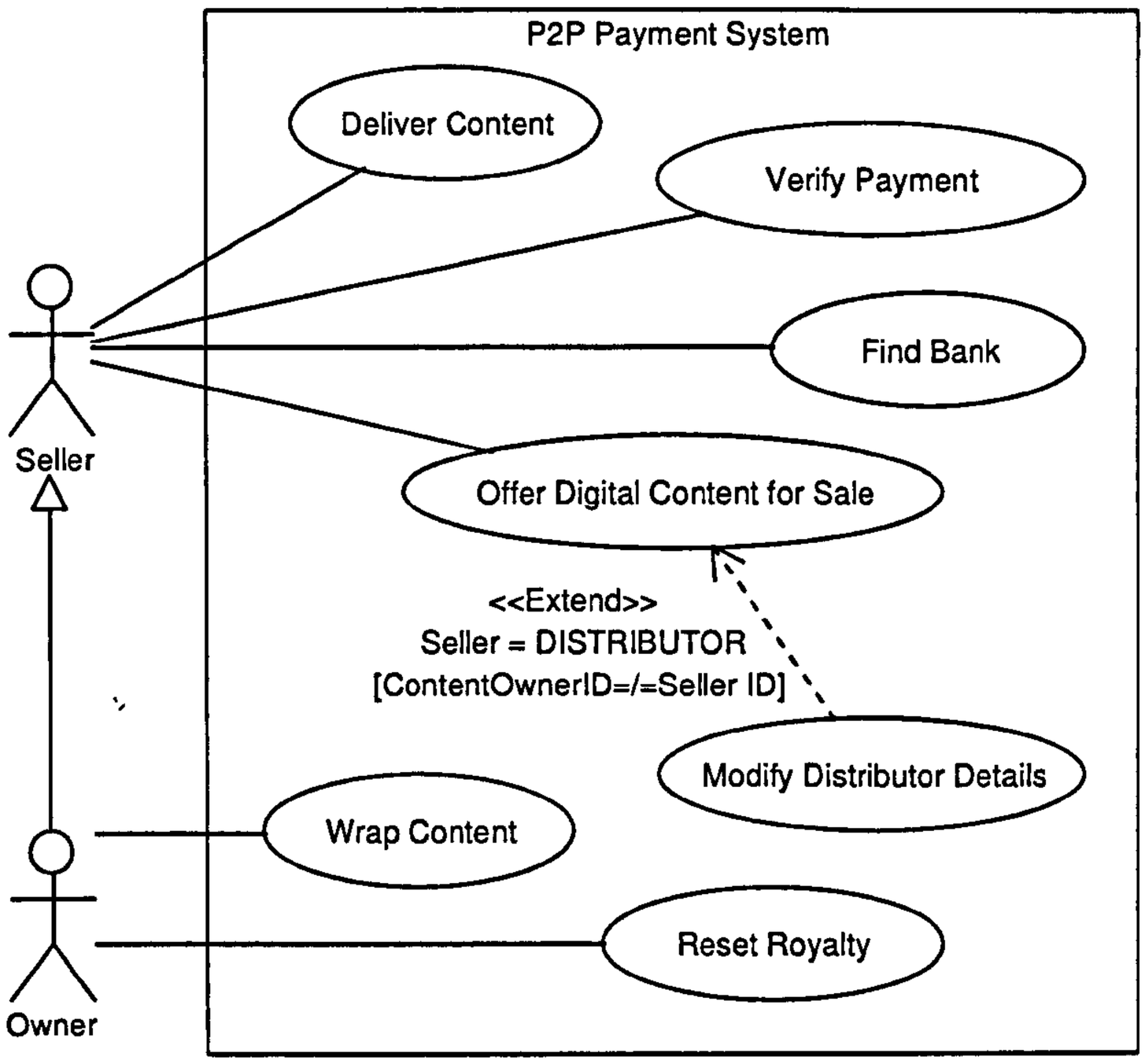


Figure A. 6: Sell content

Description:

This Use Case illustrates a typical scenario when a seller/owner wishes to sell content.

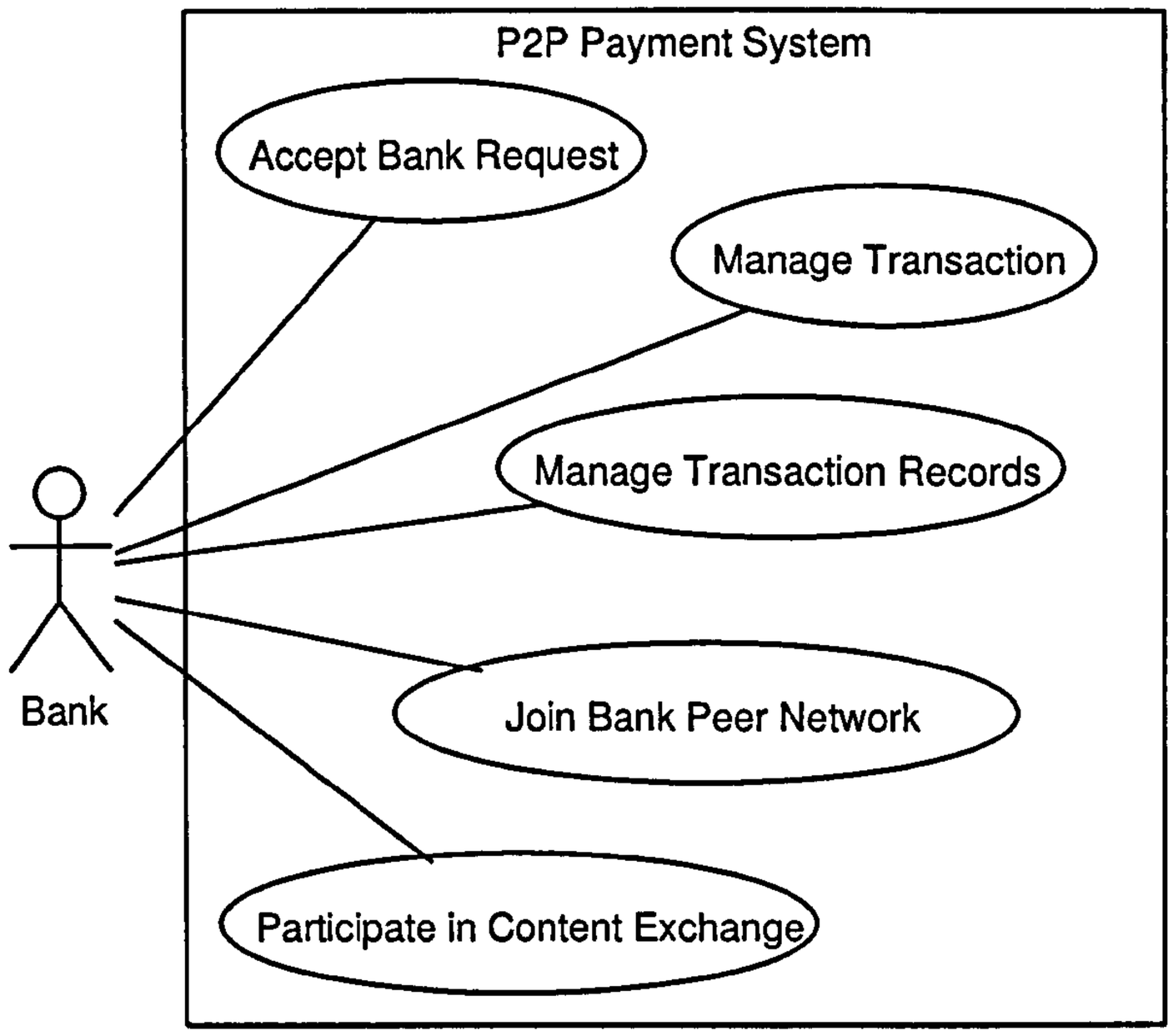


Figure A. 7: Bank peer functionality

Description:

This Use Case illustrates the typical functionality of a bank peer within this framework.

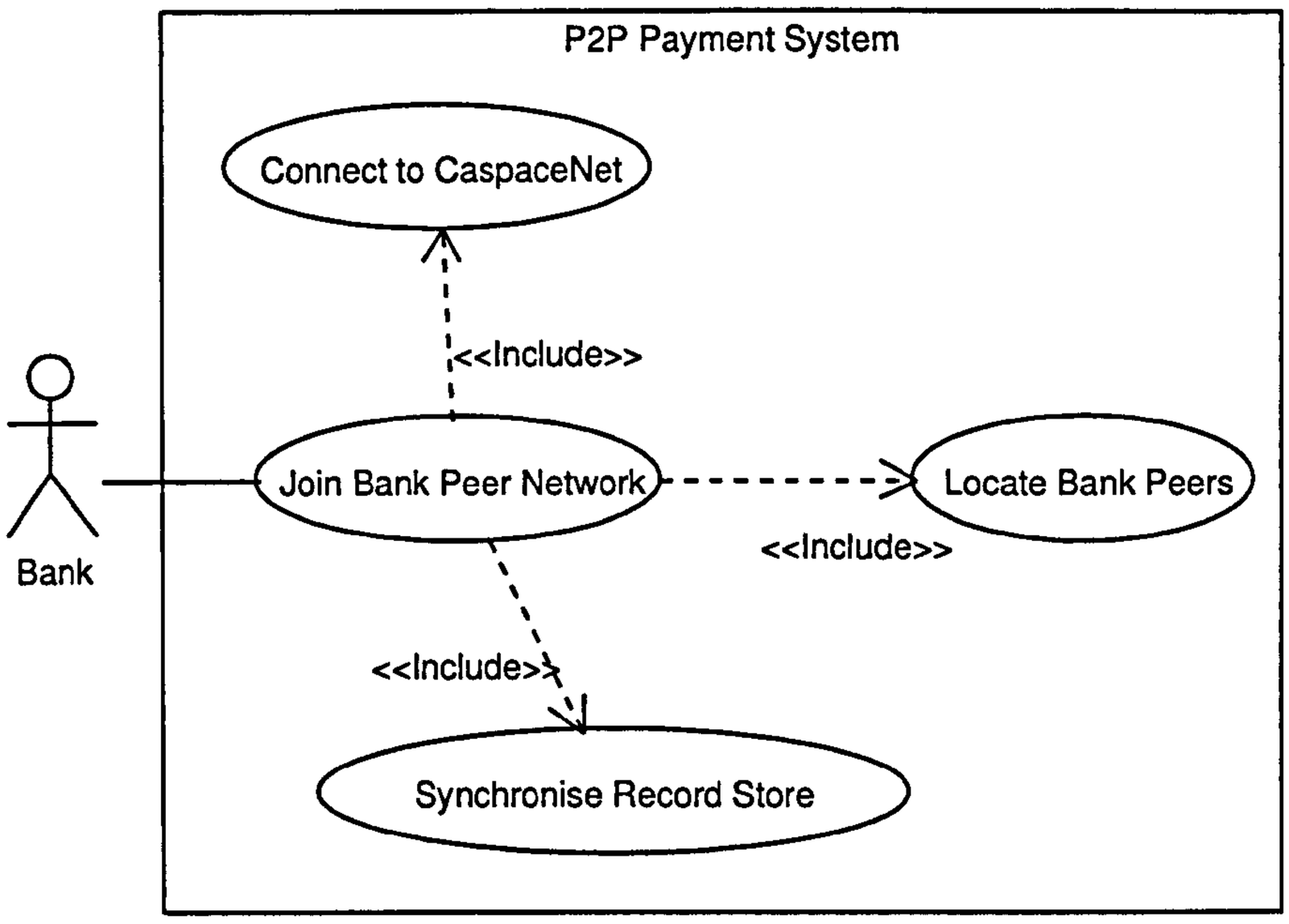


Figure A. 8: Join bank peer network

Description:

This Use Case illustrates a typical scenario when a bank peer joins the bank peer network within this framework.

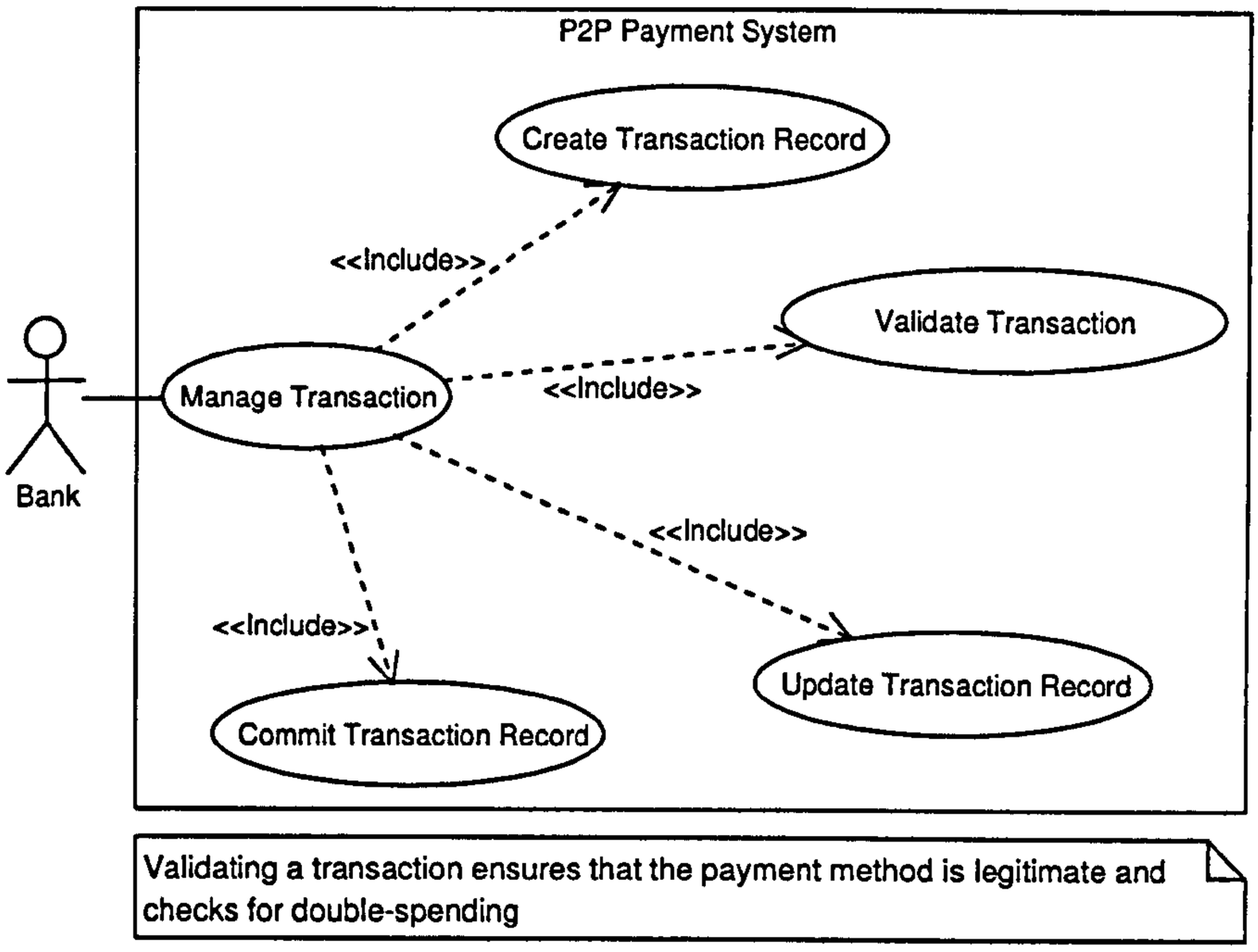


Figure A. 9: Manage transaction

Description:

This Use Case illustrates a typical scenario when a bank peer participates in a content exchange transaction.

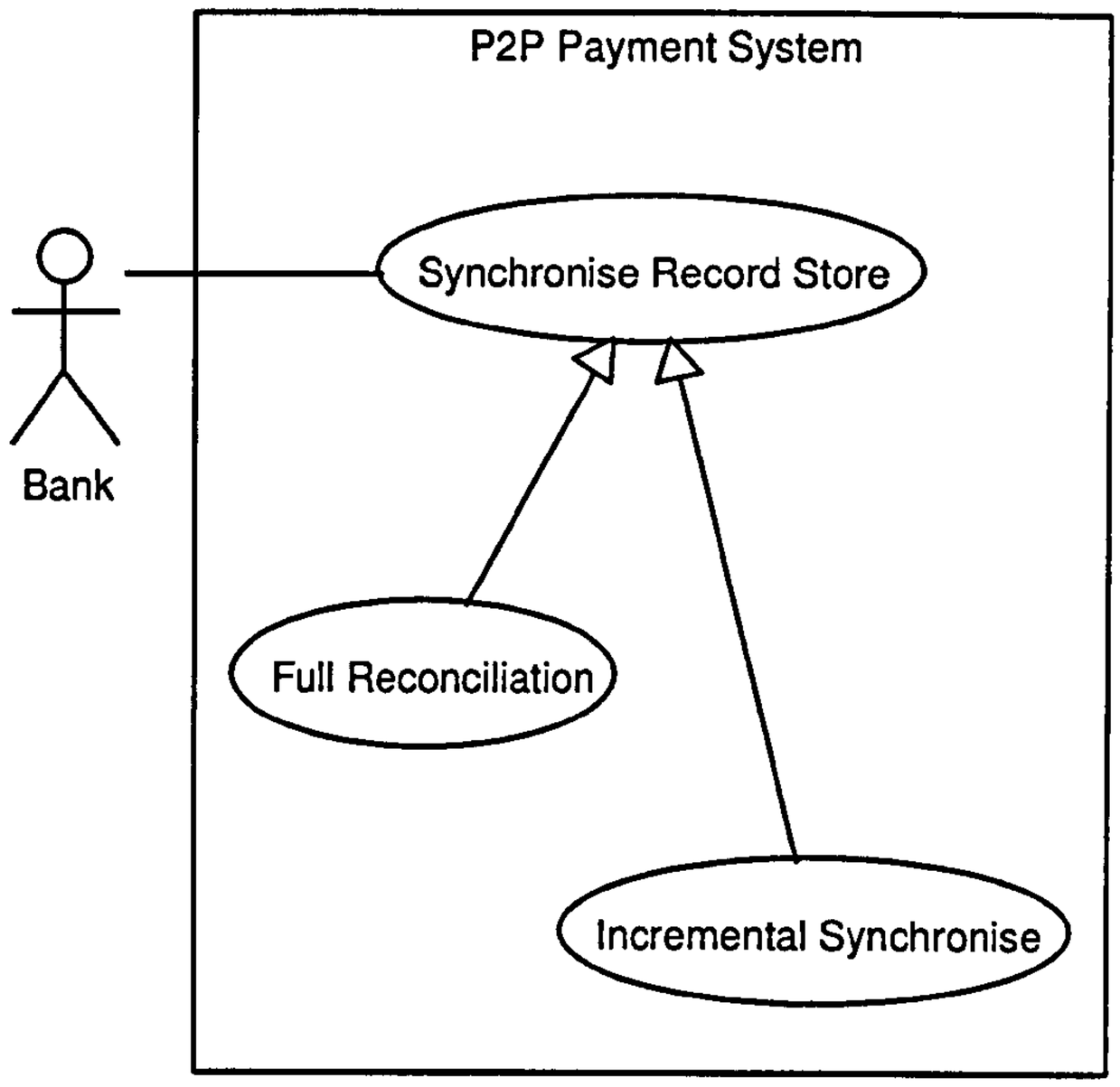


Figure A. 10: Manage transaction records

Description:

This Use Case illustrates a scenario when a bank peer has to synchronise its transaction record store while managing its data store. A full reconciliation occurs at startup and incremental synchronise is performed at regular intervals.

