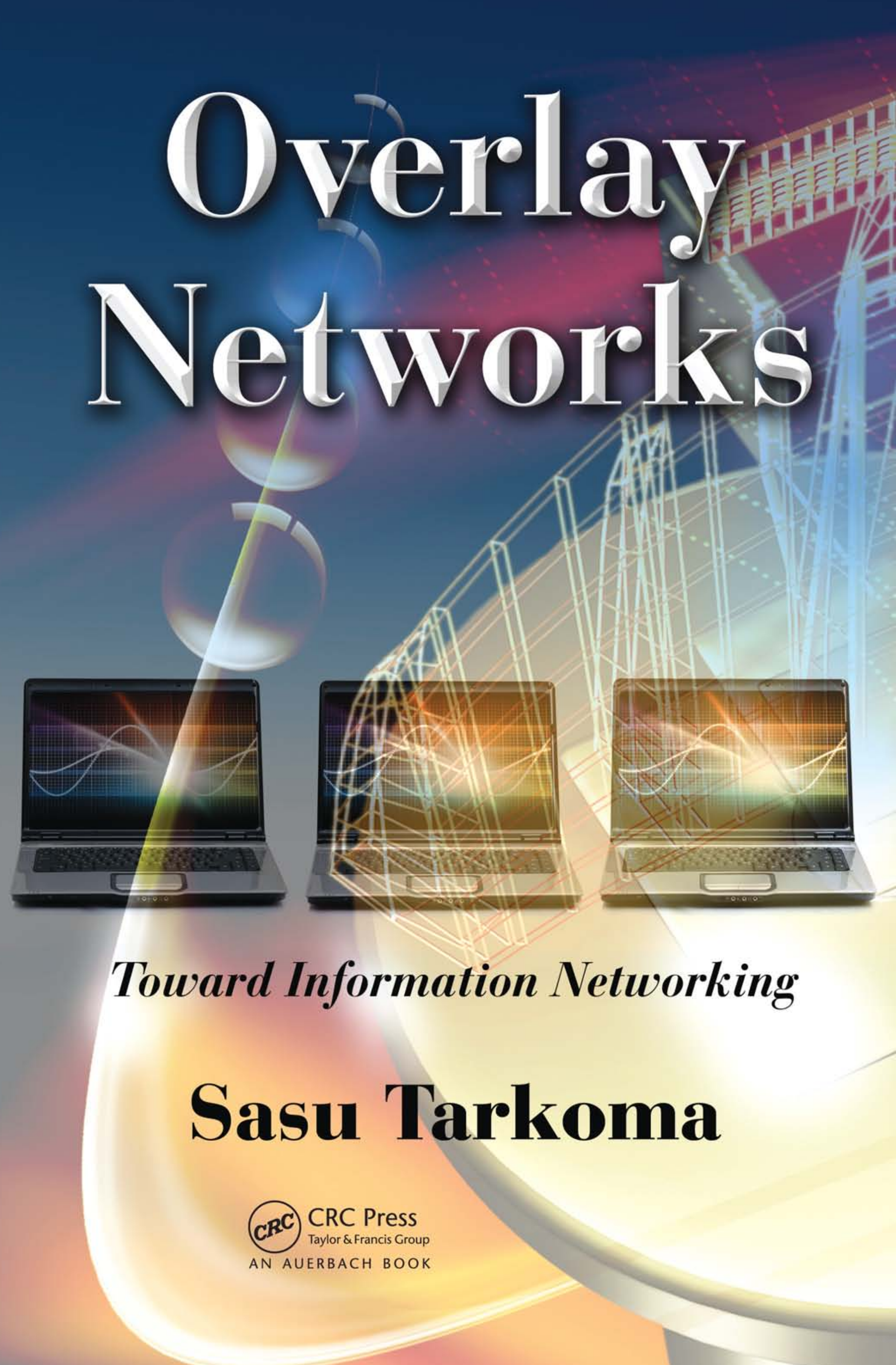

Overlay Networks

Toward Information Networking

Sasu Tarkoma

CRC Press
Taylor & Francis Group
AN AUERBACH BOOK

Overlay Networks

Toward Information Networking

OTHER TELECOMMUNICATIONS BOOKS FROM AUERBACH

Broadband Mobile Multimedia: Techniques and Applications
Yan Zhang, Shiwen Mao, Laurence T. Yang, and Thomas M Chen
ISBN: 978-1-4200-5184-1

Carrier Ethernet: Providing the Need for Speed
Gilbert Held
ISBN: 978-1-4200-6039-3

Cognitive Radio Networks
Yang Xiao and Fei Hu
ISBN: 978-1-4200-6420-9

Contemporary Coding Techniques and Applications for MobileCommunications
Onur Osman and Osman Nuri Ucan
ISBN: 978-1-4200-5461-3

Converging NGN Wireline and Mobile 3G Networks with IMS: Converging NGN and 3G Mobile
Rebecca Copeland
ISBN: 978-0-8493-9250-4

Cooperative Wireless Communications
Yan Zhang, Hsiao-Hwa Chen, and Mohsen Guizani
ISBN: 978-1-4200-6469-8

