

# Opportunistic Networks: Challenges in Specializing the P2P Paradigm

Leszek Lilien, *Senior Member, IEEE*, Z. Huma Kamal, *Student Member, IEEE*, Ajay Gupta, *Senior Member, IEEE*

*Abstract*—We first introduce a new category of peer-to-peer (P2P) networks, which we call *opportunistic networks* or *oppnets*. Initially, a relatively small *seed oppnet* is deployed, which grows into a bigger *expanded oppnet*. Oppnet growth starts with detection and benefit evaluation of diverse systems existing in its relative vicinity. Systems with best evaluations are invited by oppnet to become its *helpers*. Joining helpers are “integrated” well enough so that oppnet is able to leverage their vast collective capabilities and resources. Helpers are employed by oppnet to execute diverse tasks in support of its goals. Second, we sketch scenarios for application of oppnets. In particular, oppnet characteristics make them a natural fit for emergency response applications. Third, we discuss challenges in the development and use of the oppnet technology. Overcoming the challenges will result in successfully specializing the more general paradigm of P2P systems to the more focused paradigm of oppnets, and contributing to the state of the art in P2P networks. To the best of our knowledge, we were the first to define and are now the first to investigate oppnets.

*Index Terms*—Peer-to-peer systems, P2P, opportunistic networks, oppnets, privacy.

## I. INTRODUCTION

WE propose to chart a new direction within the area of peer-to-peer (P2P) computing, by exploring *opportunistic networks* or *oppnets*—a new category of P2P systems invented by us. By developing this new paradigm and technology within the P2P area, we add to this workshop, which—as stated by the organizers—“*will focus on high quality contributions on general theme of P2P Computing [...]*.” To the best of our knowledge, oppnets have not been studied by others. (They were first signaled as opportunistic sensor networks by one of the co-authors in his portion of an earlier paper [BLRW04].)

We ask the reader to temporarily accept at the face value our claim that *oppnets are P2P systems*. We have to present oppnets in some detail before we can convincingly argue that they are indeed a category of P2P networks. Therefore, the arguments are presented only in Section V.

Oppnets differ from traditional networks, in which the nodes of a single network are all deployed together, with the size of

the network and locations of its nodes pre-designed (either in a fully “deterministic” fashion, or with a certain degree of randomness, as is the case with ad hoc networks). The initial *seed oppnet* grows into an *expanded oppnet* by taking in foreign nodes that become its *helpers* in realization of the oppnet’s goals. The size of the expanded network and locations of all but a subset of its “seed” nodes can not be even approximately predicted. Oppnets can be powerful, autonomous, able to self-organize, adapt to changing environments, and self-heal when faced with component failures or malicious attacks.

We invented oppnets since they will facilitate existing applications and enable new ones. In particular, oppnets are a natural fit for emergency preparedness and response applications in homeland security.

In the rest of the paper, Section II describes basic oppnet operations, as well as benevolent and malevolent uses for oppnets. Section III sketches related work. Challenges confronting oppnets are presented in Section IV. Section V shows why oppnets are P2P networks. Finally, Section VI concludes the paper and sketches directions for future work.

## II. BASICS OF OPPNET OPERATION

### A. Seed Oppnet and Its Growth

Each oppnet grows from a pre-designed *seed oppnet*, or simply a *seed*, which is a set of nodes employed together at the time of the initial oppnet deployment. It can have just a few (possibly powerful) nodes, in the extreme even a single node. A seed can be wireless and ad hoc—with nodes not carefully pre-positioned but, for example, thrown out of a plane or a car in the general disaster area.

Once the seed becomes operational, its first task is to detect a set of “foreign” entities—devices, clusters, networks, or other systems—which it deems useful. The detected entities are candidates for becoming helpers for the oppnet. Each such *candidate helper* (or simply *candidate*) has a potential to provide oppnet with communication, computing, sensing, or other capabilities or resources. Detection can be done by any means possible, both traditional and novel (cf. [GuAA05]), including cellular-based, radio-based, or satellite-based detection. It can search for systems in the disaster area using the range of Internet addresses, known as IP addresses, assigned to its own geographical area. It can even use artificial intelligence techniques for visual detection of systems and appliances with embedded chips. For example, it can visually detect a car within the surveillance area of a helper that already joined the oppnet, read the license plate, checked if the car was equipped with the On-

This work was supported in part by the National Science Foundation under Grant IIS-0242840, and in part by the U.S. Department of Commerce under Grant BS123456.