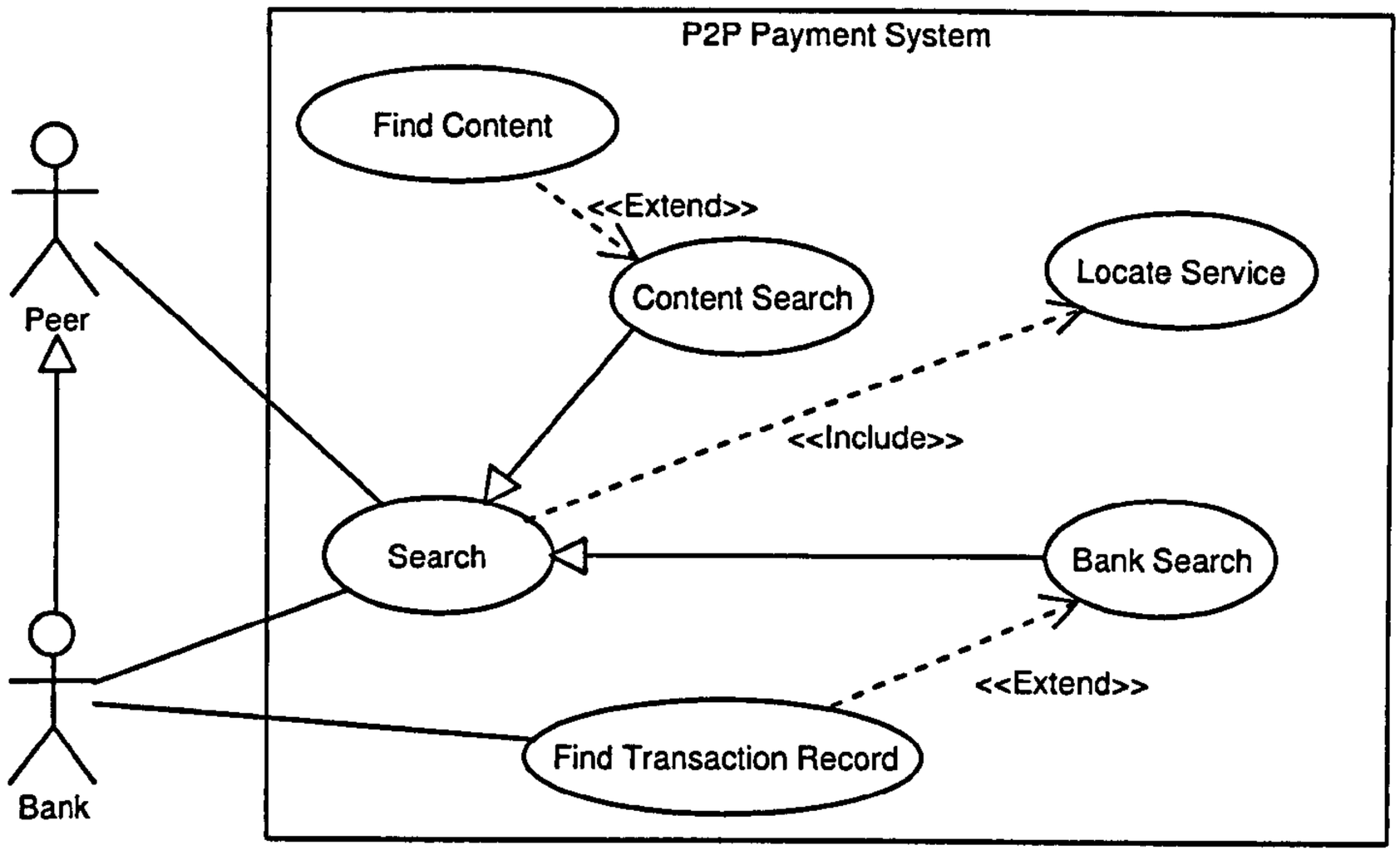


Figure A. 11: Search

Description:

This Use Case illustrates the different search scenarios possible within this framework.

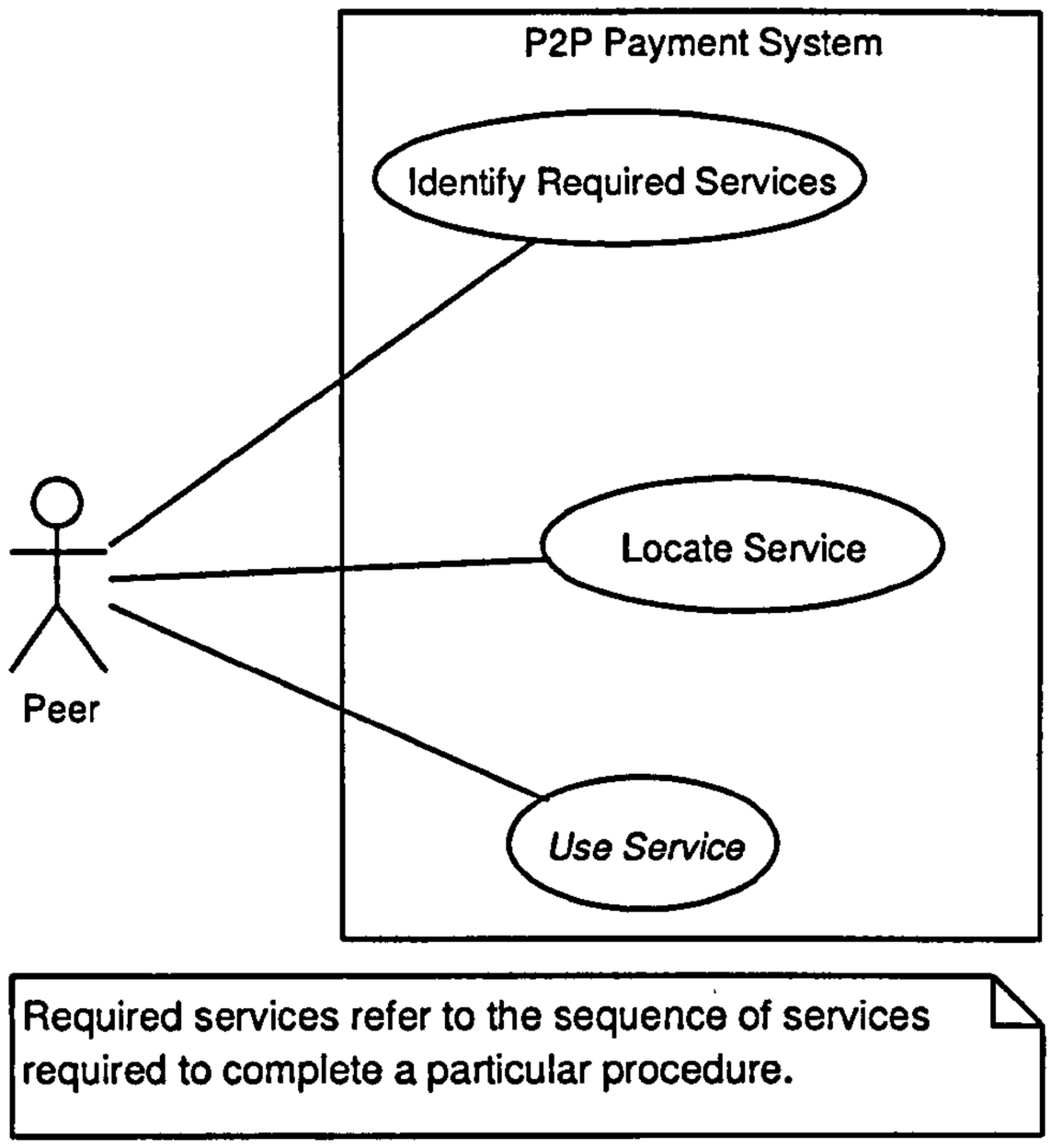


Figure A. 12: Service location and utilisation

Description:

This Use Case illustrates a typical scenario when a peer wishes to use a service within this framework.

e.g. Function$_1$ => S_1, S_2, S_3, S_4..... S_n

Where S_1, S_2, S_3.....S_n belong to the set of services and Function refers to desired functionality that is achieved by the utilisation of these services in a specific order.

Appendix B. Activity diagrams

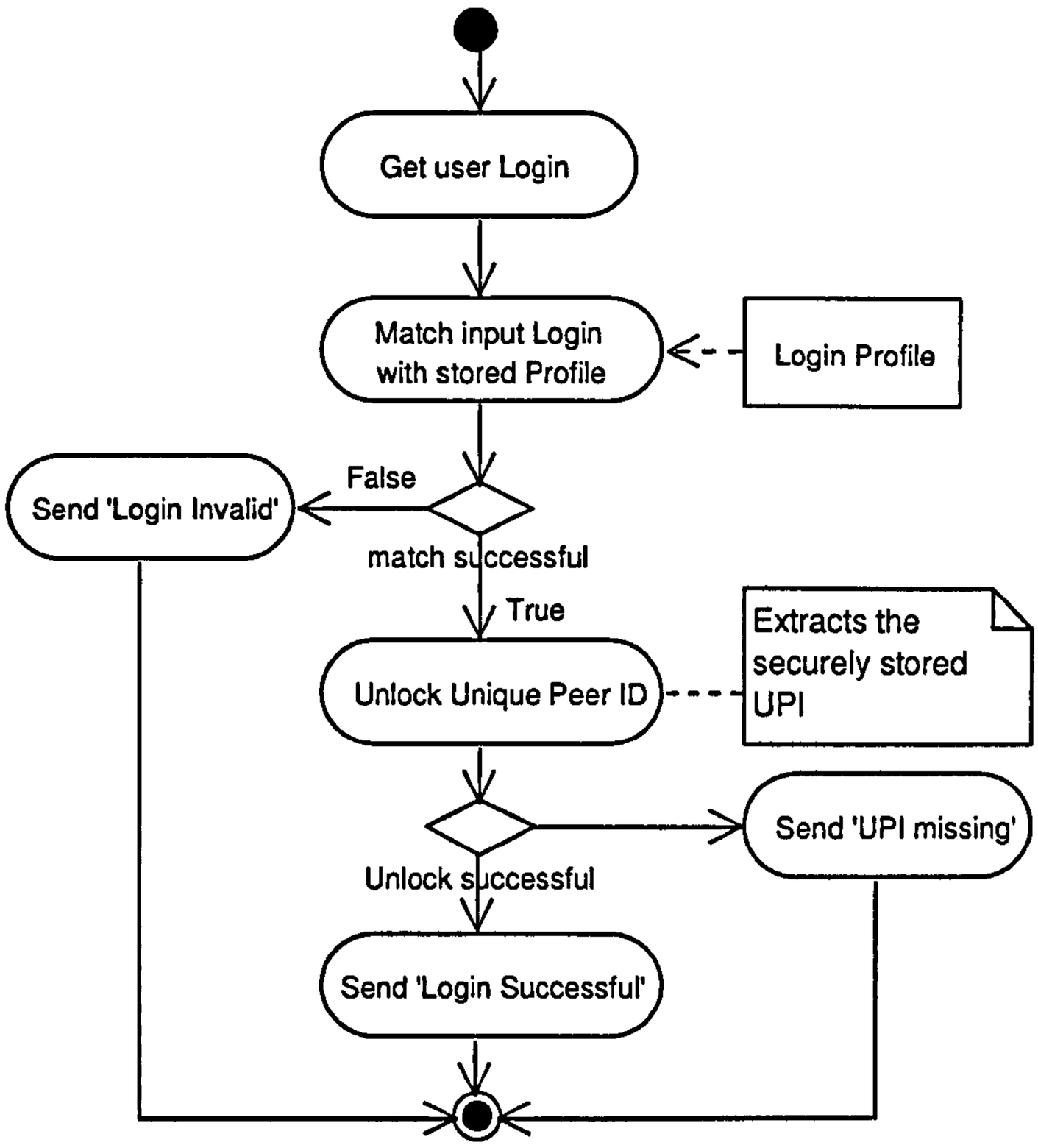


Figure B. 1: Authenticate User

Description:

This Activity Diagram illustrates what happens at user login.

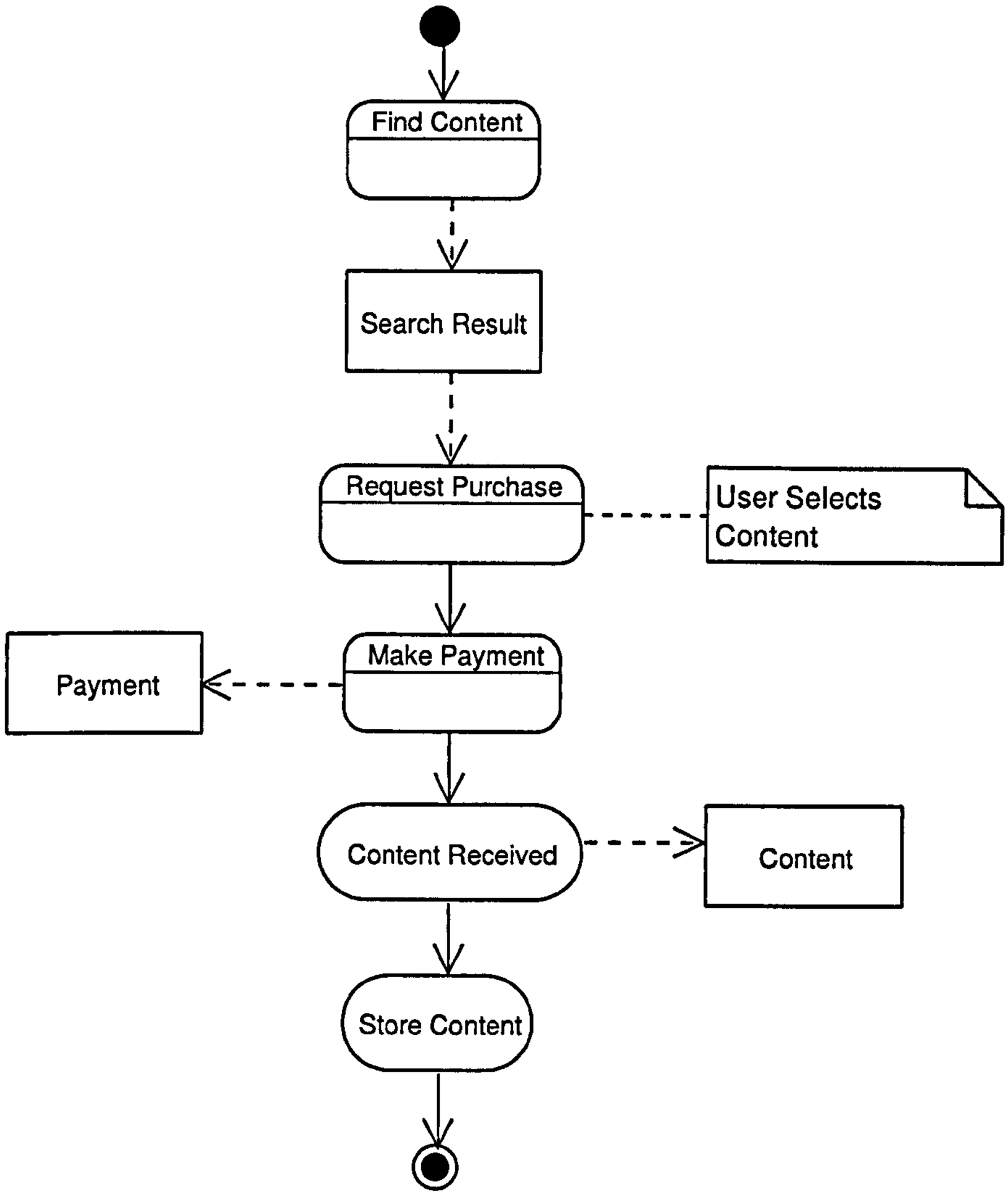


Figure B. 2: Buy content

Description:

This Activity Diagram illustrates a content purchase within the framework.

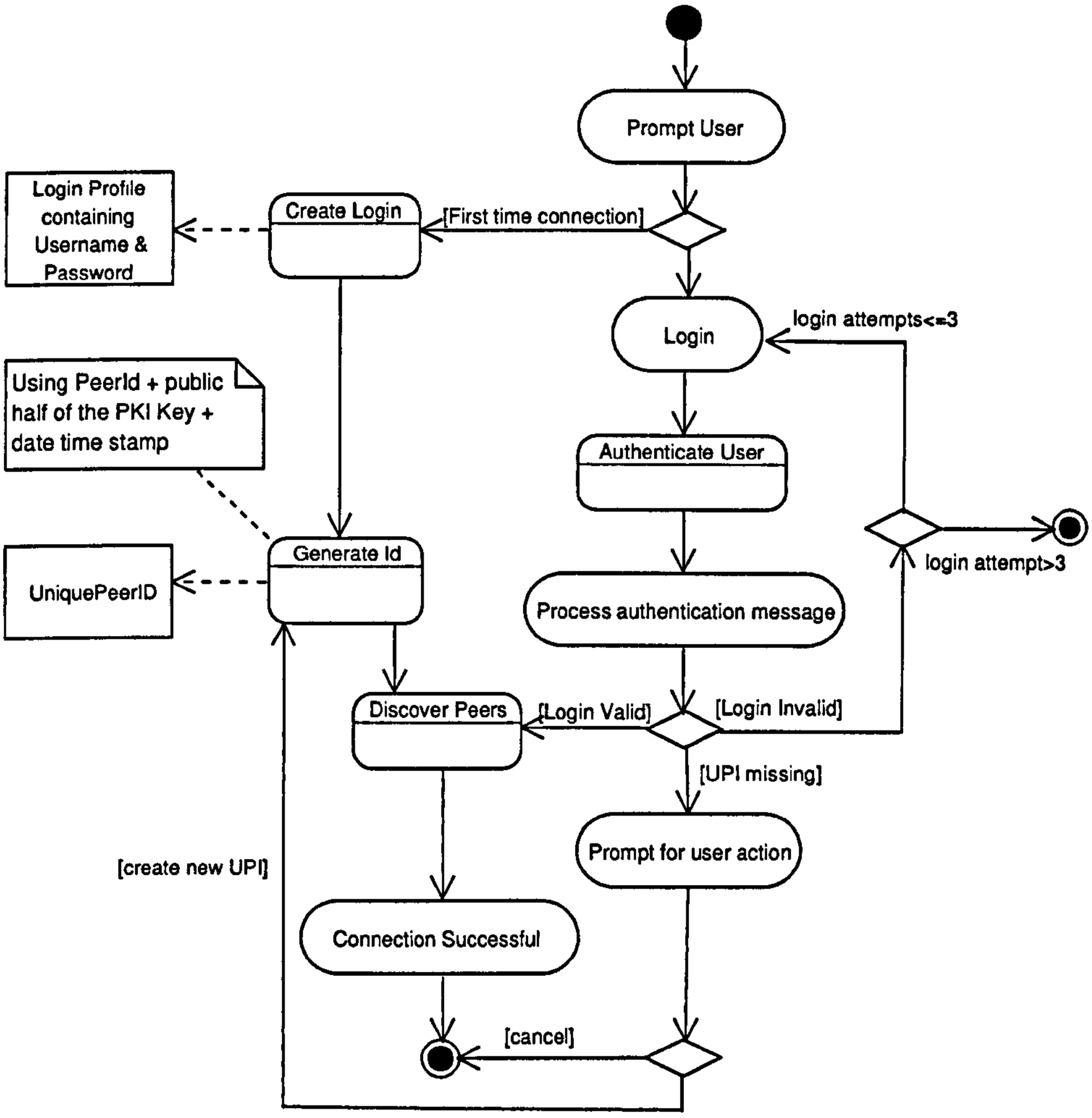


Figure B. 3: Connect to CaspaceNet

Description:

This Activity Diagram illustrates the peer's connection to the CasPaCENet.

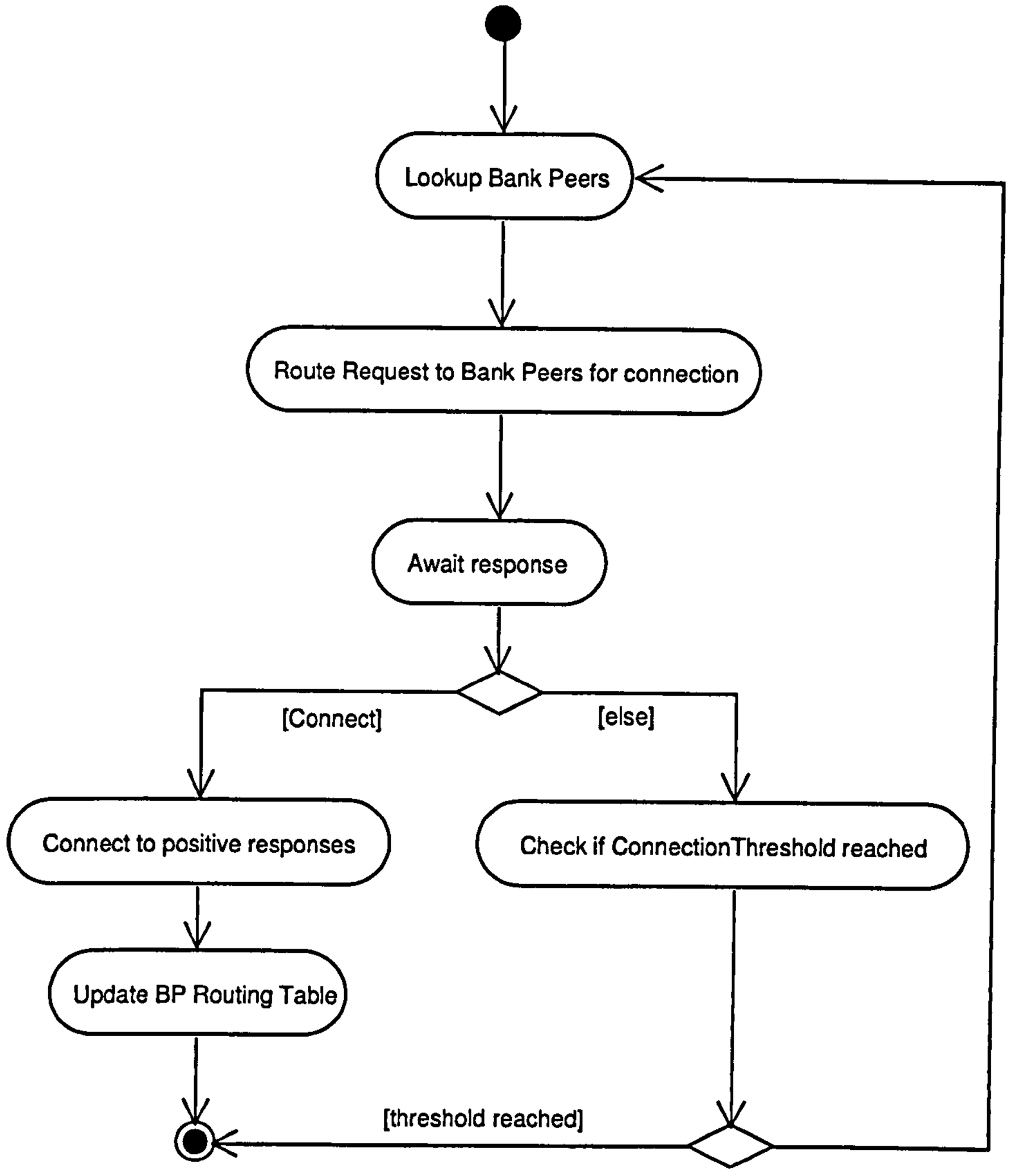


Figure B. 4: Connect to bank peer network

Description:

This Activity Diagram illustrates the connection to the bank peer overlay newtork.