Data Scheduling and Transmission Strategies in Asymmetric Telecommunication Environments
Abhishek Roy and Navrati Saxena
ISBN: 978-1-4200-4655-7

Encyclopedia of Wireless and Mobile Communications
Borko Furht
ISBN: 978-1-4200-4326-6

IMS: A New Model for Blending Applications
Mark Wuthnow, Jerry Shih, and Matthew Stafford
ISBN: 978-1-4200-9285-1

The Internet of Things: From RFID to the Next-Generation Pervasive Networked Systems
Lu Yan, Yan Zhang, Laurence T. Yang, and Huansheng Ning
ISBN: 978-1-4200-5281-7

Introduction to Communications Technologies: A Guide for Non-Engineers, Second Edition
Stephan Jones, Ron Kovac, and Frank M. Groom
ISBN: 978-1-4200-4684-7

Long Term Evolution: 3GPP LTE Radio and Cellular Technology
Borko Furht and Syed A. Ahson
ISBN: 978-1-4200-7210-5

MEMS and Nanotechnology-Based Sensors and Devices for Communications, Medical and Aerospace Applications
A. R. Jha
ISBN: 978-0-8493-8069-3

Millimeter Wave Technology in Wireless PAN, LAN, and MAN
Shao-Qiu Xiao and Ming-Tuo Zhou
ISBN: 978-0-8493-8227-7

Mobile Telemedicine: A Computing and Networking Perspective
Yang Xiao and Hui Chen
ISBN: 978-1-4200-6046-1

Optical Wireless Communications: IR for Wireless Connectivity
Roberto Ramirez-Iniguez, Sevia M. Idrus, and Ziran Sun
ISBN: 978-0-8493-7209-4

Satellite Systems Engineering in an IPv6 Environment
Daniel Minoli
ISBN: 978-1-4200-7868-8

Security in RFID and Sensor Networks
Yan Zhang and Paris Kitsos
ISBN: 978-1-4200-6839-9

Security of Mobile Communications
Noureddine Boudriga
ISBN: 978-0-8493-7941-3

Unlicensed Mobile Access Technology: Protocols, Architectures, Security, Standards and Applications
Yan Zhang, Laurence T. Yang, and Jianhua Ma
ISBN: 978-1-4200-5537-5

Value-Added Services for Next Generation Networks
Thierry Van de Velde
ISBN: 978-0-8493-7318-3

Vehicular Networks: Techniques, Standards, and Applications
Hassnaa Moustafa and Yan Zhang
ISBN: 978-1-4200-8571-6

WiMAX Network Planning and Optimization
Yan Zhang
ISBN: 978-1-4200-6662-3

Wireless Quality of Service: Techniques, Standards, and Applications
Maode Ma and Mieso K. Denko
ISBN: 978-1-4200-5130-8

AUERBACH PUBLICATIONS
www.auerbach-publications.com
To Order Call: 1-800-272-7737 • Fax: 1-800-374-3401
E-mail: orders@crcpress.com

Overlay Networks

Toward Information Networking

Sasu Tarkoma

CRC Press
Taylor & Francis Group
Boca Raton London New York

CRC Press is an imprint of the
Taylor & Francis Group, an **informa** business
AN AUERBACH BOOK

CRC Press
Taylor & Francis Group
6000 Broken Sound Parkway NW, Suite 300
Boca Raton, FL 33487-2742

Version Date: 20140514

International Standard Book Number-13: 978-1-4398-1373-7 (eBook - PDF)

Visit the Taylor & Francis Web site at
http://www.taylorandfrancis.com

and the CRC Press Web site at
http://www.crcpress.com

Contents

Preface

Data and media delivery have become hugely popular on the Internet, with well over 1 billion Internet users. Therefore scalable and flexible information dissemination solutions are needed. Much of the current development pertaining to services and service delivery happens above the basic network layer and the TCP/IP protocol suite because of the need to be able to rapidly develop and deploy them.

In recent years, various kinds of overlay networking technologies have emerged as an active area of research and development. Overlay systems, especially *peer-to-peer* systems, are technologies that can solve problems in massive information distribution and processing tasks. The key aim of many of these technologies is to be able to offer deployable solution for processing and distributing vast amounts of information, typically petabytes and more, while at the same time keeping the scaling costs low.

The aim of this book is to present the state of the art in overlay technologies, examine the key structures and algorithms used in overlay networks, and discuss their applications. Overlay networks have been a very active area of research and development during the last 10 years, and a substantial amount of scientific literature has formed around this topic.

This book has been inspired by the teaching notes and articles of the author in content-based routing. The book is designed not only as a reference for overlay technologies, but also as a textbook for a course in distributed overlay technologies and information networking at the graduate level.

About the Author

Sasu Tarkoma received his M.Sc. and Ph.D. degrees in Computer Science from the University of Helsinki, Department of Computer Science. He is currently professor at Helsinki University of Technology, Department of Computer Science and Engineering. He has been recently appointed as full professor at University of Helsinki, Department of Computer Science. He has managed and participated in national and international research projects at the University of Helsinki, Helsinki University of Technology, and Helsinki Institute for Information Technology (HIIT). He has worked in the IT industry as a consultant and chief system architect, and he is principal member of research staff at Nokia Research Center. He has over 100 publications, and has also contributed to several books on mobile middleware.