The authors are with the Department of Computer Science, Western Michigan University, Kalamazoo, MI 49008. E-mail: {llilien, zkamal, gupta}@cs.wmich.edu

Star system [OnSt05], and attempted to contact its embedded computing systems if it did.

Candidates are evaluated by the oppnet, and the best ones are *invited* by the oppnet to join it. They can either accept or refuse the invitation. However, in emergency life-or-death situations, candidates are *ordered* to join, and must agree to be conscripted in the spirit of citizens called to arms (or suffer the consequences of going AWOL). A candidate willing (or ordered) to join still needs to ask for permission to join—the need of the oppnet for helpers is very dynamic. The oppnet admits only candidates that it needs at the moment of its admission decision. Admitted entities become oppnet *helpers*.

By admitting candidates, a seed grows into an *expanded oppnet*. E.g., an expanded oppnet include the following helpers: (a) a computer network—contacted via a wired Internet link; (b) a cellular infrastructure—via oppnet’s cellphone peripheral; (c) a satellite system—via a direct satellite link; (d) a HAN (home area network)— via wireless-equipped embedded processor in a refrigerator; (e) a microwave network—via a microwave relay; (f) a BAN (body area network) of an occupant of an overturned car—via an OnStar™ system.

All helpers collaborate on realization of its oppnet’s goals. A helper may be allowed by an oppnet manager to invite other systems. The more helpers are allowed to invite foreign nodes the faster oppnet grows.

### B. Oppnet Helpers

*1) Potential Oppnets Helpers:* The set of helpers includes even entities not usually thought of as network nodes, both wired and wireless, free-standing and embedded. Even nodes with no sensing capabilities, such as networked mainframes from LANs or wireless-equipped processors embedded in cars, can significantly contribute to processing or communication capabilities of an oppnet. After all, any networked PC or embedded processor has some useful sensing, processing, or communication capabilities. For example, information about user’s presence or absence, her work habits and Internet access patterns can be collected by her desktop and her PDA; information about user’s location – by his cellphone (even one without GPS can be triangulated); and data about food consumed by user’s household – by a processor embedded in a refrigerator and RFID-equipped food packages and containers.

*2) Helper Functionalities:* In general, working in the “disaster mode” does not require any new functionalities from the helpers. For example, in case of fire monitoring tasks, the weather sensornet that became a helper can be simply told to stop collecting precipitation data, and use the released resources to increase the sampling rates for temperature and wind direction.

It is possible that more powerful helpers could be reprogrammed on the fly. Also, oppnet nodes might be built with excess general-purpose communication, computation, storage, sensing, and other capabilities useful in case of unforeseen emergencies. For example, excess sensing capabilities could be facilitated by multisensor devices.

### C. Applications for Oppnets

We see important applications for oppnets in all kinds of emergency situations, for example in man-made or natural

disaster recovery for homeland security emergencies. They have the potential to significantly improve efficiency and effectiveness of relief and recovery operations. For predictable disasters (like hurricanes or firestorms, whose path can be predicted with some accuracy), seed oppnets can be put into action and their build-up started (or even completed) *before* the disaster, when it is still much easier to locate and invite other nodes and clusters into the oppnet. The first invited helpers could be the sensornets deployed for structural damage monitoring and assessment in buildings, roads, and bridges.

As most technologies, oppnets can be used to either benefit or harm humans, their artifacts, and technical infrastructure they rely upon (for more details on malevolent oppnets are counteracting them cf. [LKBG06]).

## III. RELATED WORK

Oppnets might be perceived as networks that lie within the intersection of ad hoc, P2P, and sensor networks. After necessary adjustments, they can use node localization and self-organization techniques from ad hoc networks, growth-by-joining approaches from P2P systems, and data aggregation algorithms from sensornets. Hence, the fact that a lot of related work comes from these three areas should not be surprising.

We look at three more categories of related work on technologies potentially useful for oppnets: (1) Interoperability research—on highly heterogeneous wired and wireless types of communication media, networks, devices, and protocols; (2) Grid computing—for resource integration and management; and (3) Benevolent Trojans—for helper search. There is a tremendous amount of knowledge and experience in all six areas that we can learn from—but cannot employ ‘as-is’ in oppnets, due to their unique characteristics.

We omit the detailed discussion of related areas due to space limitations.

## IV. CHALLENGES IN OPPNETS

In this section we delineate the challenges in development and use of oppnets.