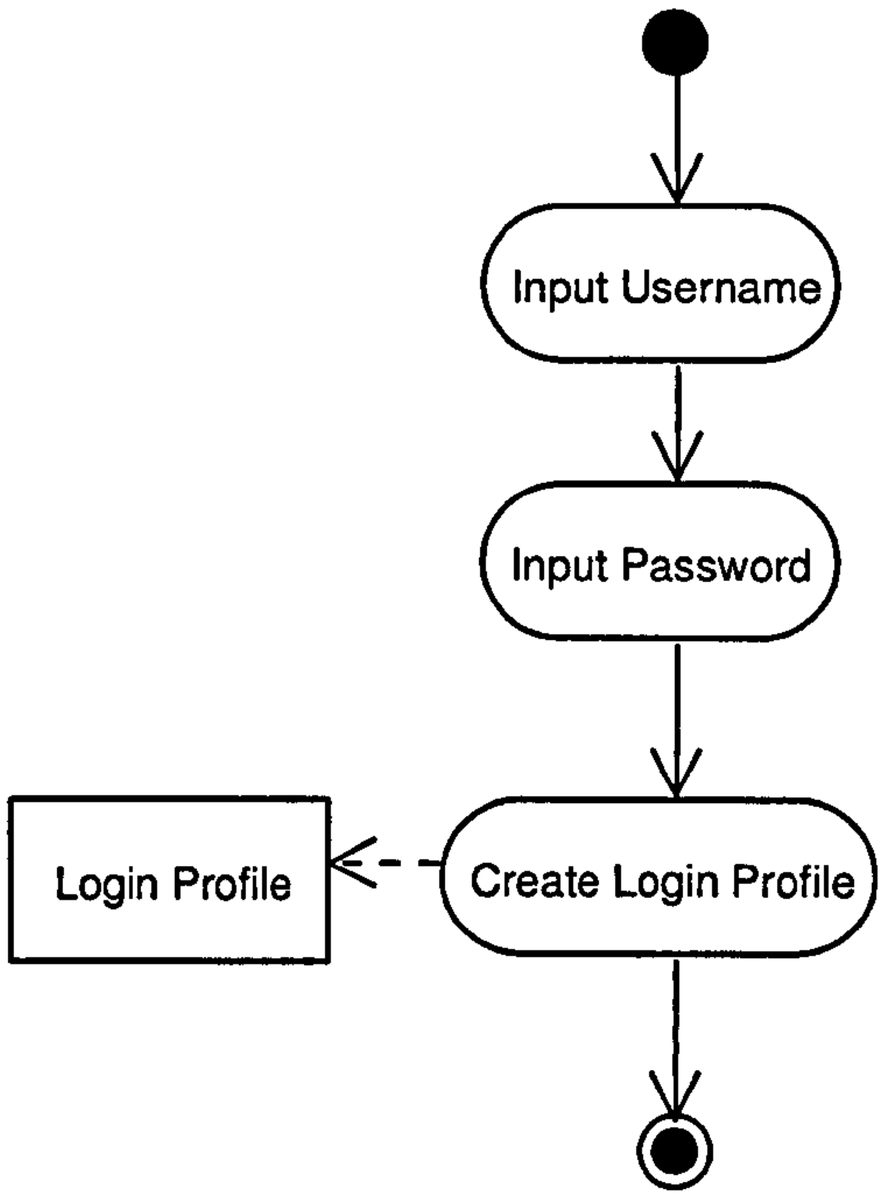


Figure B. 5: Create Login

Description:

This Activity Diagram illustrates the login creation process.

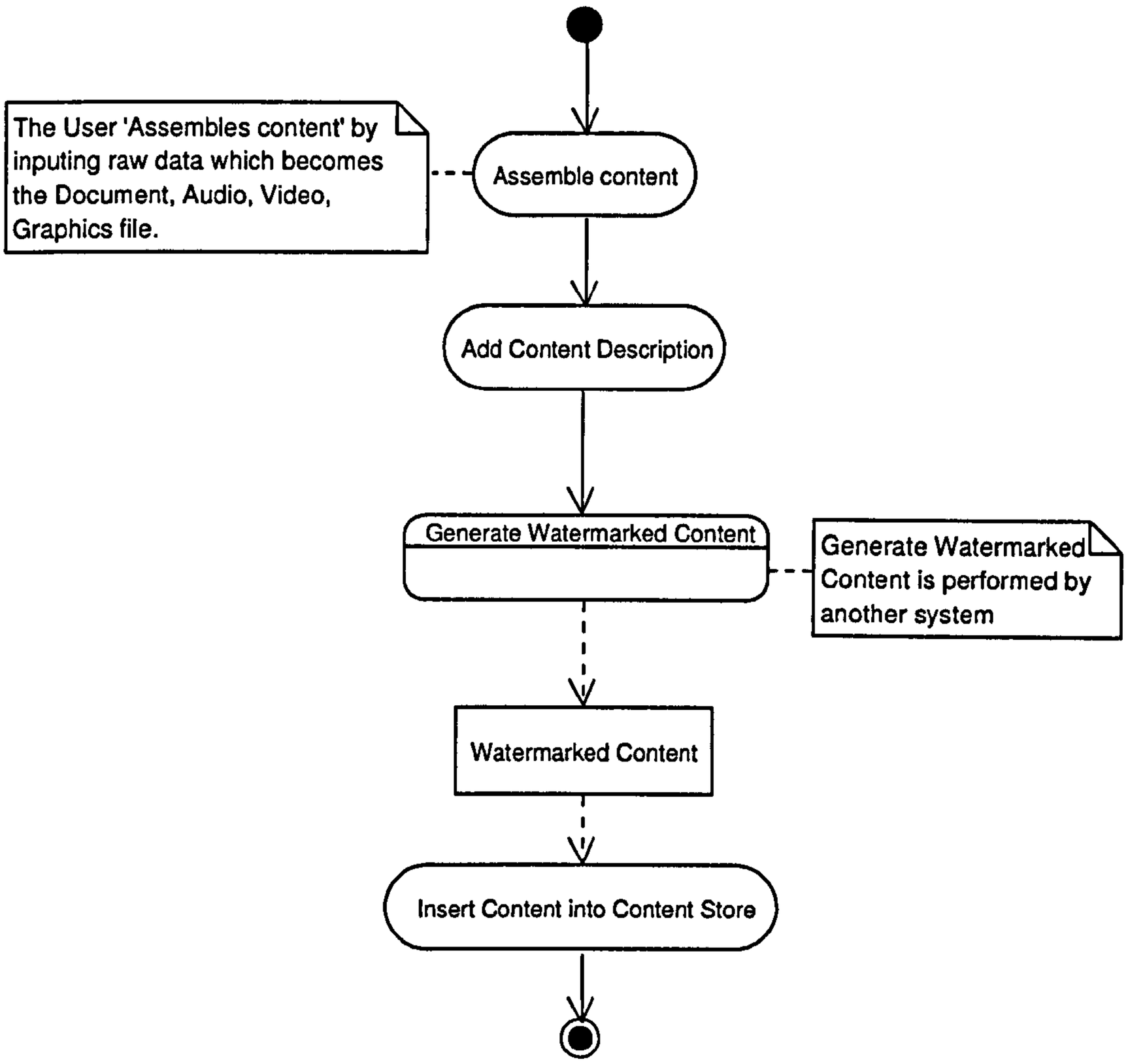


Figure B. 6: Create caspace enabled content

Description:

This Activity Diagram illustrates the caspace enabled content creation within the framework.

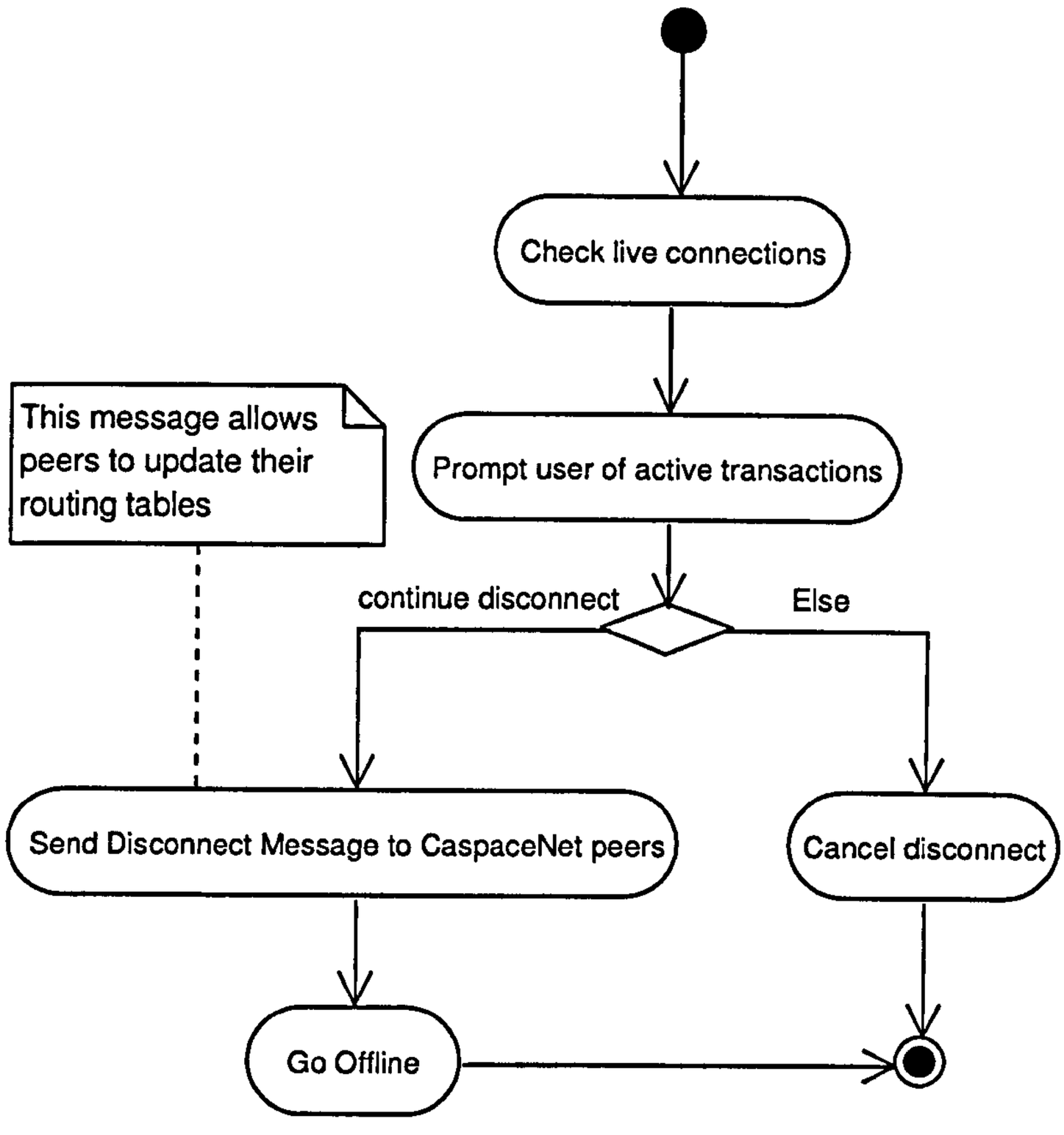
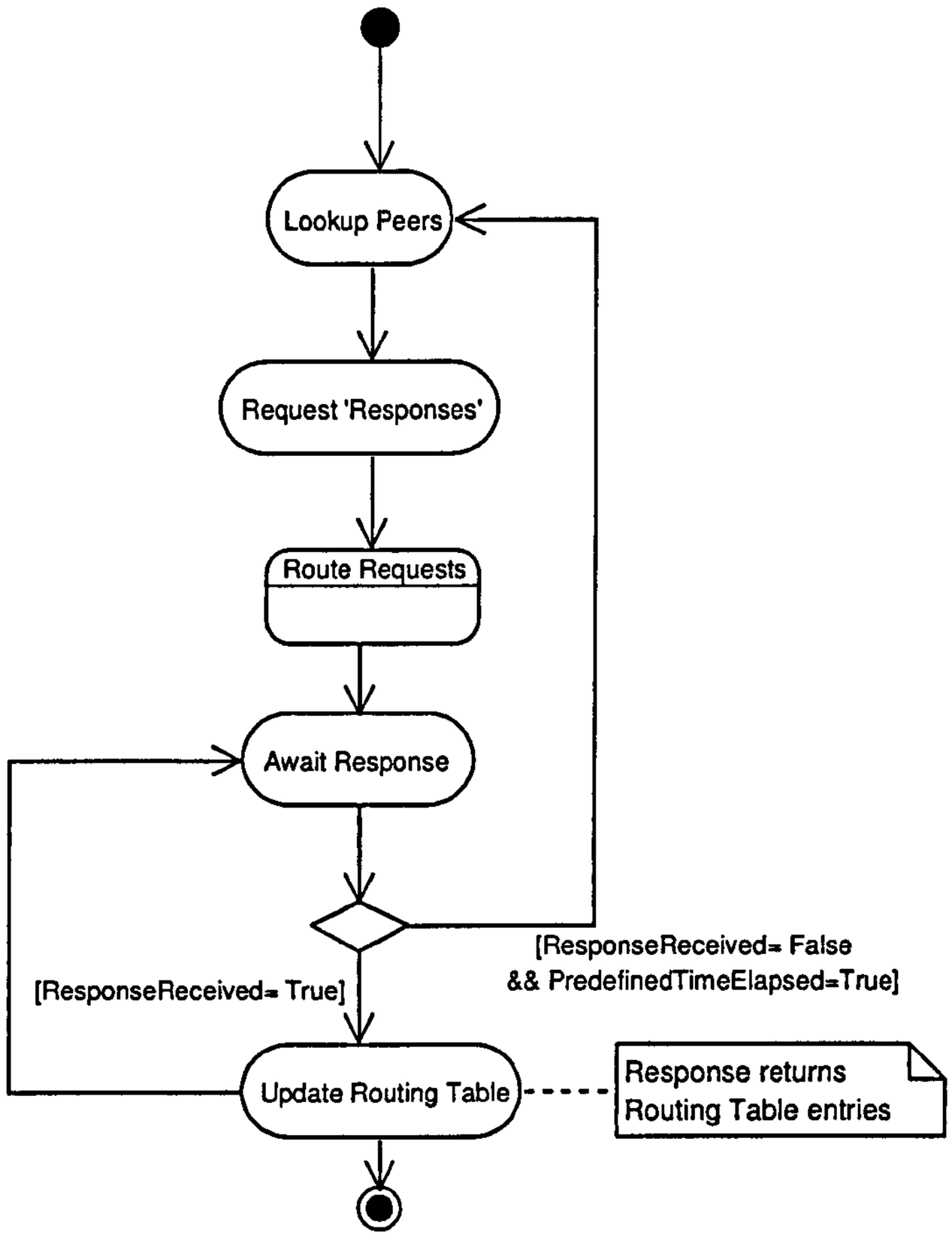


Figure B. 7: Disconnect from CaspaceNet

Description:

This Activity Diagram illustrates disconnection from the CaspaceNet.



If a DHT routing protocol is used then we perform similar operations however PeerIDs are not looked up based on previous experience, but on the circular ID plane as in Chord.

Figure B. 8: Discover peers

Description:

This Activity Diagram illustrates the discovery of peers within this framework.

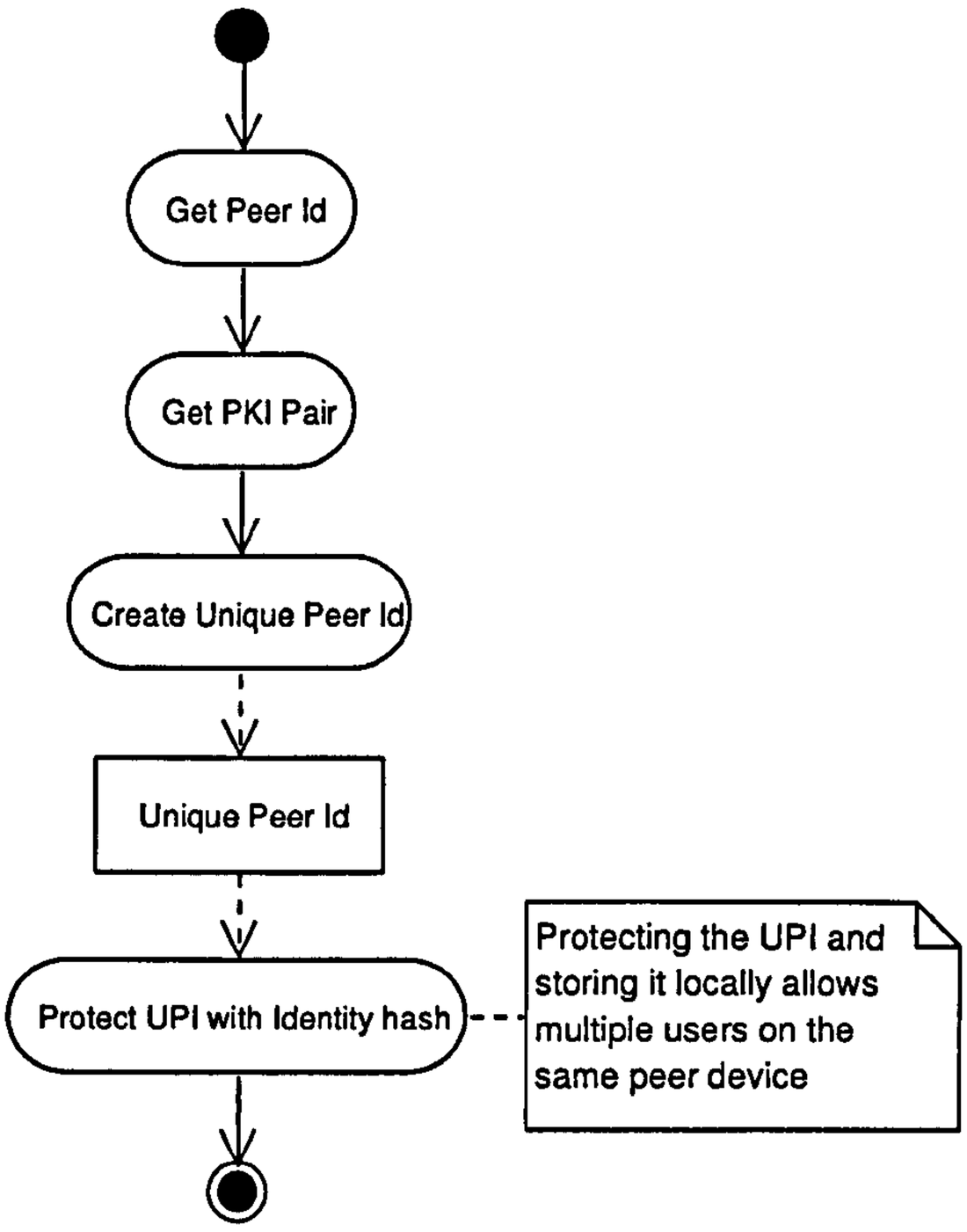


Figure B. 9: Generate UPI

Description:

This Activity Diagram illustrates the creation of a unique peer ID within this framework.

