

*Meirong Liu*

# EFFICIENT SUPER-PEER- BASED COORDINATED SERVICE PROVISION

UNIVERSITY OF OULU GRADUATE SCHOOL;  
UNIVERSITY OF OULU,  
FACULTY OF INFORMATION TECHNOLOGY AND ELECTRICAL ENGINEERING,  
DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING





ACTA UNIVERSITATIS OULUENSIS  
C Technica 486

*MEIRONG LIU*

**EFFICIENT SUPER-PEER-BASED  
COORDINATED SERVICE PROVISION**

Academic dissertation to be presented with the assent of the Doctoral Training Committee of Technology and Natural Sciences of the University of Oulu for public defence in OP-sali (Auditorium L10), Linnanmaa, on 15 April 2014, at 12 noon

UNIVERSITY OF OULU, OULU 2014

Copyright © 2014  
Acta Univ. Oul. C 486, 2014

Supervised by  
Professor Mika Ylianttila  
Professor Timo Ojala

Reviewed by  
Professor Sasu Tarkoma  
Professor Zheng Yan

Opponent  
Professor Roch Glitho

ISBN 978-952-62-0399-7 (Paperback)  
ISBN 978-952-62-0400-0 (PDF)

ISSN 0355-3213 (Printed)  
ISSN 1796-2226 (Online)

Cover Design  
Raimo Ahonen

JUVENES PRINT  
TAMPERE 2014

## **Liu, Meirong, Efficient super-peer-based coordinated service provision.**

University of Oulu Graduate School; University of Oulu, Faculty of Information Technology and Electrical Engineering, Department of Computer Science and Engineering

*Acta Univ. Oul. C 486, 2014*

University of Oulu, P.O. Box 8000, FI-90014 University of Oulu, Finland

### ***Abstract***

Peer-to-Peer (P2P) networks have been applied in many applications for sharing resources such as storage space, media files or network bandwidth. Their main benefits include decentralization, self-organization, and scalability. Moreover, P2P technologies are evolving towards hybrid systems, where P2P networks are used in those parts of a larger system to leverage the decentralization most efficiently. The examples include cloud computing, where P2P networks are used in sharing computing resources, and Machine-to-Machine communication, where P2P networks are used for resource discovery.

In super-peer overlays, the nodes are either regular nodes or super nodes that are located higher in the node hierarchy. This type of overlay explores the heterogeneity of peers in the overlay network to enable applications to run more efficiently. Leveraging the advantage of a super-peer overlay for service provision is an important issue. This thesis contributes to the research and development of super-peer-based coordination service provision from three aspects.

Firstly, a super-peer-based coordinated service provision framework is proposed to coordinate different service groups and service peers in resource sharing, aiming to enable service groups to adapt to dynamic service demands. The proposed framework is evaluated using the following performance metrics: service response time, scalability, robustness, and communication traffic, in comparison to related work.

Secondly, an efficient algorithm for rapidly constructing a robust super-peer overlay is proposed. The algorithm introduces a super-peer candidate based method for super-peer selection and a two-hop search method for finding client peers. Performance evaluation takes into account the convergence time of building a super-peer overlay, communication overhead, scalability, robustness. A comparison to related work is also conducted.

Thirdly, the architecture of distributed resource discovery based on P2P overlay for Machine-to-Machine service provision is proposed. The architecture supports heterogeneous devices using different communication protocols in resource registration and discovery for achieving interoperability. As a part of the thesis, a functional real-world prototype is implemented and verified with a simple demonstration. Preliminary evaluation on the prototype indicates that caching can improve the response time of resource lookup dramatically.

*Keywords:* CoAP, coordination, distributed resource directory, Machine-to-Machine, overlay construction, Peer-to-Peer, resource discovery, service provision, super-peer



## **Liu, Meirong, Tehokas vertaisverkon ylinoodien koordinoima palveluntarjonta.**

Oulun yliopiston tutkijakoulu; Oulun yliopisto, Tieto- ja sähkötekniikan tiedekunta, Tietotekniikan osasto

*Acta Univ. Oul. C 486, 2014*

Oulun yliopisto, PL 8000, 90014 Oulun yliopisto

### ***Tiivistelmä***

Vertaisverkkoja on hyödynnetty resurssien kuten tallennustilan, mediasisältöjen ja tietoliikennekapasiteetin jakamisessa. Niiden etuja perinteisiin keskitettyihin järjestelmiin verrattuna ovat hajautettu arkkitehtuuri, itseorganisoituvuus ja skaalautuvuus. Vertaisverkkoja käytetään yhä useammin järjestelmän osien toteuttamisessa, joissa hajautettujen resurssien hyödyntämisellä saavutetaan suurimmat edut. Esimerkkeinä ovat pilvilaskenta, jossa vertaisverkkoa käytetään laskentaresurssien jakamiseen, sekä laitteidenvälinen kommunikaatio, jossa vertaisverkkoja käytetään resurssien löytämiseen.

Hierarkkisissa vertaisverkoissa niihin kytkeytyneet laitteet jaotellaan laitteiden kapasiteetin mukaan tavallisiin noodeihin ja näiden yläpuolella hierarkiassa toimiviin ylinoodeihin. Ylinoodeihin perustuva vertaisverkon kuoriverkko hyödyntää yksittäisten verkon nooidien eli laitteiden erilaisuutta, jotta verkko voisi toimia tehokkaammin. Tämän ominaisuuden hyödyntäminen on erityisen tärkeää palvelun tarjonnassa. Tässä työssä on tutkittu ylinoodeihin perustuvan vertaisverkon palvelun tarjontaa kolmesta näkökulmasta.

Ensimmäiseksi, työssä ehdotetaan ylinoodien koordinoimaa palveluntarjonnan toimintamallia resurssien jakamisessa. Toimintamallissa palveluryhmät ja palvelunoodit adaptoituvat dynaamisesti palvelupyyntöjen tarpeisiin. Tämän ratkaisun suorituskykyä arvioidaan palvelun vasteajan, skaalautuvuuden, robustisuuden ja tietoliikennemäärän suhteen verrattuna aiempiin ratkaisuihin.

Toiseksi, työssä esitellään tehokas algoritmi robustin ylinoodikuoriverkon nopeaan muodostamiseen. Algoritmi käyttää ylinoodiehdokasmenetelmää ja kahden hypyn hakumetodia asiakasnooidien etsimisessä. Suorituskyvyn arvioinnissa otetaan huomioon ylinoodikuoriverkon konvergoitumisaika, tietoliikenneviestinnän aiheuttama ylimääräinen kuormitus, sekä järjestelmän skaalautuvuus ja robustisuus. Esitetyn algoritmin tehokkuutta arvioidaan vertaamalla näitä suorituskykymittareita aiempiin ratkaisuihin.

Kolmanneksi, työssä esitellään hajautettu resurssihakemiston arkkitehtuuri, joka perustuu laitteiden välisen kuoriverkon palveluntarjontaan. Arkkitehtuuri tukee erilaisten laitteiden ja niiden käyttämien protokollien resurssien rekisteröintiä ja löytämistä yhteensopivuuden saavuttamiseksi. Väitöskirjatyön osana on toteutettu toimiva prototyyppi, jonka toimivuus on todennettu demonstraation avulla. Prototyypillä tehdyt mittaukset antavat perustellun syyn olettaa, että esitetyn ratkaisun mukainen välimuistin käyttö voi merkittävästi lyhentää resurssien etsimisen vastaikaa.

*Asiasanat:* hajautettu resurssihakemisto, koordinaatio, kuoriverkon muodostaminen, laitteiden välinen verkkopalvelu, palveluntarjonta, resurssien hakupalvelu, vertaisverkot, ylinoodi





## Preface

The research work presented in this thesis has been carried out at the MediaTeam Oulu research group, Department of Computer Science and Engineering, University of Oulu. Specifically, this thesis work was supported by the ITEA2-EXPESHARE, DECICOM, SOPSCC, and MAMMOTH projects.

I would like to express my deep gratitude to my supervisors, Professor Mika Ylianttila and Professor Timo Ojala, for supervising this thesis and guiding me throughout my post-graduate studies. Without their direction, guidance, and support, this thesis would not have been possible. I would also like to thank Professor Jukka Riekkı for the project work and his valuable suggestions on writing a journal paper.

I am grateful to the official reviewers, Professor Sasu Tarkoma from the University of Helsinki and Professor Zheng Yan from Aalto University, Finland and Xidian University, China, for their valuable feedback. I would also like to thank Professor Roch Glitho for serving as the opponent in the defense.

I owe thanks to all the team members whom I have worked with, including Mr. Xiang Su, Ms. Archana Ramalingam, Ms. Ekaterina Gilman, Mr. Pauli Närhi. I would specially like to thank Mr. Erkki Harjula, Dr. Timo Koskela, Mr. Teemu Leppänen, Dr. Ou ZhongHong, and Dr. Jiehan Zhou for close co-operation, discussions and other contribution in the original publications in this thesis, and the former program manager Ms. Marika Leskelä for providing me much help and encouragement during my research. Thanks to all the members in the laboratory for creating a pleasant atmosphere.

Lastly, I dedicate this work to my beloved parents and family for their support.

Oulu, December, 2013

Meirong Liu



## List of abbreviations

6LoWPAN	IPv6 over Low power Wireless Personal Area Networks
ARPANET	Advanced Research Projects Agency Network
ARQ	Automatic Repeat Request
BFS	Breadth-First
CASA	Collaborative Adaptive Sensing of the Atmosphere
CoAP	Constrained Application Protocol
CoRE	Constrained RESTful Environments
CPU	Central Processing Unit
CS	Client-Server
DCAS	Distributed Collaborative Adaptive Sensing
DFS	Depth-First Search
DHT	Distributed Hash Table
DNS	Domain Name System
DRD4M	Distributed Resource Directory for Machine-To-Machine
EC2	Elastic Compute Cloud
GC	Global-State-Based Centralized
GSPS	Gossip-Based Super-Peer Selection Algorithm
HONet	Hybrid Overlay Networks
HTTP	Hypertext Transfer Protocol
ID	Identifier
IP	Internet Protocol
LDAP	Lightweight Directory Access Protocol
LTE	Long Term Evolution
IETF	Internet Engineering Task Force
I/O	Input/Output
MANET	Mobile Ad hoc Network
M2M	Machine-to-Machine
NFC	Near Field Communication
OS	Operating System
PC	Personal Computer
PDG	Perfect Difference Graph
RFID	Radio Frequency Identification
P2P	Peer-to-Peer
QoS	Quality of Service
RAM	Random Access Memory

RD	Resource discovery
RLC	Resource lookup component
RRC	Resource registration component
RTTs	Round Trip Times
SCSP	Super-Peer-Based Coordinated Service Provision
SD	Service Discovery
SLP	Service Location Protocol
SNG	Square Network Graph
SOA	Service Oriented Architecture
SON	Semantic Overlay Network
SPChord	Structured Super-Peer Overlay
SRRT	Response Time of a Service Group to a Service Request
SSNG	Self-Similar Square Network Graph
SWOP	Small World Overlay Protocol
TAG	Tiny Aggregation
TTL	Time-To-Live
UDDI	Universal Description, Discovery, and Integration
UPnP	Universal Plug and Play
URL	Uniform Resource Locator
VMs	Virtual machines
VoIP	Voice over Internet Protocol
WLAN	Wireless Local Area Network
WSN	Wireless Sensor Network
XMHT	Extendible Metadata Hash Table
YAPPERS	Yet Another Peer-to-PEER System

## List of original publications

This thesis is based on the following original publications, which are referred to in the text by Roman numerals (I-V):

- I Liu M, Koskela T, Ou Z, Zhou J, Riekkilä J & Ylianttila M (2011) Super-Peer-based Coordinated Service Provision. *Journal of Network and Computer Applications* 34 (4): 1210–1224.
- II Liu M, Koskela T & Ylianttila M (2011) An Efficient Collaborative Mechanism for Enterprise Cloud Applications. *Proc IEEE International Conference on Networking, Sensing and Control*. Delft, Netherlands: 163–168.
- III Liu M, Zhou J, Koskela T & Ylianttila M (2009) A Robust Algorithm for the Membership Management of Super-Peer Overlay. *Proc IFIP/IEEE International Conference on Management of Multimedia and Mobile Networks and Services*, Springer Lecture Notes in Computer Science 5842. Venice, Italy: 132–143.
- IV Liu M, Harjula E & Ylianttila M (2013) An efficient selection algorithm for building a super-peer overlay. *Journal of Internet Services and Applications* 4(4): 1–12.
- V Liu M, Leppänen T, Harjula E, Ou Z, Ramalingam A, Ylianttila M & Ojala T (2013) Distributed resource directory architecture in Machine-to-Machine communications. *Proc IEEE International conference on Wireless and mobile computing, networking and communication Workshop on the Internet of Things Communications and Technologies*. Lyon, France: 319–324.

Papers I-II focus on the super-peer-based coordination framework for service provision. Papers III and IV address the research issues on the super-peer overlay construction. Finally, Paper V focuses on designing a distributed resource discovery architecture based on P2P overlay for Machine-to-Machine (M2M) service provision.

In addition, the author has also actively authored or co-authored eight conference papers in the relevant research areas, including Machine-to-Machine communication, wireless sensor network, Web Services, and overlay construction.



# Table of contents

<b>Abstract</b>	
<b>Tiivistelmä</b>	
<b>Preface</b>	<b>7</b>
<b>List of abbreviations</b>	<b>9</b>
<b>List of original publications</b>	<b>11</b>
<b>Table of contents</b>	<b>13</b>
<b>1 Introduction</b>	<b>15</b>
1.1 Background .....	15
1.2 Motivation and research problems .....	18
1.3 Research methodology .....	21
1.4 Contributions of the thesis .....	23
1.5 Organization of the thesis.....	25
<b>2 Literature review</b>	<b>27</b>
2.1 P2P overlay networks.....	27
2.1.1 Key concepts of P2P.....	27
2.1.2 Taxonomy of P2P overlays.....	29
2.1.3 Super-peer overlay.....	30
2.1.4 P2P networks versus mobile ad hoc networks.....	36
2.2 Super-peer-overlay-based service provision .....	37
2.2.1 Overview of P2P applications .....	37
2.2.2 Coordinated service provision leveraging super-peer overlay.....	38
2.3 Super-peer overlay construction.....	41
2.3.1 Overview of overlay construction .....	42
2.3.2 Super-peer overlay construction .....	46
2.4 P2P-overlay-based service discovery.....	50
2.4.1 Overview of service discovery .....	50
2.4.2 P2P-based service discovery for constrained devices .....	53
<b>3 Summary of research contributions</b>	<b>57</b>
3.1 Super-peer-based coordinated service provision.....	57
3.2 An efficient algorithm for constructing a super-peer overlay .....	60
3.3 Distributed resource discovery architecture based on a P2P overlay for Machine-to-Machine service provision .....	63
3.4 Discussion and future work.....	65
<b>4 Conclusions</b>	<b>67</b>
	13

<b>References</b>	<b>69</b>
<b>Original publications</b>	<b>81</b>



# 1 Introduction

## 1.1 Background

Peer-to-Peer (P2P) technology has been applied in many applications, such as personal communication systems, e.g. Skype (Baset & Schulzrinne 2006), decentralized content (file) sharing systems, e.g. BitTorrent (Cohen 2009), streaming media delivery systems, e.g. PPLive (PPLive 2009), and PPStream (Liang *et al.* 2009). According to Labovitz *et al.* (2010), P2P applications generate at least 20% of all Inter-domain traffic on the Internet. Its overall traffic volume continues to increase even though P2P traffic does not dominate Internet traffic today (Moya 2009) (Piatek *et al.* 2009) (Iliofotou *et al.* 2011) (Mondal *et al.* 2012). P2P-based applications, especially the P2P video streaming, continue gaining popularity in Asian countries according to the traffic report published by Ericsson in 2012 (Ericsson 2012). Moreover, P2P is evolving and applied in the new emerging paradigms such as Internet of Things (Kortuem *et al.* 2010), content-centric networking (Helgason *et al.* 2010) (Mondal *et al.* 2012), opportunistic computing (Conti *et al.* 2010), cloud computing (Ranjan & Rajkumar 2010) (Ranjan & Zhao 2013) and Machine-to-Machine (M2M) computing (Tanganelli *et al.* 2013) because of its strengths in self-organization, decentralization and scalability.

In P2P-based systems, computers (or nodes or peers) are connected in a distributed way. Peers can communicate with each other directly for sharing resources without any central control, which is opposed to client-server systems in which clients can only access resources hosted by centralized servers. Thus, P2P systems do not suffer from a single-point of failure. Traditional P2P systems mainly focus on simple applications such as sharing files, content, or CPU cycles. The target of collaboration among peers nowadays has expanded beyond that. Emerging collaborative P2P systems target at bringing in a group of diverse peers with unique capabilities (e.g. hardware, software, services, and data) to work in cooperation in order to accomplish greater tasks beyond those that can be accomplished by individual peers, yet are beneficial to all the peers (Bandara & Jayasumana 2013). Such collaborations among peers involve diverse application specific resources and dynamic goals on Quality of Service (QoS). Coordination mechanisms are always needed for managing such diverse resources in order to achieve the expected QoS goals.

Collaborative P2P systems are applicable in a wide variety of contexts such as Distributed Collaborative Adaptive Sensing (DCAS) (Brotzge *et al.* 2004) (Lee *et al.* 2012), and cloud computing (Ranjan *et al.* 2010). For example, DCAS applications need to detect hazardous atmospheric conditions (e.g. tornados and severe storms) in real time, which involve a large number of distributed, resourceful and high-data-rate sensors. The data generation rate at each of the radars (sensors) ranges from tens to hundreds of Mbps. Such a large number of generated data by sensors can cause network congestion close to the server (Zhang & Li 2010). The conventional client-server framework has limited scalability and suffers from a single-point-of-failure, which can be catastrophic in CASA (Collaborative Adaptive Sensing of the Atmosphere) systems that perform critical functions under hazardous weather conditions (Lee *et al.* 2012). The distributed collaborative P2P-based framework fits naturally to a large-scale DCAS system because it can enable DCAS systems to aggregate and utilize unused resources from anywhere in the system to achieve better performance and QoS while enhancing the overall resource utilization (Anderson *et al.* 2002) (Lee *et al.* 2012). Lee *et al.* (2012) verified that a P2P-based collaboration framework for CASA multi-sensor data fusion achieved a significant performance improvement compared with traditional client-server paradigm.

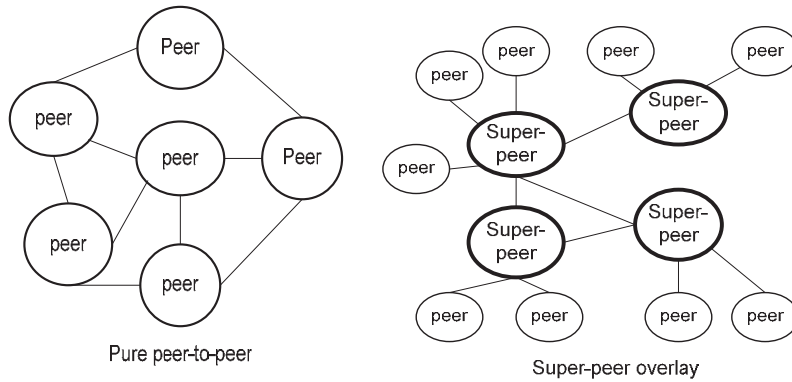
Clouds (e.g. Amazon EC2 (Varia 2008) and Google App Engine (Zahariev 2009)) have evolved as popular platforms to provide wide-area on-demand service provision of computing, and storage for applications requiring high availability and performance. Cloud computing enables government agencies and financial organizations to cut their costs by leasing IT services hosting and maintenance from an external cloud. Cloud computing also provides organizations a more cost effective platform to manage peak-load by using the cloud. The components of a cloud service platform (i.e. services, VMs, physical servers, storage) may be distributed. However, the overall service monitoring, discovery, and load-balancing functionalities usually employ centralized approaches (Amazon cloudwatch service 2010) (Amazon load balancer service 2010) (Windows azure platform 2010). These centralized approaches usually suffer from scalability, performance, and reliability (e.g. single point of failure), which arise from the management of multiple service queues and large volume of service requests (Ranjan *et al.* 2008) (Quiroz *et al.* 2009). Thus, the P2P-based approach has been investigated to manage the cloud service provision (Ranjan *et al.* 2010) (Ranjan & Rajkumar 2010) (Ranjan & Zhao 2013). Evaluation using the Amazon EC2 by Ranjan & Zhao (2013) demonstrated that managing services based on P2P

routing and information dissemination offers significant performance benefits as regards to overall system reliability, scalability, and self-management.

P2P-based service provision usually adopts different overlay structures, which vary in the degrees of centralization or the type of network structure. For example, P2P overlay structures can be categorized into hierarchical or flat, structured or unstructured. Hierarchical overlays organize peers in more than one layer, and provide advantages such as local traffic not affecting the other layers, network proximity, efficient caching, etc.