### Challenge 1: *Optimizing the seed oppnet infrastructure.*

Measures and criteria for optimization of oppnets in their deployment environments are needed. They must allow quantitative specification of at least communication, computational, sensing, and energy resources. Measures and criteria should enable optimization of oppnet size and of the quality of its localization.

Next, researchers need to provide techniques for optimization of oppnets as required to most successfully achieve oppnet goals (e.g., determining optimal oppnet sizes and optimal locations of seed nodes). The techniques might use probabilistic terms if an oppnet is deployed without precise positioning—for instance, when dispersed from the air.

In addition to the optimal configuration, researchers need to characterize the minimal seed oppnet configuration that assures a credible execution of oppnet tasks (e.g., the minimal size of the seed oppnet and the minimal required seed node capabilities). Node capabilities are to be defined at least in

terms of their communication, computational, sensing, and energy resources.

**Challenge 2:** *Developing methods for detecting nearby communication, computing, and sensing facilities by an oppnet.*

Researchers must devise efficient algorithms for detection of potential helpers, whether they are individual nodes, clusters and entire networks. For example, current solutions for detection (via localization) of neighboring nodes in sensor networks use a single technology – based on GPS, radio, ultrasound, IP address or cellular network. Oppnets should attempt to detect any useful systems available in their neighborhood. To the best of our knowledge, there are no solutions, which fuse the localization data obtained by all diverse technologies in a way that would satisfy the needs of oppnets.

**Challenge 2.1:** *Designing an integrated medium of communication.*

This challenge is logically a subset of the preceding one (this is indicated by using a two-part numbering for it) but, due to its importance and effort required, warrants to be listed separately.

Providing means of communications between an oppnet and all potential helper systems is a significant problem. Each of these systems may have distinct and disjoint media of communication, with unique protocols for information transfer. It may be as complex as 802.11b, or as simplistic as communicating binary values of *motion* or *no\_motion* coming from a motion sensor.

An oppnet should be able to detect and contact potential helpers, and then be able to communicate with them over their preferred medium and with their preferred protocols. This approach would be similar to a multi-agent system [Flor03], where agents try to collaborate to achieve a common goal. However, oppnets cannot use a common language or protocol or format for communications. Oppnets do not share an ontology with all potential helpers, so a standard lightweight protocol would need to be developed enabling a shared ontology.

Utilizing the emergency communication channels should be considered. In particular, researchers should investigate whether the rescue and recovery operations of oppnets should be segregated from other emergency control and monitoring tasks, and whether oppnets should switch between modes when directed by the operations command center.

**Challenge 3:** *Designing methods for inviting candidate systems, and methods for controlling systems that joined.*

Researchers must design protocols for inviting and admitting candidate systems to oppnets. First, they need to develop algorithms to determine which candidates from the pool of available ones to invite. To this end, systems useful for oppnets must be prioritized based on their capabilities and functionalities. The potential helpers should be evaluated at least in terms of their communication, computational, sensing, and energy resources. The priorities could determine the order in which oppnets will invite different classes of devices to join. For example, an oppnet may decide that inviting communication systems (such as providers of WiFi spots or cellular infra-

structures) is the highest priority, since they extend oppnet coverage considerably.

A prioritization scheme will enable oppnets to encompass a systematic approach to inviting helpers. In a simple case, oppnet could start with a search for communication systems, then move to systems with critical processing abilities, finally to sensing systems. In a more complex case, oppnet would start with a search for a system with communication capabilities characterized by a measure  $C1$  with the value at least equal to  $c1$ , and with sensing capabilities characterized by a measure  $S2$  with the value at least equal to  $s2$ .

Once a prioritization scheme is available, researchers will be able to develop algorithms and protocols for finding which systems from the identified pool of available ones to invite. Priorities will show which candidates can help, and which ones can help more than others. (It should be noticed that even a very primitive device can help. For example, a position of a simple light switch in an office can “sense” a presence of a person: the “off” position means with some probability that the occupant of the office is absent.) The algorithms might select candidates to assure the best area and functional coverage with a minimal number of invited systems, or with a certain degree of coverage redundancy for fault tolerance.

Whenever possible, oppnet algorithms should perform a priori evaluation of trustworthiness of helper candidates, to avoid inviting unreliable or malicious ones.

