Peer-3-Peer: Disaster Proof Offline Messaging

Simran Arora
CIS, UPenn
Philadelphia, PA
simarora@wharton.upenn.edu

Jenna Barton
CMPE, UPenn
Philadelphia, PA
bartonje@wharton.upenn.edu

Ben Judd
CIS, UPenn
Philadelphia, PA
benjudd@seas.upenn.edu

Ramya Rao
CIS, UPenn
Philadelphia, PA
ramyarao@wharton.upenn.edu

David Wang
CMPE, UPenn
Philadelphia, PA
daviwang@seas.upenn.edu

Abstract

Internet and cellular connectivity are frequently compromised after disasters. Communication is central to recovery efforts, allowing civilians to contact their loved ones, aid organizations to manage the supply and distribution of resources, and governments officials to provide important news and updates.

Peer-3-Peer is a next-generation peer-to-peer communication ecosystem for emergency situations. Peer-to-peer smartphone messaging and long-range radio broadcast (e.g. walkie talkies) are two contemporary solutions for offline communication. Peer-to-peer solutions, which leverage onboard WiFi Direct and Bluetooth (e.g. AirDrop), are easy to deploy, but operate on limited range (~ 0.005-0.25 mi.). Meanwhile, radio communication devices have long-range coverage (~ 50 mi.) but are prohibitively expensive and complex to deploy. Thus, existing approaches have seen limited effectiveness and adoption in disaster situations.

Peer-3-Peer integrates radio broadcasting with peer-to-peer communication to create a high-coverage and easy to use solution. Portable hardware amplifiers bridge gaps in the peer-to-peer network, caused by the limited range of wireless communications, creating extended "service networks". These amplifiers interface locally with smartphones and forward messages to other amplifiers via radio. This enables long-range communication without requiring users to purchase specialized hardware components on an individual basis.

Peer-3-Peer is a robust offline communication ecosystem that provides civilians, aid organizations, and governments with reliable messaging when cellular and internet access are unavailable.

1 Motivation

There has been a proliferation of disaster relief technologies due to the innovation of new protocols and improved support for development using these protocols. In particular, there are several SMS-based platforms which allow two-way communication between those affected by disasters and aid agencies, real-time maps from social media updates, deployed IoT sensors, and peer-to-peer solutions that rely on the Bluetooth communication protocol.

The key challenge is that disaster situations often leave a region without cell service and Internet access. The existing Bluetooth solutions rely on a high volume of app users, which is often not the case. Given the low relative communication range of Bluetooth, without enough users, messages will not be quickly propagated from the sender to receiver.

Our solution is based on the reality that post-disaster regions lack cellular and Internet services, and we respond to this need for low-latency Bluetooth messaging. We incorporate the concept of “strengthening hubs” to provide a reliable core in our network from which the users with peer-to-peer connections can descend. The hubs are beneficial for coordinating routing information in the distributed system; speeding up message propagation; and providing fault tolerance. This combination-network is unique, and we detail our technical implementation, evaluation, and business plan in the following sections.

---

2 Technical Approach

2.1 Mobile Application

The mobile application is what users interact with in order to send messages to one another. The GUI resembles a traditional messaging app and has an address book that lists all current chats and the classic "sent and received" style message portrayal once a particular chat is selected. Sample screen captures can be found in Appendix A.

2.2 Custom Underlying Protocols

Underneath the GUI lies a custom routing protocol for message transmission and a custom protocol for how phones self-organize within the network.

2.2.1 Tree Structure and Message Passing

Peer-3-Peer uses a custom protocol to take the mesh network and overlay a simple tree structure upon the connected nodes. When a new client wants to join the network, it sends an adoption request to the first node it connects to in order to establish a parent-child relationship. This relationship is approved through a handshake between the nodes only if the addition of the new node and its descendants (whose identities are passed to the parent during the handshake) maintains a valid tree structure with no loops. The end result of this handshake is that each node has a single parent and possibly many children. Any given node will be fully aware of the nodes in the subtree below it, and thus the root of the tree is fully aware of all nodes in its particular mesh.

A node will transmit messages by following two rules: it either (1) sends the message to a child who has the final destination of the message as a descendant or (2) sends the message up to its parent. If a node does not have a parent and does not have the final destination as a descendant, the message is stored locally until one of these options becomes available.