Super-peer overlays are an important type of hierarchical P2P overlays, which take into account the heterogeneity of peers in the overlay and organize peers into a super-peer layer and a client-peer layer (e.g. Gnutella (Klingberg & Manfredi 2002)). In other words, a super-peer overlay network is simply a P2P overlay network that consists of the super-peers and their client peers. Each client peer should connect to a super-peer in order to communicate with other peers in the overlay. Client peers submit queries to their super-peers and receive results from those super-peers. A super-peer acts as a centralized server for its client peers and connects to other super-peers in the same way as in a pure P2P network for routing messages over the overlay network (Yang & Garcia-Molina 2003). Fig. 1 demonstrates a pure P2P system versus a super-peer-overlay-based system. In a pure P2P system, peers take the same responsibility and connect with each other randomly. In a super-peer overlay, super-peers take more responsibilities than client peers. Client peers with low capacity are shielded from massive query traffic by super-peers, which improves the scalability of the system and enables other peers (e.g. mobile devices) to join the P2P network as client peers. Super-peer overlays explore the heterogeneity of nodes in the overlay network to enable applications to run more efficiently. For example, in Skype voice streaming, and live video streaming applications, the performance of these applications is improved through assigning nodes with high network bandwidth, long on-line time, or high processing capability as super-peers (Wang *et al.* 2008).

In brief, P2P technology has been utilized to achieve decentralization, self-organization, and scalability for the emerging network service provision. Meeting special QoS (e.g. response time and resource efficiency) in coordinated service provision using P2P technology is a challenging topic. This thesis investigates three different aspects of coordinated service provision using a P2P overlay: the coordination mechanism, an algorithm for constructing a super-peer overlay, and distributed resource discovery based on a P2P overlay.



**Fig. 1. A pure P2P system versus a super-peer-overlay-based system.**

## 1.2 Motivation and research problems

Web services have been a popular middleware to provide dynamic integration and interaction of heterogeneous software artifacts (Alonso *et al.* 2004). One of the challenges in Web service provision is how to make the Web service providers to adapt themselves to satisfy requirements in dynamic service demand (Cuenca-Acuna & Nguyen 2004) (Foster 2005) (Pacifci *et al.* 2005) (Papazoglou *et al.* 2007) (Chen *et al.* 2008) (Brazier *et al.* 2009). This challenge widely exists in the applications that are computational intensive and have dynamically fluctuating service demands, such as in the fields of cosmology, climate and grid services (Butt *et al.* 2006) (Goel *et al.* 2007) (Ranjan *et al.* 2008), bioinformatics analysis (Chakravarti *et al.* 2005), and distributed collaborative adaptive sensing systems (Lee *et al.* 2012).

In these applications, on the one hand, simple Web services are required; on the other hand, a considerable amount of computation (i.e. a large number of Web services) is required to process the large data sets (Foster 2005). Furthermore, as service entities are autonomous and heterogeneous, connecting and coordinating them is a delicate and time-consuming task (Benatallah *et al.* 2002). At last, these applications usually involve a large number of service entities. Thus, efficient coordinate mechanisms are needed to coordinate these service providers to adapt themselves to meet the dynamically changing service demands (Foster 2005) (Pacifci *et al.* 2005) (Papazoglou *et al.* 2007) (Chen *et al.* 2008) (Brazier *et al.* 2009).

The P2P approach has been widely employed to support a large scale of Internet service provision, including enterprise applications, workflow management, coordination or integration of distributed enterprises peers (Gu & Nahrstedt 2006) (Goel *et al.* 2007) (Votis *et al.* 2008) (Ranjan *et al.* 2008). Super-peer overlay networks explore the heterogeneity of peers and have been utilized in service provision in order to improve service performance (e.g. Skype).

Based on the above-mentioned observations, the following research question is addressed in this thesis:

*Q1:* How to utilize a super-peer overlay in the design of the coordination mechanism? What kind of efficient coordinated service provision mechanism can coordinate different service groups for sharing resources in order to handle the service provision with dynamic service demands?

Besides investigating the design of an efficient coordination mechanism based on a P2P overlay, the mechanism of constructing a P2P overlay should also be considered because the structure of a utilized P2P overlay in the service provision has an impact on the performance of service provision. Many studies on service provision always assume that a P2P overlay already exists or utilize a pre-configured overlay (Ganesh *et al.* 2003) (Xiao *et al.* 2005) (Min *et al.* 2006). Thus, an efficient algorithm for constructing a super-peer overlay for service provision is needed. In a P2P overlay, peers usually have heterogeneous resources in network bandwidth, processing capability, online time, battery life, etc. Both the criteria for selecting super-peers and the mechanism of quickly selecting super-peers should be studied. Furthermore, in a dynamic network environment, super-peers could suffer from failure and client peers need to find a new super-peer when their super-peers are failed. It is also common that peers join or leave the overlay independently (called churn) (Stutzbach & Rejaie 2006). The robustness of a super-peer overlay to failure of super-peers and to the churn of peers (dynamics of peer participation) is a common performance requirement that should be investigated in the design of an overlay construction algorithm (Stutzbach & Rejaie 2006).

Based on the above-mentioned observations, another research question is addressed in this thesis:

*Q2:* What is an efficient algorithm that can quickly construct a super-peer overlay that is robust to the failure of peers given a random connected topology? What kind of criteria should be applied in selection of super-peers? How does the algorithm scale up in constructing a large overlay, e.g. 10000 nodes? It should be

noted that in this thesis a robust super-peer overlay means a super-peer overlay that is robust to the failure of super-peers and churn of nodes.

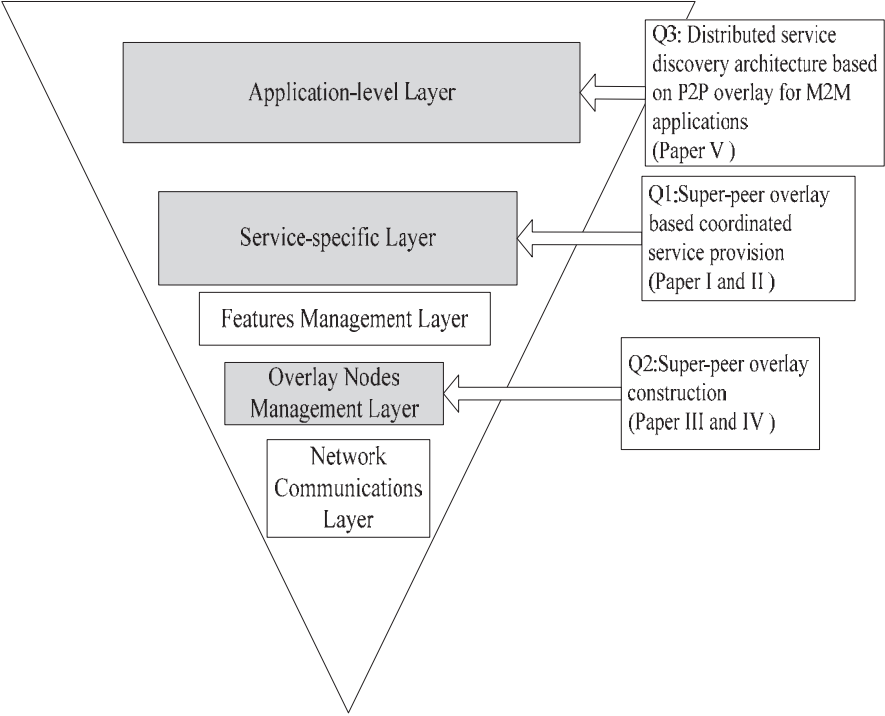
Because of the rapid development of the Internet and telecommunication technology, more smart devices and embedded devices are able to participate in service provision. These devices are able to connect to the Internet, expose their services into the Internet and access services provided by other devices (Kortuem *et al.* 2010) (Tarkoma & Ailisto 2013). These devices feature limited processing capabilities, limited battery lifetime, processing memory or constrained network connections (Leppanen *et al.* 2012). These constrained devices interact with each other and construct a highly distributed embedded network, referred to as Machine-to-Machine (M2M) communication network. Although many studies have been done on service discovery using P2P technology, such as Banaei-Kashani *et al.* (2004) and Sioutas *et al.* (2008), these solutions cannot fulfill the requirements in service discovery for constrained devices in M2M communication network. For example, it is not feasible to directly lookup resources provided by constrained devices when devices are in sleeping mode or have intermittent connection to the network, and disperse networks (Shelby *et al.* 2012). In addition, a new communication protocol CoAP (Constrained Application Protocol), proposed for these embedded devices for saving energy, should be supported in the resource discovery design. The issue on designing an architecture that can meet the requirements of service discovery for constrained devices in M2M service provision should be investigated.

Motivated by above-mentioned observations, a third research question is addressed in this thesis:

*Q3*: What kind of distributed resource discovery architecture can enable heterogeneous constrained devices to register and lookup resource descriptions using different communication protocols (e.g. HTTP and CoAP (Shelby *et al.* 2013)) for M2M service provision? What are the benefits of utilizing P2P overlay in the architecture?

Fig. 2 demonstrates the three research areas investigated in this thesis according to the P2P overlay network layered architecture as proposed by Lua *et al.* (2005). Specifically, this thesis contributed to the Service-specific Layer (Papers I and II), the Overlay Nodes Management Layer (Papers III and IV), and Application-level Layer (Paper V). According to Lua *et al.* (2005), P2P overlay networks span a wide spectrum of the communication framework and consist of five layers. The Overlay Nodes Management Layer focuses on the management of peers, including discovery of peers and routing algorithms for optimization.

The Services-specific Layer focuses on supporting application-specific components through scheduling of parallel and computation-intensive tasks, content and file management. Application-level layer focuses on tools, applications, and services, which are implemented with specific functionalities on top of the underlying P2P overlay infrastructure. More details about the topics on other layers can be found in Lua *et al.* (2005).



**Fig. 2. The mapping of the original publications to the research areas and the research questions of this thesis according to the architecture of P2P overlay network as proposed by Lua *et al.* (2005).**

### 1.3 Research methodology

This thesis focuses on using the super-peer overlay for coordinated service provision, including a coordination mechanism design and an efficient algorithm for constructing a super-peer overlay. Meanwhile, a distributed resource discovery architecture based on a P2P overlay for heterogeneous constrained devices to

register and lookup resource descriptions is investigated. In order to solve the research questions specified in Section 1.2, three different methodological approaches are employed in this thesis.

Firstly, a simulation model is utilized for performance evaluation. Specifically, a simulation model is employed in Papers I and II to evaluate the performance of super-peer-overlay-based coordinated service provision framework. The performance evaluation utilized the metrics (or measures) that are common to P2P-based service provision, namely robustness to the failure of peers, scalability, service response time, and the traffic generated by the coordination mechanism (i.e. communication messages) (Chen *et al.* 2008) (Montresor & Jelasity 2009). The simulation model is also utilized in Papers III and IV for examining the performance of the proposed algorithm for constructing a super-peer overlay. The performance metrics (or measure) utilized in Papers III and IV are robustness to the failure of peers, scalability, communication overhead in terms of constructing the overlay, the convergence time of constructing a super-overlay, and parameters' impact on the convergence time of constructing a super-overlay (Montresor & Jelasity 2009) (Garbacki *et al.* 2010). Simulations are carried out on the PeerSim platform (Montresor & Jelasity 2009), which is a software that can be deployed with P2P simulation models to simulate P2P overlay networks for performance evaluation. The simulation model is very useful for the evaluation of P2P applications with a large number of peers because it is not easy to deploy and evaluate a real-world prototype with a large number of peers (e.g. 10000). Using the PeerSim platform, deploying applications with a large number of peers in an overlay becomes realistic. Both the number of peers and the initial overlay utilized for constructing a super-peer overlay can be set in the PeerSim platform. A comparison between PeerSim and other P2P simulators can be found in Naicken *et al.* (2007).

Secondly, a mathematical analysis method is used in Paper I for analyzing the traffic generated by the coordination mechanism in order to make a comparison with related work. Traffic overhead is one of the most important performance metrics that should be taken into account in performance evaluation. Mathematical analysis provides a method to seek approximate solutions while maintaining reasonable bounds on errors. Mathematical analysis makes it feasible to compare the traffic overhead generated by our proposed P2P-based service provision with related work.

Thirdly, a real-world prototype of a distributed resource directory based on a P2P overlay for M2M service provision (DRD4M) is implemented for verifying



its feasibility and functionality in Paper V. The author of this thesis developed the prototype. The prototype is deployed in the embedded device, Gumstix Overo™ (Gumstix). The constrained devices used for testing the DRD4M include Atmel 1284P microcontroller-based embedded sensor devices with 16KB of RAM (Leppanen *et al.* 2012) and Android smart phone Samsung Galaxy SIII. The wireless sensor uses CoAP messages for registration and look-up resources from the DRD4M. The smart phone uses HTTP messages over Wi-Fi network to communicate with the DRD4M and wireless sensors. The wireless sensor and the smartphone application are not developed by the author, but are used for testing the prototype of the DRD4M. Preliminary evaluation was conducted. The performance metric of response time on looking-up resources was evaluated.

#### **1.4 Contributions of the thesis**

This thesis contributes from three different aspects. A brief overview of the contributions regarding original papers is given below. Specific contributions of each original paper are elaborated in more detail in Chapter 3.

The first contribution of this thesis is design and evaluation of an efficient coordinated service provision framework for handling service provision that is computational intensive and has dynamic service demands. Papers I and II contribute to this research area. Specifically, Paper I proposes a super-peer-based coordinated service provision framework (SCSP) to enable collaboration and resource sharing among different service groups. The SCSP consists of an S-labor-market model (super-peer-based labor-market model), a recruiting protocol based on a weighting mechanism, and an optimal dispatch algorithm. The market model has been widely applied to regulate the coordination among distributed entities because it provides an analogy between coordinated systems and the social recruiting structure (Buyya *et al.* 2000) (Li & Li 2009). In this thesis, the S-labor-market model is designed to build the coordination among service groups by employing the proposed recruiting protocol. The optimal dispatch algorithm is designed to select the optimal service peers within a service group to process service requests. Each service group consists of a super-peer and several client peers. Performance of the SCSP is evaluated in terms of service response time, scalability, and robustness to the peer joins and leaves using the PeerSim platform. In the evaluation, the SCSP is also compared with Chen *et al.* (2008), a structured DHT-overlay-based service provision. In Papers I, the author was responsible for designing and implementing the SCSP framework, carrying out simulations, and

writing the paper. Dr. Koskela worked on traffic analysis, participated in technical discussion and paper finalization. Dr. Ou participated in architecture discussion. Dr. Zhou and Prof. Riekkilä participated in paper finalization. Prof. Ylianttila was the supervisor. Paper II presents an efficient coordination mechanism for cloud service provision. The author was responsible for the design and implementation of the coordination mechanism, carrying out simulation and writing the paper. Dr. Koskela provided comments. Prof. Ylianttila was the supervisor.

The second contribution of this thesis is designing an efficient algorithm for quickly constructing a super-peer overlay that is robust to failure of peers. Papers III and IV contribute to this research area. Specifically, Paper III presents a gossip-based super-peer selection (GSPS) algorithm for quickly constructing a super-peer overlay. The GSPS algorithm introduces the concept of super-peer candidates to speed up the selection of super-peers. The performance of the GSPS is evaluated using the PeerSim platform with different application scenarios. The performance metrics take into account the convergence time of constructing the super-overlay, scalability, and robustness to the failure of peers. In the evaluation, the proposed GSPS algorithm is also compared with the related work SG-1 (Montresor 2004). The author of this thesis was responsible for designing and implementing the GSPS algorithm, carrying out the performance evaluation and writing the paper. Dr. Zhou and Dr. Koskela participated in finalizing the paper. Prof. Ylianttila was the supervisor. Paper IV continues the work in Paper III and carries out additional experiments to evaluate the performance of the GSPS algorithm in terms of communication overhead, scalability, and robustness. Comparison between the GSPS algorithm and SG-1 is also made. The author conducted the performance evaluation and wrote the paper. Mr. Harjula participated in paper finalization. Prof. Ylianttila was the supervisor.

The third contribution of this thesis is designing a distributed resource directory based on a P2P overlay that enables heterogeneous constrained devices to register and lookup resource descriptions using different communication protocols. Paper V proposes the DRD4M that supports both CoAP and HTTP protocols. A P2P overlay is utilized to connect directory peers for prevention of a single-point of failure. A real-world prototype of the DRD4M is implemented. The feasibility and functionality of the DRD4M is evaluated with an application involving a smartphone application and wireless sensors. The author designed the architecture, implemented the prototype, carried out functionality testing, and wrote the paper. Mr. Leppänen participated in the design of devices communication interface, technical discussion, setting up wireless sensors for

functionality testing and finalizing the paper. Mr. Harjula provided discussion on architecture design and participated in finalizing the paper. Dr. Ou provided technical discussion and participated in finalizing the paper. Ms. Ramalingam set up the smartphone for the functionality testing of the DRD4M. Prof. Ylianttila and Prof. Ojala were the supervisors.

## **1.5 Organization of the thesis**

This thesis is organized as follows: Chapter 2 presents a literature review on P2P overlay networks, P2P-overlay-based service provision, super-peer overlay construction, and service discovery based on a P2P overlay. Chapter 3 summarizes the main contributions of the original papers, provides discussion and ideas for future work. This chapter consists of four parts: a super-peer-based coordinated service provision framework with performance evaluation, an efficient algorithm for quickly constructing a robust super-peer overlay with performance evaluation, distributed resource discovery architecture based on a P2P overlay for heterogeneous constrained devices in M2M service provision as well as discussion and future work. Chapter 4 concludes this thesis.



## 2 Literature review

This chapter provides an overview of the literature on super-peer-based coordinated service provision. The chapter consists of four parts. Firstly, key concepts regarding P2P overlay networks and super-peer overlays are presented in Section 2.1. Then, an overview of P2P applications and coordinated service provision leveraging P2P overlays is provided in Section 2.2. Section 2.3 presents related studies on constructing P2P overlays with different targets and for different applications. Section 2.4 discusses service discovery using the P2P overlays.

### 2.1 P2P overlay networks

In this section, key concepts related to P2P overlay networks are first introduced and then different categories of P2P overlays are presented, followed by a review of super-peer overlays.

#### 2.1.1 Key concepts of P2P

*Peer:* A peer is an autonomous entity that participates in a P2P overlay network, acting as both a client and a server at the same time. Peer is referred to as servant sometimes. The terms peer and node are used interchangeably in this thesis.

*P2P system:* P2P systems are 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 (Androutsellis-Theotokis & Spinellis 2004).

*Overlay:* Overlay network is a virtual network that is built above an existing physical network. Most overlay networks are constructed on the application layer and on top of the TCP/IP networking suite. The overlay hides the networking and communication details of the underlying networks (called underlay). Typically, the underlay is the Internet. A link between two overlay nodes may take several hops in the underlying network (Tarkoma 2010).

*Super-peer:* A super-peer is a peer that is usually more powerful and stable than a peer. A super-peer usually takes more responsibilities than a peer does. A

super-peer can submit and answer queries on behalf of their peers. More details about super-peers are presented in Section 2.1.3.

*Structured P2P network:* The topology of a structured P2P overlay is tightly controlled and content is not placed at random peers, but rather at specified locations that makes subsequent queries more efficient (Lua *et al.* 2005). A structured P2P network usually utilizes a DHT (distributed hash table) for object storage and retrieval (Stoica *et al.* 2003). Specifically, in DHT-based P2P overlay networks, the uniformity of the hash function is used to generate object Ids, which is used in resource storage and retrieval. Chord (Stoica *et al.* 2003) and Kademia (Maymounkov & Mazieres 2002), Pastry (Rowstron & Druschel 2001) are examples of this category. Structured overlays, especially DHT-based P2P overlay networks, are more suited to searching for rare items (a key with a few values) than popular items (a key with a large number of values) (Ganesan *et al.* 2003).

*Unstructured P2P network:* an unstructured P2P network organizes peers using a random graph in either a flat or a hierarchical way. Unstructured P2P networks usually use flooding, random walks or expanding Time-to-Live (TTL) search methods to query content stored by overlay peers (Lua *et al.* 2005). Gnutella v.0.4 (Ripeanu 2001) is a typical example of an unstructured P2P network. The key difference between a structured and an unstructured overlay is whether the organization of the P2P overlay topology and the positions of the content are controlled by some rules or not.

*Hierarchical P2P network:* a hierarchical P2P network, as the name indicates, consists of at least two levels of hierarchy. Peers are organized into different levels based on their capabilities, reliability, network bandwidth, etc. Peers take different roles, for example, a super-peer and a peer, as aforementioned. Usually, peers on the high level are named as super-peers and peers on the bottom level are named as peers. Different network topologies can be utilized at each level of the overlays. For example, the upper level can be a structured P2P overlay, while the lower level can be an unstructured P2P overlay. Hierarchical unstructured P2P systems include Kazaa (Liang *et al.* 2004) and Gnutella v.0.6 (Zoels *et al.* 2008).