Customized protocols for inviting selected systems to join an oppnet are needed. Their design should be preceded by investigation of usefulness of existing protocols for contacting and inviting other systems by oppnets. Researchers should consider use of incentives to encourage systems reluctant to join. They might also consider use of penalties for systems that refuse to join in life-or-death situations.

In the future, lightweight facilities to answer to invitations from oppnets could become a standard portion of any operating system on any desktop, laptop, handheld, or embedded device.

**Challenge 4:** *Developing methods for deciding which tasks should be “offloaded” by oppnet to its helpers, and techniques for coordinating these tasks by oppnet.*

Oppnet tasks need be classified according to their suitability for offloading to a “helper” system. A special consideration must be given to identifying and exploiting idiosyncrasies of basic tasks in oppnets. This means finding out whether they are different and how different from the basic tasks in non-oppnets, especially P2P or ad hoc networks (such basic tasks include self-organization, reorganization, localization of neighboring nodes, and aggregation [IyBr03]).

All systems that join the oppnet become its helpers and collaborate on realization of its goals. Oppnet must be aware that different helpers might be better suited for different jobs. Helper should be classified w.r.t. their capabilities for completing different classes of tasks to be offloaded. For example, some of them can perform computationally intensive operations very well, while others can store information efficiently and reliably. Special consideration must be given to identify-

ing and exploiting characteristics of the tasks that will ensure the best use of the capabilities of helpers.

Some helpers may be more privileged than others. The privileges are determined based on the level of trust established between the seed oppnet and helpers. For example, only most trusted helpers may be permitted by the seed oppnet to contact and invite other systems to join.

Protocols for offloading the tasks to helpers and controlling their execution must be devised. Helpers should probably be allowed to maintain their low-level control, while assuring the high-level supervision and decision-making by oppnets. Again, special consideration must be given to identifying and exploiting idiosyncrasies of the typical oppnet tasks.

Due to their computing needs, aggregation tasks might be a special category of tasks for offloading. To be effective, oppnet algorithms must be able to identify information that needs to be aggregated, efficient aggregating techniques, and appropriate helpers able to use these techniques for these data. The algorithms must recognize data that needs preprocessing before it can yield useful information—such as image rectification, enhancement and classification, and identify helpers capable of handling such procedures.

The algorithms must also consider scenarios where helpers as powerful as needed do not exist in the oppnet. Finding new helpers might be necessary. If new helpers can not be found quickly enough, a decision must be made what to do with overwhelming information that can not be aggregated. For example, the oppnet algorithms must decide between two options: (a) immediately relaying information to the oppnet's controller—which is a significant communication effort; and (b) immediately applying restricted data mining techniques to it—which is still a significant computation effort.

**Challenge 5:** *Proposing ways of managing oppnets, including control of privacy and security problems in oppnets.*

Management algorithms for controlling oppnet nodes are needed, including algorithms for identifying suspicious or inefficient members of an oppnet, and dismissing them when necessary (even members of the original seed oppnet can be dismissed).

Once the goals of an oppnet have been achieved and its activity is no longer needed, it should release its helpers and contacted candidates. Furthermore, the released systems should be able to quickly return to their normal operations, without any unwelcome residue from the tasks performed as helpers in an oppnet.

Oppnets are pervasive computing systems. Since huge privacy risks are associated with pervasive computing, privacy is the “make it or break it” issue for oppnets, discussed in more detail in our earlier paper [LKBG06]. Privacy solutions can be divided into three categories for protecting oppnets, helpers, and the environment from each other: (1) protecting oppnets from helpers and the environment, (2) protecting helpers from oppnets and the environment, and (3) protecting the environment from oppnets and helpers. For example, privacy of entities under oppnet surveillance can be protected by assuring their anonymity or pseudonymity.

It is important to note that some relaxation of the strictest privacy protection standards might be permissible in emergency situation. Especially in life-and-death situations, saving a life of one person takes precedence over strict privacy of another. For example, a victim searching for help will probably not object to an oppnet taking over her Body Area Network (BAN), controlling devices on and within her body.

Security controls methods to protect oppnets and to disable their malicious uses are needed. Some are discussed in our earlier paper [LKBG06]. This might be done by planting spies in suspicious networks, as well as by using the honeypot approach [ChBe02]. Malevolent oppnets, which can either hide their malicious activities, or masquerade as benevolent oppnets, must also be considered. Algorithms for their detection and uncovering of their real goals are needed.