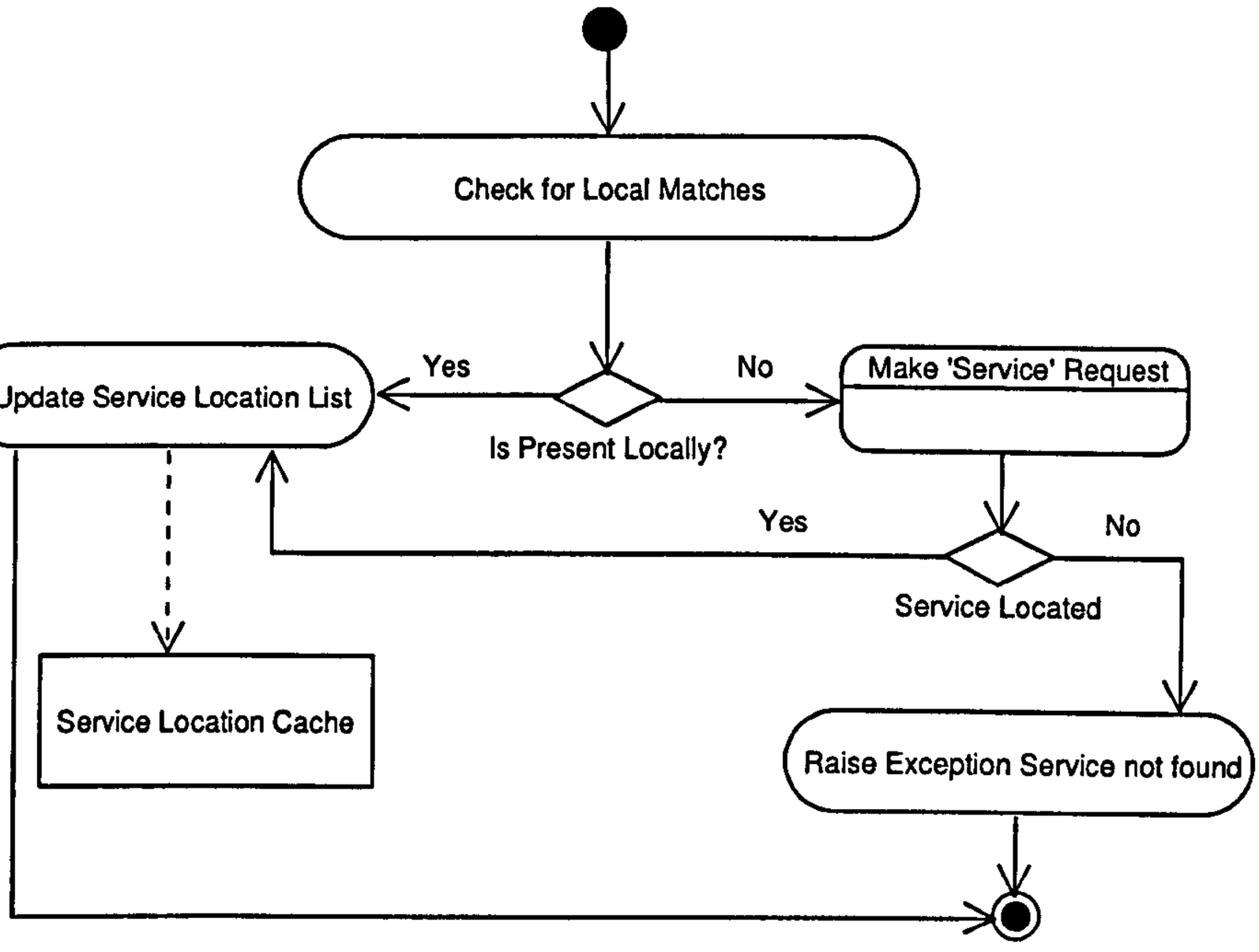


Figure B. 10: Locate service

Description:

This Activity Diagram illustrates service location within this framework.

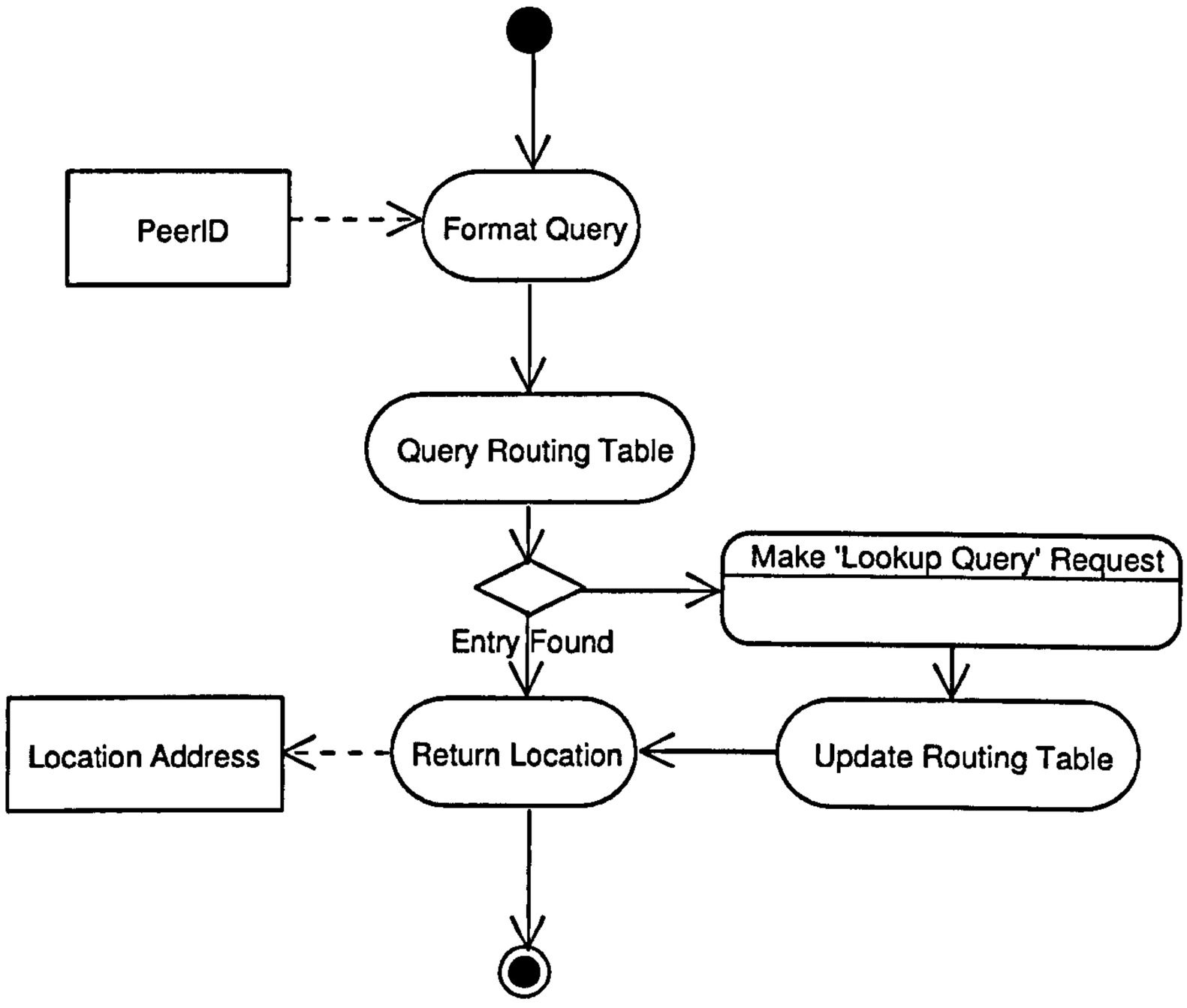


Figure B. 11: Lookup peer addresses

Description:

This Activity Diagram illustrates the lookup of the peer addresses within this framework.

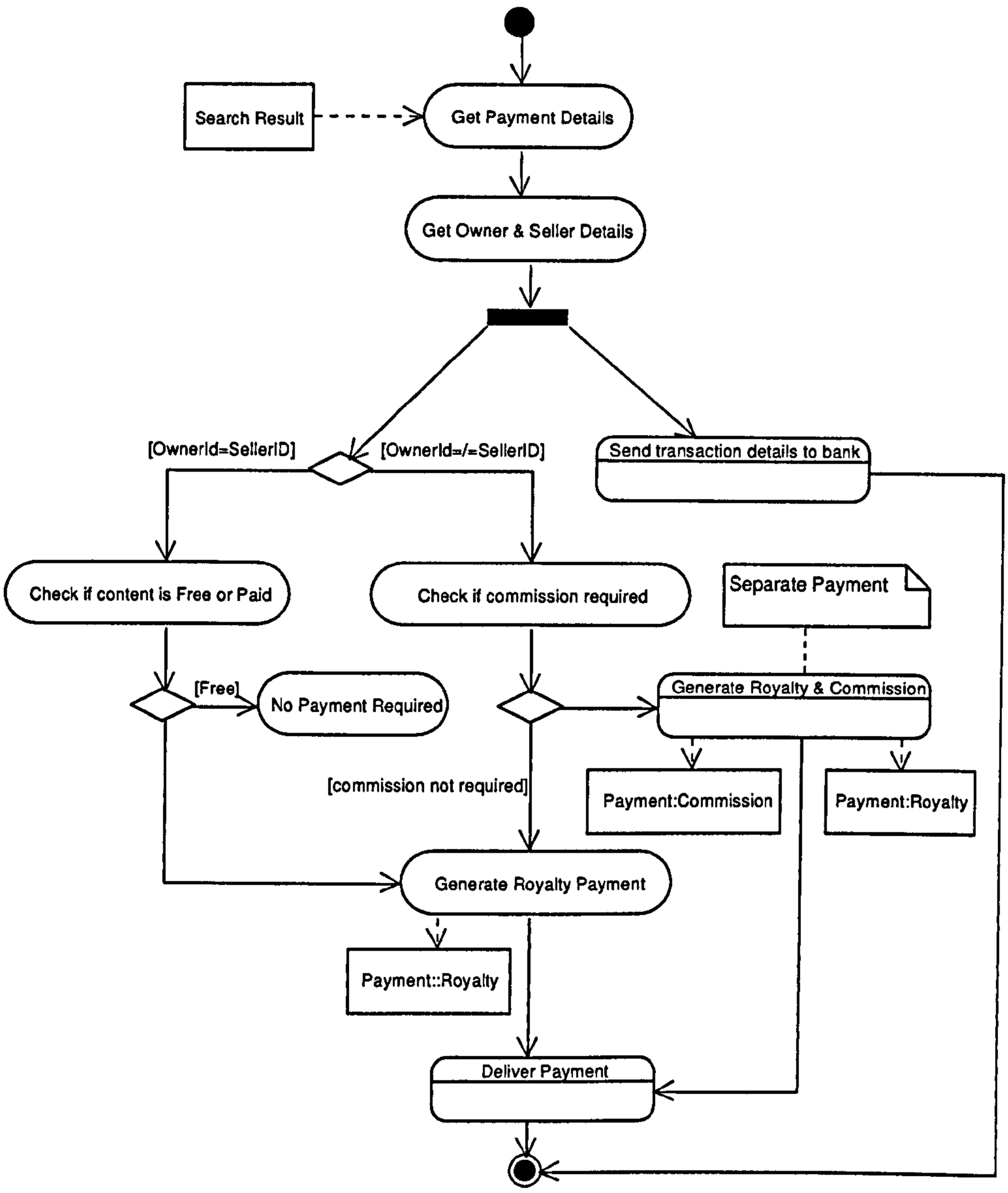


Figure B. 12: Make payment

Description:

This Activity Diagram illustrates how content is paid for and how the payments are generated at payment separation within this framework.

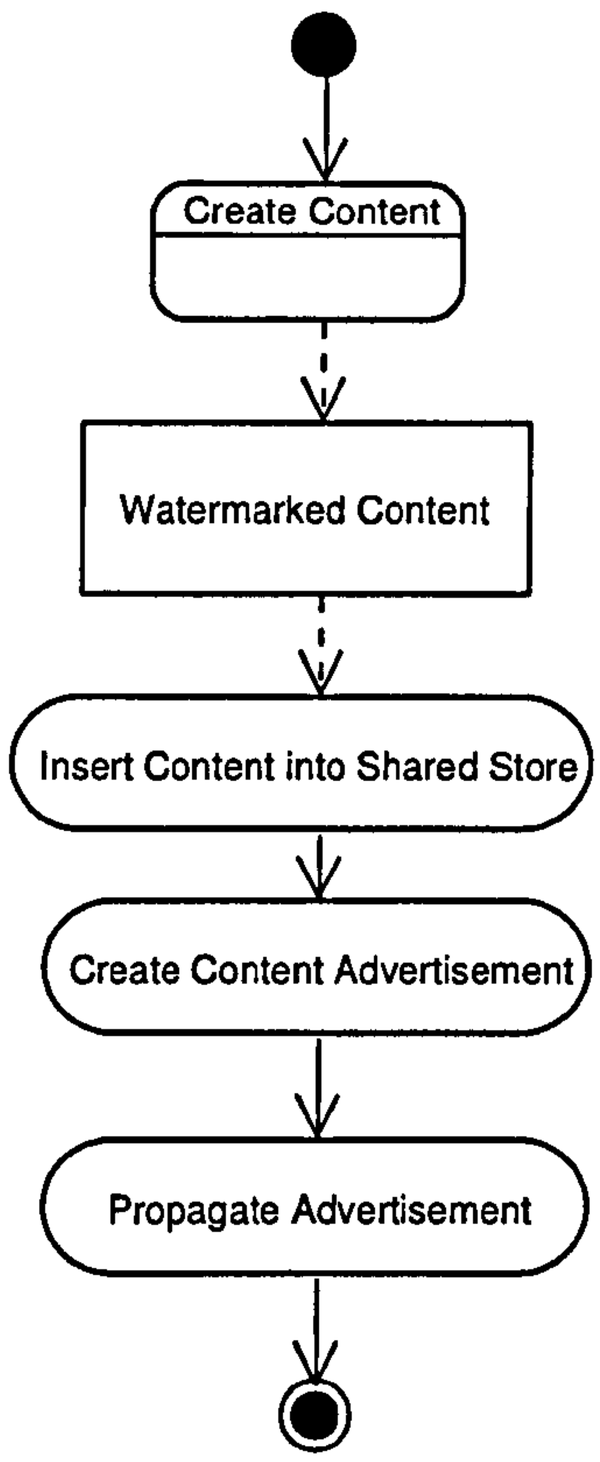


Figure B. 13: Offer content for sale

Description:

This Activity Diagram illustrates how content is offered for sale within this framework.

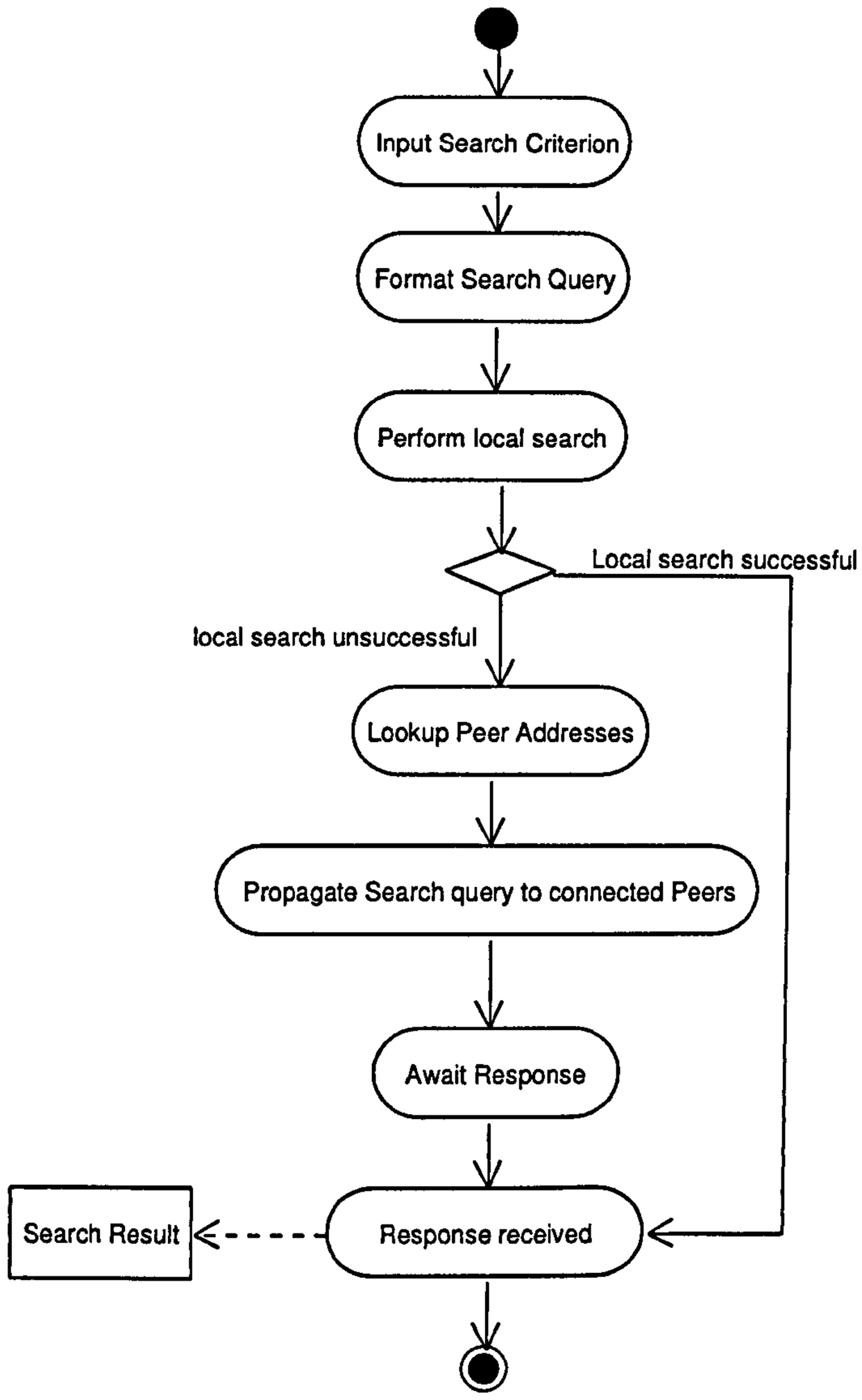


Figure B. 14: Search

Description:

This Activity Diagram illustrates a typical search process within this framework.

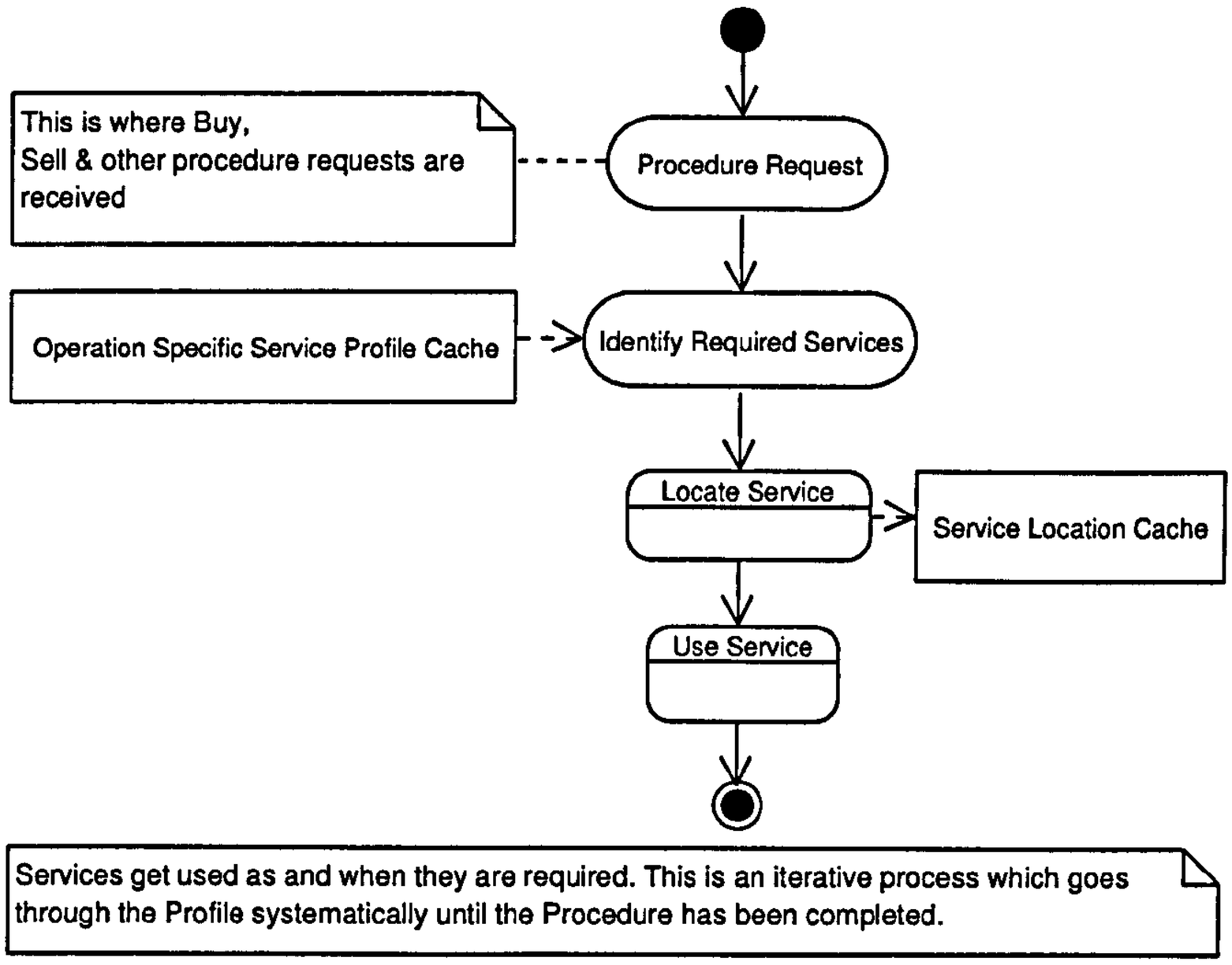


Figure B. 15: Service location and utilisation

Description:

This Activity Diagram illustrates how redundant services are discovered and used by peers within this framework.