Ms. Nelli Tarkoma produced most of the diagrams used in this book.

1

Introduction

1.1 Overview

In recent years, various kinds of overlay networking technologies have emerged as an active area of research and development. Overlay systems, especially *peer-to-peer (P2P)* systems, are technologies that can solve problems in massive information distribution and processing tasks. The key aim of many of these technologies is to be able to offer deployable solution for processing and distributing vast amounts of information, typically petabytes and more, while at the same time keeping the scaling costs low.

Data and media delivery have become hugely popular on the Internet. Currently there are over 1.4 billion Internet users, well over 3 billion mobile phones, and 4 billion mobile subscriptions. By 2000 the Google index reached the 1 billion indexed web resources mark, and by 2008 it reached the trillion mark.

Multimedia content, especially videos, are paving the way for truly versatile network services that both compete with and extend existing broadcast-based medias. As a consequence, new kinds of social collaboration and advertisement mechanisms are being introduced both in the fixed Internet and also in the mobile world. This trend is heightened by the ubiquitous nature of digital cameras. Indeed, this has created a lot of interest in community-based services, in which users create their own content and make it available to others.

These developments have had a profound impact on network requirements and performance. Video delivery has become one of the recent services on the Web with the advent of YouTube [67] and other social media Web sites. Moreover, the network impact is heightened by various P2P services. Estimates of P2P share of network traffic range from 50% to 70%. Cisco's latest traffic forecast for 2009–2013 indicates that annual global IP traffic will reach 667 exabytes in 2013, two-thirds of a zettabyte [79]. An exabyte (EB) is an SI unit of information, and 1 EB equals 10^{18} bytes. Exabyte is followed by the zettabyte ($1\ Z = 10^{21}$) and yottabyte ($1\ Y = 10^{24}$). The traffic is expected to increase some 40% each year. Much of this increase comes from the delivery of video data in various forms. Video delivery on the Internet will see a huge increase, and the volume of video delivery in 2013 is expected to be 700 times the capacity of the US Internet backbone in 2000. The study anticipates that video traffic will account for 91% of all consumer traffic in 2013.

According to the study, P2P traffic will continue to grow but will become a smaller component of Internet traffic in terms of its current share. The current P2P systems in 2009 are transferring 3.3 EB data per month. The recent study indicates that the P2P share of consumer Internet traffic will drop to 20% by 2013, down from the current 50% (at the end of 2008). Even though the P2P share may drop, most video delivery solutions, accounting for much of the traffic increase, will utilize overlay technologies, which makes this area crucial for ensuring efficient and scalable services.

A P2P network consists of nodes that cooperate in order to provide services to each other. A pure P2P network consists of equal peers that are simultaneously clients and servers. The P2P model differs from the *client-server* model, where clients access services provided by logically centralized servers.

To date, P2P delivery has not been successfully combined with browser-based operation and media sites such as YouTube. Nevertheless, a number of businesses have realized the importance of scalable data delivery. For example, the game company Blizzard uses P2P technology to distribute patches for the *World of Warcraft* game. Given the heavy use of network, P2P protocols such as BitTorrent offer to reduce network load by peer-assisted data delivery. This means that peer users cooperate to transfer large files over the network.

1.2 Overlay Technology

Data structures and algorithms are central for today's data communications. We may consider circuit switching technology as an example of how information processing algorithms are vital for products and how innovation changes markets. Early telephone systems were based on manual circuit switching. Everything was done using human hands. Later systems used electromechanical devices to connect calls, but they required laborious preconfiguration of telephone numbers and had limited scalability. Modern digital circuit switching algorithms evolved from these older semiautomatic systems and optimize the number of connections in a switch. The nonblocking minimal spanning tree algorithm enabled the optimization of these automatic switches. Any algorithm used to connect millions of calls must be proven to be correct and efficient. The latest development changes the fundamentals of telephone switching, because information is forwarded as packets on a hop-by-hop basis and not via preestablished physical circuits. Today, this complex machinery enables end-to-end connectivity irrespective of time and location.

Data structures are at the heart of the Internet. Network-level routers use efficient algorithms for matching data packets to outgoing interfaces based on prefixes. Internet backbone routers have to manage 200,000 routes and more in order to route packets between systems. The matching algorithms include *suffix trees* and *ternary content addressable memories (TCAMs)* [268], which have to balance between matching efficiency and router memory. Therefore, just as with telephone switches, optimization plays a major role in the development of routers and routing systems.

The current generation of networks is being developed on top of TCP/IPs network-layer (layer 3 in the *open systems interconnection (OSI)* stack). These so-called overlay networks come in various shapes and forms. Overlays make many implementation issues easier, because network-level routers do not need to be changed. In many ways, overlay networks represent a fundamental paradigm shift compared to older technologies such as circuit switching and hierarchical routing.