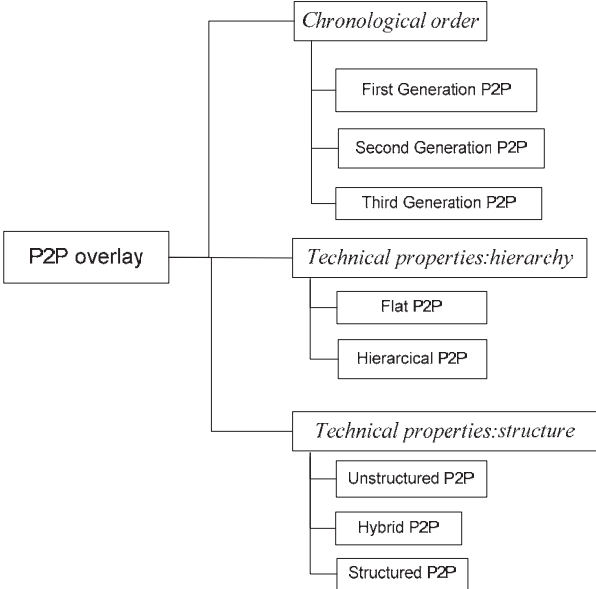
*Flat P2P network:* a flat P2P network, as its name indicates, organizes all the peers at only one level without using hierarchy. All the peers participating in the overlay network have equal responsibility. Gnutella v.0.4 (the earlier version of Gnutella) (Ripeanu 2001) and the basic form of Chord are examples of this category. The distinction between a hierarchical and a flat P2P overlay network depends on the number of levels utilized in the network topology.

*Hybrid P2P network*: hybrid P2P networks combine different concepts in the overlay network organization in order to improve search efficiency for both popular and rare items. The term hybrid is more general than the term hierarchical (Zoels *et al.* 2008). Both structured and unstructured concepts can be used in hybrid overlay organizations. A detailed case study on hybrid P2P networks can be found in Loo *et al.* (2005).

*Robustness*: robustness refers to the ability to adapt to changes in the network environment. Dynamicity in a P2P network is caused by both the churn (constant joining and leaving of nodes) and the failures of nodes and network entities. Churn in many cases affects the performance, leads to the data loss and causes extra network maintenance messaging (Leonard *et al.* 2007). Robustness is an important requirement in P2P overlay networks.

**2.1.2 Taxonomy of P2P overlays**

P2P overlay systems are typically classified according to their generations and technical properties. In the latter classification, P2P overlay systems are grouped based on two properties, hierarchy, and structure. Fig. 3 shows a detailed classification of P2P overlays.



**Fig. 3. A taxonomy of P2P overlay architectures.**

Generation-based classification refers to three generations of P2P systems (Zhang *et al.* 2006), as shown on the top layer in Fig. 3. Specifically, the first generation of P2P systems depended on centralized servers for indexing. The second generation of P2P systems feature the utilization of flooding for message propagation. The third generation of P2P systems are structured systems in which each node has a numeric identifier and the identifiers are arranged in a topology for message routing (Zhang *et al.* 2006). A similar three-generation-based classification method is given by Foster & Iamnitchi (2003). However, this generation-based classification has the drawback that the generation names, such as “the second generation”, are not self-explanatory.

Different from the aforementioned generation-based classification, the second classification approach employs technical properties of P2P systems as the criteria for categorization. On the one hand, as shown on the second layer of Fig. 3, some studies classify P2P overlay networks into hierarchical and flat ones according to the overlay hierarchy. On the other hand, as shown on the third layer of Fig. 3, many studies classify P2P overlay networks into unstructured, structured, and hybrid ones based on the overlay topology, and the construction and organization of network connections.

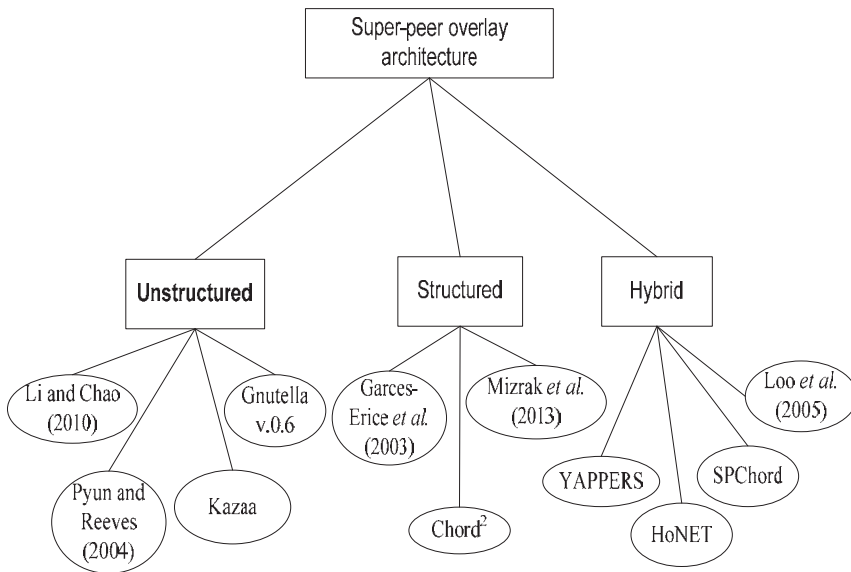
Hierarchical P2P systems utilize multiple levels to distribute the overlay nodes and each level can have a different topology. Consequently, hierarchical P2P systems can be further classified into three categories: structured hierarchical P2P systems, unstructured hierarchical P2P systems, and hybrid hierarchical P2P systems, according to the adopted overlay topologies at each level. Other technical properties for P2P system classifications include the type of indexing used for nodes or resources in the network, the type of searching method implied by the selected indexing method, and the content delivery method used in the network. The details about these properties are not presented in this thesis because these properties are not the main criteria.

### **2.1.3 Super-peer overlay**

A super-peer overlay is a two-layer hierarchical overlay, in which the lower layer consists of client peers and the higher layer consists of super-peers. A client peer only maintains a connection with its super-peer, while a super-peer maintains connections to its own client peers, as well as to other super-peers in the overlay. Super-peers are usually more powerful and stable than client-peers and thus take more responsibilities than client peers. For example, super-peers may act as



centralized servers to their client peers and handle routing, storing and forwarding information on behalf of client peers, while client peers just initiate service requests and receive associated responses (Yang & Garcia-Molina 2003). Super-peers are usually selected based on metrics, such as CPU capacity, network connectivity, reliability, as well as other metrics, such as security, privacy and trust, according to application requirements (Kleis *et al.* 2005). In brief, a super-peer overlay exploits the heterogeneity of nodes and combines the efficiency of a centralized search with the advantage of autonomy, load balancing, and robustness provided by the distributed search in P2P overlay. Consequently, the use of super-peers is proposed to improve the performance of both structured and unstructured P2P networks. Super-peer overlays can be categorized into structured, unstructured, and hybrid ones, as shown in Fig. 4.



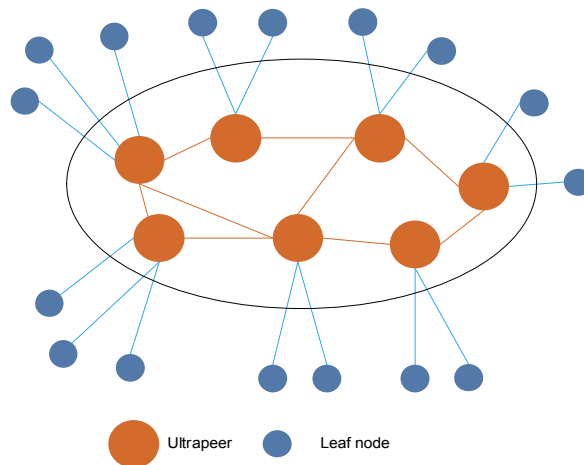
**Fig. 4. Categorization of Super-peer overlay architectures.**

### *Unstructured super-peer architecture*

Unstructured super-peer systems employ an unstructured topology at each level of the overlays. They are proposed to solve the shortcomings of the flooding-based one-layer (i.e. flat) P2P networks, such as Gnutella v.0.4 (Ripeanu 2001).

Specifically, a large population of Gnutella v.0.4 applications (deployed around the world) generated a huge number of search traffic across different administrative domains (Ou *et al.* 2010), and degraded the search recall ratio significantly (Zoels *et al.* 2008). Herein, the search recall ratio refers to the ratio of the number of search results to the total number of available copies of the searched object. To solve these problems, two-layer hierarchical P2P systems (i.e. super-peer-based P2P system), such as Gnutella v.0.6 (Xie *et al.* 2008) and Kazaa (Liang *et al.* 2004) were proposed.

The structure of Gnutella v.0.6 is shown in Fig. 5. Gnutella v.0.6 classifies nodes into leaf nodes and ultrapeer. A leaf node connects to its own ultrapeer, and an ultrapeer connects to both its own leaf nodes and other ultrapeers. One ultrapeer can connect with multiple leaf nodes, depending on its capability. Leaf nodes just initiate service requests, receive associated responses, and respond to the requests that they can exactly answer. Besides having the normal functionality of leaf nodes, ultrapeers handle routing, storing and forwarding information on behalf of leaf nodes (Yang & Garcia-Molina 2003). Flooding-based mechanism is usually utilized for forwarding lookup requests among ultrapeers. Kazaa utilizes a similar overlay topology as Gnutella v.0.6 (Xie *et al.* 2008). However, Kazaa dynamically elects super-peers that are more powerful to form an unstructured overlay network, and a regular node can connect to one or more super-peers to query the network resources (Liang *et al.* 2004).



**Fig. 5. Architecture of Gnutella v.0.6.**

Along with improving the search recall ratio, scalability is also one important performance metric in the design of unstructured super-peer overlays. The degree of super-peers has an important impact on the scalability of an unstructured super-peer overlay. Specifically, when the degree of super-peers becomes significantly large, super-peers can sustain heavy workload and consequently suffer from the restricted scalability. Herein, the degree of super-peer refers to the number of connections to neighbors that super-peers need to maintain. Many studies were made to tackle the problem of scalability. For example, Pyun & Reeves (2004) presented a balanced, scalable,  $\Theta(\log N / \log \log N)$ -diameter unstructured super-peer topology (SUPS) using random graph theory (where  $N$  is the number of nodes in the overlay topology). Li & Chao (2010) presented a Perfect-Difference-Graph-based (PDG) unstructured super-peer overlay with a smaller network diameter for reducing the number of lookup messages and the average delay, and achieved a good scalability. Teng *et al.* (2011) proposed utilizing a self-similar square network graph (SSNG) to construct a more scalable super-peer overlay topology than the PDG-based overlay.

There are also many research efforts addressing other issues of unstructured super-peer overlays. Leveraging unstructured super-peer overlays for service provision is discussed in Section 2.2. Studies on constructing unstructured super-peer overlays are presented in Section 2.3. Search performance (Lv *et al.* 2002), security and trust (Dimitriou *et al.* 2008), the topology mismatch (Hsiao *et al.* 2009), and reputation (Chhabra *et al.* 2004) have been studied as well. Details about these issues are not presented because these are out of the scope of this thesis.

### *Structured super-peer architecture*

Structured super-peer systems employ only a structured topology at both levels of the overlays. Structured hierarchical P2P systems combine the merits of hierarchical P2P overlays and structured P2P overlays. Specifically, introducing a hierarchy provides the benefits in scalability and fuzzy search. Meanwhile, utilizing a DHT-based structured overlay has the advantages in precise search and search efficiency. Structured super-peer systems have been studied a lot. Representative examples of this category include Garces-Erice *et al.* (2003), Mizrak *et al.* (2003), and Joung & Wang (2007).

Garces-Erice *et al.* (2003) proposed a two-tier hierarchical lookup design that groups peers based on network latency and uses a super-peer layer for

communication between groups. Two different DHT algorithms are used at each level of the overlays. Lookup messages are first routed to a destination group, and then sent to a destination peer using an intra-group overlay. Their study shows that by using the most reliable peers at the top level, the hierarchical design significantly reduces the expected number of lookup hops. Mizrak *et al.* (2003) presented a structured super-peer overlay design in order to achieve constant-time  $O(\log M)$  lookup (where  $M$  is the number of super-peers), in which super-peers maintain more information (the addresses of managed peers and mapping information of other super-peers). Joung & Wang (2007) presented a two-layer Chord, Chord<sup>2</sup>, to reduce the overlay maintenance cost. They imposed a conduct layer as a super-peer layer (a Chord ring) over the existing regular Chord ring. Powerful and stable peers were promoted to be super-peers. In brief, some studies have constructed a structured super-peer overlay by leveraging the heterogeneity of peers (e.g. processing capability, stableness, and proximity) in the overlay construction, but other studies have focused on other factors (e.g. node degree) without considering the heterogeneity in reducing lookup time.

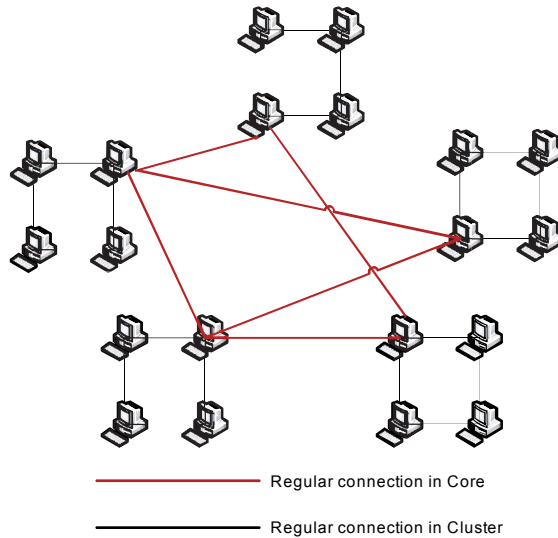
### *Hybrid super-peer architecture*

Hybrid super-peer systems utilize both unstructured and structured overlay topologies on the tiers in order to make full use of the benefits of both. Typical examples of hybrid super-peer architecture include YAPPERS (Ganesan *et al.* 2003), Tian *et al.* (2005), Loo *et al.* (2005), and Yang & Yang (2010).

Ganesan *et al.* (2003) introduced YAPPERS (Yet Another Peer-to-PEER System) that combines both structured and unstructured P2P networks to provide a scalable lookup service over an arbitrary topology. For each node in the search network, YAPPERS builds a small DHT overlay consisting of nearby nodes and then provides an intelligent search mechanism that can traverse all the small DHT overlays. YAPPERS reduces the number of nodes contacted for a lookup, and allows rapid searching of nearby nodes. YAPPERS is efficient in partial lookup that only returns partial values of data without flooding the request to all the peers in the same color (Yang & Yang 2010).

Tian *et al.* (2005) proposed a two-level hybrid overlay networks (HONet) to integrate the scalability of structured overlays with the connection flexibility of unstructured overlays. Fig. 6 demonstrates the structure of HONet. At the lower level, nodes self-organize into structured clusters based on network locality. At the upper layer, the root nodes in each cluster build connections with each other

adaptively using a random-walk method, which aims to reduce network delay and bandwidth consumption. The number of random connections built by a super-peer is decided by its service capacity.



**Fig. 6. Architecture of HONet (Tian et al. 2005).**

Loo *et al.* (2005) proposed a two-level hybrid P2P search infrastructure to locate rare object items with structured DHT techniques and highly replicated contents with unstructured flooding. Their work was motivated by their finding that Gnutella v.0.6 is effective for locating highly replicated items, but less suited for locating rare items. Loo *et al.* (2005) organized all ultrapeers into a DHT overlay and these ultrapeers were responsible for identifying and publishing rare items into the DHT overlay on behalf of their leaf nodes. A resource lookup query was first performed using conventional flooding techniques, and then the query was reissued as a DHT query if no enough number of results was returned within a predefined period.

Yang & Yang (2010) also proposed a two-level hybrid P2P system. The lower level of the systems consists of multiple unstructured P2P overlays. The upper level of the system is a structured ring-based core P2P network that forms the backbone of the hybrid system. Each unstructured P2P overlay on the lower level is attached to one super-peer at the upper level. According to Yang & Yang (2010), the main difference between these two studies, Loo *et al.* (2005) and Yang & Yang (2010), is that the former utilized the structured overlay as a supplement for

the unsuccessful flooding lookup, while the latter utilized the structured overlay to connect all the lower level unstructured networks and transmit queries among them. From the perspective of this thesis, there is no significant difference between these two solutions. Artigas *et al.* (2007) analyzed different types of two-layer DHT overlays, including hierarchical and hybrid types. Their analysis found that when the top level utilizes a structured DHT overlay, at the lower level, a simple star-like connection among peers was superior to a DHT or a fully meshed connection.

In this thesis, the super-peer overlay refers to an unstructured two-level hierarchical overlay. In other words, in the super-peer overlay, super-peers connect with each other randomly. The rationale is that firstly, two-layer is a pragmatic choice for the number of layers in a hierarchical P2P overlay (Xu *et al.* 2003). Secondly, an unstructured super-peer overlay is more robust and efficient in handling node joins and leaves, while a structured super-peer overlay suffers from high maintaining cost and the churn from node joins and leaves. In addition, compared with flat unstructured P2P systems and flat DHT-based P2P systems, super-peer-based P2P systems have the advantage of (1) making use of the heterogeneity of peers and distributing network traffic load in a more reasonable way; (2) making use of network proximity and local traffic on one layer not affecting the other layers; and (3) decreasing lookup latency.

#### **2.1.4 P2P networks versus mobile ad hoc networks**

There are many commonalities between P2P overlay networks and mobile ad hoc networks (MANET). According to Oliveira *et al.* (2005), the commonalities between P2P overlay networks and MANET can be summarized as follows: (1) Both network architectures are decentralised in nature. (2) The connectivity of nodes or peers in both MANET and P2P networks is transient. In MANET, this is due to node mobility. In P2P networks, this is primarily due to the lack of permanent Internet connectivity or a static IP address. (3) The resources hosted by the nodes or peers in P2P networks and MANET are heterogeneous because different kinds of devices can participate in the networks.

The main differences between P2P networks and MANET can be summarized as follows: (1) The network sizes of MANET are generally limited to a few hundred nodes, while P2P networks usually run over the Internet, and therefore, potentially involve a larger number of peers up to millions of peers. (2) MANET deals with application independent network issues, the network layer in

particular. Conversely, P2P networks generally deals with application-oriented network issues. (3) The connection established in P2P networks is a logical infrastructure for providing a service whilst the connection in MANET is a physical infrastructure for providing connectivity. (4) The connection between two nodes in P2P networks are mainly fixed medium and direct (based on IP address). In contrast, the connection between two nodes in MANET is wireless and indirect. A more detailed comparison can be found in Borg (2003), Hu *et al.* (2003) and Oliveira *et al.* (2005).

## **2.2 Super-peer-overlay-based service provision**

P2P overlays have been widely applied in different kinds of distributed applications and services due to its scalability, self-organization and adaptability (Androutsellis-Theotokis & Spinellis 2004). Super-peer overlays address the heterogeneity of nodes (e.g. network bandwidth, online time, and processing capability) to improve the performance of service provision. Thus, the super-peer overlay has been widely utilized in file sharing, e.g. Gnutella (Gnutella 2002), the distributed communication (e.g. Skype) and enterprise workflow management applications (Gu & Nahrstedt 2006) (Goel *et al.* 2007) (Ranjan *et al.* 2008) (Votis *et al.* 2008).

### **2.2.1 Overview of P2P applications**

According to Androutsellis-Theotokis & Spinellis (2004), P2P systems can be classified into the following categories based on their application domains:

*Communication and collaboration*: this category mainly focuses on providing an infrastructure for facilitating direct, real-time communication and collaboration between participating peers. Typical examples include instant messaging applications, Jabber (Saint-Andre 2007), Skype, PPLive and QQ (Cheng *et al.* 2007).

*Distributed computation*: this category mainly targets at utilization of distributed computing resources, e.g. CPU cycles and memory capacity. Representative examples include SETI@home (Search for Extra-Terrestrial Intelligence) (Korpela *et al.* 2001) and Genome@home16. In this kind of systems, a central coordination is required for breaking up tasks, distributing tasks, and collecting results.

*Internet service support*: this category includes the applications that are based on the underlying P2P infrastructure in order to support a variety of Internet services, for example, application-level multicast system (Castro *et al.* 2002) and the Internet indirection infrastructure (Stoica *et al.* 2004).

*Database systems*: this category primarily concentrates on providing distributed database service by utilizing P2P concepts and infrastructures. Typical examples include Wolff & Schuster (2004), Dynamo (a highly available key-value storage system) (DeCandia *et al.* 2007) and Arai *et al.* (2007).