Appendix C. Class diagrams

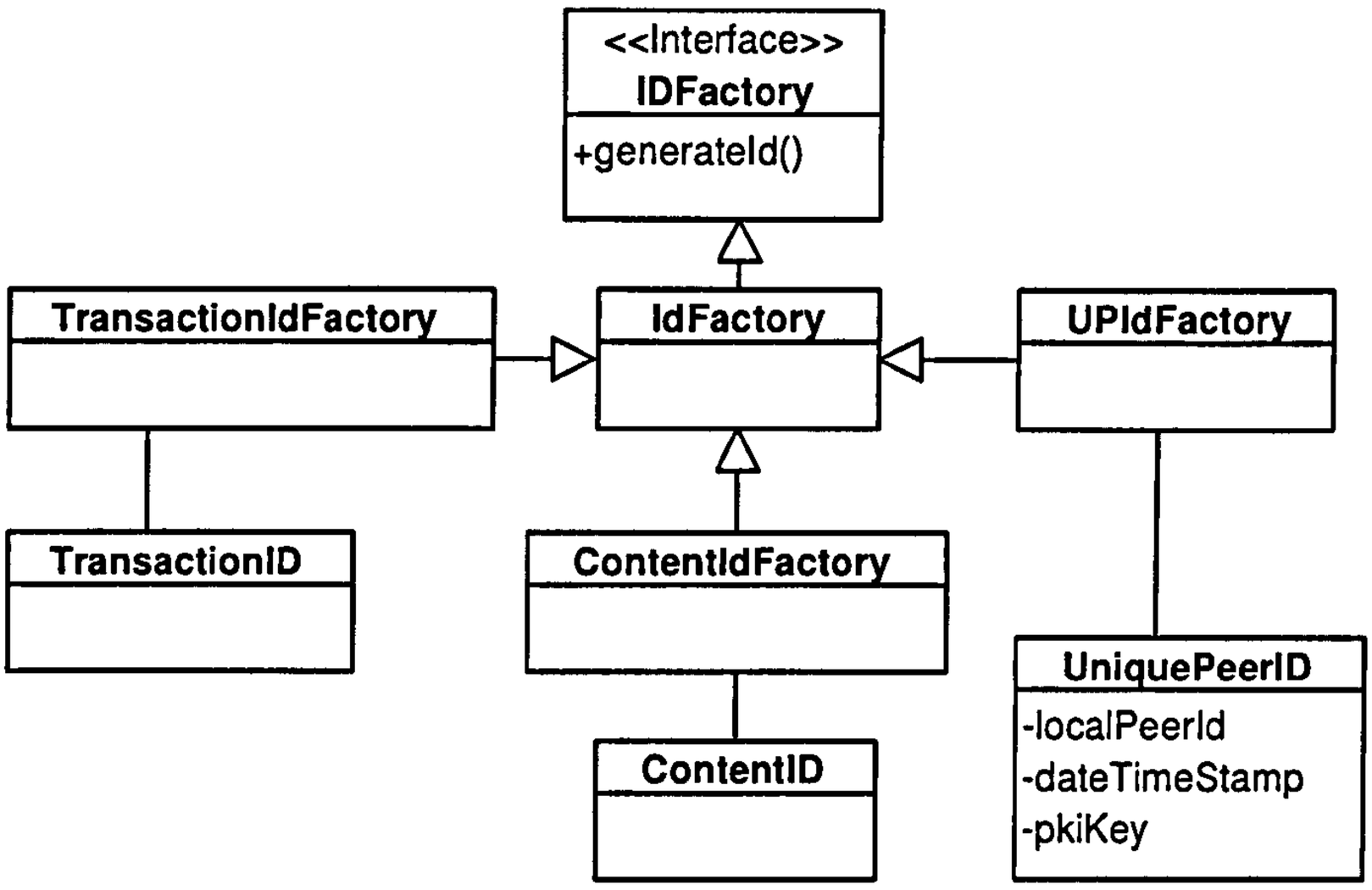


Figure C. 1: IdFactory and the caspace identifiers

Description:

This Class Diagram illustrates the classes used to create the identifiers within this framework.

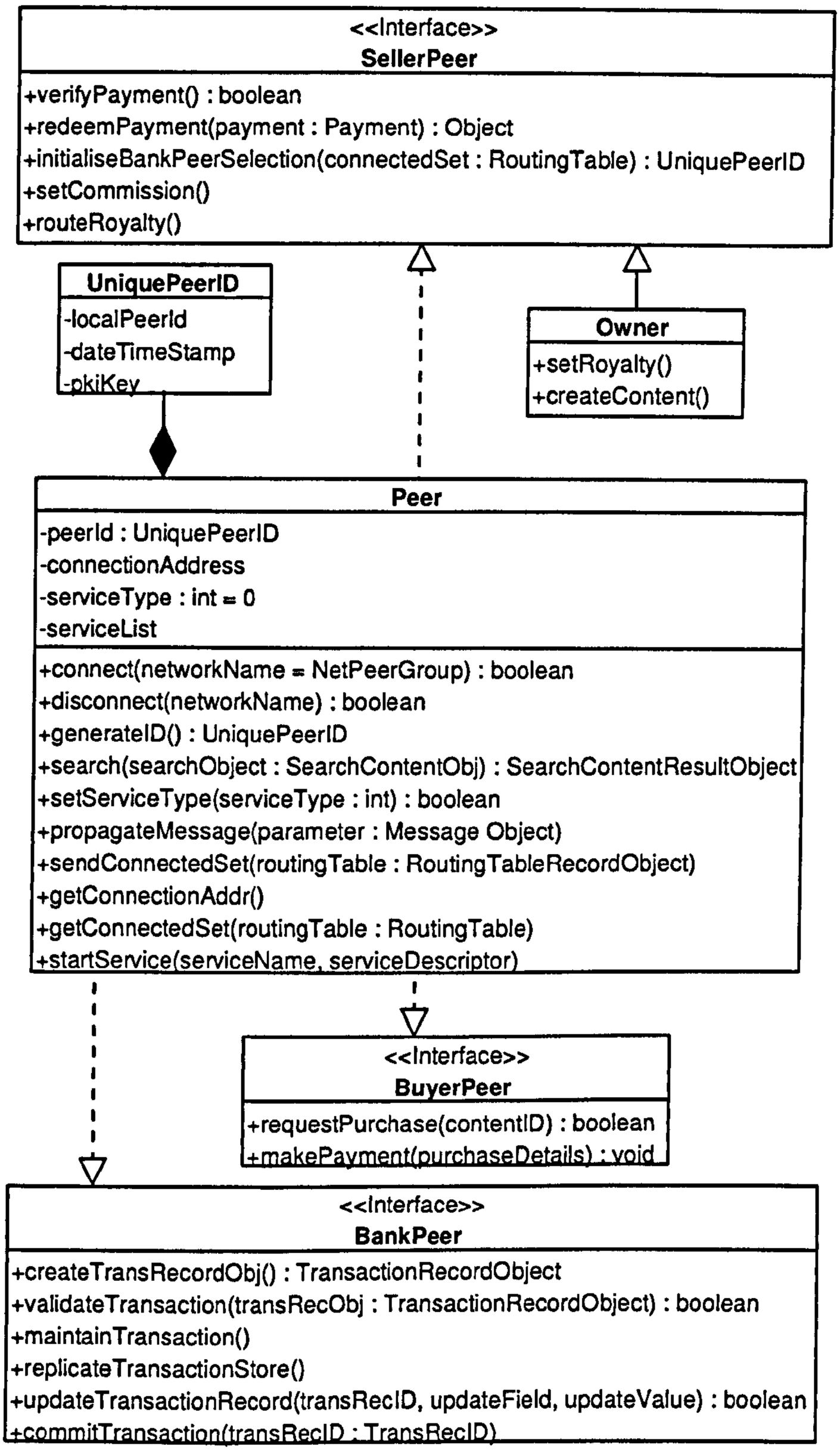


Figure C. 2: Peers

Description:

This Class Diagram illustrates the peer classes used within this framework.

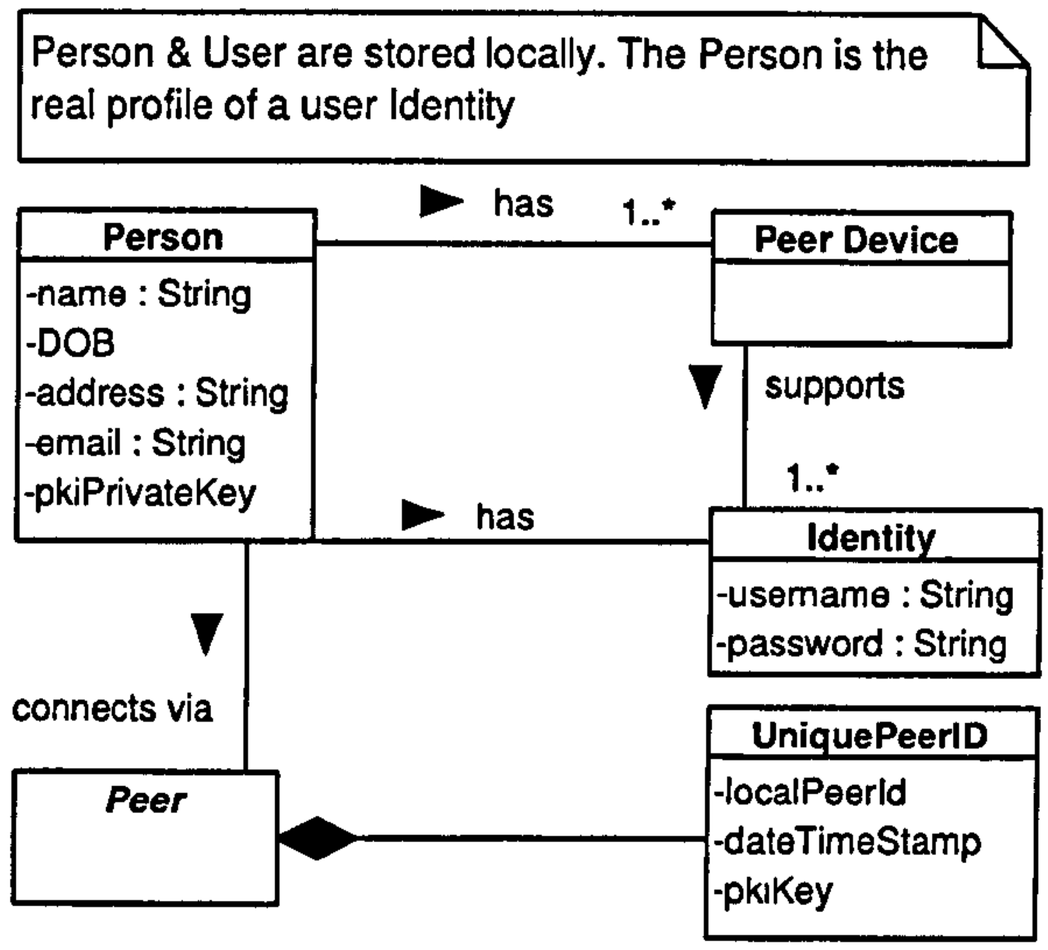


Figure C. 3: UPI and person relationship

Description:

This Class Diagram illustrates the relationships between peers, users and their login profiles within this framework.

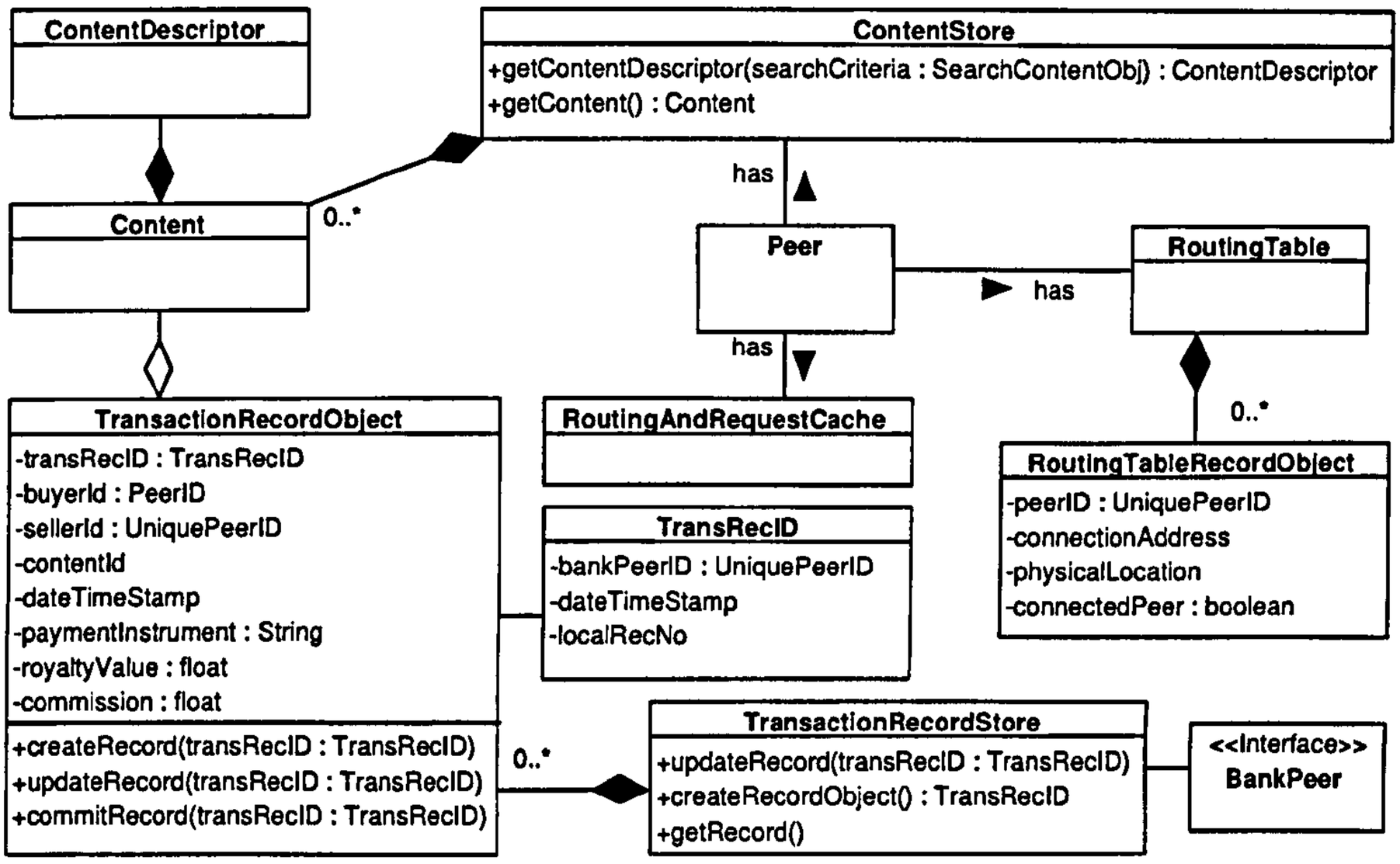


Figure C. 4: Storage classes

Description:

This Class Diagram illustrates the classes used to implement the storage capabilities within this framework.

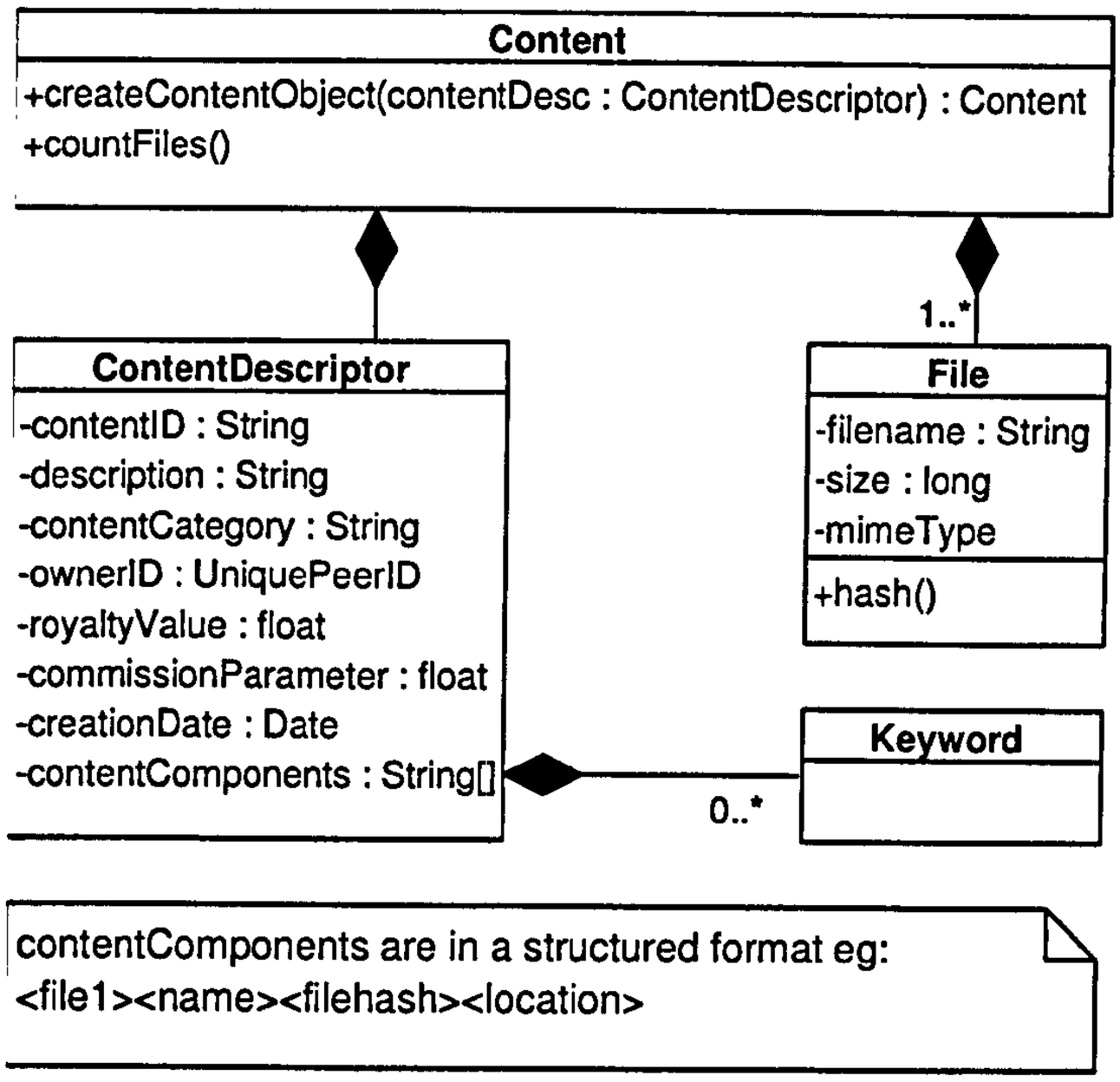


Figure C. 5: Content classes

Description:

This Class Diagram illustrates the classes used to create content within this framework.