Overlay networks are useful both in control and content plane scenarios. This division of traffic into control and content is typical of current telecommunications solutions such as the *session initiation protocol (SIP)*; however, this division does not exist on the current Internet as such. As control plane elements, overlays can be used to route control messages and connect different entities. As content plane elements, they can participate in data forwarding and dissemination.

> An *overlay network* is a network that is built on top of an existing network. The overlay therefore relies on the so-called *underlay* network for basic networking functions, namely routing and forwarding. Today, most overlay networks are built in the application layer on top of the TCP/IP networking suite. Overlay technologies can be used to overcome some of the limitations of the underlay, at the same time offering new routing and forwarding features without changing the routers. The nodes in an overlay network are connected via logical links that can span many physical links. A link between two overlay nodes may take several hops in the underlying network.

An overlay network therefore consists of a set of distributed nodes, typically client devices or servers, that are deployed on the Internet. The nodes are expected to meet the following requirements:

1. Support the execution of one or more distributed applications by providing infrastructure for them.
2. Participate in and support high-level routing and forwarding tasks. The overlay is expected to provide data-forwarding capabilities that are different from those that are part of the basic Internet.
3. Deploy across the Internet in such a way that third parties can participate in the organization and operation of the overlay network.

Figure 1.1 presents a layered view to overlay networks. The view starts from the underlay, the network that offers the basic primitives of sending and receiving messages (packets). The two obvious choices today are UDP and TCP as the transport layer protocols. TCP is favored due to its connection-oriented nature, congestion control, and reliability.

After the underlay layer, we have the custom routing, forwarding, rendezvous, and discovery functions of the overlay architecture. Routing pertains to the process of building and maintaining routing tables. Forwarding is the process of sending messages toward their destination, and rendezvous is a function that is used to resolve issues regarding some identifier or node—for example, by offering indirection support in the case of mobility. Discovery is an integral part of this layer and is needed to populate the routing table by discovering both physically and logically nearby neighbors.

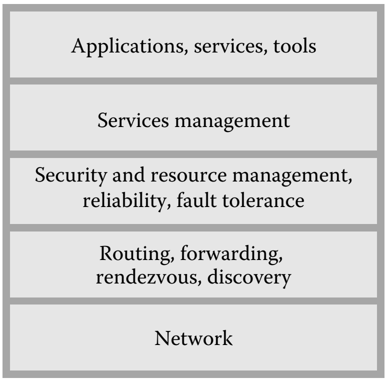
Applications, services, tools
Services management
Security and resource management,
reliability, fault tolerance
Routing, forwarding,
rendezvous, discovery
Network

FIGURE 1.1
Layered view to overlay networks.

The next layer introduces additional functions, such as security and resource management, reliability support, and fault tolerance. These are typically built on top of the basic overlay functions mentioned above. Security pertains to the way node identities are assigned and controlled, and messages and packets are secured. Security encompasses multiple protocol layers and is responsible for ensuring that peers can maintain sufficient level of trust toward the system. Resource management is about taking content demand and supply into account and ensuring that certain performance and reliability requirements are met. For example, relevant issues are data placement and replication rate. Data replication is also a basic mechanism for ensuring fault-tolerance. If one node fails, another can take its place and, given that the data was replicated, there is no loss of information.

Above this layer, we have the services management for both monitoring and controlling service lifecycles. When a service is deployed on top of an overlay, there need to be functions for administering it and controlling various issues such as administrative boundaries, and data replication and access control policies.

Finally, in the topmost layer we have the actual applications and services that are executed on top of the layered overlay architecture. The applications rely on the overlay architecture for scalable and resilient data discovery and exchange.

An overlay network offers a number of advantages over both centralized solutions and solutions that introduce changes in routers. These include the following three key advantages:

Incremental deployment: Overlay networks do not require changes to the existing routers. This means that an overlay network can be grown node by node, and with more nodes it is possible to both monitor and control routing paths across the Internet from one overlay node to another. An overlay network can be built based on standard network protocols and existing APIs—for example, the Sockets API of the TCP/IP protocol stack.

Adaptable: The overlay algorithm can utilize a number of metrics when making routing and forwarding decisions. Thus the overlay can take application-specific concerns into account that are not currently offered by the Internet infrastructure. Key metrics include latency, bandwidth, and security.

Robust: An overlay network is robust to node and network failures due to its adaptable nature. With a sufficient number of nodes in the overlay, the network may be able to offer multiple independent (router-disjoint) paths to the same destination. At best, overlay networks are able to route around faults.

The designers of an early overlay system called *resilient overlay network (RON)* [361] used the idea of alternative paths to improve performance and to route around network faults. Figure 1.2 illustrates how overlay technology can be used to route around faults. In this example, there is a problem with the normal path between A and B across the Internet. Now, the overlay can use a so-called *detour path* through C to send traffic to B. This will result in some networking overhead but can be used to maintain communications between A and B.