*Content distribution*: this category includes systems and infrastructures designed for sharing and distributing of digital content and other data. Most of the current P2P systems fall into this category. This category can be subdivided into two categories. The first sub-category consists of P2P file exchange systems, and P2P content publishing and storage systems. The second sub-category is P2P infrastructures that consist of the following services: routing, location, anonymity, and reputation management. Some representative examples include Chord (Stoica *et al.* 2003), eDonkey (Tutschku 2004) and Kazaa (Kazaa 2009).

This thesis mainly focuses on the first category of P2P-based collaboration systems and the fourth category of P2P-based content distribution, including P2P overlay construction and P2P-overlay-based resource storage and location.

### **2.2.2 Coordinated service provision leveraging super-peer overlay**

As already mentioned, this thesis focuses on collaboration among peers. A cooperative P2P system is a system that consists of responsible peer users who honestly share resources with each other for the common good of everyone, such as enterprise P2P systems (Castro *et al.* 2003).

One challenge in cooperative P2P service provision is to enable the service providers to automatically monitor their resources and tune themselves to meet QoS requirements (e.g. short service response time) by deploying new instances of services or removing existing ones under dynamic service demands (Cuenca-Acuna & Nguyen 2004) (Foster 2005) (Pacifci *et al.* 2005) (Papazoglou *et al.* 2007) (Chen *et al.* 2008) (Brazier *et al.* 2009). This challenge is common in application areas that are computational intensive with dynamic fluctuations in service demands, such as in the fields of cosmology, bioinformatics analysis (Chakravarti *et al.* 2005), grid computing (Goel *et al.* 2007) (Ranjan *et al.* 2008) and cloud computing (Parashar *et al.* 2006) (Ranjan & Rajkumar 2010).



Connecting and coordinating a large number of heterogeneous service entities are challenging tasks (Benatallah *et al.* 2002).

Some specifications have been proposed for enabling cooperative services provision, such as WS-Coordination (Burdett & Kavantzias 2004) and WS-Choreography (Cabrera *et al.* 2005). WS-Coordination describes a generic framework for application services to create a shared context to propagate an activity to other services and to register for coordination protocols. WS-Choreography aims to create interoperable collaborations among different service parties by defining their common and complementary behaviors and tracking public message exchanges. However, these specifications focus on defining application logics to implement a complex functionality or task, but do not involve a large number of service entities and do not taken into account the dynamicity of service entities and service requests for performance improvement. Once the task is completed, the collaboration among service parties dissolves and service parties find new transactions to join.

Applying P2P technologies for coordinating or integrating different peers distributed across multiple enterprises has been investigated in some studies. These studies focused on the design of data conversion rules between service peers, shared data repositories, and provider (peer) selection policies. Gu & Nahrstedt (2006) proposed and compared different algorithms for cooperative P2P streaming by composing given applications on the initial stage (i.e. global-state-based centralized (GC) versus local-state-based distributed algorithms). They utilized mathematical methods to map a function graph into the best service graph. They made some strong assumptions for cooperative service provision. Firstly, a service discovery system for finding candidate service instances matching a required service function is assumed to exist. Secondly, each peer is assumed to be able to monitor available network bandwidth on its adjacent overlay links using measurement tools. This thesis investigates the mechanism of finding proper services and monitoring the peers' workload using a super-peer overlay instead of making an assumption. Thirdly, service instances are assumed to be described using high-level specification languages based on a common ontology. Fourthly, an available translator is assumed to map the application-level QoS specifications into low-level resource requirements (e.g. CPU, memory, and network bandwidth). All these assumptions need to be tackled in a real-world system implementation. This thesis focuses on coordination of real-world service instances instead of abstracted high-level language specified services. The coordinate mechanism enables services to adapt dynamically to meet the

application requirements instead of just concentrating on the initial stage of coordinated service provision as in Gu & Nahrstedt (2006).

Goel *et al.* (2007) proposed a P2P-overlay-based service-centric architecture for scheduling service entities to achieve complex business and engineering transactions in a collaborative way. Each service provider is defined as an independent self-sustaining entity performing a specific network activity in a choreographed sequence. All services have standardized open interfaces. Once the transaction is completed, the federation among service providers dissolves and the service providers seek other transactions to join. Their service centric architecture seeks to find proper service providers to finish a complex task. It should be noted that this thesis concentrates on coordinating service groups for dynamic sharing of resources and handling dynamic service demands instead of finishing transaction-based tasks. The coordination relationship between service providers does not dissolve.

Similar to the work of Goel *et al.* (2007), Ranjan *et al.* (2008) utilized a P2P-based coordination space (Zhen & Parashar 2005) to coordinate the application schedules of distributed workflow brokers in the way of publish and subscribe. Workflow brokers posted resource demands by injecting a resource claim object into the P2P-based coordination space. Resource providers updated resource information by injecting a resource ticket object into the P2P-based coordination space. The P2P-based coordination space sent notification messages to resource claimers once a resource ticket matched with one or more resource claims. Extending the work of Ranjan *et al.* (2008), Rahman *et al.* (2010) proposed to compute a reputation value for a resource provider to decide which resource object should be invoked when a list of resource claims were matched with the submitted resource ticket object. Their work focuses on designing a third-party shared data repository in order to coordinate service providers. It should be noted that this thesis builds the coordination by utilizing the structure of super-peer overlay from the service providers themselves, not a third-party repository. The coordination mechanism is derived from a market model.

P2P overlays are also utilized for cloud service provision to facilitate creation of wide-area on-demand leasing of computing or storage services for hosting application services that experience highly variable workloads and require high availability and performance. Ranjan *et al.* (2010) presented a layered P2P-based cloud service provision architecture that connects distributed public/private clouds with a structured overlay of FreePastry (Druschel *et al.* 2001). A distributed indexing technique, a variant of peer-to-peer MX-CIF Quad tree

(Tanin *et al.* 2007), was implemented to support multi-dimensional queries (cloud service type, host utilization, instance OS type, host cloud location, host processor speed). Their work concentrated on one of the building blocks of coordinated service provision: the service discovery supporting multi-dimensional query. This thesis focuses on the coordination of service entities in terms of sharing resources and supports only simple service queries instead of multi-dimensional queries.

The efficiency of coordinating service peers among service groups has been studied as well. Cuenca-Acuna & Nguyen (2004) proposed a distributed resource management framework for self-managing federated service provision in P2P environments. Each service node relies on third-party service management agents and node monitoring agents to monitor the state of service nodes and to adjust the configurations of these nodes, e.g. the number of services and their positions. In contrast, Paper I employs a super-peer overlay to organize service nodes and selects super-peers from the service providers instead of third-party agents.

Chen *et al.* (2008) proposed a service provision framework to coordinate service groups to adapt to dynamic service demands. They utilized a DHT-based P2P overlay, in which each service peer has the same responsibilities. Service groups consist of service peers. They proposed a market model, in which each service peer takes both the roles of a recruiter (recruit or dismiss peers) and an employee (process service requests). Paper I proposes utilizing a super-peer-based P2P overlay in the coordination of service provision so that super-peers take more responsibilities than service peers. Service groups consist of both super-peers and service peers. Paper I proposes an S-labor-market model, in which super-peers take the role of a recruiter, due to their high capacity, and service peers take the role of an employee. Consequently, the coordination between service groups is different from Chen *et al.* (2008). Adopting the super-peer overlay shortens the time needed for coordination and reduces communication overhead.

There are other issues that should be taken into account in cooperative P2P service provision, for example, trust, and security management. Security is a big issue for P2P based service provision, which is out of the scope of this thesis. More studies can be found in Yan & Zhang (2006), Dimitriou *et al.* (2008), Rahman *et al.* (2010) and Meland *et al.* (2011).

### **2.3 Super-peer overlay construction**

Super-peer overlays enable applications to run more efficiently by exploring the heterogeneity of nodes in the overlay network. In typical examples, Skype voice

streaming (Baset & Schulzrinne 2006), and live video streaming applications (PPLive 2009), the performance is improved by assigning nodes with high network bandwidth, long on-line time, or high processing capability as super-peers. Thus, the criteria for selecting super-peers is an important issue in building a super-peer overlay. In addition, in a dynamic network environment, it is common that peers may join or leave a super-peer overlay randomly (e.g. a failure of a super-peer) (Stutzbach & Rejaie 2005) (Voulgaris *et al.* 2005) (Baset & Schulzrinne 2006). The construction of a super-peer overlay should handle the failure of peers. In brief, it is an important task to design an efficient super-peer selection method that can quickly build a robust super-peer overlay. This section first reviews related studies on P2P overlay construction and then investigates super-peer selection methods.

### **2.3.1 Overview of overlay construction**

Many studies have built various types of P2P overlays in order to satisfy different purposes or to achieve different performance targets. According to the structure of P2P overlays, overlay constructions can be generally categorized into (1) structured P2P overlay construction, and (2) unstructured P2P overlay construction. Structured P2P overlay constructions (e.g. Chord, Pastry (Rowstron & Druschel 2001) and Tapestry (Zhao *et al.* 2004)) usually use DHT-based algorithms to build connections between peers and guarantee locating a value associated with a key within a bounded number of hops. A bootstrap node is usually needed to provide node information during the overlay construction. However, structured overlays utilize complex algorithms and sustain heavy maintenance cost for node joins and leaves. In addition, structured overlays have poor performance in keyword-based search. In unstructured P2P overlays, connections between peers are ad hoc and data placement is flexible and independent of the overlay topology. Unstructured overlays are more commonly used in applications than structured overlays and many more studies have been done on the constructions of versatile unstructured overlays than structured overlays (Lua *et al.* 2005).

Neither unstructured nor structured P2P overlays can simultaneously offer better performance in both response time and resilience against network dynamics without having to reconsider some of fundamental design tradeoffs (Pyun & Reeves 2004). Low-diameter P2P overlay networks were proposed to address both the performance and resilience, in which most nodes are within a small number of

hops from each other. A low-diameter overlay network can be constructed from power law graphs (also known as scale free graphs). For example, Wouhaybi & Campbell (2004) proposed Phenix to build a low-diameter and resilient P2P overlay utilizing power-law graphs and the degree distribution of nodes follows a power-law distribution. Phenix hid the identities of major hubs and was fully distributed without requiring any central servers, which makes it resilient to malicious attacks. A low-diameter overlay can also be constructed through random augmentation (i.e. by adding random additional links between nodes) (Erdős & Renyi 1959) instead of power-law graphs. For example, Pyun & Reeves (2004) proposed a scalable unstructured P2P system (SUPS) for construction of a low-diameter topology with balanced link distribution for super-peers. In SUPS, interconnections between super-peers were selected to approximate a random graph and super-peers were organized in a fully distributed way.

Small-world graphs share many features of random graphs and are also utilized for constructing low-diameter overlays, e.g. Hui *et al.* (2006) and Liu *et al.* (2007). Liu *et al.* (2007) presented a small-world characterized architecture for P2P networks (SWAN) for resource discovery in multi-group P2P systems. Each peer was connected to some neighbouring nodes in the same peer group and peer groups were connected by a small number of inter-group links for keeping long-range links to distant nodes in the network. A semi-structured P2P algorithm was proposed to create and find long-range shortcuts between different peer groups. Every peer can easily find out which nodes have external connections to a specific peer group in order to communicate with remote groups. NEWSCAST (Jelasy *et al.* 2003) proposed a totally different approach of randomly exchanging neighborhood information among peers using a gossip-based protocol to construct an overlay exhibiting small-world characteristics. BLATANT-R (Brocco *et al.* 2009) proposed to dynamically reorganize connections between peers without introducing hierarchies for constructing a diameter-bounded network overlay with a fair distribution of links. The reorganization process utilized ant-inspired distributed swarm intelligence to monitor the network and collect information, which ensures adaptability and robustness.

In both structure and unstructured P2P overlays, nearby nodes in the overlay networks may actually be far away in the underlying networks and each routing hop may take a message to a node with a random location in the Internet, which results in a long message delay and unnecessary wide-area network traffic (Zhang *et al.* 2004). Locality-aware (or topology-aware) overlays have been proposed to

decrease the communication cost and to reduce message delay between nodes. Locality-aware (or topology-aware) overlay constructions take into account the locality of nodes and choose the nodes that are closely located in the underlying physical network as the neighboring hosts in the overlay. Many studies have explored locality in both structured and unstructured overlay constructions to reduce message delay and communication overhead, e.g. Ratnasamy *et al.* (2002), Harvey *et al.* (2003), Zhang *et al.* (2004), Artigas *et al.* (2005) and Jesi *et al.* (2006).

Evaluating the distances between nodes (i.e. locality) is a key problem. The locality can be measured with different criteria, such as physical location, the number of hops between hosts, and link latency. Landmark approach (both static and dynamic landmark approaches) is a widely used method for measuring the distances between peers, for example, Ratnasamy *et al.* (2002) and Zhang *et al.* (2004). A virtual coordinate mechanism has also been utilized in a locality-aware P2P network to compute distance between peers, e.g. Yu *et al.* (2005). This mechanism usually introduces extra overhead into overlay construction. Another approach is dynamical computation of distances among nodes by sending messages, such as Time-To-Live (TTL), and Round Trip Times (RTTs). A bootstrapping mechanism has also been used to compute the distance between peers in order to find a place for a new peer to join the network (Miers *et al.* 2010). In this mechanism, tracker nodes maintain a list of all the active peers in the network and compute the distances between peers when needed, which suffers from a single point of failure. More information about locality-aware overlay construction can be found in Miers *et al.* (2010).

Besides investigation on reducing message delay, another issue of the poor performance for approximate, range, or text queries has been studied in the overlay construction as well. Semantic-based overlay construction has addressed this issue. Semantic-based overlays organize peers in such a way that semantically or socially similar peers become members of the same semantic overlay network (SON), which can improve text or range query performance (Nejdl *et al.* 2003) (Löser *et al.* 2004) (Crespo & Garcia-Molina 2005) (Doulkeridis *et al.* 2007). Desired content in P2P applications is often available in the proximity of interested users because of content language and geographical regions of interest (Rasti *et al.* 2006). In SONs, resource queries are forwarded to only those peers that contain the documents within specific topics. Some studies focused on designing distributed algorithms for constructing semantic overlays so that peers having common preferences are clustered together (Nejdl *et al.* 2003)

(Löser *et al.* 2004) (Tempich *et al.* 2004) (Crespo & Garcia-Molina 2005) (Doulkeridis *et al.* 2007). Some studies focused on constructing semantic overlays that exhibit small-world graph properties (Li *et al.* 2004) (Jin *et al.* 2006) (Hsiao & Su 2012). More information about semantic-based P2P overlays can be found in Doulkeridis *et al.* (2010) and Hsiao & Su (2012). However, most semantic-based overlay construction studies did not address the scalability of the P2P overlay with respect to the heterogeneity of peers in terms of their network resources and participation times.

As already mentioned, peers are highly diverse in terms of their processing capabilities, network resources (such as access bandwidth, CPU, disk I/O) and participation times (or online time) (Sarioiu *et al.* 2002) (Wang *et al.* 2008). Capability-based (or heterogeneity-aware) overlay constructions have been investigated to address the heterogeneity of peers (Sarioiu *et al.* 2002) (Lanham & Shenker 2003) (Montresor 2004) (Li *et al.* 2005) (Baset & Schulzrinne 2006) (Kwong & Tsang 2008) (Wang *et al.* 2008) (PPLive 2009) (Kurve *et al.* 2013). The load imposed on each node should be proportional to the capacity of the node in order to explore of the heterogeneity of nodes and to achieve load balancing. The load consists of two parts: the number of messages (or requests) handled by a node and the node degree (i.e. the number of links connected to the node). The load generated from the second part comes from (1) the cost of maintaining links with other nodes (e.g. periodic keep-alive messages to neighbor nodes), and (2) application specific load that is also related or proportional to node degree (Vishnumurthy & Francis 2006). For example, in Gnutella, the load of a node is related to the node degree (Lanham & Shenker 2003). Thus, it is important to control the node degree in heterogeneous overlay construction. Controlling the frequency of visiting the nodes is also important because nodes experience load each time when other nodes visit them. For example, Sarioiu *et al.* (2002) constructed a heterogeneity-aware multitier P2P overlay to better balance the load of peers with heterogeneous capacities and to prevent performance degradation by low-capability nodes. Many studies have focused on improving application performance (e.g. short response time and low latency) by exploring the heterogeneity of peers in the P2P overlay, e.g. P2P Live streaming (Wang *et al.* 2008) (Liu *et al.* 2009) (Couto da Silva *et al.* 2011) and selecting the most reliable nodes as the super-peers (capacity-based super-peer selection), e.g. Montresor (2004), Li *et al.* (2005) and Kwong & Tsang (2008). This thesis focuses on construction of a super-peer overlay by exploring the heterogeneity of peers. Further analysis is provided in Section 2.3.2.

It should be noted that there are also other studies combining some of these different technologies (such as locality, semantics, capability-based and low-diameter) together for P2P overlay constructions, e.g. Shen & Xu (2007). Some studies on overlay construction focus on security and trust (Dimitriou *et al.* 2008), reputation (Chhabra *et al.* 2004) and other issues. Details about these studies are not presented because these are out of the scope of this thesis.

### **2.3.2 Super-peer overlay construction**

A super-peer overlay is a special hierarchical and heterogeneous-aware P2P overlay with two layers. Super-peers are assumed to have high workload than client peers. In the construction of super-peer-based overlays, the selection of super-peers and building the association between super-peers and client peers are key issues for study. Many criteria can be used for selecting super-peers, including locality, semantic, capacity, network connectivity, reliability, security, privacy and trust as well as other metrics, according to application requirements (Kleis *et al.* 2005).

Some studies focused on constructing locality-aware super-peer overlays. One approach first clusters peers into groups according to their localities, and then selects peers with high capacity or short distance to other peers inside a group as super-peers. For example, HONet (Hybrid Overlay Networks), proposed by Tian *et al.* (2005), first clustered nodes based on their network coordinates and then selected the node with the highest capability within a cluster as a super-peer. Super-peers were connected with each other using a random walk and a node chose to join the cluster with a short distance. Another approach first selects peers with high capacity as super-peers and then builds the connections between super-peers and client peers by connecting client peers to the closet super-peers based on distance or RTT, e.g. Kleis *et al.* (2005), Jesi *et al.* (2006) and Merz *et al.* (2008). Specifically, Kleis *et al.* (2005) first selected stable peers with enough resources to serve other peers as super-peers and these super-peers acted as landmarks. After that, they assigned nodes into clusters that had the nearest centroid (mean) distance based on RTTs using a K-means clustering algorithm. Client peers joined the cluster with the closest super-peer. Their work relied on a coordinate system to compute and control the super-peer topology. SG-2, proposed by Jesi *et al.* (2006), first employed a biology-inspired task allocation mechanism to promote the best nodes to super-peers and then associated super-peers with clients whose RTTs were within a specified threshold. SG-2 utilized a

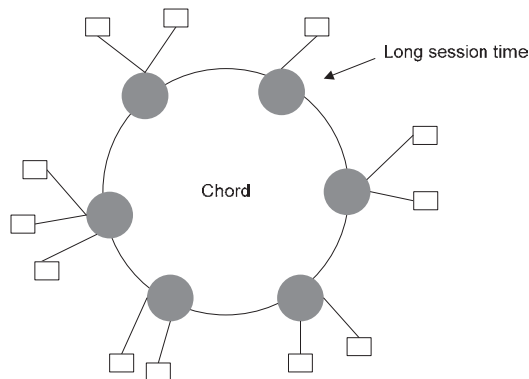


gossip-based protocol to spread messages to nearby nodes. SG-2 aims to cover a space with a small number of super-peers and to minimize the latency between clients and super-peers. Super-peers also acted as landmarks for measuring the distance between peers to build a locality-aware super-peer overlay. Merz *et al.* (2008) presented an SPSA algorithm to construct a locality-aware super-peer overlay by using a network coordinate system (a Vivaldi system). SPSA first selected the peers that were able to deliver super-peer service at lower cost as super-peers. A super-peer dynamically relinquished its role if the number of its attached client peers was too low or too high (two thresholds are used). A client peer was always connected to the nearest super-peer and changed its super-peer if it found another closer super-peer. SPSA strived to minimize the number of super-peers to be  $O(\sqrt{n})$  ( $n$  is the total number of peers in the overlay). SPSA needs a smaller number of super-peers than SG-2 to cover a particular space. A coordinate system is needed in the above studies to assign network coordinates to nodes in order to connect client peers to a super-peer, which generates extra workload on super-peers. It should be noted that in Papers III and IV, client peers are connected to super-peers through gossip-based information exchange without a coordinate system.