**Challenge 6:** *Analyzing performance of oppnet algorithms and protocols for localization, invitation, task offloading and coordination.*

Metrics for evaluating efficiency and effectiveness of the above-mentioned oppnet algorithms and protocols are needed. In addition to the metrics, researchers need to develop methods and guidelines for carrying out theoretical analyses, simulations, testbed experiments, etc.

## V. OPPNETS ARE P2P NETWORKS

According to Milojevic *et al.* [MKLN02], “*Conceptually, P2P computing is an alternative to the centralized and client-server models of computing, where there is typically a single or small cluster of servers and many clients. [...] In its purest form, the P2P model has no concept of server; rather all participants are peers.*” Since oppnets are neither centralized nor client-server systems, they are P2P systems.

Oppnets satisfy also these P2P characteristics [Shir00]:

- 1) The definition: *P2P is a class of applications that takes advantage of [previously unused] resources -- storage, cycles, content, human presence -- available at the edges of the Internet. Because accessing these decentralized resources means operating in an environment of unstable connectivity and unpredictable IP addresses, P2P nodes must operate outside the DNS system and have significant or total autonomy from central servers.*
- 2) The litmus test: *(1) Does it treat variable connectivity and temporary network addresses as the norm? (2) Does it give the nodes at the edges of the network significant autonomy? An application is P2P if and only if the answer to both of those questions is “yes.”*
- 3) Additional ownership test: Another way to examine “P2P or not P2P” is to think about ownership: “Who owns the hardware that the service runs on?” Most of the hardware that makes a P2P system work is owned and managed by system users. In other words, P2P is a way of decentralizing not just features, but costs and administration as well.

Oppnets satisfy also the following descriptive definition of P2P systems [MKLN02]: *The term “peer-to-peer” refers to a class of systems and applications that employ distributed resources to perform a critical function in a decentralized manner. The resources encompass computing power, data (stor-*

age and content), network bandwidth, and presence (computers, human, and other resources). The critical function can be distributed computing, data/content sharing, communication and collaboration, or platform services. Decentralization may apply to algorithms, data, and meta-data, or to all of them. This does not preclude retaining centralization in some parts of the systems and applications if it meets their requirements. Typical P2P systems reside on the edge of the Internet or in ad-hoc networks. P2P enables:

- valuable externalities, by aggregating resources through low-cost interoperability, the whole is made greater than the sum of its parts
- lower cost of ownership and cost sharing, by using existing infrastructure and by eliminating and distributing the maintenance costs
- anonymity/privacy, by incorporating these requirements in the design and algorithms of P2P systems and applications, and by allowing peers a greater degree of autonomous control over their data and resources.

Oppnet characteristics are direct matches to all properties mentioned in the above definition, possibly with the exception of anonymity. It is not critical to oppnets, but can be used in oppnets to protect privacy of helpers.

Steinmetz and Wehrle [StWe05] state: “The P2P approach [...] reflects the paradigm shift from coordination to cooperation, from centralization to decentralization, and from control to incentives.” Oppnets wholeheartedly participate in this paradigm shift.

All these definitions can be complemented by adding a few more P2P system characteristics, some of them relaxing the definition, thus increasing its scope [Peer06]:

- 1) P2P networks are typically used for connecting nodes via largely *ad hoc* connections.
- 2) *Pure P2P networks* do not have the notion of clients or servers, but only equal *peer* nodes that function as “*servents*” (simultaneously “clients” and “servers”) to the other nodes on the network. This differs from the client-server model where communication is usually to and from a central server. Pure P2P networks are rather rare.
- 3) Most P2P networks and applications actually contain or rely on some non-peer elements, such as DNS. Also, real world applications often use multiple protocols and act as client, server, and peer simultaneously, or over time.
- 4) *Hybrid P2P systems* or *mixed P2P systems* (such as Napster, OpenNAP, or IRC@find) use a client-server structure for some tasks (e.g., searching) and a peer-to-peer structure for others. In particular, many P2P systems use stronger peers (super-peers, super-nodes) as servers and client-peers are connected in a star-like fashion to a single super-peer.

## VI. CONCLUSIONS

After presenting the concept of *opportunistic networks* or *oppnets*, this paper discusses research challenges that must be overcome to successfully specialize the more general paradigm of P2P systems to the more focused paradigm of oppnets.