Overlay networks face also a number of challenges and limitations. The three central challenges include the following:

- The real world: In practice, the typical underlay protocol, IP, does not provide universal end-to-end connectivity due to the ubiquitous nature of firewalls and *network address translation (NAT)* devices. This means that special solutions are needed to overcome reachability issues. In addition, many overlay networks are oblivious to the current organizational and management structures that exist in applications

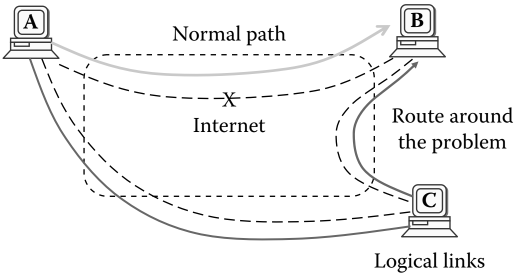

FIGURE 1.2
Improving resiliency using overlay techniques.

and also in network designs. For example, most of the overlay solutions presented in this book do not take Internet topology into account from the viewpoint of the *autonomous systems (ASs)* and inter-AS traffic.

- Management and administration: Practical deployment requires that the overlay network have a management interface. This is relatively easy to realize for a single administrative domain; however, when there are many parties involved, the management of the overlay becomes nontrivial. Indeed, at the moment most overlays involve a single administrative domain.

  The administrator of an overlay network is typically removed from the actual physical devices that participate in the overlay. This requires advanced techniques for detecting failed nodes or nodes that exhibit suspect behaviors.
- Overhead: An overlay network typically consists of a heterogeneous body of devices across the Internet. It is clear that the overlay network cannot be as efficient as the dedicated routers in processing packets and messages. Moreover, the overlay network may not have adequate information about the Internet topology to properly optimize routing processes.

Figure 1.3 presents a taxonomy of overlay systems. Overlays can be router-based, or they can be completely implemented on top of the underlay, typically TCP/IP. Router-based overlays typically employ IP Multicast [107, 130] and IP Anycast [106] features; however, given the fact that deployment of the next version of the IP protocol, IPv6 [106], has not progressed according to most optimistic expectations, these extensions are not

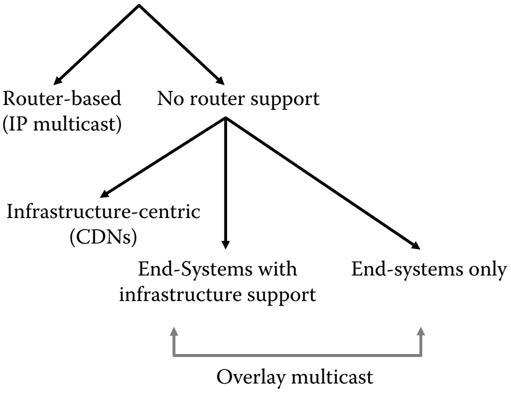
Router-based (IP multicast)
No router support
Infrastructure-centric (CDNs)
End-Systems with infrastructure support
End-systems only
Overlay multicast

FIGURE 1.3
Taxonomy of overlay networks.

globally supported on the Internet. If the routers only provide basic unicast end-to-end communication, information networking functions need to be provided by the overlay.

> *Content delivery networks (CDNs)* are examples of overlay networks that cache and store content and allow efficient and less costly ways to distribute data on a massive scale. CDNs typically do not require changes to end-systems, and they are not P2P solutions from the viewpoint of the end clients.

The two remaining categories illustrated in Figure 1.3 are end-systems with and without infrastructure support, respectively. The former combines fixed infrastructure with software running in the end-systems in order to realize efficient data distribution. The latter category does not involve fixed infrastructure, but rather establishes the overlay network in a decentralized manner.

Overlay networks allow the introduction of more complex networking functionality on top of the basic IP routing functionality. For example, *filter-based routing*, *onion routing*, *distributed hash tables (DHTs)*, and *trigger-based forwarding* are examples of new kinds of communication paradigms. DHTs are a class of decentralized distributed algorithms that offer a lookup service. DHTs store (key, value) pairs, and they support the lookup of the value associated with a given key. The keys and values are distributed in the system, and the DHT system must ensure that the nodes have sufficient information of the global state to be able to forward and process lookup requests properly.

The DHT algorithm is responsible for distributing the keys and values in such a way that efficient lookup of the value corresponding to a key becomes possible. Since peer nodes may come and go, this requires that the algorithm be able to cope with changes in the distributed system. In addition, the locality of data plays an important part in all overlays, since they are executed on top of an existing network, typically the Internet. The overlay should take the network locations of the peers into account when deciding where data is stored, and where messages are sent, in order to minimize networking overhead.

Figure 1.4 illustrates the key DHT API functions that allow peers to insert, look up, and remove values associated with a key. Typically, the key is a hash value, so-called *flat label*, which realizes essentially a flat namespace that can be used by the DHT algorithm to optimize processing.

> DHTs are a class of decentralized distributed systems. They provide a logically centralized lookup service similar to hash tables. A DHT stores (key, value) pairs and allows a client to retrieve a value associated with a given key. The DHT is typically realized as a structured P2P network in which peers cooperate to provide the service across the Internet.