Constructing semantic-based super-peer overlays has also been investigated so that peers having common preferences are clustered together (Nejdl *et al.* 2003) (Löser *et al.* 2004) (Qiao *et al.* 2007) (Tan *et al.* 2012). Particularly, Löser *et al.* (2004) proposed hashing the whole taxonomy hierarchy to be super-peers based on DHT algorithm, and connected these super-peers as a Chord protocol. Their approach had the drawback of loose semantics of data within a super-peer and increased communication overhead between super-peers. Qiao *et al.* (2007) employed a taxonomy hierarchy to describe the contents of objects, and presented a taxonomy-based approach for constructing a semantics-based super-peer overlay. Their approach dynamically clustered peers in taxonomy-based semantic space, and organized clusters into semantic routing overlays. Their work achieved a competitive trade-off between search latencies and overheads, and maintained a load balancing among super-peers.

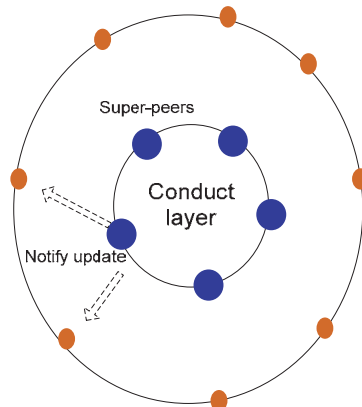
Some studies have focused on constructing capacity-based super-peer overlays by exploring the heterogeneous capacities of peers to improve application performance. For example, Li *et al.* (2005) presented a DHT-based capacity-aware super-peer overlay (SPChord) in order to improve stability and data availability. The topology of SPChord is shown in Fig. 7. The nodes with longer session time (on-line time) were selected as super-peers and other dynamic

unstable nodes joined the super-peers and acted as a member in super-peer clusters. This method improves the stability of the super-peer overlay and reduces maintenance overhead because the frequency of nodes joins/leaves the overlay is reduced. However, their study only takes into account the session time as a criterion for selecting super-peers. Thus, the heterogeneity of processing capabilities is not fully investigated. This thesis investigates also processing capability, and network bandwidth, in addition to online time as criteria for selecting super-peers. It should be noted that this thesis focuses on constructing a super-peer overlay in a fixed network environment. Mobile devices are becoming more powerful and can connect with each other to build mobile P2P overlays for mobile service provision. When constructing such mobile P2P overlays for mobile service provision, important factors related to mobile devices (e.g. power of a device, multiple available connections) should be taken into account in the super-peer selection.



**Fig. 7. Topology of the SPChord.**

Joung & Wang (2007) proposed a structured capacity-aware super-peer overlay Chord<sup>2</sup> (i.e. a two-layer Chord) shown in Fig. 8. Powerful and stable peers were promoted to super-peers by measuring their bandwidth, storage space, and CPU speed. A super-peer layer (a conduct layer) was imposed as a Chord ring overlay over an existing regular Chord ring. A binary tree was constructed over the regular Chord ring to store the identifiers of found super peers. The binary tree was also utilized by a new super-peer in finding existing super-peers. The conduct super-peer layer maintained the overlay and notified affected nodes to update their fingers when it received information of node joins or leaves at the bottom layer. The proposed super-peer layer reduced cost of overlay maintenance.



**Fig. 8. Chord<sup>2</sup> (Joung & Wang 2007).**

Li & Chao (2010) proposed to employ the Perfect Difference Graph (PDG) to construct a small-network-diameter super-peer overlay, which generated a small number of lookup messages and short average delay. A bootstrap server was designed to maintain a super-peer table and decided whether a peer joining the network should become a super-peer or not by comparing its bandwidth with other super-peers stored in the super-peer table. The bootstrap server also sent the corresponding forward and backward connections to the peer (or super-peer) to enable this peer (or super-peer) to build connections. A PDG-based forwarding algorithm flooded lookup messages from an originator super-peer to other super-peers to guarantee that each super-peer received only one message. Teng *et al.* (2011) proposed to utilize the self-similar square network graph (SSNG) to construct a more scalable and efficient super-peer overlay topology than the PDG-based overlay proposed by Li & Chao (2010). A bootstrap server kept a SSNG table in order to maintain the super-peer overlay topology. The SSNG table stored information of each SNG (square network graph). Peers were selected as super-peers by taking into account their localities. When a peer was selected as an active super-peer, it could spawn a child SNG (i.e. a client-peer layer). Specifically, the bootstrap server sent information of the contact peer of a SNG to the newly promoted super-peer, which sent a join message to that contact peer. After receiving neighbors information of the child SNG from the contact peer, the newly promoted super-peer established connections with those neighbors and became the super-peer of the SNG. However, the SNG contact peer bore heavy workload and was not scalable. Both PDG-based and SSNG-based overlay constructions need a bootstrap server for storing overlay information, which has a

single point of failure. It should be noted that in Papers III and IV, no contact peer is needed to collect all the peers' information for selecting super-peers and constructing a super-peer overlay, which improves scalability. In this thesis, information dissemination and peer sampling were conducted in a distributed manner using gossip without a server.

Montresor (2004) proposed a gossip-based algorithm, SG-1, for quickly building super-peer overlays. In SG-1, peers with high capacity are selected as super-peers. Super-peer selection is based on the comparing the capacity of a node with that of its randomly sampled neighbor peer. While the super-peer selection method is simple, it takes a long time to converge. In addition, super-peer searches only its one-hop neighbors in order to build connections with client peers, which is time consuming. Compared with SG-1, Papers III and IV proposed to use a set of super-peer candidates in super-peer selection, which is faster and reduces the time needed to select super-peers. Further, a conditional two-hop search method was proposed to speed up establishing the connections between super-peers and client peers.

## **2.4 P2P-overlay-based service discovery**

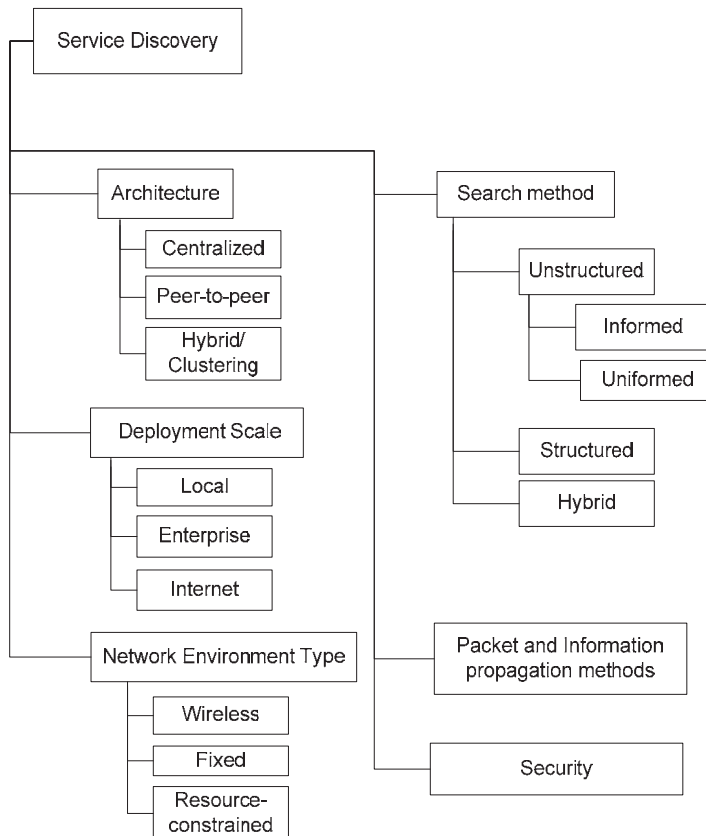
Service discovery has become critically important due to the growth of computer networks and the ubiquity of mobile and wireless networks (Meshkova *et al.* 2008). This section first provides an overview of service discovery and then reviews studies on P2P-based service discovery for constrained devices.

### **2.4.1 Overview of service discovery**

Service discovery (i.e. SD) is considered to be the capability to find specific services such as applications or well-defined networked services that are not pure abstractions (Meshkova *et al.* 2008). The terms service discovery and resource discovery are used interchangeably in this thesis.

A taxonomy of SD systems (Meshkova *et al.* 2008) is depicted in Fig. 9. In terms of architecture, SD systems can be classified into centralized, P2P, and hybrid ones (Vanthournout *et al.* 2005) (Meshkova *et al.* 2008). UDDI (Curbera *et al.* 2002) is a typical example of a centralized SD system, which consists of multiple repositories for handling service queries. Such a centralized approach may suffer from overload when the number of services grows. Consequently, a number of decentralized P2P-based SD systems have been proposed for services

discovery, e.g. Banaei-Kashani *et al.* (2004), Mastroianni *et al.* (2005) and Ranjan *et al.* (2008). Hybrid and clustering architectures for SD systems have been also investigated. For example, Puppin *et al.* (2005) proposed a super-peer-based architecture, where nodes are grouped into clusters and each cluster contains at least one super-peer. Super-peers act as servers within their clusters, handle service queries, and forward service queries to other clusters. Resource information is published by agents.



**Fig. 9. A taxonomy of service discovery systems (adopted from Meshkova *et al.* (2008)).**

In terms of search method, SD systems can be classified into unstructured, structured, and hybrid. Structured approach controls the locations of services by

algorithms and service lookup latency is bounded and usually shorter than that of unstructured SDs. Unstructured approach places services randomly and supports both partial-match and complex service queries. It is robust to node failures and node churn. However, performance of unstructured SDs depends heavily on employed search mechanisms and it cannot discover rare services as efficiently as popular services. The hybrid approach, in its turn, combines the advantages of structured and unstructured approaches to enhance the overall system performance. The original Gnutella (Klingberg & Manfredi 2002) used the unstructured approach, whereas DHT-based P2P systems (e.g. Chord) use the structured approach. Gnutella v.0.6 (Gnutella 2002) and Kazaa (Kazaa 2009) are examples of the hybrid approach.

The deployment scale of SD systems varies from local area networks to enterprise-scale networks and Internet-scale networks. Specifically, Internet-scale SD systems (e.g. for a million users) usually require high scalability and therefore utilize hybrid P2P overlays, such as eMule (Kulbak & Bickson 2005). Enterprise-scale SD systems such as Jini (Waldo 1999) use simple protocols and fewer servers, and put high emphasis on security (Meshkova *et al.* 2008). Enterprise-scale SD systems typically utilize a client–server infrastructure and discovered services can be particular documents or sensor readings. Local area-scale SD systems in networks (or personal area networks) usually adopt the simplest architectures, but emphasize high performance demands on service discovery because the number of services is small (Meshkova *et al.* 2008). UPnP (Universal Plug and Play) is a typical SD mechanism for local area networks that aims to provide a whole range of services in office or home environment (Miller *et al.* 2001).

In terms of network environment, SD systems can be deployed in fixed networks, Ad-hoc wireless networks, and resource-constrained networks (e.g. 6LowPan (IPv6 over Low power Wireless Personal Area Networks)). Many studies focused on service discovery in fixed network environments, in which service providers are usually not resource-constrained, for example, grid computing, cloud computing and clustering. In a fixed network environment, typical P2P-based methods for resource discovery are DHT-based structured protocols (e.g. Chord and Tapestry) and federated search based on unstructured protocols. The latter method can be further divided into systematic and random search algorithms. Systematic search methods explore search trees or graphs according to predefined rules without utilization of probabilistic or random choice (e.g. flooding (Tsoumakos & Roussopoulos 2003)). Random search methods, for

example, the random walk and probabilistic forwarding, are governed by random variables for the search depth and directions, which do not guarantee a good performance (Birman *et al.* 1999) (Kalogeraki *et al.* 2002) (Lv *et al.* 2002) (Russel & Norvig 2003) (Stauffer & Barbosa 2007). More discussion on service discovery in a fixed-network environment can be found in Meshkova *et al.* (2008). Security is a big topic for SD systems. Studies on security are not provided in this thesis because it is out of the scope of this thesis.

A resource discovery (RD) can be designed either as a third party service or as a genuinely distributed system, depending on the distinction made between resource providers and end-users (Vanthournout *et al.* 2005). Third party RD services such as DNS (Mockapetris & Dunlap 1988) and LDAP (Sermersheim 2006) are deterministic and centrally controllable systems that collect service information and handle resource queries from end-users. It should be noted that a third-party RD system could also be distributed even though it appears as a single entity.

Genuinely distributed RD service such as Gnutella (Gnutella 2002) and Tapestry (Zhao *et al.* 2004) involves both resource providers and end-users, without employing any central or coordinating components. More information on resource discovery design can be found in Vanthournout *et al.* (2005) and Meshkova *et al.* (2008).

#### **2.4.2 P2P-based service discovery for constrained devices**

This thesis focuses on P2P-based service discovery for resource-constrained devices in a constrained network. Such an environment is characterized by limited network connections and devices with limited processing capability and limited battery lifetime. A typical example is a wireless sensor network (WSN), where the embedded sensors may have only 1KB of RAM, for example, Leppanen *et al.* (2012). In such a resource-constrained environment, resource discovery faces the two challenges: (1) high data flows and sophisticated automatic repeat request (ARQ) methods are not feasible, (2) computationally heavy protocols cannot be applied. Consequently, some well-known service discovery protocols such as UPnP or SLP (service location protocol) are not directly applicable in a resource-constrained environment. For example, their packets are too large to be processed by a sensor.

There have not been many studies on service discovery systems for a resource-constrained environment (Meshkova *et al.* 2008). In some studies,

selected features of the needed protocols are sacrificed to reduce resource demands and to enable service discovery in a resource constrained environment, e.g. Shelby *et al.* (2003). TinyDB (Madden *et al.* 2005) and SwissQM (Mueller *et al.* 2007) are the examples of systems that merge service discovery with data collection functionalities. The rationale is that a majority of services (especially sensors) have a limited number of different values, which can be assembled in a single message.

Some studies utilize a P2P-based approach with other techniques, such as data aggregation and in-network processing (Fasolo *et al.* 2007) to achieve better scalability in a resource-constrained environment. For example, the TAG (Tiny Aggregation) proposed by Madden *et al.* (2002) handles data processing and aggregation on the intermediate nodes to save the bandwidth and power resources.

**Table 1. Mapping of taxonomy on some typical real-world service discovery systems (adapted from Meshkova *et al.* (2008)).**

Systems	Attributes								
	Network Environment Type			Architecture			Deployment scale		
	Fixed	Wireless	Constrained	Client-server	Peer-to-peer	Hybrid	Local	Enterprise	Internet
Bonjour	√	√	√		√		√		
TinyDB		√	√	√		√	√		
SwissQM		√	√	√			√		
UPnP	√	√			√		√		
SLP	√			√				√	
JXTA	√	√			√	√			√
eDonkey	√			√	√				√
Kazaa	√				√	√			√
BitTorrent	√				√				√
Freenet	√				√				√
Napster	√			√	√				√
Gnutella	√				√				√
UDDI	√			√				√	
Jini	√			√				√	
Tanganelli <i>et al.</i> (2013)	√	√	√		√				√



Table 1 provides a mapping of some typical real-world service discovery systems onto the taxonomy shown in Fig. 9. We can see that only a few studies (e.g. Bonjour, TinyDB, and SwissQM) focus on a resource-constrained environment.

Service discovery systems in resource-constrained WSNs have two basic models: client-server and distributed. In the client-server model, powerful devices work as servers (usually as gateways) for surrounding nodes: integrating new nodes into the network, collecting data from them and controlling their behaviors. The client-server model has been studied a lot in integrating constrained sensors into the IP networks, e.g. by Valtchev & Frankov (2002), Schramm *et al.* (2004) and Luckenbach *et al.* (2005). In the distributed model, functionalities are distributed over all the nodes as equally as possible. This yields a more scalable system, which is suitable for large-scale applications with random node distribution. The distribution is often integrated with data aggregation and in-network processing, e.g. Yao & Gehrke (2002) and Gupta *et al.* (2006).

Nowadays, besides wireless sensors, other smart devices such as smartphone are able to participate in the intelligent service provision (e.g. environment monitoring, intelligent transport systems), by exposing their services into the Internet, and accessing services provided by other devices. These constrained devices interact with each other and construct a highly distributed network of embedded systems, referred to as a M2M communication network. The aforementioned service discovery systems cannot fulfill the service discovery requirement in a M2M network environment. For example, it is not feasible to directly discover resources provided the constrained devices because the nodes may be sleeping or have only intermittent connection (Shelby *et al.* 2012).

The IETF CoRE (Constrained RESTful Environments) working group has been developing protocols that would enable utilizing of constrained devices in M2M service provision (Constrained RESTful Environment 2012). The working group has proposed the Constrained Application Protocol (CoAP) (Shelby *et al.* 2013) to support the communication for resource-constrained devices. The working group also proposed a draft of the CoRE resource directory (Shelby *et al.* 2012) for constrained devices to register and look up resource descriptions. The draft defined the methods of discovering resource directory, registering, and looking up resource descriptions. CoAP protocol was used by devices to communicate with the CoRE resource directory. The draft specifies the basic functionality of a resource directory but does not provide architecture design nor implementation. Mäenpää *et al.* (2012) presented a P2P-based rendezvous service for resource-constrained nodes in WSNs. In their work, both resource description

registration and look-up are done using CoAP. Their work does not support smart devices (e.g. a smartphone) using a different protocol (e.g. HTTP) for resource registration and lookup. In contrast, this thesis aims to support heterogeneous devices in M2M service provision. Thus, Paper V reports how the two common protocols, HTTP and CoAP, are supported in resource registration and discovery, including the lookup of a set of resources by using some resource properties in query filtering.

Tanganelli *et al.* (2013) proposed a P2P overlay-based distributed architecture for discovering and accessing resources provided by constrained devices in M2M service provision. Following a resource-oriented approach, their system also utilized CoAP as the communication protocol for smart objects. They utilized a Pastry overlay (a P2P overlay) and the extendible Metadata Hash Table (XMHT) (Andreini *et al.* 2010) for service discovery. When a client sends a request to a proxy, the proxy first looks up the server location through a P2P querying. Then, the proxy invokes the requested method on behalf of the client and sends a response back to the client. They used the CoAP URI as the resource names for replication and semantic aggregation of data. This means that different servers can host resources with the same name. However, Tanganelli *et al.* (2013) did not take into account the naming of specific devices and cannot support the discovery of a service provided by a specific device. Paper V utilizes both URI and an endpoint name to uniquely identify a resource provided by a specific device. For example, two sensors provide temperatures (laboratory1 and laboratory2), when a client wants to know the temperature in laboratory2, the client sends a query request that includes the endpoint name “laboratory2” to get its needed value.

### 3 Summary of research contributions

In this chapter, the contributions of the original publications are elaborated in detail. Furthermore, the chapter discusses how these contributions answer the specific research questions addressed in this thesis. Section 3.1 presents super-peer-based coordinated service provision. Section 3.2 presents an efficient algorithm for constructing a robust super-peer overlay. Section 3.3 presents a distributed resource discovery for Machine-to-Machine service provision. Section 3.4 provides discussion and future work.

#### 3.1 Super-peer-based coordinated service provision

The first research question of this thesis addresses efficient coordinated service provision using a super-peer overlay. Papers I and II make contributions to this research area and provide an answer to the research question.

Paper I presents the SCSP framework for coordinating different service groups in sharing resources in order to handle dynamic service demands. The SCSP framework introduces two core components: a *S-labor-market model* and a *recruiting protocol* based on a weighting mechanism to coordinate service groups in sharing resources and maintain service peers within service groups. The *S-labor-market model* defines the abstract coordination operations among service groups. Specifically, the *S-labor-market model* defines two market roles (recruiter taken by super-peers and employee taken by service peers) and two events (*Service change event* and *Resignation event*). Service groups perform the *S-labor-market model* events using the *recruiting protocol*. Particularly, by running the *recruiting protocol*, a service group can dynamically recruit service peers from other groups to reduce service response time (when the workload of the service group increases), and dismiss idle service peers within a service group to improve resource efficiency (when the workload of the service group decreases).

The architecture of the SCSP is illustrated in Fig. 10. The SCSP framework consists of three layers: the super-peer layer, the service peer layer, and the network layer. The super-peer layer is responsible for dispatching users' service requests to service peers and maintaining service peers in the service groups. Service peers providing the same services are grouped as service groups. Each service group consists of a super-peer and service peers. Different super-peers manage different services. Service groups cooperate with each other on sharing service peers to meet dynamic service demands. Instead of applying a DHT-based

P2P overlay as in Chen *et al.* (2008), the SCSP utilizes a super-peer-based P2P overlay in organizing service groups and exploiting the heterogeneity of peer capabilities. Super-peers have more responsibilities than service peers. Super-peers take the role of a recruiter, due to their high capacity, and service peers take the role of employees.

When the *service change event* for sharing resources with other service groups is performed, a super-peer in one service group selects a few other super-peers to send service change applications. A simple weighting mechanism is utilized by the super-peer to make an initial recruiting decision and to negotiate the final decision with the service peer. When performing the *Resignation event*, a service peer negotiates with its super-peer to make a final decision according to the *recruiting protocol*. Adopting the super-peer overlay reduces the time for making decisions and reduces communication traffic as well.