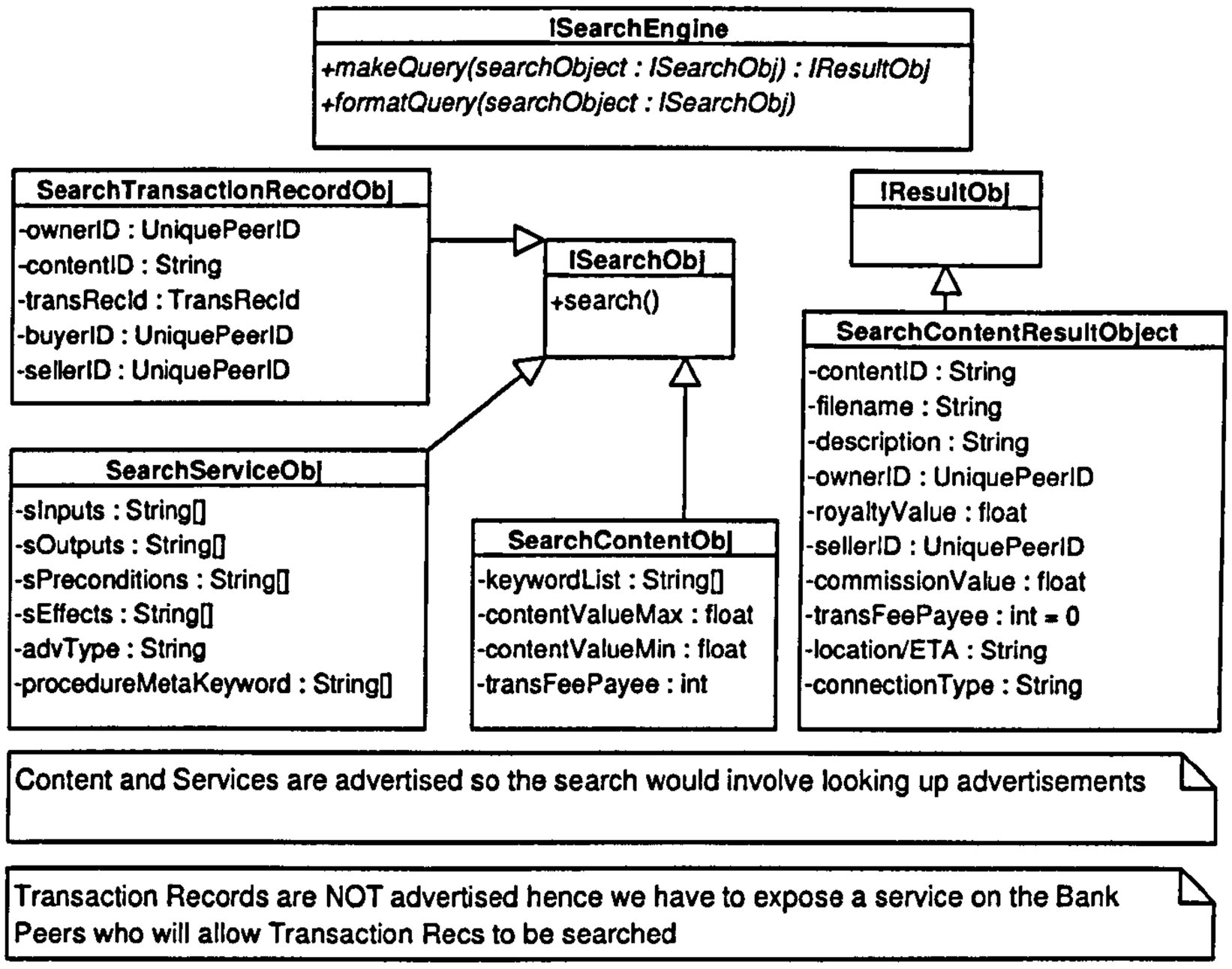


Figure C. 6: Search object classes

Description:

This Class Diagram illustrates the classes used to implement the various search and results objects within this framework.

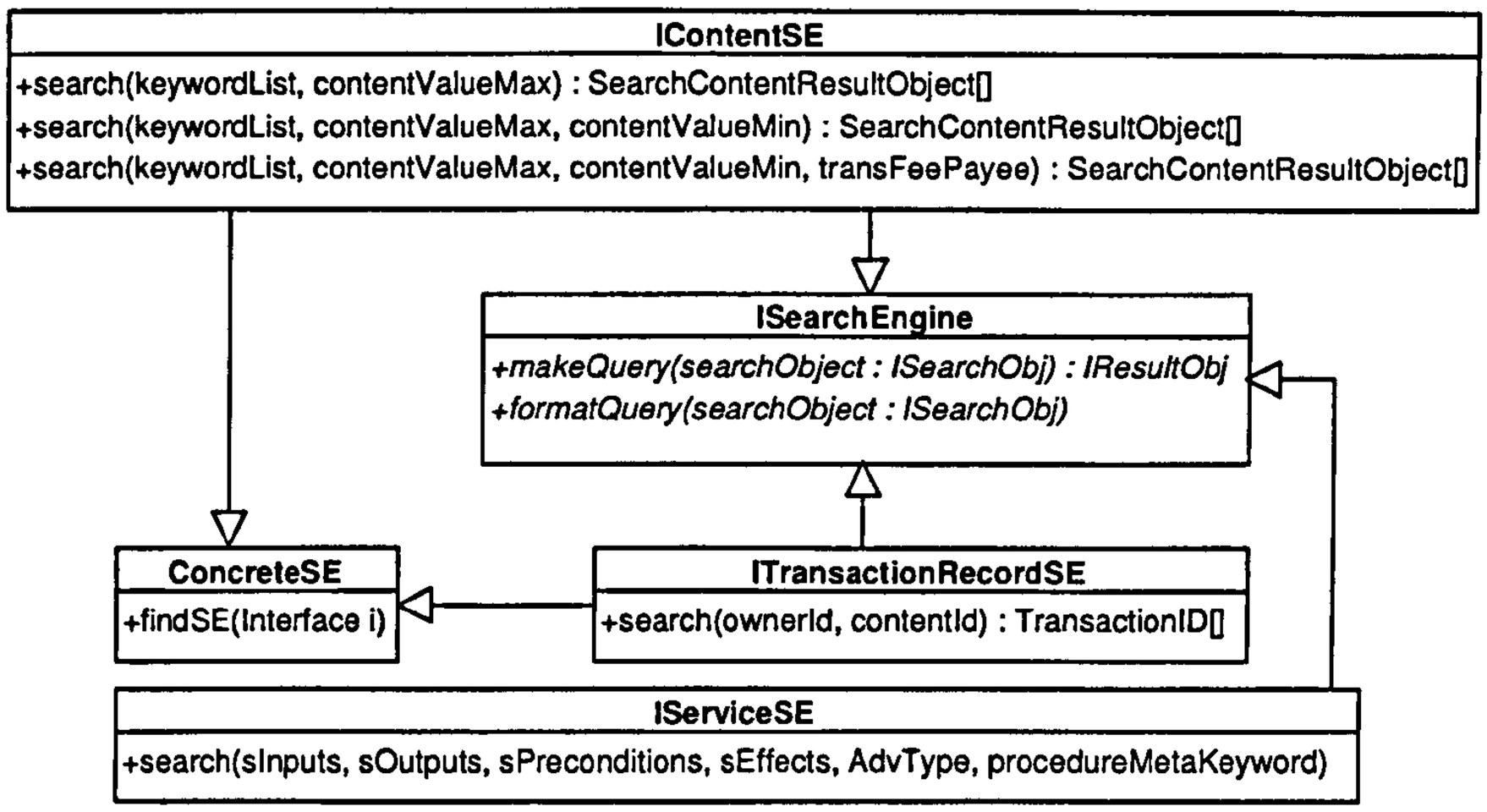


Figure C. 7: Search engine interfaces

Description:

This Class Diagram illustrates the classes used to implement the search capability within this framework.

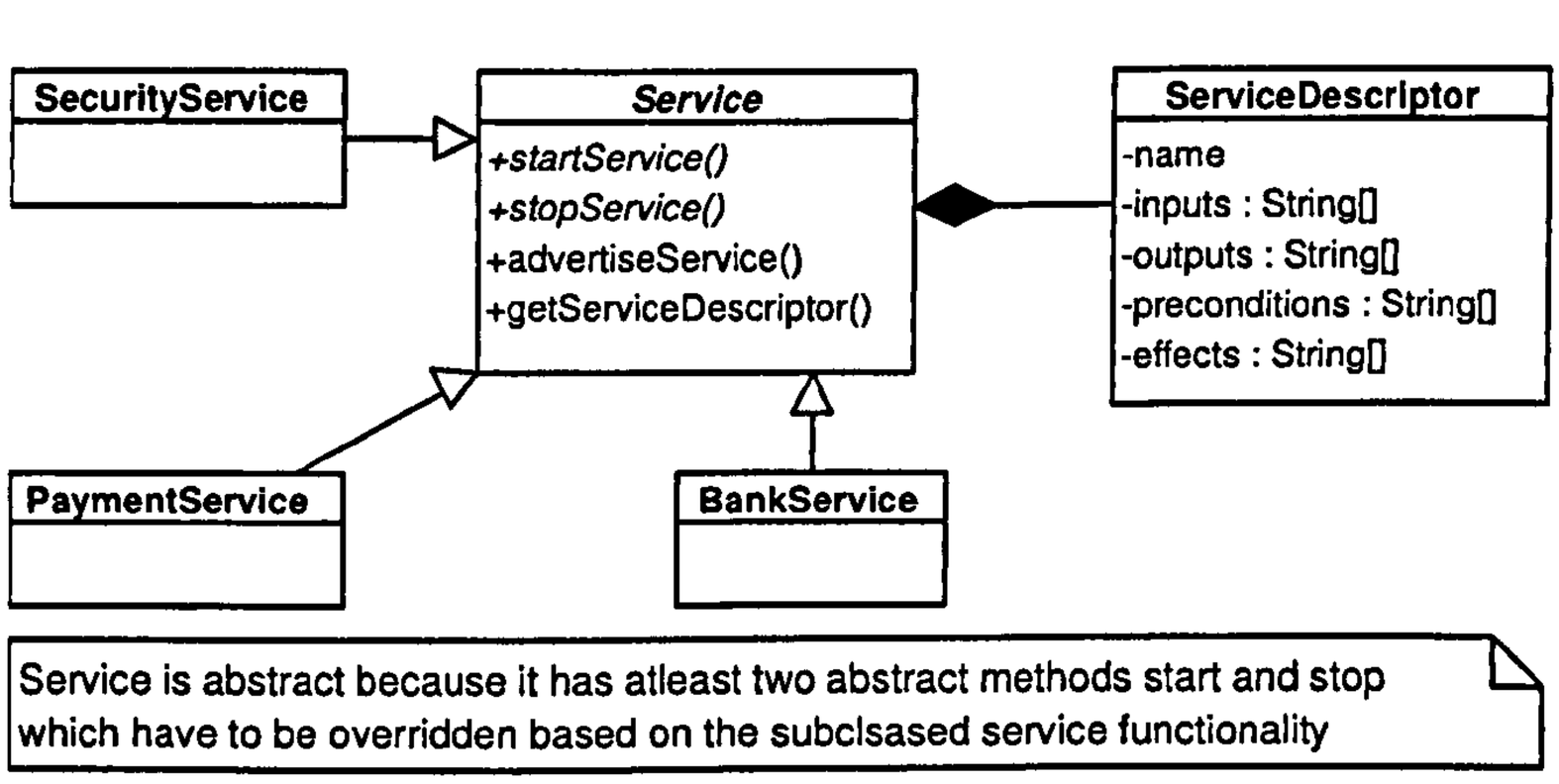


Figure C. 8: Services

Description:

This Class Diagram illustrates the relationships between the various services within this framework.

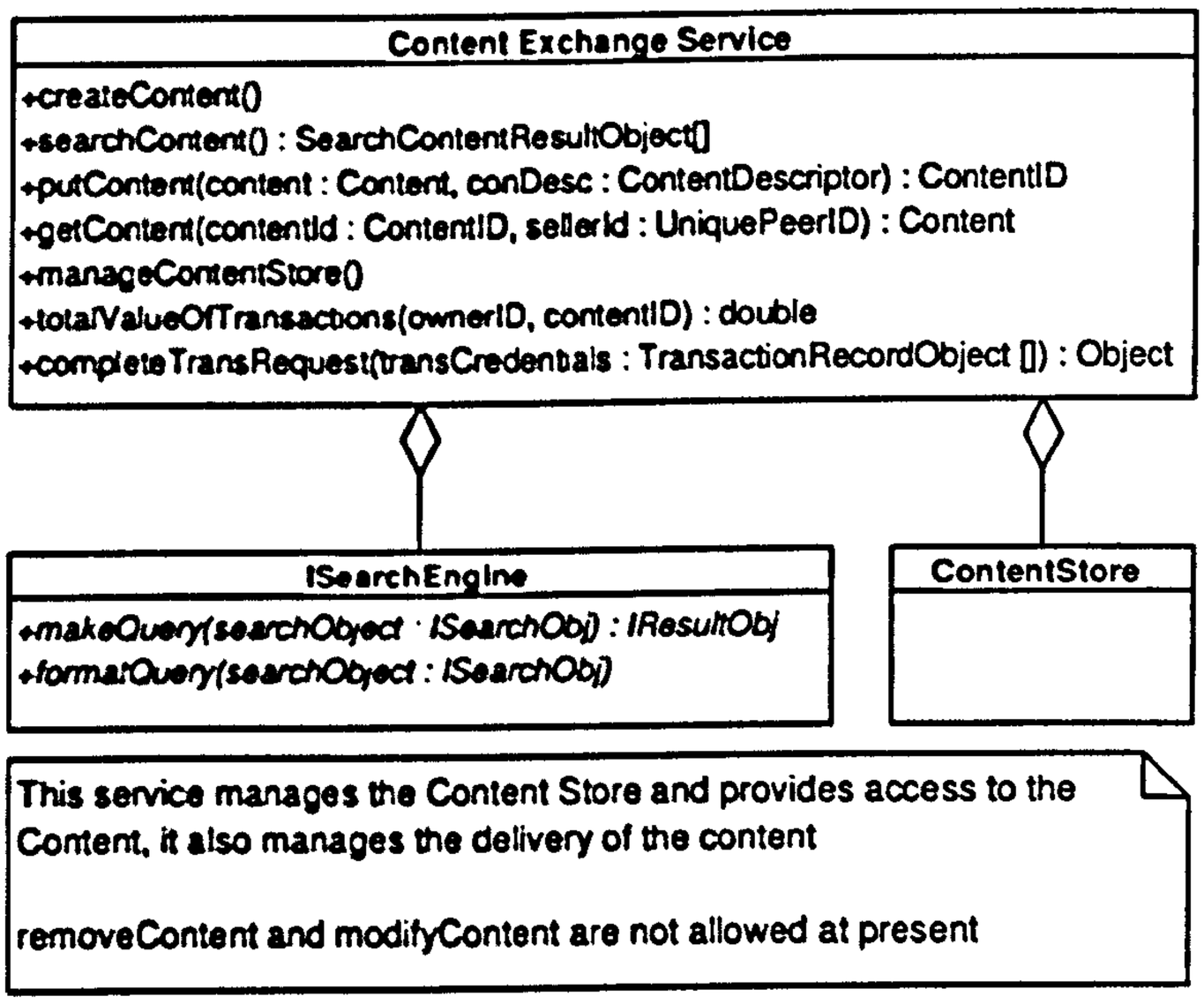


Figure C. 9: Content Exchange Service

Description:

This Class Diagram illustrates the classes required to create the Content Exchange Service within this framework.

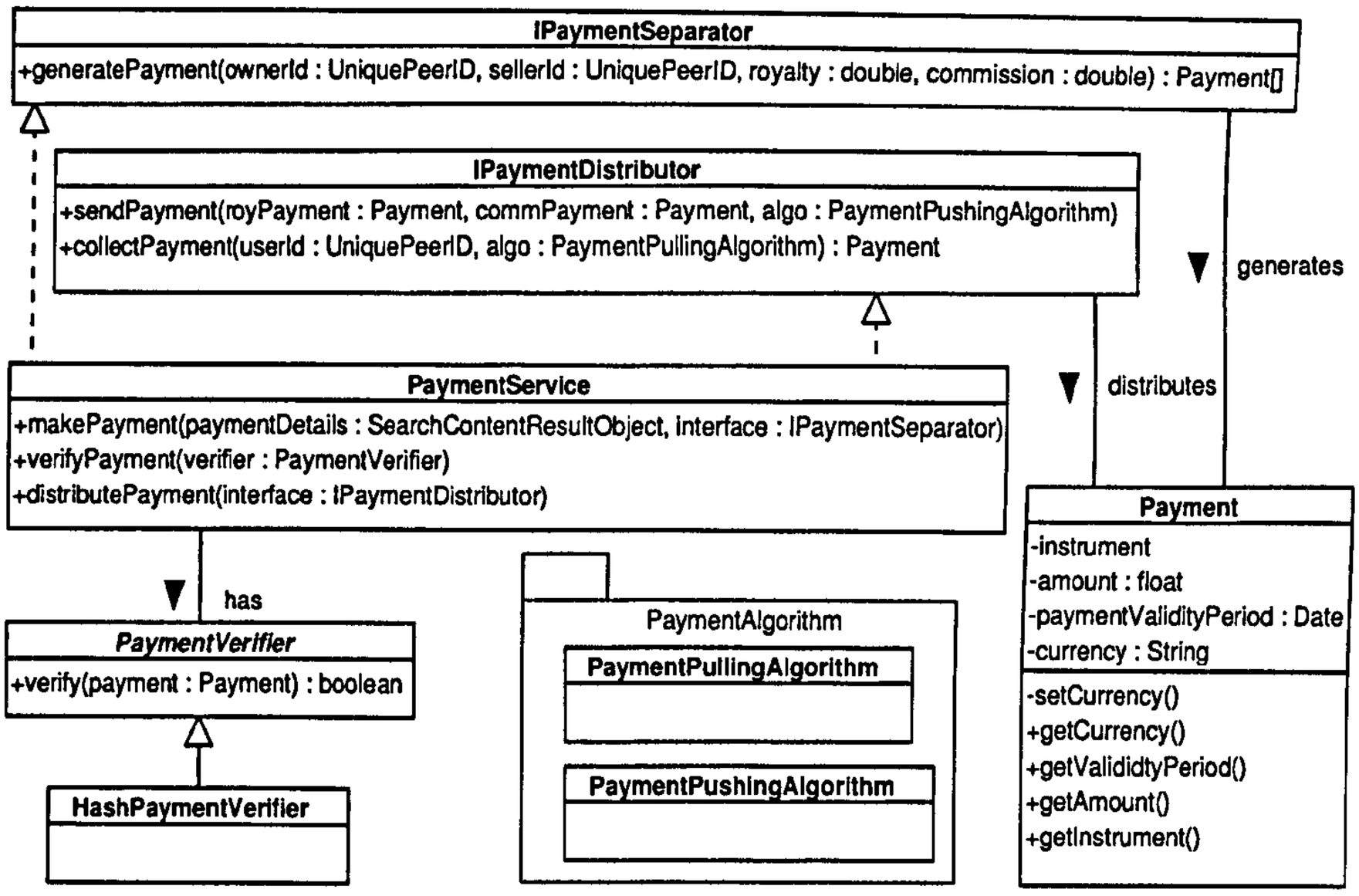


Figure C. 10: Payment service

Description:

This Class Diagram illustrates the classes required to create the Payment Service within this framework.

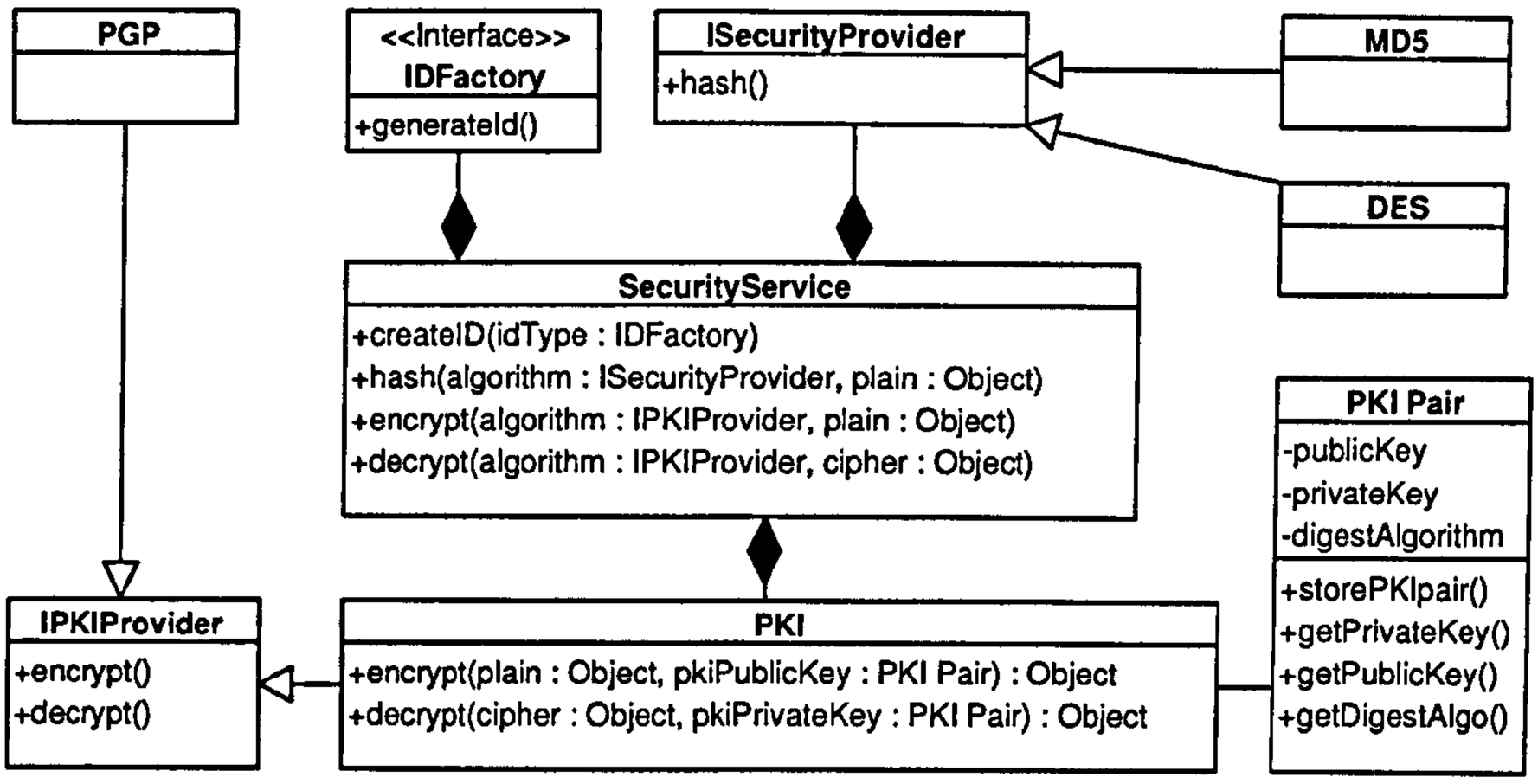


Figure C. 11: Security service

Description:

This Class Diagram illustrates the classes required for the Security Service within this framework.