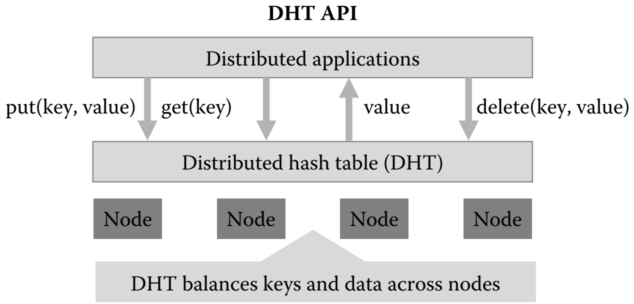

FIGURE 1.4
DHT API.

There are two main classes of P2P networks, *structured* and *unstructured*. In the former type, the overlay network topology is tightly controlled by the P2P system and content is distributed in such a way that queries can be made efficiently. Many structured P2P systems utilize DHT algorithms in order to map object identifiers to distributed nodes. Unstructured P2P networks do not have such tightly controlled structure, but rather they utilize flooding and similar opportunistic techniques, such as *random walks* and *expanding-ring time-to-live (TTL)* search, for finding peers that host interesting data. Each peer receiving a query can then evaluate the query locally using its own content. This allows unstructured P2P systems to support more complex queries than are typically supported by structured DHT-based systems.

Unstructured P2P algorithms are called *first generation* and the structured algorithms are called *second generation*. They can also be combined to create *hybrid* systems. The key-based structured algorithms have a desirable property: namely, that they can find data locations within a bounded number of overlay hops [162]. The unstructured broadcasting-based algorithms, although resilient to network problems, may have large routing costs due to flooding, or may be unable to find available content [274].

Another approach to P2P systems is to divide them into two classes, *pure* and *hybrid* P2P systems. In the former, each peer is simultanously a client and a server, and the operation is decentralized. In the latter class, a centralized component is used to support the P2P network.

Figure 1.5 illustrates the inherent trade-off between completeness and expressiveness of an overlay system. By completeness we mean the ability of the system to guarantee the location and retrieval of a piece of data. Expressiveness pertains to the system's ability to reason about the data—for example, how complex queries can be used to locate data elements. DHTs and other structured overlays typically guarantee completeness, whereas unstructured systems, such as Gnutella and Freenet, do not provide such guarantees. As an inherent limitation, structured systems support less complex queries, typically the lookup of keys. Unstructured systems, on the other hand, can support complex query processing. In this book, we cover both structured and unstructured systems and highlight their key properties.

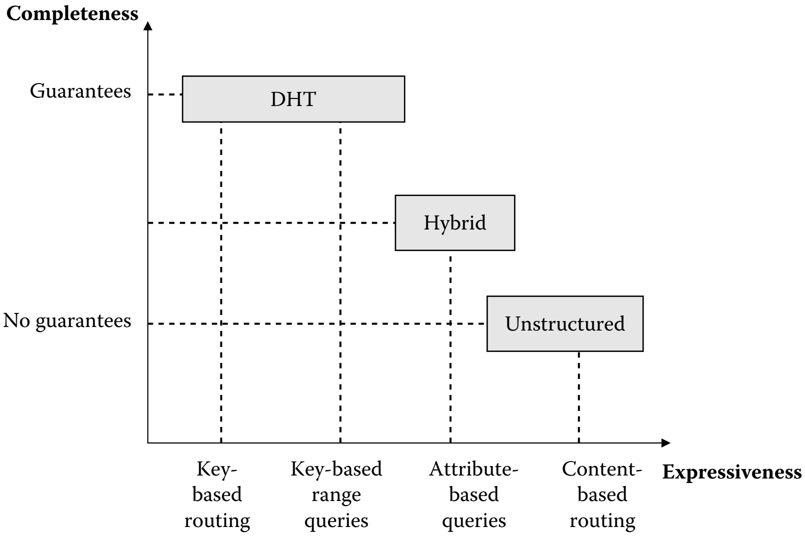

FIGURE 1.5
Balancing completeness and expressiveness in overlays.

1.3 Applications

Many overlay networks have been proposed both in the research community and by Internet and Web companies. Overlay networks can be categorized into the following classes [80]:

- P2P file sharing: For sharing media and data. For example, Napster, Gnutella, KaZaA.
- CDN: Content caching to reduce delay and cost. For example, Akamai and LimeLight.
- Routing and forwarding: Reduce routing delays and cost, resiliency, flexibility. For example, resilient overlay network (RON), Internet indirection infrastructure (i3).
- Security: To enhance end-user security and offer privacy protection. For example, virtual private networks (VPNs), onion routing, anonymous content storage, censorship resistant overlays.
- Experimental: Offer testing ground for experimenting with new technologies. For example, PlanetLab.
- Other: Offer enhanced communications. For example, e-mail, VoIP, multicast, publish/subscribe, delay tolerant operation, etc.

Currently a significant amount of content is being served using decentralized P2P overlays. Most of the deployed algorithms are based on unstructured overlays. The unstructured P2P protocol BitTorrent has become a popular content distribution protocol over the recent years.