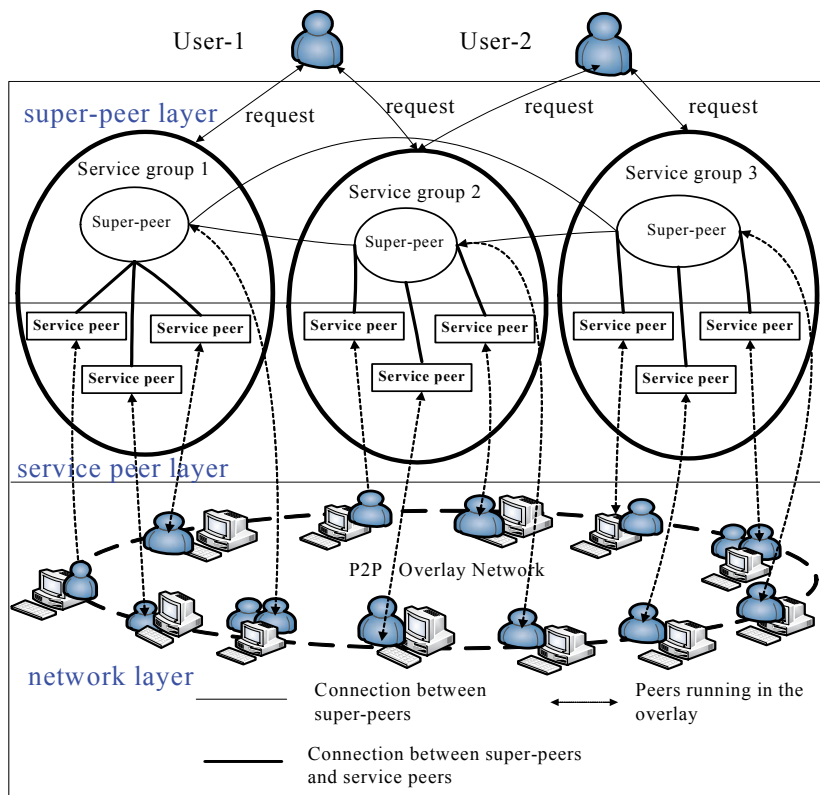


Fig. 10. The SCSP architecture in Paper I. 2011 Elsevier Journal of Network and Computer Applications. Reprinted with permission.

The performance of the SCSP was evaluated with simulations on the PeerSim platform. Peers were classified into three categories, the high capability class, the middle capability class, and the low capability class to demonstrate the heterogeneity of service peers' capabilities. Capabilities of service peers were assumed to follow normal distribution. Five service groups were used in the experiments. Each service group provides a distinct service, i.e. different from the services provided by other service groups. Peers were randomly assigned to one of the service groups. The arrival rate of service requests to each service group was assumed to follow Poisson distribution.

The coordination among different service groups built by the SCSP was evaluated. Simulation results showed that the variations in the response time of a service group to a service request (SRRT) among the five service groups in the SCSP were small. This result verifies that the coordination mechanism in the SCSP works well among different service groups. Furthermore, the variations in the SRRTs among the five service groups in the SCSP were smaller than those of Chen *et al.* (2008).

The scalability of the SCSP was examined with three sets of simulations. The number of peers in the system increased from 200 to 2000. Results show that the response time (i.e. SRRTs) varied marginally among the three sets of simulations when the size of service groups and service peers increase. The average group sizes fluctuated limitedly among the three sets of simulations as well. In other words, the SCSP achieved a good scalability.

The robustness of the SCSP to node joins or leaves was also evaluated. Two sets of simulations were conducted and five service groups were employed in these simulations. The first simulation simulated the activity of service peer joins. Results showed that the service response time reduces significantly when new peers join the system. The second simulation simulated the activity of service peer leaves. Results showed that the number of peers in one service group (e.g. service group SG<sub>2</sub>) decreases but the service response time does not fluctuate when the total number of peers in the five service groups decreases. The reason for this result is twofold. First, in the beginning, SG<sub>2</sub> employs enough service peers with high capability since the total number of initial service peers in the five service groups is large. Second, the leaving peers are chosen from the resigned service peers and most of them have relatively low and middle capabilities according to the *recruiting protocol*. In other words, most of the service peers with a high capability remain in SG<sub>2</sub> even when peers continue to leave.

Therefore, the performance of  $SG_2$  is not affected much by the leaves of service peers. These results prove that the SCSP is robust to service peers joins and leaves.

In order to make a fair comparison with Chen *et al.* (2008), traffic (i.e. communication message) generated by coordination mechanism was analyzed. The analysis shows that when the total number of peers in a service provision system is relatively small (e.g. 250), the SCSP generated fewer messages than that of Chen *et al.* (2008). When the total number of peers in a system is relatively large (e.g. 2000) and the number of service groups is relatively small (i.e. the service group has a relatively large number of peers), the number of messages generated by a peer in a service change operation in the SCSP is smaller than that in Chen *et al.* (2008).

Paper II presents the coordination mechanism for cloud service provision (SPCM). An enterprise cloud (e.g. Aneka (Chu *et al.* 2007)) consists of a number of distributed heterogeneous resources (nodes, virtual computers, applications), which are combined together to form a massive environment (Ranjan & Rajkumar 2010). Due to dynamic resource demands and less cooperation among users' applications, some resources in enterprise clouds could be frequently required and bear a heavy workload. SPCM manages the resources (e.g. the cloud nodes and virtual computers) of enterprise clouds efficiently and enables enterprise clouds to collaborate with each other in sharing resources to satisfy QoS requirements (e.g. short response time of a service request). Whenever the load of an enterprise cloud managed by a super-peer increases, the super-peer of that enterprise cloud can reserve cloud nodes from other enterprise clouds for processing service requests. The service response time of enterprise cloud applications is reduced by those reserved cloud nodes. Consequently, the overall response time of the enterprise cloud applications is reduced. Simulations were carried out to evaluate the coordination performance of the SPCM with two application scenarios. The results demonstrate that the SPCM enables enterprise clouds to collaborate with each other efficiently.

### **3.2 An efficient algorithm for constructing a super-peer overlay**

The second research question of this thesis addresses the rapid construction of a robust super-peer overlay given a random connected topology. Papers III and IV make contributions to this research area.

Paper III presents a gossip-based super-peer selection algorithm (GSPS) for constructing a super-peer overlay. The GSPS selects the nodes with the highest

capacity as super-peers; thus exploiting the heterogeneity of nodes to improve performance. The capacity of a node is determined by its capabilities, including CPU cycles, bandwidth, storage, and online time. The capacity of a node is computed as  $\sum_{k=1}^K (v_k \times w_k)$ , where  $K$  is the resource types ( $K=4$ );  $v_k$  is the value of the  $k^{th}$  resource metric;  $w_k$  is the weight of the  $k^{th}$  resource metric and  $\sum_{k=1}^K w_k = 1$ . The value of  $w_k$  can be set experimentally according to the requirements of specific applications.

The basic idea of the GSPS is as follows: First, each node constructs a set of super-peer candidates based on the gossip communication, in which a node exchanges information with its neighbors in order to select nodes with high capacity as super-peers candidates. After that, if the node is in the list of the super-peer candidates, the node adopts the role of a super-peer. Finally, each node executes corresponding operations, either joining a super-peer or recruiting a client node, according to their roles. A conditional two-hop search method is employed by super-peers to find and add client peers, which reduces the convergence time for building an overlay in a worst-case scenario. Herein, the worst case is that a super-peer manages some client peers and finds out that all of its neighbors have joined super-peers after it searches its neighbors. The worst case would increase the convergence time of building a super-peer overlay dramatically if only the one-hop search method would be used. In the case of a disaster (i.e. the failure of a super-peer), the client peers reconstruct their sets of super-peer candidates, select a new number of required super-peers, and then join these super-peers.

Simulations using PeerSim were conducted to evaluate the performance of the GSPS algorithm. The evaluation focused on three aspects: scalability, robustness to the failure of super-peers, and performance comparison with related protocol. The initial simulation topology was a random graph, in which all the nodes took the role of a client. This simulation setting provides a reliable way to verify the efficiency of the GSPS algorithm because the initial topology is farthest from the target super-peer overlay. The number of simulation rounds was used to measure the time needed to construct a super-peer overlay. An abstracted capacity of a node, denoted as the number of client nodes that can be managed by a node, was used to represent the capacity of a node.

The scalability of the GSPS was measured by increasing the overlay size from 1000 to 100000 nodes. Results show that the time for building the target

super-peer overlay increases only a little with respect to the increase in network size. In other words, the increase in the number of the nodes (in the overlay) has only a small impact on the convergence time of the GSPS and the GSPS achieves a good scalability.

The GSPS was compared with SG-1 in terms of the influence of super-peers' maximum capacity on the convergence time of constructing a super-peer overlay. The results showed that as the super-peers' maximum capacity increases, the GSPS takes a few more simulation rounds to converge to the target overlay, peaking at 8.5 rounds on average when the capacity is equal to 600. The SG-1 takes a little longer simulation time than the GSPS with the increase of the super-peers' maximum capacity, peaking at 14 rounds when the capacity is equal to 600 on average. Further details on the comparison can be found in Paper III.

The robustness of the GSPS was evaluated as well. Three catastrophic scenarios used in the evaluation are as follows: (1) 10% of the super-peers are removed at simulation round six; (2) 20% of the super-peers are removed at round six; and (3) 30% of the super-peers are removed at round six. Results showed that even when the number of failed super-peers increases, it almost takes the same number of simulation rounds for the GSPS algorithm to reach the steady overlay state again.

Paper IV continues the work in Paper III and proposes the SPS algorithm with additional simulations for evaluating its performance. Firstly, the communication overhead of the SPS was compared with that of the SG-1 through simulation. Three types of communication overhead were evaluated: (1) the total number of probes per node for query on the load of neighbor super-peers, (2) the number of gossip messages per node for building super-peer candidates, and (3) the number of client peer transfers per node. Results showed that the proposed SPS algorithm is comparable to SG-1 in terms of the total communication overhead. Particularly, maintaining the set of super-peer candidates in the algorithm generates more communication overhead; nevertheless, it has a significant positive effect on reducing the number of client peer transfers.

Secondly, the scalability of the SPS algorithm is compared with that of the SG-1 in terms of convergence time of constructing a super-peer overlay. Simulation results show that the SPS algorithm achieves better scalability than the SG-1 when the number of nodes increases from 1000 to 10000.

Thirdly, the robustness of the SPS is compared with that of the SG-1 in case of the failure of super-peers. Results show that robustness of the SPS is comparable to the SG-1 in the face of super-peer failure when taking into account



both the convergence time and the impact of super-peer failure on client peers. Particularly, the number of client peers that have joined a super-peer in the SPS decreases sharply compared to a slight decrease in the SG-1. This means that the failure of super-peers causes more client peers losing super-peers. However, a smaller number of simulation rounds is needed to restore stable state in the SPS algorithm than in the SG-1. This means that the SPS algorithm restores the overlay faster than that of the SG-1.

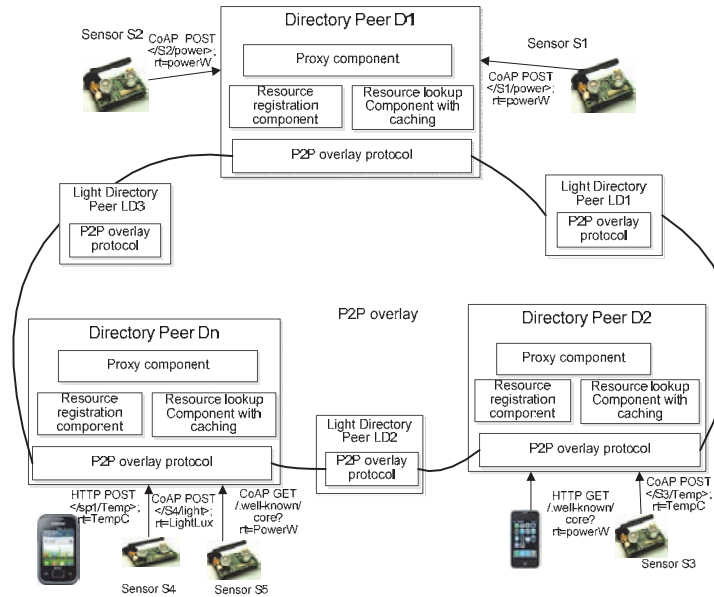
### **3.3 Distributed resource discovery architecture based on a P2P overlay for Machine-to-Machine service provision**

The third research question of this thesis focuses on designing a distributed resource discovery for M2M service provision. Paper V contributes to this research area and provides an answer to the research question.

Paper V proposes a distributed resource directory architecture for M2M service provision, referred to as the DRD4M. It is not feasible to assume that heterogeneous devices involved in M2M service provision use the same communication protocol. Therefore, the DRD4M supports both HTTP and CoAP protocols to enable interoperability with heterogeneous devices (the constrained devices and the common Internet of things devices (e.g. a smartphone)) and resource access by constrained devices from disparate networks, including the Internet. A P2P overlay is introduced to connect resource peers and light resource peers to avoid a single point of failure and to achieve load balancing.

The architecture of the DRD4M is illustrated in Fig. 11. The DRD4M consists of a number of directory peers and light directory peers, which are connected via a P2P overlay. A directory peer handles the registrations of resource description and lookups of resource description for these constrained devices. Each directory peer consists of three layers. The first layer is a proxy component that handles CoAP and HTTP requests. The second layer consists of a resource registration component (RRC) and a resource lookup component (RLC) with caching. The RRC component is responsible for registering resource descriptions into the P2P overlay when getting parsed requests from the proxy. The RLC component with caching functionality takes care of resource queries and processes lookup results returned by the P2P overlay protocol (if needed). The bottom layer runs a P2P protocol to connect to other directory peers, and provides basic resource storage and resource lookup from the overlay. A light directory peer only runs the P2P protocol to store and lookup resources from the overlay,

but does not handle requests from clients (i.e. not parsing HTTP and CoAP requests). The reason is that while some mobile devices can participate in the overlay as light directory peers to store resources and forward lookup requests (Ou *et al.* 2009), they cannot act as both HTTP and CoAP servers.



**Fig. 11. Architecture of the DRD4M in Paper V. 2013 IEEE conference on Wireless and mobile computing, networking and communication, workshop on Internet of Things Communications and Technologies. Reprinted with permission.**

The RLC employs a caching mechanism to reduce the response times and communication overhead of resource lookups. Specifically, if the RLC receives a query including filters, the RLC first looks up cached results (stored in the overlay) to find queried resources. If a matching response can be found in the cache, the RLC returns the cached results. Otherwise, the RLC looks up the resources from the overlay and then caches the response with a probability. If some resources are queried more frequently, these resources are assigned a higher probability to be cached into the overlay.

A functional real-world prototype of the DRD4M was implemented and its feasibility was verified with a real-world demo. In the demo, the DRD4M enables a smartphone to access resources provided by an embedded wireless sensor deployed on 6LoWPAN (IPv6 over Low power Wireless Personal Area Networks).

The constrained sensor uses Atmel 1284P microcontroller-based embedded devices with 16 KB of RAM (Leppanen *et al.* 2012), which communicates using CoAP over 6LoWPAN. Samsung Galaxy SIII is used as a general Internet device for querying resources provided by the wireless sensor by sending HTTP messages over a Wi-Fi network. A preliminary evaluation of the response time on looking up resources based on the prototype was conducted. Evaluation results indicate that the caching functionality can dramatically reduce the response time (from 62.17 ms to 17.49 ms) on a resource lookup query without containing resource path information. The caching functionality reduces the response time on a resource lookup query that includes resource path information only a little bit. This result can be used as a guidance to improve the caching functionality. Further details about this evaluation can be found in Paper V.

### **3.4 Discussion and future work**

Cloud computing targets at providing infrastructure, platform and software as a service to end users. Thus, clouds host many heterogeneous resources to serve the customers. The SCSP coordination framework presented in Paper I could also be applied (with modification) to coordinate the hosted different resources in the clouds, to improve their resource utilization and provide better quality of service (e.g. short response time) to end users. The SCSP framework considers only those peers that are independent from each other as the service entities (service peers). Many studies have focused on composing existing services to reduce deployment cost and increase revenue (Komorita *et al.* 2012). Coordinating the provision of composed services requires investigation on the interdependencies and transaction property of individual services.

More and more powerful mobile devices actively participate in service provision. For example, a mobile device can participate in mobile computing or mobile cloud computing as a client peer. However, it is very challenging to build a super-overlay for a mobile cloud that supports multiple access networks (3G, LTE (Long Term Evolution), Bluetooth and Wi-Fi) and provides a timing sensitive service such as mobile streaming (Lee *et al.* 2013). The overlay construction algorithm published in Papers III and IV is limited to the setting where nodes or peers are not resource-constrained and nodes have usually fixed network connection. Thus, the proposed overlay construction algorithm is not directly applicable to mobile cloud-based services. For example, in a multi-access network environment, the selection of the connections between peers is an

important issue for investigation. It is not feasible to maintain a link for seamless transmission between mobile devices in a wireless P2P network due to users' mobility. A mobile P2P overlay construction needs to take into account the network connectivity status of neighboring devices, the status of multi-access networks (e.g. Bluetooth, Wi-Fi, and 3G) in order to generate less traffic and deliver services with acceptable response times. Further, the criteria on super-peer selection should take into account relevant factors, for example, the battery power level of devices.

The P2P-based distributed resource discovery architecture for constrained devices (DRD4M) proposed in Paper V provides a fundamental building block for M2M service provision. The DRD4M is implemented with a DHT-based P2P overlay, which suffers from heavy maintenance cost, especially when peers join or leave the overlay frequently. In addition, the DHT-based P2P overlay is efficient in handling simple service queries but cannot resolve multi-dimensional queries. Thus, further research is needed to achieve scalability with multi-dimensional queries, which is an important performance requirement in large-scale M2M service provision. A super-peer overlay can be introduced to reduce the response time of a resource query. Specifically, if a query contains a filter, e.g. an endpoint name, the super-peer can first process the filter to decide whether it should handle the query or forward the query to another super-peer. The current implementation of the DRD4M supports two different communication protocols, CoAP and HTTP, to achieve interoperability with heterogeneous devices, for example smartphone and wireless sensors. However, many pervasive applications also use RFID (Radio Frequency Identification) or NFC (Near field communication) communication technologies. Supporting devices using RFID and NFC to register and lookup of resource descriptions is another area of future work that would allow integrating additional heterogeneous devices into M2M service provision.

## 4 Conclusions

The foundation of P2P networking was laid over four decades ago with the birth of the predecessor of the Internet, the ARPANET. The term P2P has been widely used for over ten years, symbolized by the emergence of popular P2P applications such as Napster, BitTorrent, and Skype. Although P2P applications (especially file sharing applications) generate a large proportion of Internet traffic (Moya 2009) (Piatek *et al.* 2009) (Mondal *et al.* 2012), many studies have demonstrated that P2P traffic can be decreased by using ISP-friendly P2P designs (Mondal *et al.* 2012). Moreover, applying P2P technologies in the emerging domains, such as cloud computing, Internet of Thing, and M2M communication, pours new vigor into P2P-based service provision.

This thesis investigated three aspects of super-peer-based service provision. Firstly, a super-peer-based coordinated service provision framework (SCSP) was proposed to satisfy dynamic service demands in large-scale computing applications, e.g. cloud computing, grid services. The SCSP framework employs a labor-market model and a recruiting protocol to coordinate service groups and enable resource sharing to achieve short service response time and high resource efficiency. The performance of the SCSP was evaluated in terms of scalability, robustness, and service response time. The SCSP achieved shorter response times and a comparable resource efficiency when compared with existing work (Chen *et al.* 2008). In addition, the SCSP achieved good scalability and robustness in a network with 2000 peers. The SCSP could be modified for coordinating the hosted resources of a cloud to reduce response time and to improve resources utilization. The SCSP could also be utilized for resource management in applications that have computationally intensive and dynamically fluctuating service demands, such as bioinformatics analysis and distributed collaborative adaptive sensing systems.

This thesis proposes a GSPS algorithm for constructing a super-peer overlay on a given overlay. The GSPS introduces a super-peer candidate-based method for selecting super-peers and a conditional two-hop search method to reduce the time needed to build a super-peers overlay. The performance of the GSPS was compared with the SG-1 in terms of the convergence time, communication overhead, scalability, and robustness. The comparison showed that the GSPS converges faster with a comparable performance in robustness and communication traffic. The super-peer selection algorithm could be enhanced for mobile environment to build a mobile P2P overlay for mobile service provision.

Network connectivity status and selection of multi-access networks should be taken into account.

Finally, this thesis proposes the DRD4M, a distributed resource discovery architecture utilizing a P2P overlay to address the challenge of discovering resources provided by a large number of heterogeneous devices in the M2M service provision. The DRD4M supports constrained devices with the CoAP and HTTP protocols. A real-world prototype of the DRD4M was implemented to verify the feasibility of the proposed architecture. A preliminary evaluation of the response time of resource lookup was conducted using the prototype. The results showed that the caching functionality dramatically reduces the time on looking up resource descriptions for constrained devices. The current design supports only simple one-dimensional queries. Thus, adding support for multi-dimensional queries is an important feature extension. In addition, adding support for devices using RFID and NFC to register and lookup of resource descriptions would allow integrating other heterogeneous devices into M2M service provision.