2.2.2 Network Updates

One key element of Peer-3-Peer is the fact that nodes can dynamically leave and rejoin the network, requiring nodes to be able to update the tree as these changes are made. In order to accomplish this, child nodes send regular heartbeat messages to parents and parent nodes send heartbeat acknowledgements in response. These heartbeats allow children to keep parents updated regarding the state of their descendants: the heartbeat contains information regarding additions and deletions made to the subtree. All nodes then regularly perform timestamp checks to confirm whether their connections are active or not by analyzing when the last heartbeat and heartbeat acknowledgement were received for each of their connections. If the logged timestamp differs from the current time by more than a set threshold, the connected node is considered inactive, or "dead."

These timeliness checks allow nodes to detect when a parent or child has died. If a child dies, all nodes below that particular child are considered lost, and the fact that these nodes are unreachable is propagated up the tree in the next heartbeat update. If a parent dies, the child immediately begins looking for another parent but maintains its subtree. The process followed in rejoining the network is unique to Peer-3-Peer, since priority for a new parent is given to nodes that have an upstream connection to a backbone. This behavior ensures that backbone nodes maintain their location at the root of the tree. Similar to when a node first joins the network, the node and the target parent perform a handshake that ensures the validity of the tree structure. Upon approval, the parent is now aware of the node and all of its descendants, and heartbeat updates resume as before.

2.3 The Backbone

The key innovation of the Peer-3-Peer ecosystem is the integration of strengthening hubs, or hardware "backbone" nodes, with long range radio communications into the client peer-to-peer meshes. The backbone serves a few key functions to elevate the traditional peer-to-peer into something more: it stabilizes individual mesh networks, links disparate client meshes, and mailboxes messages for delayed delivery.

The primary function of backbones, namely linking disparate meshes, is done through their unique ability to interface with the other backbone nodes in the deployment. Through these connections, the backbones are able pass messages from any client of one mesh to its destination in another. This greatly extends the range of individual clients as now they can reach users much further away and in another mesh. The range between meshes is also much larger than would be otherwise achievable because the technologies available for backbone to backbone communication (as compared to end users' mobile devices) have ranges on the scale of tens of kilometers instead of tens of meters.

Aside from enabling long range inter-mesh communication, backbones increase the stability of the mesh they are a part of. As compared to clients which move and can be running on a variety of hardware, backbones run known hardware with stronger radios and don't move. This allows them to lend stability to the network, since in this structure the root of the tree will be stationary and takes load off of clients. Additionally, this known hardware and higher reliability allows the backbones to serve as mailboxes. Due to the transient nature of clients (they may move out of range or experience temporary disconnections) it may be necessary to delay the sending of a message until a time when the recipient is back in the network. Backbones fulfill this function by holding onto these messages and proceeding with delivery later, after it sees that the target is back in the network.

We have implemented the backbone software logic in Java using the Android Things operating system deployed on ARM-based NXP microprocessors. These boards interact with the clients/phones using the same WiFi direct/Bluetooth protocol that clients use to talk to each other. Due to budgetary concerns, our initial implementation of the backbones connects to each other with WiFi instead of higher power radio equipment, but the code only relies on having a channel so moving to a longer-range protocol only requires a hardware substitution with no changes to the code. Our WiFi connection effectively simulates the same performance as we would observe with a satellite uplink, but other protocols can be hotswapped as hardware is made available. This means that depending on the deployment scenario we can optimize the backbones without changing the backbone logic: for a longer-range requirement in a rural area we can use long-range XBee or LoraWAN radios whereas for a scenario that needs more throughput than range, like in a city, we can use LiFi.

3 Evaluation

To evaluate our protocol, we confirmed that the trees of local networks are efficiently induced on the nodes and that the state of all nodes gets updated and passed to the parent nodes correctly via heartbeat messages.

To evaluate our overall product, we considered components of performance, cost, and ease of use. In terms of performance, we measured the maximum distance between client nodes, maximum distance between backbone nodes, speed of joining the network and messaging latency. To evaluate cost, we minimized the expense of the hardware nodes and compared our costs to those of our competitors, ensuring that ours was reasonable and ideally lower. For ease of use, we ensured that connecting with the network was backgrounded and did not interrupt the user flow and that the app design was intuitive and easy to navigate.