P2P technologies are not commonly used with CDNs; however, they are increasingly used by end clients. P2P offers end client–assisted data distribution, in which clients acting as peers upload data. This contrasts with the traditional client-server CDN model, in which clients do not upload data. The main strength of P2P is in the delivery of massively popular data items; however, items that fall into the long tail may not be cost-efficient to distribute using P2P. This can be alleviated by storing data items on client machines using caching, but this requirement is not favored by many users.

1.4 Properties of Data

In this section, we briefly discuss the properties of data [117, 120, 228]. Data can be characterized in many ways. We consider an example taxonomy in Figure 1.6 that divides data into two parts: stored data and real-time data.

Stored data consists of bits that are stored on a system on a more permanent basis in such a way that the data can be made available later. This data can take two forms: it can be mutable or immutable. Mutable data can be shared and modified by various entities either locally or in the distributed environment. Mutable data can be made incrementally available, and it can be created and managed by multiple entities. On the other hand, mutable data is not easy to cache and it requires complicated security solutions, especially in distributed environments. Immutable data means that the full data—for example, a picture or a video file—is available, and it does not change. This data can therefore be cached and verified easily.

Real-time data is generated on the fly and transmitted over the network. The data is packetized, possibly on multiple layers, and it is transferred hop-by-hop on a store-and-forward basis. This means that, although individual packets of the data are stored in intermediate

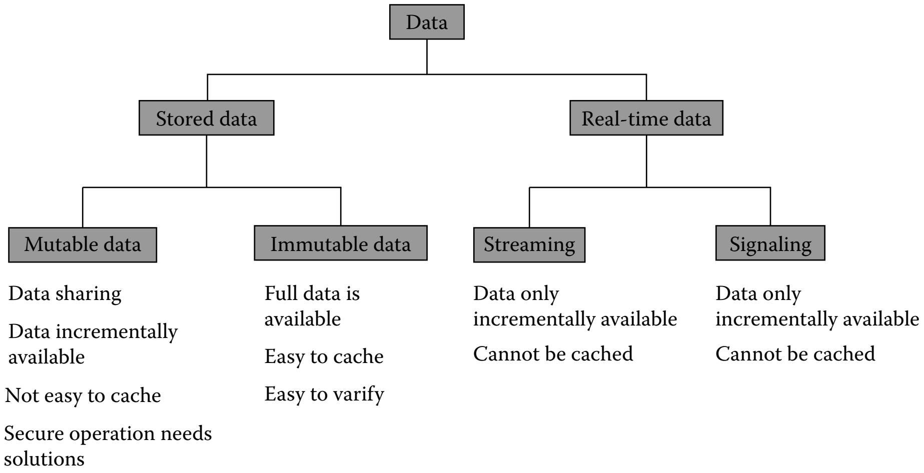

FIGURE 1.6
Taxonomy of data.

buffers, the whole data is not stored as such. In addition, with real-time data, the time when the data is inserted into the network plays a crucial part.

Streaming data is only incrementally available, and only the latest packets of this stream are important. This means that this kind of data cannot be cached. Another form of real-time data is signaling. In this case, data also becomes incrementally available and cannot be cached; however, the data packets are typically very different from streaming.

References play an important part in distributed systems. A reference encapsulates a relationship between itself and a referent defined relative to the state of some physical system. As examples we may consider memory addresses that point to some specific locations of physical memory and *universal resource locators (URLs)* that point to Web resources located on specific servers, available using a specific protocol such as the *hypertext transfer protocol (HTTP)*. If the physical system changes—for example, memory is swapped or a server is relocated—the referent changes as well. These so-called *physical references* may become invalid when the environment changes.

In order to cope with changes in the environment, the common practice is to introduce a level of *indirection* into the reference system. For example, the *domain name system (DNS)* binds host names to IP addresses, which allows administrators to change IP addresses without changing host names. The hierarchical and replicated structure of DNS scales well for its intended purposes, and it is at the core of the Internet.

A data element can be either mutable or immutable. In the former case it can change, and in the latter case it cannot change. It is obvious that a mutable data element can be represented by a sequence (or a graph) of immutable data elements. Given that a piece of data does not change, it can be uniquely and succinctly summarized using a hash function. We note that hashes only provide probabilistic uniqueness; however, a long enough hash bitstring results in a vanishingly small probability of collision.

A hash function is a function from a sequence of bytes to a fixed size sequence of bits, a bitstring. Hash functions can be characterized based on how easy it is to find a collision [227]:

- A hash function is *strongly collision resistant* if it is not computationally feasible to find two different input data items which have the same hash.
- A hash function is *weakly collision resistant* if, for a given data item, it is computationally not feasible to find another data item that has the same hash.

- A hash function is *probabilistically collision resistant* if, for a given input data item, the probability that a randomly chosen data item will have the same hash as the input data item is extremely small.

Semantic-free references have been proposed to achieve persistence and freedom from contention in a naming system [20, 339]. The idea is to use a reference namespace devoid of explicit semantics—for example, based on hashed identifiers. This means that a reference should not contain information about the organization, administrative domain, or network provider. Flat semantic-free references contrast with DNS-based URLs because they have no explicit structure. The semantic-free referencing method uses DHTs to map each object reference to a machine that contains object metadata. The metadata typically includes the object's current network location and other information.