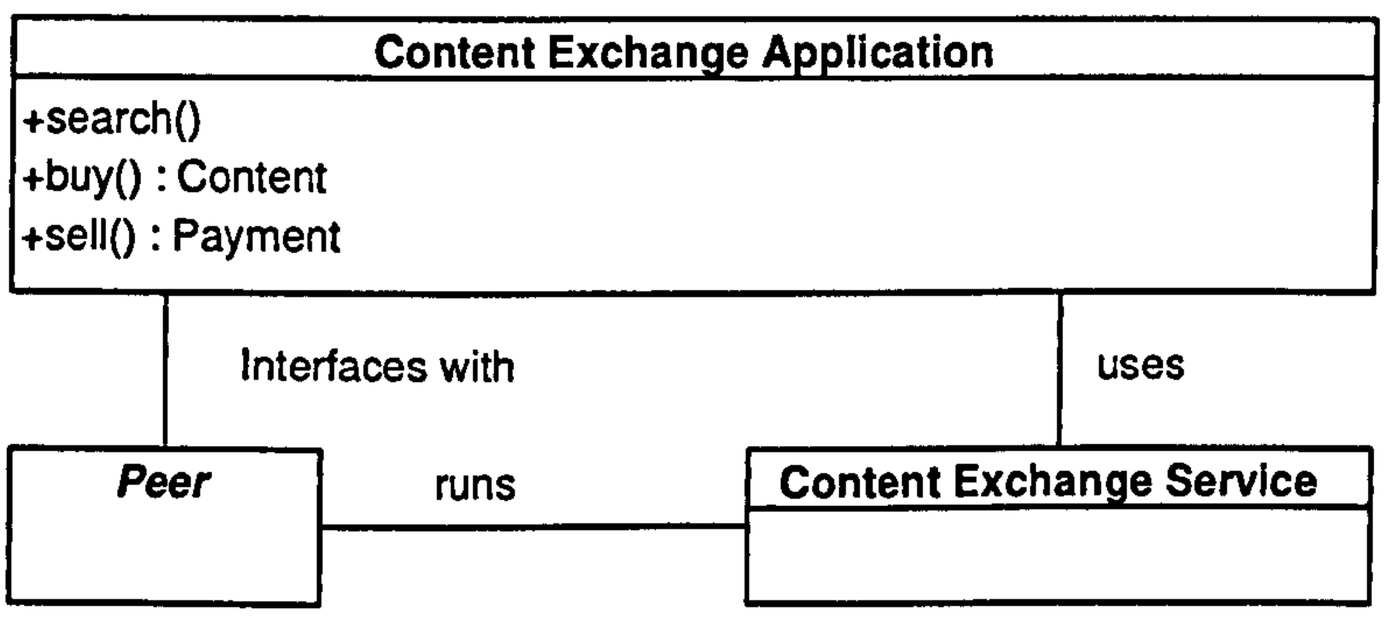


Figure C. 12: Relationship between peer, CES and CEA

Description:

This Class Diagram illustrates the relationship between the peer, the content exchange service and the content exchange application within this framework.

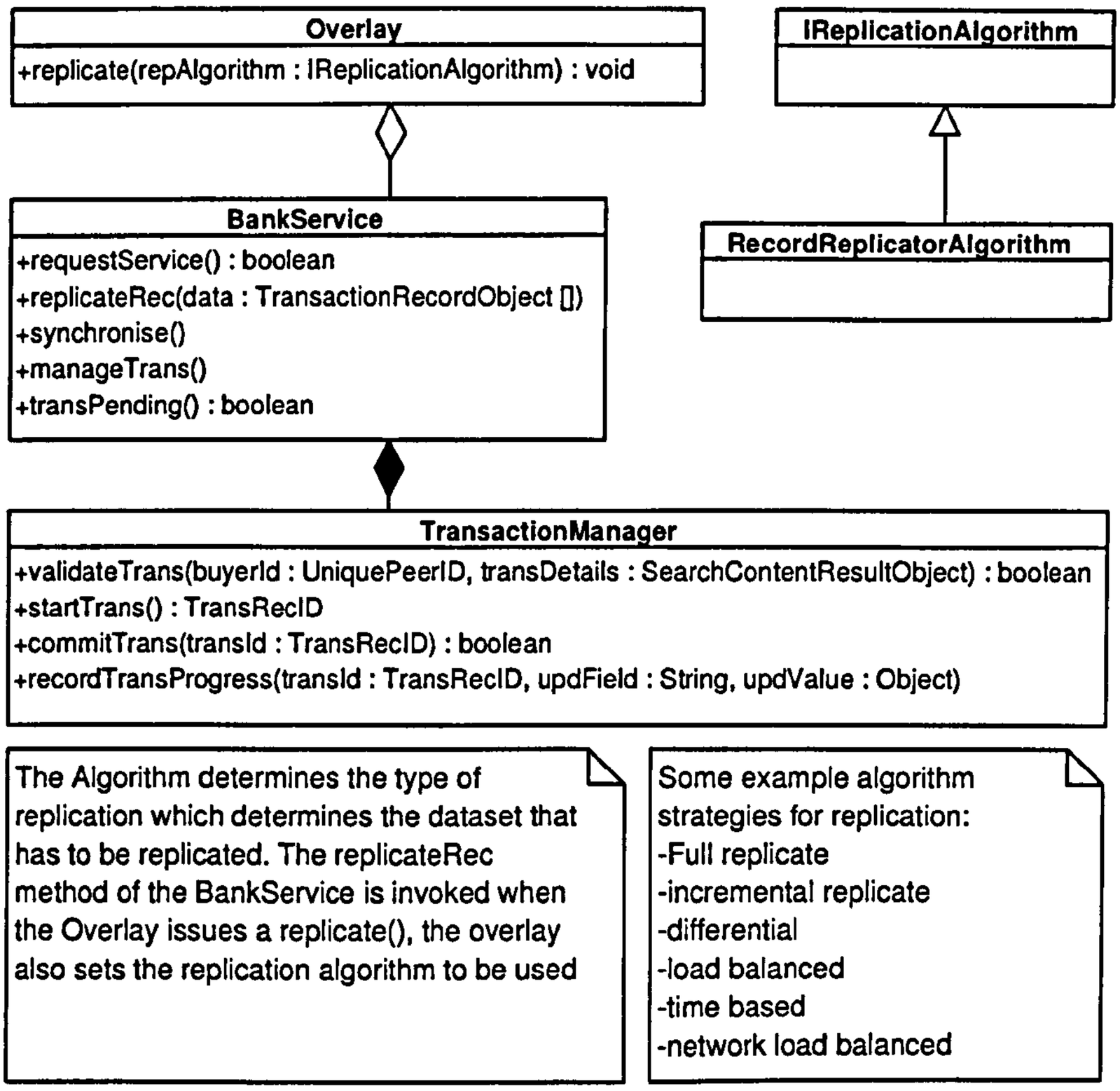


Figure C. 13: Bank service

Description:

This Class Diagram illustrates the classes required to create the Bank Service within this framework.

Appendix D. Behavioural model

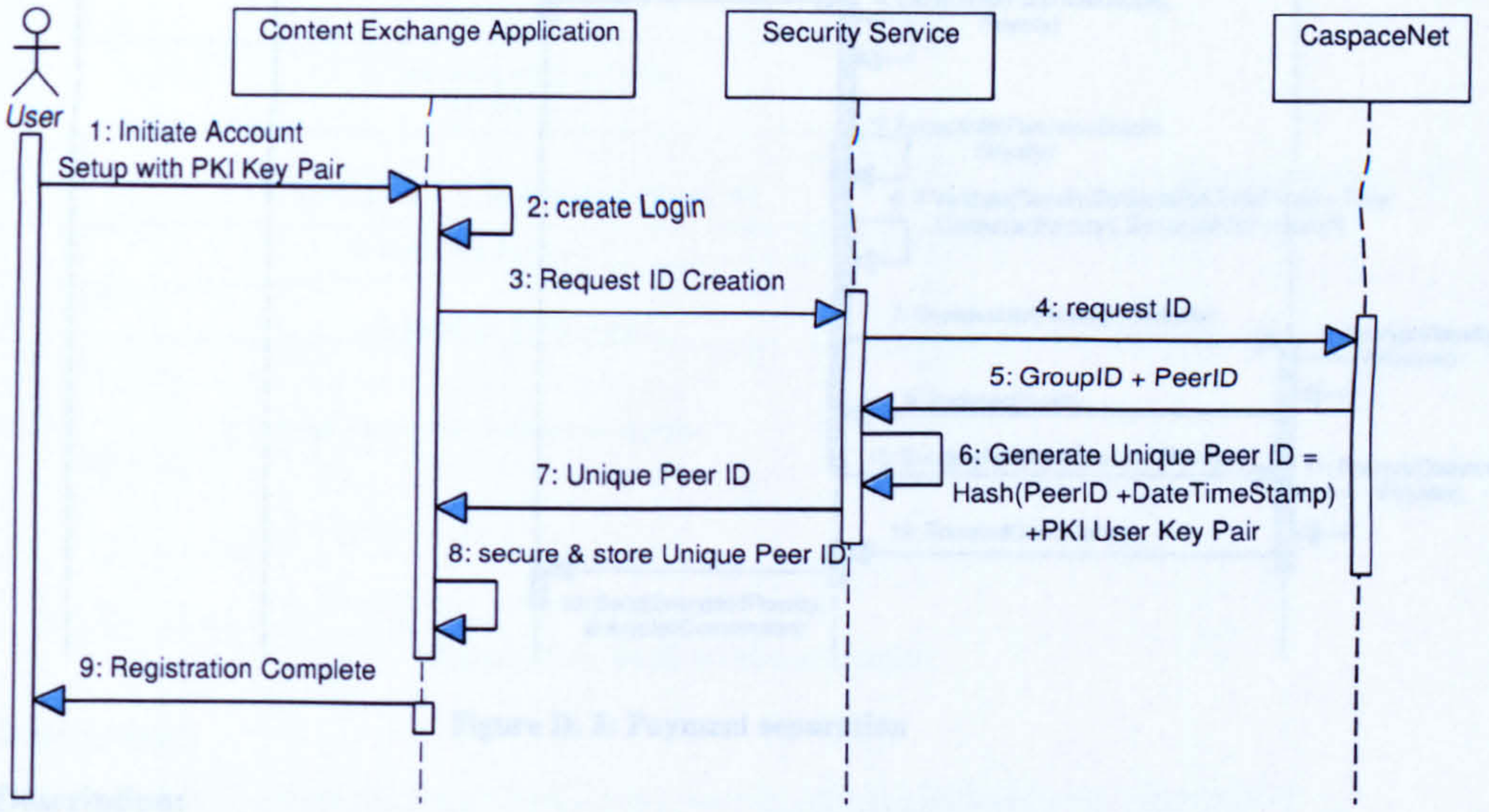


Figure D. 1: Generate ID

Description:

This Sequence Diagram illustrates the object interactions required to generate a UPI within this framework.

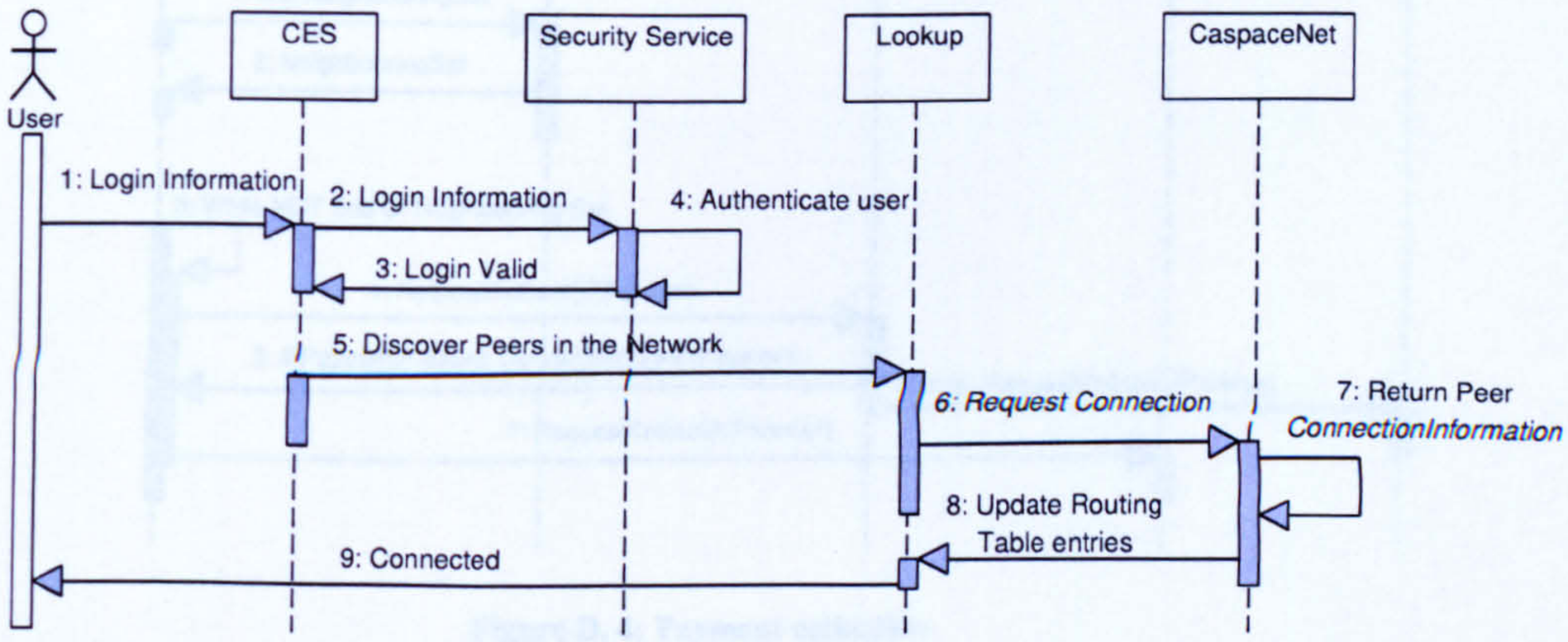


Figure D. 2: Connect to CaspaceNet

Description:

This Sequence Diagram illustrates the object interactions required to to connect to the CaspaceNet within this framework.

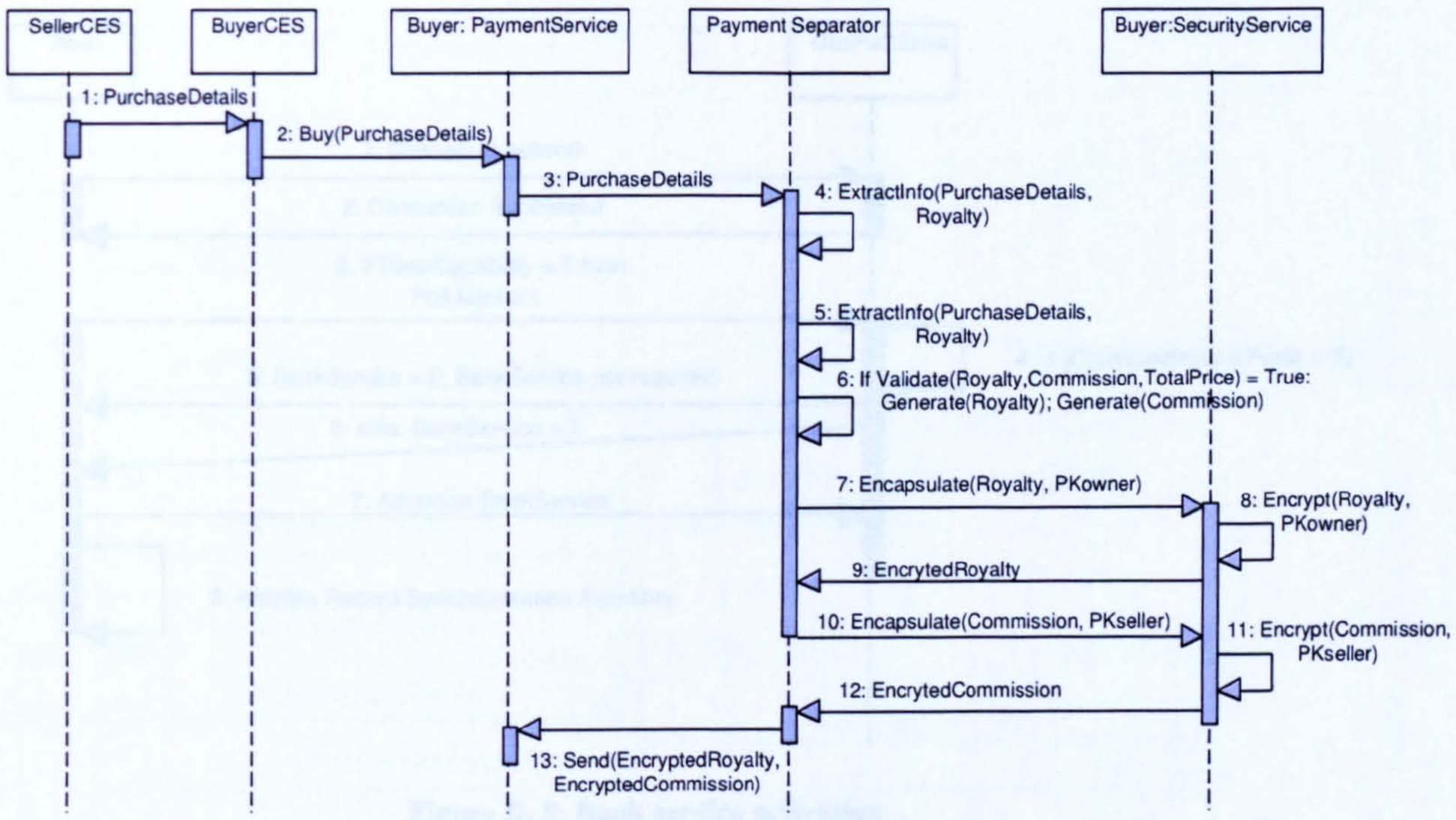


Figure D. 3: Payment separation

Description:

This Sequence Diagram illustrates the object interactions required to make royalty and commission payments within this framework.