Evaluation Metrics	
Backbone Range	WiFi: unlimited
Time to Join Mesh	< 10 seconds
Max Client Connections	7
Max Clients per Mesh	15,000
Max Distance Between Clients	285+ feet
Latency Between Clients	< 10 milliseconds
Latency Between Backbones	< 10 milliseconds

Table 1: Evaluation Metrics

4 Addressing User Needs

Disaster relief is defined as the design and transportation of first aid material, food, equipment, and rescue personnel from supply points to a large number of destinations geographically scattered over the disaster region and the evacuation and transfer of people affected by the disaster to the health care centers safely and very rapidly.[1] The coordination challenges often leads to confusion and significant inefficiencies in relief efforts, given the many different parties involved in the process.

The three key user groups in our system are governments, aid organizations, and civilians. Governments must coordinate safety and rebuilding efforts and communicate updates to civilians and aid organizations. Aid organizations require communication to identify the demand for resources by different affected regions (how much do people need?); the types of resources required (what do people need?); and fulfillment management (can we get people what they need?). It is estimated that logistics efforts account for 80 percent of disaster relief. Finally, civilians must communicate with their loved ones and resource providers. Disaster victims often cannot travel far distances due to injuries, ruined transportation infrastructure, or sheer distance.

Our platform serves these user groups. We envision renting backbones to the three parties as disasters are forecasted, offering the application for free, and enabling communication between these parties. Our solution addresses the need for communication, the bottleneck to disaster relief.

5 Discussion of Findings

We created a thorough proof of concept for a system where hardware nodes can augment the peer-to-peer network. The system supports communication between phones and through backbone nodes at less than 10 milliseconds. The time to join the mesh can be up to 10 seconds as phones try to find and connect with other phones and backbone nodes using Bluetooth connection. Since the Bluetooth protocol can manage at most 8 connections on the chip, we put a maximum of 8 Bluetooth connections per client. This implies a maximum of approximately 15,000 clients per mesh with a tree depth of five. We tested the range between client phones in various buildings and got over a 285-foot range.

We tested these metrics through multiple configurations, where the target is within line of sight, from phone to backbone to backbone to phone, from phone to backbone to backbone to phone to phone, and many others. Overall, the

---

[1] Kovács and Spens

system demonstrates low latency and supports a scalable tree structure overlay.

6 Ethical Considerations and Societal Impact

Traditional peer-to-peer networks are decentralized, giving rise to societal benefits and risks.

A key benefit of Peer-3-Peer, composed of backbone nodes and a client messaging application, is that the system improves the accessibility of disaster relief solutions. The backbone nodes are low cost and easily deployable compared to existing approaches to disaster relief. Furthermore, our application is free and can be downloaded without cellular or Internet connection through wireless transfer of the APK.

We have also prioritized accessibility in our decision to develop an Android application because Android phones are cheaper than competitors such as the iPhone. Additionally, given our focus on disaster relief, we limit the messages in our system to text content, eliminating video and photos, which cause congestion.

The risk of peer-to-peer systems is that decentralization reduces the ability to monitor the network traffic such that malicious participants can use them to communicate and sustain counterfeit or copyright-infringing operations.[2] It is estimated that 70 million people engage in online file sharing; it is estimated that 95% of these downloads are illegal and amounted to $700 million in lost revenue for the music industry in 2003.[3] Historically, this has led to the demise of systems such as Napster. Peer-3-Peer alleviates some of this risk because of the backbone nodes; since the nodes are sold to government and aid organizations, they can be used to increase the level of monitoring in the system.

Another ethical consideration for our system is secure messaging and peer-authentication. In our messaging protocol, users route their messages through other users to send a message to a particular destination. Intermediate nodes involved in this routing should not be able to view the message contents. Future directions for Peer-3-Peer include implementing end-to-end encryption for messages with a cryptographic protocol. Additionally, when nodes advertise their presence and form connections with one another we must be able to authenticate peer-identities. While this is an active area of research, one approach is the "Web of Trust" concept in which a particular node X can authenticate node Y by using validation from other peers (other than X and Y).