Until recently, there have been no good candidate solutions for resolving semantic-free names in a scalable fashion in the distributed environment. The traditional solution has been to use a partitioned set of context-specific name resolvers. The emerging overlay DHT technology can be used to efficiently store and look up semantic-free references. Indeed, the so-called self-certified flat labels have gained widespread adoption in recent overlay systems.

Self-certifying data is data whose integrity can be verified by the client accessing it [227]. A node inserting a file in the network or sending a packet calculates a cryptographic hash of the content using a known hash function. This hashing produces a file key that is included in the data. The node may also sign the hash value with its private key and include its public key with the data. This additional process allows other nodes to authenticate the original source of the data. When a node retrieves the data using the hash of the data as the key, it calculates the same hash function to verify that the data integrity has not been compromised.

A large part of the research and development on P2P systems has focused on *data-centric* operation, which emphasizes the properties of the data instead of the location of the data. Ideally, the clients of the distributed system are not interested in where a particular data item is obtained as long as the data is correct. The notion of data-centricity allows the implementation of various dynamic data discovery, routing, and forwarding mechanisms [274].

In *content-based routing* systems, hosts subscribe to content by specifying filters on messages. In content-based routing, the content of messages defines their ultimate destination in the distributed system. Information subscribers use an interest registration facility provided by the network to set up and tear down data delivery paths. Data-centric and content-based communications are currently being investigated as possible candidates for Internet-wide communications.

1.5 Structure of the Book

After the introduction chapter that motivates overlay technology and outlines several application scenarios, we start with an overview of networking technology in Chapter 2. This chapter briefly examines the TCP/IP protocol suite and the basics of networking, such as naming, addressing, routing, and multicast. The chapter forms the basis for the following chapters, because typically TCP/IP is the underlay of the overlay networks and thus understanding its features and properties is vital to the development of efficient overlay solutions.

We discuss properties of networks in Chapter 3, including the growth of the Internet, trends in networking, and how data can be modeled. Many of the overlay algorithms are based on the observation that networks exhibit power law degree distributions. This can then be used to create better routing algorithms.

In Chapter 4 we examine a number of unstructured P2P overlay networks. Many of these solutions can be seen to be part of the first generation of P2P and overlay networks; however, they can be also combined with structured approaches to form hybrid solutions. We cover protocols such as Gnutella, BitTorrent, and Freenet and present a comparison of them. This chapter places special emphasis on BitTorrent, because it has become the most frequently used P2P protocol.

Chapter 5 presents the foundations of structured overlays. We consider various geometries and their properties that have been used to create DHTs. The chapter also presents consistent hashing, which is the basis for the scalability of many DHTs. After surveying the foundations and basic cluster-based solutions, we then examine a number of structured algorithms in Chapter 6. Structured overlay technologies place more assumptions on the way nodes are organized in the distributed environment. We analyze algorithms such as the Plaxton's algorithm, Chord, Pastry, Tapestry, Kademlia, CAN, Viceroy, Skip Graphs, and others. The algorithms are based on differing structures, such as hypercubes, rings, tori, butterflies, and skip graphs. The chapter considers also some advanced issues, such as adding hierarchy to overlays.

Many P2P protocols and overlay networks utilize probabilistic techniques to reduce processing and networking costs. Chapter 7 presents a number of frequently used and useful probabilistic techniques. Bloom filters and their variants are of prime importance, and they are heavily used in various network solutions. The chapter also examines epidemic algorithms and gossiping, which are also the foundation of a number of overlay solutions.

As observed in this chapter, data-centric and content-centric operation offer new possibilities regarding data caching, replication, and location. Recently, content-based routing has become an active research area. In Chapter 8 we consider content-centric routing and examine a number of protocols and algorithms. Special emphasis is placed on distributed publish/subscribe, in which content is targeted to active subscribers.

Given the scalable and flexible distribution solutions enabled by P2P and overlay technologies, we are faced with the question of security risks. The authenticity of data and content needs to be ensured. Required levels of anonymity, availability, and access control also must be taken into account. Chapter 9 examines the security challenges of P2P and overlay technologies, and then outlines a number of solutions to mitigate the examined risks. Issues pertaining to identity, trust, reputation, and incentives need to be analyzed.

Chapter 10 considers applications of overlay technology. Amazon's Dynamo is considered as an example of an overlay system used in production environment that combines a number of advanced distributed computing techniques. We also consider *video-on-demand (VoD)* in this chapter. Much of the expected IP traffic increase in the coming years will come from the delivery of video data in various forms. Video delivery on the Internet will see a huge increase, and the volume of video delivery in 2013 is expected to be 700 times the capacity of the US Internet backbone in 2000. The remainder of the chapter examines P2P SIP for telecommunications signaling, and content distribution technologies.

Finally, we conclude in Chapter 11 and summarize the current state of the art in overlay technology and the future trends. The chapter outlines the main usage cases for P2P and overlay technologies for applications and services.