## References

- Alonso G, Casati F, Kuno H & Machiraju V (2004) *Web Services: Concepts, architectures and applications*. Springer.
- Amazon cloudwatch service (2010) Amazon cloudwatch service, <http://aws.amazon.com/cloudwatch/> (last accessed 12-12-2013)
- Amazon load balancer service (2010) Amazon load balancer service <http://aws.amazon.com/elasticloadbalancing/> (last accessed 12-12-2013).
- Anderson D P, Cobb J, Korpela E, Lebofsky M & Werthimer D (2002) SETI@ home: an experiment in public-resource computing. *Communications of the ACM* 45(11): 56–61.
- Andreini F, Crisciani F, Cicconetti C & Mambri R (2010) Context-aware location in the internet of things. *Proc IEEE Global Telecommunications Conference Workshop on Network of the Future*. Miami, Florida, USA: 300–304.
- Androutsellis-Theotokis S & Spinellis D (2004) A survey of peer-to-peer content distribution technologies. *ACM Computing Surveys* 36(4): 335–371.
- Arai B, Das G, Gunopulos D & Kalogeraki V (2007) Efficient approximate query processing in peer-to-peer networks. *IEEE Transactions on Knowledge and Data Engineering* 19(7): 919–933.
- Artigas M S, Lopez P G & Skarmeta A F (2007) A comparative study of hierarchical DHT systems. *Proc IEEE Conference on Local Computer Networks*. Dublin, Ireland: 325–333.
- Banaei-Kashani F, Chen C C & Shahabi C (2004) WSPDS: Web services peer-to-peer discovery service. *Proc International Conference on Internet Computing*. Las Vegas, USA: 733–739.
- Bandara H M N D & Jayasumana A (2013) Collaborative applications over peer-to-peer systems—challenges and solutions. *Peer-to-Peer Networking and Applications* 6(3): 257–276.
- Baset S A & Schulzrinne H G (2006) An analysis of the Skype peer-to-peer Internet telephony protocol. *Proc IEEE International Conference on Computer Communications*. Barcelona, Spain: 1–11.
- Benatallah B, Dumas M, Sheng Q Z & Ngu A H (2002) Declarative composition and peer-to-peer provisioning of dynamic web services. *Proc International Conference on Data Engineering*. California, USA: 297–308.
- Birman K P, Hayden M, Ozkasap O, Xiao Z, Budiu M & Minsky Y (1999) Bimodal multicast. *ACM Transactions on Computer Systems* 17(2): 41–88.
- Borg J (2003) *A comparative study of ad hoc & peer to peer networks*. University College London.
- Brazier F M T, Kephart J O, Parunak H V D & Huhns M N (2009) Agents and Service-Oriented Computing for Autonomic Computing. *IEEE Internet Computing* 13(3): 82–87.

- Brocco A, Frapolli F & Hirsbrunner B (2009) Bounded diameter overlay construction: A self-organized approach. Proc IEEE Swarm Intelligence Symposium. Tennessee, USA: 114–121.
- Brotzge J, Chandresakar V, Droegemeier K, Kurose J, McLaughlin D, Philips B, Preston M & Sekelsky S (2004) Distributed collaborative adaptive sensing for hazardous weather detection, tracking, and predicting. Proc International conference on Computational Science, Lecture Notes in Computer Science 3038. Kraków, Poland: 670–677.
- Burdett D & Kavantzias N (2004) The WS-Choreography Model Overview. W3C Draft, <http://www.w3.org/TR/ws-chor-model/> (last accessed 12-12-2013).
- Butt A R, Zhang R M & Hu Y C (2006) A self-organizing flock of Condors. Journal of Parallel and Distributed Computing 66(1): 145–161.
- Buyya R, Abramson D & Giddy J (2000) An Economy Driven Resource Management Architecture for Global Computational Power Grids. Proc International Conference on Parallel and Distributed Processing Techniques and Applications. Nevada, USA: 300–304.
- Cabrera L, Copeland G, Freund T, Klein J, Langworthy D, Orchard D, D S & Storey J (2005) Web Services Coordination Specification. <http://www.ibm.com/developerworks/library/specification/ws-tx/> (last accessed 12-12-2013).
- Castro M, Druschel P, Kermarrec A-M, Nandi A, Rowstron A & Singh A (2003) SplitStream: high-bandwidth multicast in cooperative environments. Proc ACM symposium on Operating systems principles. Bolton Landing, NY, USA: 298–313.
- Castro M, Druschel P, Kermarrec A M & Rowstron A I T (2002) Scribe: A large-scale and decentralized application-level multicast infrastructure. IEEE Journal on Selected Areas in Communications 20(8): 1489–1499.
- Chakravarti A J, Baumgartner G & Lauria M (2005) The organic grid: Self-organizing computation on a peer-to-peer network. IEEE Transactions on Systems Man and Cybernetics Part a-Systems and Humans 35(3): 373–384.
- Chen G, Low C P & Yang Z H (2008) Coordinated service provision in peer-to-peer environments. IEEE Transactions on Parallel and Distributed Systems 19(4): 433–446.
- Cheng W, Gong J & Ding W (2007) Identifying BT-like P2P traffic by the discreteness of remote hosts. Proc 32nd IEEE Conference on Local Computer Networks. Dublin, Ireland: 237–238.
- Chhabra S, Damiani E, De Capitani di Vimercati S, Paraboschi S & Samarati P (2004) A protocol for reputation management in super-peer networks. Proc International Workshop on Database and Expert Systems Applications. Zaragoza, Spain: 979–983.
- Chu X C, Nadiminti K, Jin C, Venugopal S & Buyya R (2007) Aneka: Next-generation enterprise grid platform for e-Science and e-Business applications. Proc IEEE International Conference on E-Science and Grid Computing. Bangalore, India: 151–159.
- Cohen B (2009) BitTorrentURL <http://www.bittorrent.com/btusers/guides/bittorrent-user-manual/chapter-02-basic-guides/basics-bittorrent> (last accessed 17-12-2009).



- Constrained RESTful Environment (2012) Constrained RESTful Environments, <https://datatracker.ietf.org/wg/core/charter/> (last accessed 12-12-2013).
- Conti M, Giordano S, May M & Passarella A (2010) From Opportunistic Networks to Opportunistic Computing. *IEEE Communications Magazine* 48(9): 126–139.
- Couto da Silva A P, Leonardi E, Mellia M & Meo M (2011) Exploiting heterogeneity in P2P video streaming. *IEEE Transactions on Computers* 60(5): 667–679.
- Crespo A & Garcia-Molina H (2005) Semantic overlay networks for P2P systems. *International Workshop on Agents and Peer-to-Peer Computing, Lecture Notes in Computer Science 3601*. New York, USA: 1–13.
- Cuenca-Acuna F M & Nguyen T D (2004) Self-managing federated services. *Proc 23rd IEEE International Symposium on Reliable Distributed Systems*. Florianopolis, Brazil: 240–250.
- Curbera F, Duftler M, Khalaf R, Nagy W, Mukhi N & Weerawarana S (2002) Unraveling the Web services web: an introduction to SOAP, WSDL, and UDDI. *IEEE Internet Computing* 6(2): 86–93.
- DeCandia G, Hastorun D, Jampani M, Kakulapati G, Lakshman A, Pilchin A, Sivasubramanian S, Voshall P & Vogels W (2007) Dynamo: amazon's highly available key-value store. *ACM SIGOPS Operating Systems Review* 205–220.
- Dimitriou T, Karame G & Christou I (2008) SuperTrust—A secure and efficient framework for handling trust in super peer networks. *Proc International Conference on Distributed Computing and Networking, Lecture Notes in Computer Science 4904*. Kolkata, India: 350–362.
- Doulkeridis C, Norvag K & Vazirgiannis M (2007) DESENT: Decentralized and distributed semantic overlay generation in P2P networks. *IEEE Journal on Selected Areas in Communications* 25(1): 25–34.
- Doulkeridis C, Vlachou A, Nørvgå K & Vazirgiannis M (2010) Distributed semantic overlay networks. *Handbook of Peer-to-Peer Networking*. Springer USA: 463–494.
- Druschel P, Engineer E, Gil R, Hu Y C, Iyer S & Ladd A (2001) FreePastry.
- Erdős P & Renyi A (1959) On random graphs I. *Publicationes Mathematicae Debrecen* 6(): 290–297.
- Ericsson E (2012) Traffic and Market Data Report [http://www.ericsson.com/res/docs/2012/traffic\\_and\\_market\\_report\\_june\\_2012.pdf](http://www.ericsson.com/res/docs/2012/traffic_and_market_report_june_2012.pdf).
- Fasolo E, Rossi M, Widmer J & Zorzi M (2007) In-network aggregation techniques for wireless sensor networks: A survey. *IEEE Wireless Communications* 14(2): 70–87.
- Foster I (2005) Service-oriented science. *Science* 308(5723): 814–817.
- Foster I & Iamnitchi A (2003) On death, taxes, and the convergence of peer-to-peer and grid computing. *Proc International workshop on IPTPS (Peer-to-Peer Systems), Lecture Notes in Computer Science 2735*. Berkeley, CA, USA: 118–128.
- Ganesan P, Sun Q & Garcia-Molina H (2003) Yappers: A peer-to-peer lookup service over arbitrary topology. *Proc Annual Joint Conference of the IEEE Computer and Communications*. California, USA: Vol 2, 1250–1260.
- Ganesh A J, Kermarrec A M & Massoulié L (2003) Peer-to-peer membership management for gossip-based protocols. *IEEE Transactions on Computers* 52(2): 139–149.

- Garces-Erice L, Biersack E W, Ross K W, Felber P A & Urvoy-Keller G (2003) Hierarchical peer-to-peer systems. *Parallel Processing Letters* 13(04): 643–657.
- Gnutella (2002) Gnutella ultrapeers. <http://rfc-gnutella.sourceforge.net/Proposals/Ultrapeer/Ultrapeers.htm/> (last accessed 12-12-2013).
- Goel S, Talya S S & Sobolewski M (2007) Service-based P2P overlay network for collaborative problem solving. *Decision Support Systems* 43(2): 547–568.
- Gu X H & Nahrstedt K (2006) On composing stream applications in peer-to-peer environments. *IEEE Transactions on Parallel and Distributed Systems* 17(8): 824–837.
- Gumstix [http://www.gumstix.com/store/product\\_info.php?products\\_id=211,2012\(oct\)](http://www.gumstix.com/store/product_info.php?products_id=211,2012(oct)).
- Gupta H, Zhou Z, Das S R & Gu Q (2006) Connected sensor cover: self-organization of sensor networks for efficient query execution. *IEEE/ACM Transactions on Networking* 14(1): 55–67.
- Helgason Ó R, Yavuz E A, Kouyoumdjieva S T, Pajevic L & Karlsson G (2010) A mobile peer-to-peer system for opportunistic content-centric networking. *Proc ACM SIGCOMM Workshop on Networking, systems, and applications on mobile handhelds*. New Delhi, India: 21–26.
- Hsiao H-C, Liao H & Huang C-C (2009) Resolving the topology mismatch problem in unstructured peer-to-peer networks. *IEEE Transactions on Parallel and Distributed Systems* 20(11): 1668–1681.
- Hsiao H-C & Su H-W (2012) On Optimizing Overlay Topologies for Search in Unstructured Peer-to-Peer Networks. *IEEE Transactions on Parallel and Distributed Systems*, 23(5): 924–935.
- Hu Y C, Das S M & Pucha H (2003) Exploiting the Synergy between Peer-to-Peer and Mobile Ad Hoc Networks. *Proc International conference on Hot Topics in Operating Systems*. Hawaii, USA: 37–42.
- Hui K Y, Lui J & Yau D K (2006) Small-world overlay P2P networks: construction, management and handling of dynamic flash crowds. *Computer Networks* 50(15): 2727–2746.
- Iliofotou M, Kim H C, Faloutsos M, Mitzenmacher M, Pappu P & Varghese G (2011) Graption: A graph-based P2P traffic classification framework for the internet backbone. *Computer Networks* 55(8): 1909–1920.
- Jelasy M, Kowalczyk W & Van Steen M (2003) Newscast computing. Technical Report IR-CS-009, Department of computer Science, Amsterdam, Netherlands.
- Jesi G P, Montresor A & Babaoglu O (2006) Proximity-aware superpeer overlay topologies. *Proc IEEE Workshop on Self-Managed Networks, Systems, and Services, Lecture Notes in Computer Science* 3996. Dublin, Ireland: 43–57.
- Jin H, Ning X & Chen H (2006) Efficient search for peer-to-peer information retrieval using semantic small world. *Proc International conference on World Wide Web*. Dinburgh, Scotland: 1003–1004.
- Joung Y-J & Wang J-C (2007) Chord2: A two-layer Chord for reducing maintenance overhead via heterogeneity. *Computer Networks* 51(3): 712–731.

- Kalogeraki V, Gunopulos D & Zeinalipour-Yazti D (2002) A local search mechanism for peer-to-peer networks. Proc International Conference on Information and Knowledge Management. McLean, VA, USA: 300–307.
- Kazaa (2009) URL: <http://www.kazaa.com/us/help/glossary/p2p-.htm> (last accessed 17-12-2009).
- Kleis M, Lua E K & Zhou X M (2005) Hierarchical peer-to-peer networks using lightweight SuperPeer topologies. Proc IEEE Symposium on Computers and Communications. Murcia, Spain: 143–148.
- Klingberg T & Manfredi R (2002) The gnutella protocol 0.6, [http://rfc-gnutella.sourceforge.net/src/rfc-0\\_6-draft.html](http://rfc-gnutella.sourceforge.net/src/rfc-0_6-draft.html). (last accessed 12-12-2013).
- Komorita S, Ito M, Yokota H, Makaya C, Falchuk B, Chee D & Das S (2012) Loosely coupled service composition for deployment of next generation service overlay networks. IEEE Communications Magazine. 50(1): 62–72.
- Kortuem G, Kawsar F, Fitton D & Sundramoorthy V (2010) Smart Objects as Building Blocks for the Internet of Things. IEEE Internet Computing 14(1): 44–51.
- Kulbak Y & Bickson D (2005) The eMule protocol specification, eMule project, <https://gnunet.org/node/951> (last accessed 12-02-2014).
- Kurve A, Griffin C, Miller D J & Kesidis G (2013) Optimizing cluster formation in super-peer networks via local incentive design. Peer-to-Peer Networking and Applications: 10.1007/s12083-12013-10206-12086.
- Kwong K-W & Tsang H (2008) Building heterogeneous peer-to-peer networks: protocol and analysis. IEEE/ACM Transactions on Networking (TON) 16(2): 281–292.
- Labovitz C, Iekel-Johnson S, McPherson D, Oberheide J & Jahanian F (2010) Internet Inter-Domain Traffic. Computer Communication Review 40(4): 75–86.
- Lanham N & Shenker S (2003) Making Gnutella-like P2P systems scalable. Computer Communication Review 33(4): 407–418.
- Lee H S, Lim K H & Kim S J (2013) A configuration scheme for connectivity-aware mobile P2P networks for efficient mobile cloud-based video streaming services. Cluster Computing 16(4): 745–756.
- Lee P, Jayasumana A P, Bandara H M N D, Lim S & Chandrasekar V (2012) A peer-to-peer collaboration framework for multi-sensor data fusion. Journal of Network and Computer Applications 35(3): 1052–1066.
- Leonard D, Yao Z M, Rai V & Loguinov D (2007) On lifetime-based node failure and stochastic resilience of decentralized peer-to-peer networks. IEEE-ACM Transactions on Networking 15(3): 644–656.
- Leppanen T, Ylioja J, Nrhi P, Rty T, Ojala T & Rieki J (2012) Holistic energy consumption monitoring in buildings with IP-based wireless sensor networks. Proc ACM Workshop on Embedded Sensing Systems for Energy-Efficiency in Buildings. Toronto, Canada: 195–196.
- Li C L & Li L Y (2009) Three-layer control policy for grid resource management. Journal of Network and Computer Applications 32(3): 525–537.

- Li J S & Chao C H (2010) An Efficient Superpeer Overlay Construction and Broadcasting Scheme Based on Perfect Difference Graph. *IEEE Transactions on Parallel and Distributed Systems* 21(5): 594–606.
- Li M, Lee W-C & Sivasubramaniam A (2004) Semantic small world: An overlay network for peer-to-peer search. *Proc IEEE Conference on Network Protocols*. Berlin, Germany: 228–238.
- Li Y, Huang X, Ma F & Zou F (2005) Building Efficient Super-Peer Overlay Network for DHT Systems. *Proc International Conference on Grid and Cooperative Computing*, *Lecture Notes in Computer Science* 3795. Beijing, China: 787–798.
- Liang J, Kumar R & Ross K W (2004) The Kazz Overlay: A measurement study. *Proc IEEE Annual Computer Communications Workshop*. Florida, USA.
- Liang W, Bi J P, Wu R, Li Z Y & Li C (2009) On Characterizing PPStream: Measurement and Analysis of P2P IPTV under Large-Scale Broadcasting. *Proc IEEE Global Telecommunications Conference*. Hawaii, USA: 3552–3557.
- Liu D, Li F & Chen S (2009) Towards optimal resource utilization in heterogeneous P2P streaming. *Proc IEEE International Conference on Distributed Computing Systems*. Montreal, Canada: 352–359.
- Liu L, Antonopoulos N & Mackin S (2007) Fault-tolerant peer-to-peer search on small-world networks. *Future Generation Computer Systems* 23(8): 921–931.
- Loo B T, Huebsch R, Stoica I & Hellerstein J M (2005) The case for a hybrid P2P search infrastructure. *Proc International Workshop on Peer-to-Peer Systems (IPTPS)*, *Lecture Notes in Computer Science* 3279. California, USA: 141–150.
- Löser A, Naumann F, Siberski W, Nejd W & Thaden U (2004) Semantic overlay clusters within super-peer networks. *Proc International Workshop on Databases, Information Systems, and Peer-to-Peer Computing (DBISP2P)*, *Lecture Notes in Computer Science* 2994. Toronto, Canada: 33–47.
- Lua E K, Crowcroft J, Pias M, Sharma R & Lim S (2005) A Survey and Comparison of Peer-to-Peer Overlay Network Schemes. *IEEE Communications Surveys and Tutorials* 7(2): 72–93.
- Luckenbach T, Gober P, Arbanowski S, Kotsopoulos A & Kim K (2005) TinyREST-a protocol for integrating sensor networks into the internet. *Proc International Workshop on Real-World Wireless Sensor Network: REALWSN*. Stockholm, Sweden.
- Lv Q, Cao P, Cohen E, Li K & Shenker S (2002) Search and replication in unstructured peer-to-peer networks. *Proc ACM International Conference on Supercomputing*. New York, USA: 84–95.
- Madden S, Franklin M J, Hellerstein J M & Hong W (2002) TAG: A tiny aggregation service for ad-hoc sensor networks. *ACM SIGOPS Operating Systems Review* 36(SI): 131–146.
- Madden S R, Franklin M J, Hellerstein J M & Hong W (2005) TinyDB: An acquisitional query processing system for sensor networks. *ACM Transactions on Database Systems* 30(1): 122–173.