7 Business Plan

7.1 Value Proposition

The value to government agencies and aid organizations is a cheaper and more effective way of communication in a disaster relief situation.

7.2 Market Opportunity

In the last ten years, the citizens of developing countries in particular were hit hard by both natural and manmade disasters. During humanitarian emergencies, residents are often without access to rescue aid or resources, and risk injuries or death. The government often does not have the resources to get systems back up in place when infrastructure is ruined. Android is the dominant operating system worldwide, with 76% market share and is increasingly used in developing countries.[4] Governments and disaster relief agencies in these countries can trust that the people being affected by a situation will be able to connect to these hardware hubs.

7.3 Stakeholders

We have three main groups of stakeholders. Emergency workers need to communicate with one another and with the public in order to coordinate and direct services to those in need. The public needs to communicate with one another and to be able to communicate with sources of help. Finally, government officials need to ensure the safety of zones at risk.

7.4 Customer Segments

Our primary customer segment is *emergency workers* and *government disaster relief organizations*. As explained our revenue model, these customers would be primarily purchasing the hardware hubs. Our secondary customer segment is *general public members* who would have the free app downloaded and could purchase hardware hubs as needed.

7.5 Market Size and Growth

There were almost 300 natural disasters in 2018.[5] Of these, developing countries were hit extremely hard, from tsunamis and earthquakes in Indonesia which collectively killed over 3,000 people, volcanic explosions in Guatemala killing over 400, and terrible floods in India which caused over 350 deaths. Natural disasters have consistently caused drastic harm to infrastructure in communities worldwide, and this will not stop anytime soon. Our technology is accessible and useful to anyone who has an Android smartphone and is affected by a disaster. With Android the dominant OS in

---

[2] Christin

[3] Bateman et al.

[4] Statista

[5] Ritchie and Roser

developing countries and growing rapidly, we expect the market for this to grow.

7.6 Differentiation

Unlike some of our competitors, we do not rely on WiFi or cellular networks to create mesh networks among individuals that have our application. This differentiates us from the typical messaging app – we are not a wrapper to make cellular messages "prettier" – we are a distinct messaging app that eliminates the need for cellular networks and WiFi.

Our main point of differentiation from other apps of this kind comes in the form of our backbone nodes. These nodes serve to augment the network in the case of disparate meshes. For example, consider two independent meshes of client nodes, Mesh A and Mesh B. Given the technology inherent on smartphones and our mobile application, members of Mesh A are able to communicate with one another. Yet due to the distance between the meshes, those in Mesh A cannot get messages to those in Mesh B. This is where our backbone nodes come in: when a backbone receives a message, it is able to propagate it to other backbones as well as to other clients. This allows messages to go from Mesh A to Mesh B.

7.7 Competition

There is a myriad of competitors that can be deployed in response to a disaster situation, each using different approaches to address the lack of communication. Though they exist, none of these resources have been able to obtain significant success in remedying the ailments of a post-disaster situation.

Cell Towers are a traditional source of competition, creating the pre-existing communication infrastructure. Although these create a strong network resistant to congestion, they are vulnerable and often destroyed during disasters. *Mobile Cell Towers* are their replacement during such situations, and when deployed they can bring the network back up by taking the place of the destroyed cell towers. This "solution" is costly and relies on the cooperation and responsiveness of the cellular networks in order to quickly and efficiently repair the infrastructure. Another potential option from network providers would be to deploy *COWs*, or *Cell on Wings*, to affected areas. Albeit a robust solution, these flying drones are prohibitively expensive.

Walkie-Talkies and *Satellite Phones* are a more easily deployed and economically feasible in comparison to the cell towers, though they are limited in how many individuals they can support. Members of a community may obtain access to a satellite phone but are likely to be required to share, limiting its functionality.

More accessible alternatives include applications like Peer-3-Peer that cater to more modern advancements in technology, specifically mobile devices. The *Serval Project* leverages a WiFi mesh network to connect individuals but is limited to areas with WiFi connectivity. *Signal* is another application used to connect individuals when cellular is not an option, but it relies on access to the internet to transfer messages and establish connections.