As shown, oppnets can be viewed as a subcategory of P2P systems. When deployed, oppnets attempt to detect systems

existing in their relative vicinity—ranging from communication to computing to sensing systems—and “integrate” them enough to be able to use their available resources. When such a system is detected, an oppnet evaluates its potential benefit, and—if the evaluation is positive—invites it to become its *helper*. In this manner, an oppnet can grow from a small *seed* into an *expanded oppnet* with vast communication, computation, and sensing capabilities.

An integrated network has been called for in various critical or emergency situations [USGo01]. Oppnets can be used to enable connectivity in areas where any existing communication or information infrastructures have been fractured or destroyed. It can connect highly heterogeneous systems—that were not designed to work together—in order to facilitate creation of a bigger and better picture of the region it is deployed in. This kind of “integration” facilitates flow of information that, for example, can assist in rescue and recovery efforts for devastated areas, or can provide more data on phenomena that are just developing, such as wild fires or flash torrents.

Answering to the research challenges identified in this paper will contribute to advancing knowledge and understanding of oppnets. This will enable re-development of many existing applications on top of oppnets, which should result in lower costs and better efficiency and performance. Many new applications should be enabled by oppnets.

Planned research will simultaneously advance the state of the art of the general-purpose peer-to-peer networks.

We will continue our investigation of oppnets, and designing oppnet architectures with their associated components: methods, protocols, and algorithms. The planned prototype oppnet will provide a proof of concept, as well as stimulation and a feedback necessary for fine-tuning oppnet architectures and their components.

## ACKNOWLEDGMENT

This work was supported in part by the National Science Foundation under Grant IIS-0242840, and in part by the U.S. Department of Commerce under Grant BS123456.

Any opinions, finding, conclusions or recommendation expressed in the paper are those of the authors and do not necessarily reflect the views of the funding agencies or institutions.

## REFERENCES

- [BLRW04] B. Bhargava, L. Lilien, A. Rosenthal, and M. Winslett, “Pervasive Trust,” *IEEE Intelligent Systems*, vol. 19(5), Sep./Oct.2004, pp. 74-77.
- [ChBe02] W. Cheswick and S. Bellovin, *Firewalls and Internet Security*, 2nd ed., Addison-Wesley, 2002.
- [Flor03] R. A. Flores-Mendez, “Towards Standardization of Multi-Agent System Frameworks,” 2003. <http://turing.acm.org/crossroads/xrds5-4/multiagent.html>
- [GuAA05] A. Gupta, D. Agrawal, and A. E. Abbadi, “Distributed Resource Discovery in Large Scale Computing,” *SAINT 2005*.

- [IyBr03] S. Iyenger and R. Brooks, *Distributed Sensor Networks*, CRC Press, Inc., 2003.
- [LKBG06] L. Lilien, Z.H. Kamal, V. Bhuse, and A. Gupta, "Opportunistic Networks: The Concept and Research Challenges in Privacy and Security," *Proc. NSF Intl. Workshop on Research Challenges in Security and Privacy for Mobile and Wireless Networks (WSPWN 2006)*, Miami, March 2006, pp.134-147.
- [MKLN02] D.S. Milojevic, V. Kalogeraki, R. Lukose, K. Nagaraja, J. Pruyne, B. Richard, S. Rollins, Z. Xu, "Peer-to-Peer Computing," Report HPL-2002-57, HP Laboratories, Palo Alto, CA, March 2002.
- [OnSt05] "On Star Explained," accessed on Nov. 26, 2005, [http://www.onstar.com/us\\_english/jsp/explore/index.jsp](http://www.onstar.com/us_english/jsp/explore/index.jsp)
- [Peer06] Peer-to-peer, Wikipedia, accessed on 3/2/06, <http://en.wikipedia.org/wiki/P2p>.
- [Shir00] C. Shirky, What Is P2P... And What Isn't., O'Reilly Network, November 2000, [www.openp2p.com/lpt/a/p2p/2000/11/24/shirky1-whatisp2p.html](http://www.openp2p.com/lpt/a/p2p/2000/11/24/shirky1-whatisp2p.html).
- [StWe05] Ralf Steinmetz, Klaus Wehrle (Eds.), "Peer-to-Peer Systems and Applications," ISBN 3-540-29192-X, Lecture Notes in Computer Science, Volume 3485, Sep. 2005, <http://www.peer-to-peer.info>.
- [USGo01] U.S. Government Printing Office via GPO Access, "Combating Terrorism: Assessing the Threat of a Biological Weapons Attack." Online Resource last accessed on December 15, 2005.