- Mäenpää J, Bolonio J J & Loreto S (2012) Using RELOAD and CoAP for wide area sensor and actuator networking. *EURASIP Journal on Wireless Communications and Networking* 2012(1): 1–22.
- Mastroianni C, Talia D & Verta O (2005) A super-peer model for resource discovery services in large-scale grids. *Future Generation Computer Systems* 21(8): 1235–1248.
- Maymounkov P & Mazieres D (2002) Kademlia: A peer-to-peer information system based on the XOR metric. *Proc International Workshop on Peer-to-Peer Systems (IPTPS)*, Lecture Notes in Computer Science 2429. Massachusetts, USA: 53–65.
- Meland P H, Guerenabarrena J B & Llewellyn-Jones D (2011) The challenges of secure and trustworthy service composition in the future internet. *Proc International Conference on System of Systems Engineering*. New Mexico, USA: 329–334.
- Merz P, Ubben J & Priebe M (2008) On the Construction of a Super-Peer Topology underneath Middleware for Distributed Computing. *Proc IEEE International Symposium on Cluster Computing and the Grid*. Lyon, France: 590–595.
- Meshkova E, Riihijärvi J, Petrova M & Mähönen P (2008) A survey on resource discovery mechanisms, peer-to-peer and service discovery frameworks. *Computer Networks* 52(11): 2097–2128.
- Miers C, Simplício M, Gallo D, Carvalho T, Bressan G, Souza V, Karlsson P & Damola A (2010) I2TS01-A Taxonomy for Locality Algorithms on Peer-to-Peer Networks. *IEEE (Revista IEEE America Latina) Latin America Transactions* 8(4): 323–331.
- Miller B A, Nixon T, Tai C & Wood M D (2001) Home networking with Universal Plug and Play. *IEEE Communications Magazine* 39(12): 104–109.
- Min S-H, Holliday J & Cho D-S (2006) Optimal super-peer selection for large-scale p2p system. *Proc International Conference on Hybrid Information Technology*. Jeju Island, Korea: 588–593.
- Mizrak A, Cheng Y, Kumar V & Savage S (2003) Structured superpeers: leveraging heterogeneity to provide constant-time lookup. *Proc IEEE Workshop on Internet Applications (WIAPP)*. California, USA: 104–111.
- Mockapetris P V & Dunlap K (1988) Development of the Domain Name System. *Proc ACM symposium on communications architectures and protocol (SIGCOMM)*. California, USA: 123–133.
- Mondal A, Trestian I, Qin Z & Kuzmanovic A (2012) P2P as a CDN: A new service model for file sharing. *Computer Networks* 56(14): 3233–3246.
- Montresor A (2004) A robust protocol for building superpeer overlay topologies. *Proc Fourth International Conference on Peer-to-Peer Computing*. Zurich, Switzerland: 202–209.
- Montresor A & Jelasity M (2009) PeerSim: A Scalable P2P Simulator. *Proc IEEE International Conference on Peer-to-Peer Computing*. Washington, USA: 99–100.
- Moya J (2009) STUDY: P2P Still Generates Most Traffic Worldwide <[http://www.zeropaaid.com/news/10021/study\\_p2p\\_still\\_generates\\_most\\_traffic\\_worldwide/](http://www.zeropaaid.com/news/10021/study_p2p_still_generates_most_traffic_worldwide/)>.(last accessed 04–02-2014).

- Mueller R, Alonso G & Kossmann D (2007) SwissQM: Next generation data processing in sensor networks. Proc Biennial Conference on Innovative Data Systems Research. California, USA: 1–9.
- Naicken S, Livingston B, Basu A, Rodhetbhai S, Wakeman I & Chalmers D (2007) The state of peer-to-peer simulators and simulations. ACM SIGCOMM Computer Communication Review 37(2): 95–98.
- Nejdl W, Wolpers M, Siberski W, Schmitz C, Schlosser M, Brunkhorst I & Löser A (2003) Super-peer-based routing and clustering strategies for RDF-based peer-to-peer networks. Proc International Conference on World Wide Web. Budapest, HUNGARY: 536–543.
- Oliveira L B, Siqueira I G & Loureiro A A (2005) On the performance of ad hoc routing protocols under a peer-to-peer application. Journal of Parallel and Distributed Computing 65(11): 1337–1347.
- Ou Z, Harjula E, Kassinen O & Ylianttila M (2009) Feasibility evaluation of a communication-oriented P2P system in mobile environments. Proc International Conference on Mobile Technology, Application & Systems. Nice, France 10.1145/1710035.1710078.
- Ou Z, Harjula E, Koskela T & Ylianttila M (2010) GTPP: General truncated pyramid peer-to-peer architecture over structured DHT networks. Mobile Networks and Applications 15(5): 729–749.
- Pacifici G, Spreitzer M, Tantawi A N & Youssef A (2005) Performance management for cluster-based web services. IEEE Journal on Selected Areas in Communications 23(12): 2333–2343.
- Papazoglou M P, Traverso P, Dustdar S & Leymann F (2007) Service-oriented computing: State of the art and research challenges. Computer 40(11): 38–45.
- Parashar M, Liu H, Li Z, Matossian V, Schmidt C, Zhang G & Hariri S (2006) AutoMate: enabling autonomic applications on the Grid. Cluster Computing—the Journal of Networks Software Tools and Applications 9(2): 161–174.
- Piatek M, Madhyastha H V, John J P, Krishnamurthy A & Anderson T (2009) Pitfalls for ISP-friendly p2p design. Proc ACM Workshop on Hot Topics in Networks (HotNets). New York, USA.
- PPLive (2009) PPLive <http://www.pplive.com> (last accessed 17-12-2009).
- Puppin D, Moncelli S, Baraglia R, Tonello N & Silvestri F (2005) A grid information service based on Peer-to-Peer. Proc European conference on Parallel Processing, Lecture Notes in Computer Science 3648. Caparica, Portugal: 454–464.
- Pyun Y J & Reeves D S (2004) Constructing a balanced (log(N)/loglog(N))-Diameter super-peer topology for scalable P2P systems. Proc International Conference on Peer-to-Peer Computing. Zurich, Switzerland: 210–218.
- Qiao B, Wang G & Xie K (2007) A taxonomy-based approach for constructing semantics-based super-peer networks. Proc International Workshops on Advances in Web and Network Technologies, and Information Management, Lecture Notes in Computer Science 4537. HuangShan, China: 122–134.

- Quiroz A, Kim H, Parashar M, Gnanasambandam N & Sharma N (2009) Towards autonomic workload provisioning for enterprise grids and clouds. Proc IEEE/ACM International Conference on Grid Computing. Alberta, Canada: 50–57.
- Rahman M, Ranjan R & Buyya R (2010) Reputation-based dependable scheduling of workflow applications in Peer-to-Peer Grids. *Computer Networks* 54(18): 3341–3359.
- Ranjan R, Rahman M & Buyya R (2008) A Decentralized and Cooperative Workflow Scheduling Algorithm. Proc IEEE International Symposium on Cluster Computing and the Grid. Lyon, France: 1–8.
- Ranjan R & Rajkumar B (2010) Decentralized Overlay for Federation of Enterprise Clouds. *Handbook of Research on Scalable Computing Technologies*. IGI Global: 191–217.
- Ranjan R & Zhao L (2013) Peer-to-peer service provisioning in cloud computing environments. *Journal of Supercomputing* 65(1): 154–184.
- Ranjan R, Zhao L, Wu X, Liu A, Quiroz A & Parashar M (2010) Peer-to-Peer Cloud Provisioning: Service Discovery and Load-Balancing. *Cloud Computing*. Springer, London: 195–217.
- Rasti A H, Stutzbach D & Rejaie R (2006) On the long-term evolution of the two-tier gnutella overlay. Proc IEEE International Conference on Computer Communications (INFOCOM). Catalunya, Spain: 1–6.
- Ratnasamy S, Handley M, Karp R & Shenker S (2002) Topologically-aware overlay construction and server selection. Proc IEEE Conference on Computer Communications (INFOCOM). New York, USA: 1190–1199.
- Ripeanu M (2001) Peer-to-peer architecture case study: Gnutella network. Proc International Conference on Peer-to-Peer Computing. Linkoping, Sweden: 99–100.
- Rowstron A & Druschel P (2001) Pastry: Scalable, decentralized object location, and routing for large-scale peer-to-peer systems. Proc IFIP/ACM International Conference on Distributed Systems Platforms (Middleware), Lecture Notes in Computer Science 2218. Heidelberg, Germany: 329–350.
- Russel S & Norvig P (2003) *Artificial Intelligence A Modern Approach*. Upper Saddle River, NJ, USA, Pearson Education.
- Saint-Andre P (2007) Jingle: Jabber does multimedia. *IEEE Multimedia* 14(1): 90–94.
- Saroiu S, Gummadi P K & Gribble S D (2002) Measurement study of peer-to-peer file sharing systems. Proc SPIE/ACM conference on Multimedia Computing and Networking. California, USA: 156–170.
- Schramm P, Naroska E, Resch P, Platte J, Linde H, Stromberg G & Sturm T (2004) A service gateway for networked sensor systems. *IEEE Pervasive Computing* 3(1): 66–74.
- Sermersheim J (2006) Lightweight directory access protocol (LDAP): The protocol, <http://tools.ietf.org/html/rfc4511>. (last accessed 12-02-2014).
- Shelby Z, Bormann C & Kroc S (2012) CoRE Resource Directory, <http://datatracker.ietf.org/doc/draft-shelby-core-resource-directory/>, Work in Progress Internet Draft, [Expires September 13, 2012]

- Shelby Z, Hartke K, Bormann C & B. Frank (2013) Constrained Application Protocol (CoAP), [https://datatracker.ietf.org/doc/draft-ietf-core-coap/Internet Engineering Task Force, Work in Progress Internet Draft, \[Expires June 9, 2013\]](https://datatracker.ietf.org/doc/draft-ietf-core-coap/Internet%20Engineering%20Task%20Force,%20Work%20in%20Progress%20Internet%20Draft,%20[Expires%20June%209,%202013].).
- Shelby Z, Mahonen R, Riihijarvi J, Raivio O & Huuskonen P (2003) NanoIP: The Zen of embedded networking. Proc IEEE International Conference on Communications. Alaska, USA: 1218–1222.
- Shen H Y & Xu C Z (2007) Locality-aware and churn-resilient load-balancing algorithms in structured peer-to-peer networks. IEEE Transactions on Parallel and Distributed Systems 18(6): 849–862.
- Sioutas S, Sakkopoulos E, Drossos L & Sirmakessis S (2008) Balanced distributed web service lookup system. Journal of Network and Computer Applications 31(2): 149–162.
- Stauffer A O & Barbosa V C (2007) Probabilistic heuristics for disseminating information in networks. IEEE/ACM Transactions on Networking 15(2): 425–435.
- Stoica I, Adkins D, Zhuang S, Shenker S & Surana S (2004) Internet indirection infrastructure. IEEE/ACM Transactions on Networking 12(2): 205–218.
- Stoica I, Morris R, Liben-Nowell D, Karger D R, Kaashoek M F, Dabek F & Balakrishnan H (2003) Chord: A scalable peer-to-peer lookup protocol for Internet applications. IEEE/ACM Transactions on Networking 11(1): 17–32.
- Stutzbach D & Rejaie R (2005) Characterizing the two-tier gnutella topology. ACM SIGMETRICS Performance Evaluation Review 33(1): 402–403.
- Stutzbach D & Rejaie R (2006) Understanding churn in peer-to-peer networks. Proc ACM SIGCOMM conference on Internet measurement. Rio de Janeiro, Brazil: 189–202.
- Tan Y-H, Lü K & Lin Y-P (2012) Organisation and management of shared documents in super-peer networks based semantic hierarchical cluster trees. Peer-to-Peer Networking and Applications 5(3): 292–308.
- Tanganelli G, Mingozzi E, Vallati C & Cicconetti C (2013) A Distributed Architecture for Discovery and Access in the Internet of Things. IEEE International Conference on Computer Communications (INFOCOM) Demo. Turin, Italy.
- Tanin E, Harwood A & Samet H (2007) Using a distributed quadtree index in peer-to-peer networks. The VLDB Journal (International Journal on Very Large Data Bases) 16(2): 165–178.
- Tarkoma S (2010) Overlay Networks: Toward Information Networking. Auerbach Publications, New York, USA.
- Tarkoma S & Ailisto H (2013) The internet of things program: the finnish perspective. IEEE Communications Magazine 51(3): 10–11.
- Tempich C, Staab S & Wranik A (2004) Remindin': semantic query routing in peer-to-peer networks based on social metaphors. Proc International Conference on World Wide Web. New York, USA: 640–649.
- Teng H Y, Lin C N & Hwang R H (2011) SSNG: A Self-Similar Super-Peer Overlay Construction Scheme for Super Large-Scale P2P Systems. Proc IEEE Conference on Parallel and Distributed Systems. Tainan, Taiwan: 782–787.



- Tian R X, Xiong Y Q, Zhang Q, Li B, Zhao B Y & Li X (2005) Hybrid overlay structure based on random walks. Proc International Workshop on Peer-to-Peer Systems IV, Lecture Notes in Computer Science 3640. NewYork, USA: 152–162.
- Tsoumakos D & Roussopoulos N (2003) A comparison of peer-to-peer search methods. Proc International Workshop on the Web and Databases (WebDB). California, USA: 61–66.
- Tutschku K (2004) A Measurement-Based Traffic Profile of the eDonkey Filesharing Service. Proc International Workshop on Passive and Active Network Measurement, Lecture Notes in Computer Science 3015. Antibes Juan-les-Pins, France: 12–21.
- Valtchev D & Frankov I (2002) Service gateway architecture for a smart home. IEEE Communications Magazine 40(4): 126–132.
- Vanhournout K, Deconinck G & Belmans R (2005) A taxonomy for resource discovery. Personal and Ubiquitous Computing 9(2): 81–89.
- Varia J (2008) Cloud architectures, White Paper of Amazon, [aws.amazon.com/public/cloudarchitectures-varia.pdf](http://aws.amazon.com/public/cloudarchitectures-varia.pdf) (last accessed 14-02-2014).
- Vishnumurthy V & Francis P (2006) On heterogeneous overlay construction and random node selection in unstructured P2P networks. Proc IEEE International Conference on Computer Communications (INFOCOM). Catalunya, Spain: 1119–1130.
- Voulgaris S, Gavidia D & Van Steen M (2005) Cyclon: Inexpensive membership management for unstructured P2P overlays. Journal of Network and Systems Management 13(2): 197–217.
- Waldo J (1999) The Jini architecture for network-centric computing. Communications of the ACM 42(7): 76–82.
- Wang F, Liu J C & Xiong Y Q (2008) Stable peers: Existence, importance, and application in peer-to-peer live video streaming. Proc IEEE Conference on Computer Communications (INFOCOM). Arizona, USA: 2038–2046.
- Windows azure platform (2010) Windows azure platform <http://www.microsoft.com/azure/> (Accessed in 2010).
- Wolff R & Schuster A (2004) Association rule mining in peer-to-peer systems. IEEE Transactions on Systems Man and Cybernetics Part B-Cybernetics 34(6): 2426–2438.
- Wouhaybi R H & Campbell A T (2004) Phenix: Supporting resilient low-diameter peer-to-peer topologies. Proc IEEE Computer and Communications Societies. Hong Kong, China: 108–119.
- Xiao L, Zhuang Z Y & Liu Y H (2005) Dynamic layer management in superpeer architectures. IEEE Transactions on Parallel and Distributed Systems 16(11): 1078–1091.
- Xu Z, Min R & Hu Y (2003) HIERAS: a DHT based hierarchical P2P routing algorithm. Proc International Conference on Parallel Processing. Kaohsiung, Taiwan: 187–194.
- Yan Z & Zhang P (2006) Collaboration in P2P Systems Based on Trusted Computing Platforms. WSEAS transactions on Information Science and Applications 3(2): 275–282.
- Yang B & Garcia-Molina H (2003) Designing a super-peer network. Proc 19th International Conference on Data Engineering. Bangalore, India: 49–60.

- Yang M & Yang Y Y (2010) An Efficient Hybrid Peer-to-Peer System for Distributed Data Sharing. *IEEE Transactions on Computers* 59(9): 1158–1171.
- Yao Y & Gehrke J (2002) The cougar approach to in-network query processing in sensor networks. *ACM SIGMOD Record* 31(3): 9–18.
- Yu Y, Lee S & Zhang Z-L (2005) Leopard: A locality aware peer-to-peer system with no hot spot. *Proc International IFTP-TC6 Networking Technologies, Services, and Protocols; Performance of Computer and Communication Networks; Mobile and Wireless Communications Systems, Lecture Notes in Computer Science* 3462. Waterloo, Canada: 27–39.
- Zahariev A (2009) Google app engine. Helsinki University of Technology.
- Zhang X & Li J (2010) VLBI data transfer over fast long-distance hybrid network. *Proc IEEE International Conference on Computer Science and Information Technology (ICCSIT)*. Chengdu, China: 491–496.
- Zhang X Y, Zhang Q, Zhang Z S, Song G & Zhu W W (2004) A construction of locality-aware overlay network: mOverlay and its performance. *IEEE Journal on Selected Areas in Communications* 22(1): 18–28.
- Zhang Y F, Chen C J & Wang X P (2006) Recent advances in research on P2P networks. *Proc International Conference on Parallel and Distributed Computing, Applications and Technologies*. Lasvegas, USA: 278–284.
- Zhao B Y, Huang L, Stribling J, Rhea S C, Joseph A D & Kubiatowicz J D (2004) Tapestry: A resilient global-scale overlay for service deployment. *IEEE Journal on Selected Areas in Communications* 22(1): 41–53.
- Zhen L & Parashar M (2005) Comet: a scalable coordination space for decentralized distributed environments. *Proc International Workshop on Hot Topics in Peer-to-Peer Systems*. California, USA: 104–111.
- Zoels S, Despotovic Z & Kellerer W (2008) On hierarchical DHT systems-An analytical approach for optimal designs. *Computer Communications* 31(3): 576–590.

## Original publications

- I Liu M, Koskela T, Ou Z, Zhou J, Riekkki J & Ylianttila M (2011) Super-Peer-based Coordinated Service Provision. *Journal of Network and Computer Applications* 34 (4): 1210–1224.
- II Liu M, Koskela T & Ylianttila M (2011) An Efficient Collaborative Mechanism for Enterprise Cloud Applications. *Proc IEEE International Conference on Networking, Sensing and Control*. Delft, Netherlands: 163–168.
- III Liu M, Zhou J, Koskela T & Ylianttila M (2009) A Robust Algorithm for the Membership Management of Super-Peer Overlay. *Proc IFIP/IEEE International Conference on Management of Multimedia and Mobile Networks and Services, Springer Lecture Notes in Computer Science* 5842. Venice, Italy: 132–143.
- IV Liu M, Harjula E & Ylianttila M (2013) An efficient selection algorithm for building a super-peer overlay. *Journal of Internet Services and Applications* 4(4): 1–12.
- V Liu M, Leppänen T, Harjula E, Ou Z, Ramalingam A, Ylianttila M & Ojala T (2013) Distributed resource directory architecture in Machine-to-Machine communications. *Proc IEEE International conference on Wireless and mobile computing, networking and communication Workshop on the Internet of Things Communications and Technologies*. Lyon, France: 319–324.

Reprinted with permission from Elsevier (I), IEEE (II, V) and Springer (III, IV).

Original publications are not included in the electronic version of the thesis.



470. Czajkowski, Jakub (2013) Optical coherence tomography as a characterization method in printed electronics
471. Haapalainen, Mikko (2013) Dielectrophoretic mobility of a spherical particle in 2D hyperbolic quadrupole electrode geometry
472. Bene, József Gergely (2013) Pump schedule optimisation techniques for water distribution systems
473. Seelam, Prem Kumar (2013) Hydrogen production by steam reforming of bio-alcohols : the use of conventional and membrane-assisted catalytic reactors
474. Komulainen, Petri (2013) Coordinated multi-antenna techniques for cellular networks : Pilot signaling and decentralized optimization in TDD mode
475. Piltonen, Petteri (2013) Prevention of fouling on paper machine surfaces
476. Juuso, Esko (2013) Integration of intelligent systems in development of smart adaptive systems : linguistic equation approach
477. Lu, Xiaojia (2013) Resource allocation in uplink coordinated multicell MIMO-OFDM systems with 3D channel models
478. Jung, Sang-Joong (2013) Personal machine-to-machine (M2M) healthcare system with mobile device in global networks
479. Haho, Päivi (2014) Learning enablers, learning outcomes, learning paths, and their relationships in organizational learning and change
480. Ukkonen, Kaisa (2014) Improvement of recombinant protein production in shaken cultures : focus on aeration and enzyme-controlled glucose feeding
481. Peschl, Michael (2014) An architecture for flexible manufacturing systems based on task-driven agents
482. Kangas, Jani (2014) Separation process modelling : highlighting the predictive capabilities of the models and the robustness of the solving strategies
483. Kemppainen, Kalle (2014) Towards simplified deinking systems : a study of the effects of ageing, pre-wetting and alternative pulping strategy on ink behaviour in pulping
484. Mäklin, Jani (2014) Electrical and thermal applications of carbon nanotube films
485. Niemistö, Johanna (2014) Towards sustainable and efficient biofuels production : Use of pervaporation in product recovery and purification

S E R I E S E D I T O R S

**A**  
**SCIENTIAE RERUM NATURALIUM**

*Professor Esa Hohtola*

**B**  
**HUMANIORA**

*University Lecturer Santeri Palviainen*

**C**  
**TECHNICA**

*Postdoctoral research fellow Sanna Taskila*

**D**  
**MEDICA**

*Professor Olli Vuolteenaho*

**E**  
**SCIENTIAE RERUM SOCIALIUM**

*University Lecturer Veli-Matti Ulvinen*

**F**  
**SCRIPTA ACADEMICA**

*Director Sinikka Eskelinen*

**G**  
**OECONOMICA**

*Professor Jari Juga*

**EDITOR IN CHIEF**

*Professor Olli Vuolteenaho*

**PUBLICATIONS EDITOR**

*Publications Editor Kirsti Nurkkala*

ISBN 978-952-62-0399-7 (Paperback)

ISBN 978-952-62-0400-0 (PDF)

ISSN 0355-3213 (Print)

ISSN 1796-2226 (Online)