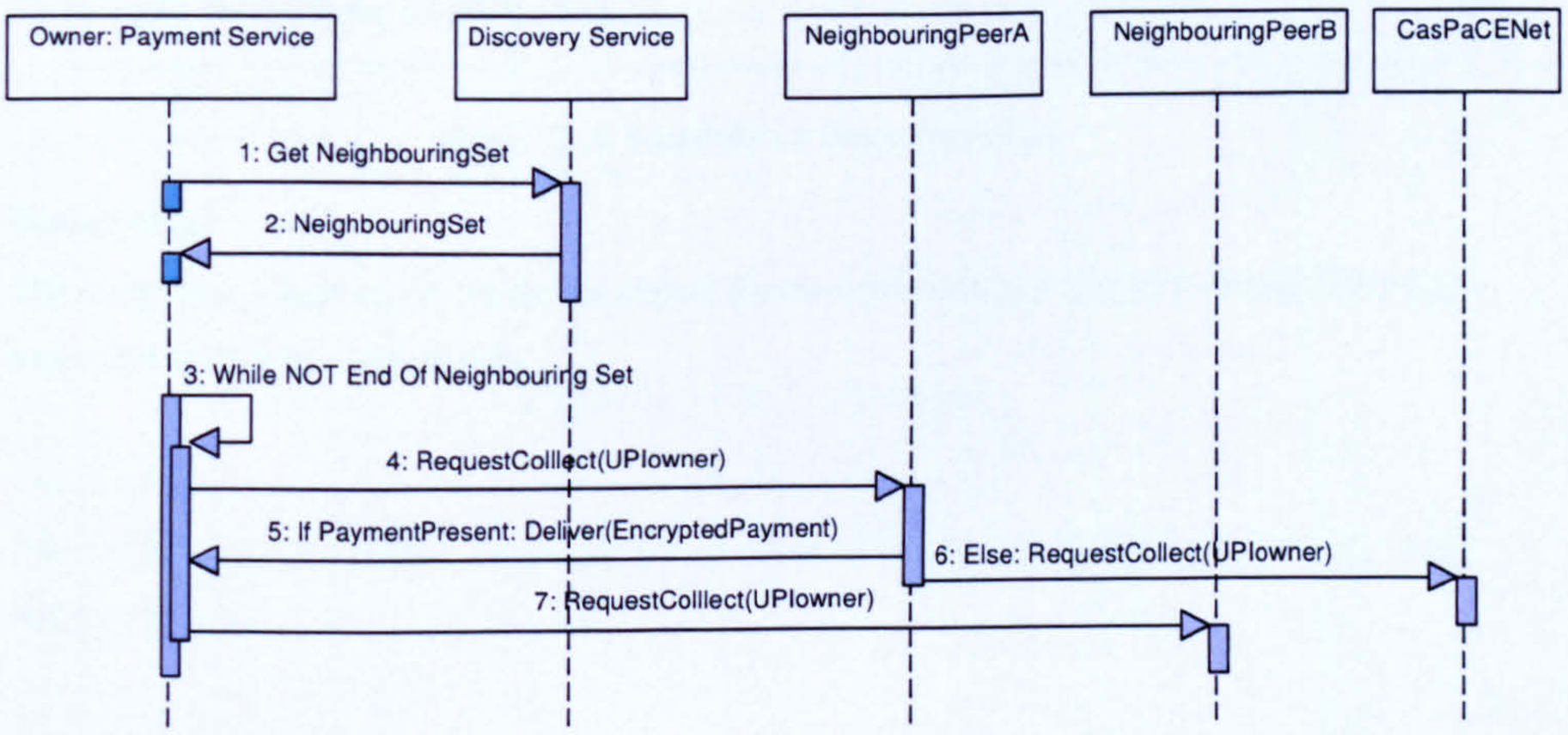


Figure D. 4: Payment collection

Description:

This Sequence Diagram illustrates the object interactions required to collect payments within this framework.

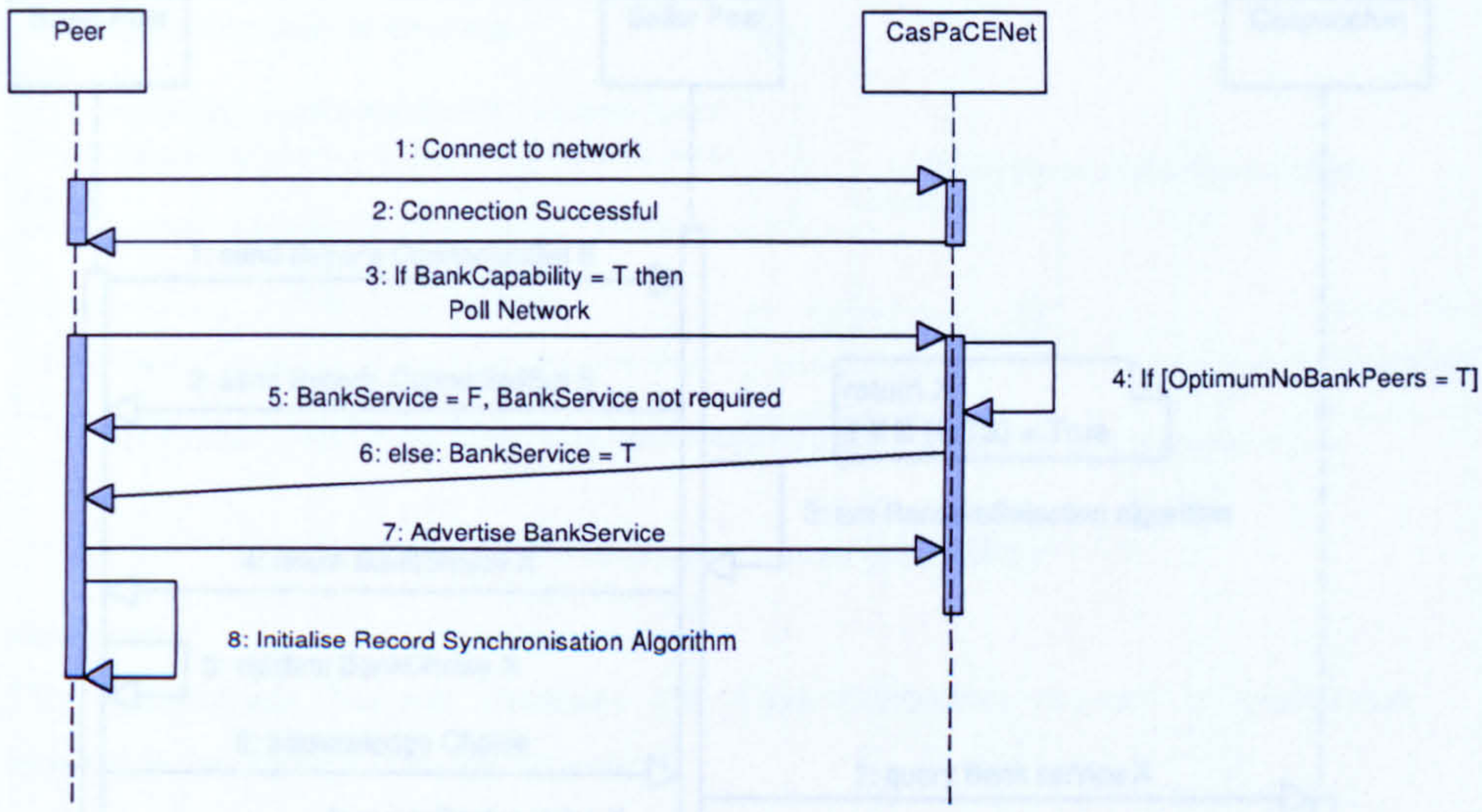


Figure D. 5: Bank service activation

Description:

This Sequence Diagram illustrates how a bank service is activated within this framework.

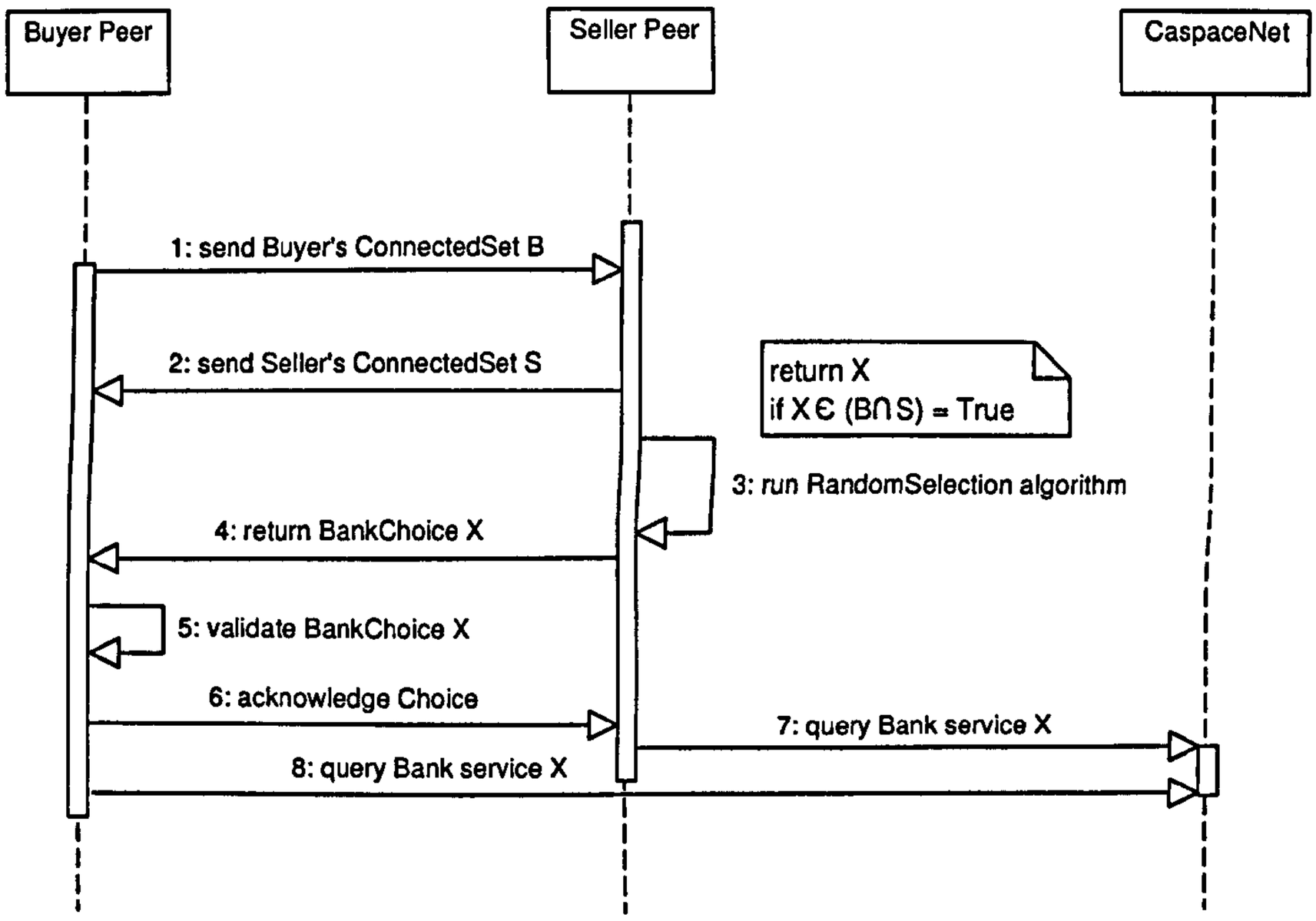


Figure D. 6: Random bank peer selection

Description:

This Sequence Diagram illustrates the object interactions required when randomly selecting a bank peer within this framework.

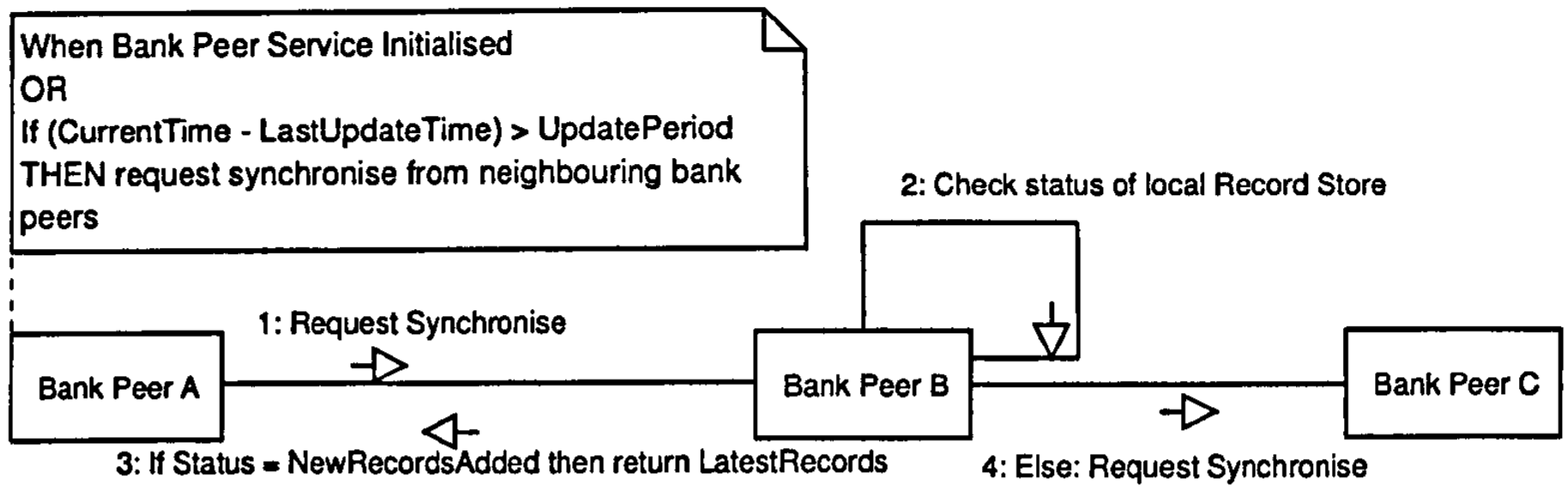


Figure D. 7: Synchronise in bank peer overlay

Description:

This Collaboration Diagram illustrates the object interactions required to synchronise transaction data stores within this framework.

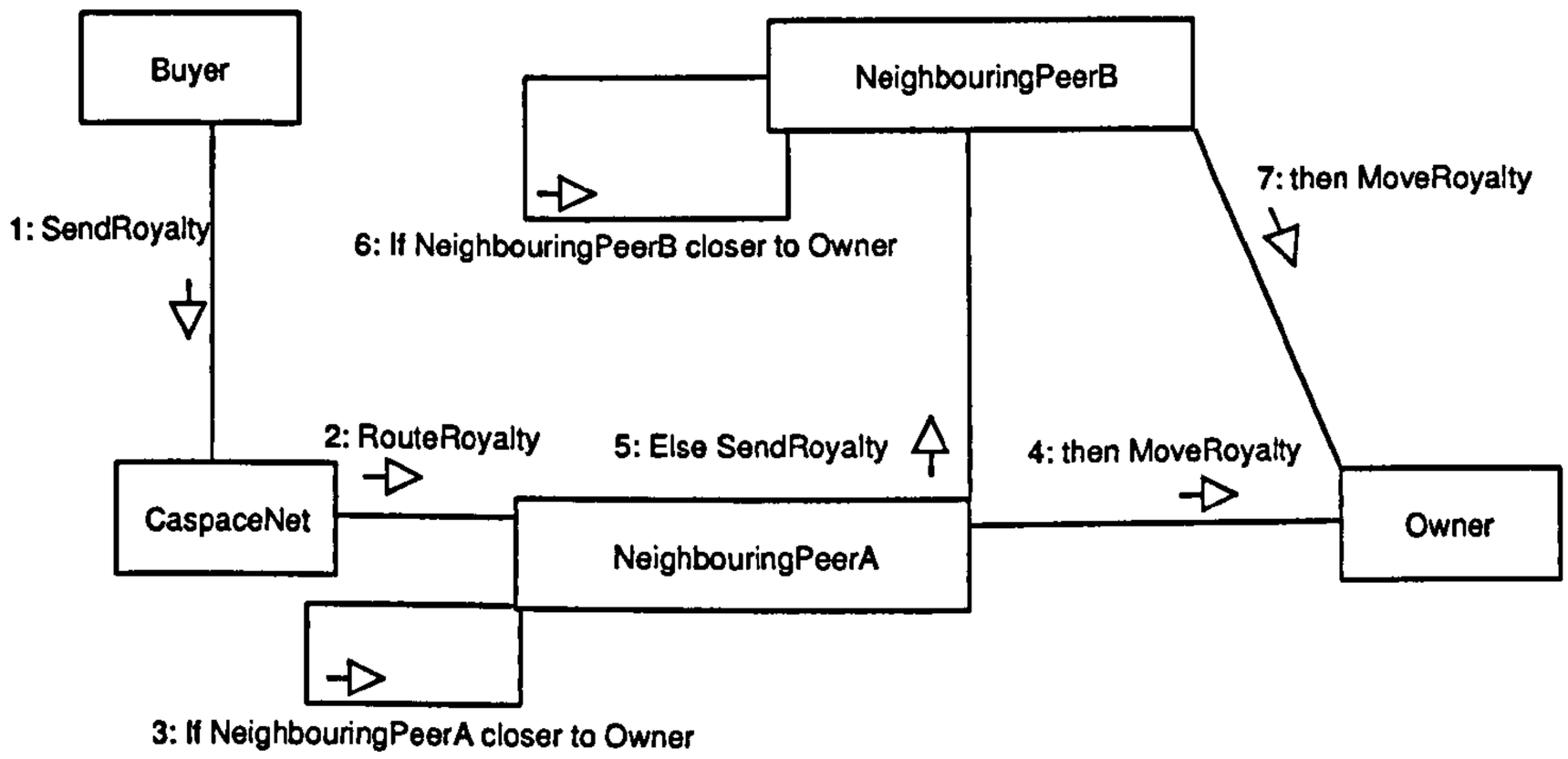


Figure D. 8: Payment pushing

Description:

This Collaboration Diagram illustrates the object interactions during payment pushing within this framework.

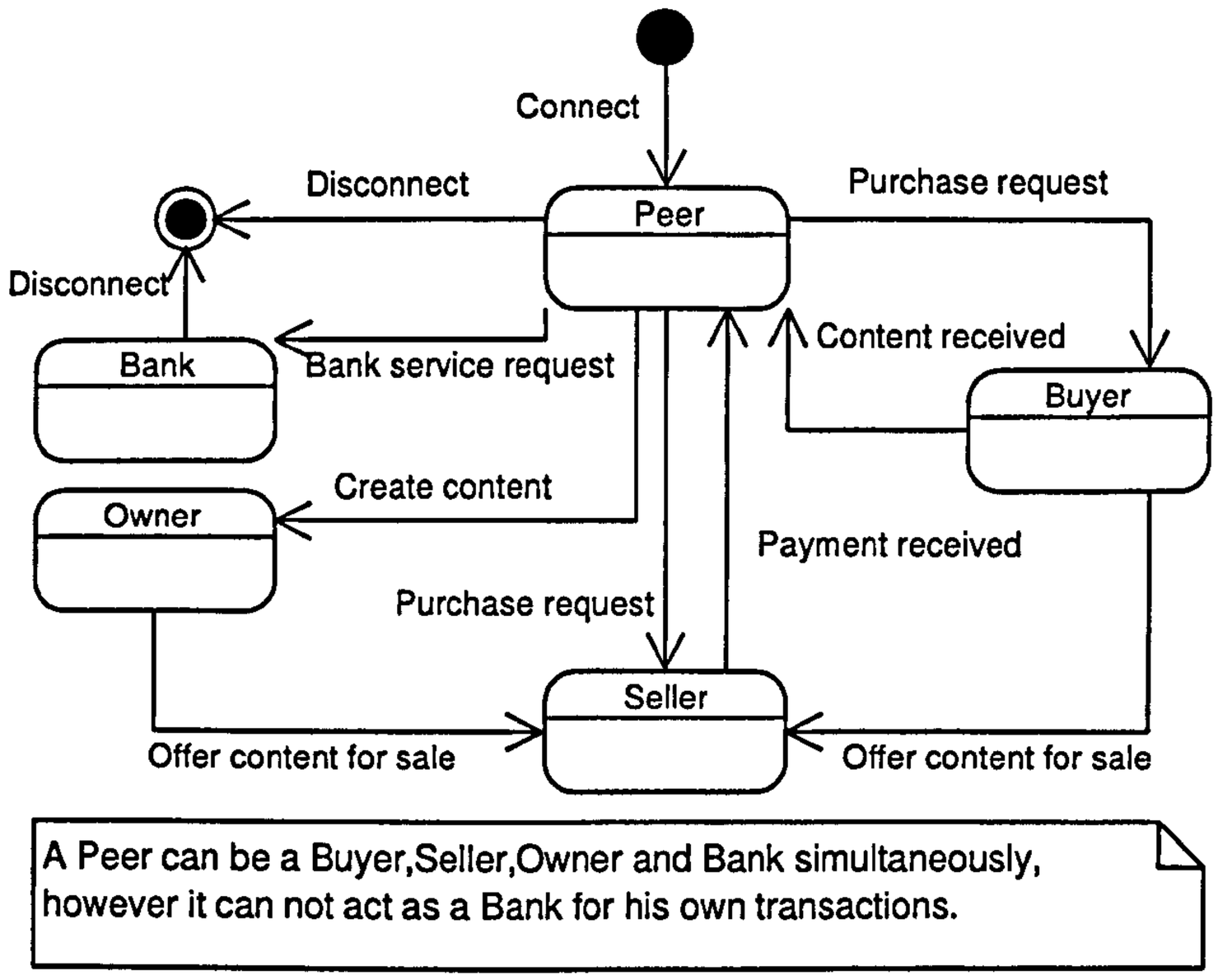


Figure D. 9: Peer roles state transition diagram

Description:

This State Transition Diagram illustrates the transitions between the various peer roles within this framework.