One of our strongest competitors is *FireChat*, which like Peer-3-Peer uses Bluetooth and WiFi to create a mesh network between individuals and thus does not require access to cellular networks or the internet. Communication is guaranteed between those in the mesh, though FireChat does not support communication outside of the mesh like Peer-3-Peer does.

Finally, *goTenna* competes with Peer-3-Peer in the area of hardware augmented communication. Like Peer-3-Peer, the hardware nodes help create more robust mesh networks, but unlike Peer-3-Peer the business model is not user-friendly: the devices cost more than twice as much as Peer-3-Peer's and the application is monetized such that features are restricted without a paid annual subscription.

For a graphical representation of the competitive landscape, please see Appendix B.

7.8 Intellectual Property

The intellectual property we would secure is the design of the hardware nodes and the network protocol. The mobile app portion will be freely distributed.

7.9 Costs

The mobile application needs to be maintained and updated to handle any issues uncovered over time.

The hardware costs include the board itself and its packaging. We've selected Weatherproof Injection Molded ABS Casing for its durability and will incur a $20-30K tooling cost and a recurring unit cost of $1.00 in order to package our backbone nodes. The board used in our hardware nodes is the $32.00 CL-SOM-iMX7 NXP i.MX7 System-on-Module. This yields a final cost of $33.00 per unit.

7.10 Revenue Model

We have explored numerous potential business models, each of which have different costs and benefits of implementing. The optimal model is to (1) make the application free to download and use and (2) sell or rent the backbone nodes to local governments and NGOs.

The free nature of the application eliminates barriers to adoption, which will help in creating larger, more effective meshes. Any model that required purchase of the application was ruled out due to the fact that purchasing the app is a disincentive for the public to use it. To optimize message transfer, our goal is to get as many people as possible using the application, especially during a time when a disaster has struck. This means that we want user buy-in to be as high as possible.

Since our target use case is disaster relief, governments will have the incentive to make the network as robust as possible in order to reach all affected individuals. This incentive validates their need to purchase the backbone nodes.

For a full breakdown of the financial analysis, please see Appendix C.

8 Conclusion

Peer-3-Peer is a disaster relief communication solution that leverages peer-to-peer, enhancing it with hardware radios, to enable unprecedented connectivity and coverage while maintaining ease of deployment. The contributions of this system are a client application and backbone nodes. The clients communicate locally via a short-range protocol (e.g. Bluetooth), and the backbones are engineered with a chip to support the desired long-range protocol (e.g. Zigbee, WiFi, LoRa). Our distributed system constructed with multiple types of nodes and multiple types of protocols is a relatively unexplored approach, and so we have designed a custom protocol for routing and messaging. Peer-3-Peer will enable communication for governments, aid organizations, and civilians, improving the efficiency and effectiveness of disaster relief.

Acknowledgements The authors would like to thank their advisor, Dr. Vincent Liu, for his guidance and support. Additionally, they would like to thank Professor Ani Nenkova and the University of Pennsylvania's School of Engineering Computer and Information Science Department for their support.

References

Bateman, Connie, Valentine, Sean, and Rittenburg, Terri. 2013. Ethical Decision Making in a Peer-to-Peer File Sharing Situation: The Role of moral Absolutes and Social Consensus. *Journal of Business Ethics,* Vol. 115 Issue: 2. 229-240.

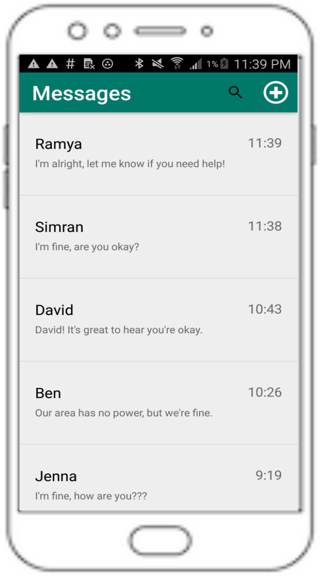
Christin, Nicolas. 2010. Peer-to-Peer Networks: Interdisciplinary Challenges for Interconnected Systems. *Information Assurance and Security Ethics in Complex Systems: An Interdisciplinary Perspective.*

Kovács, Gyöngyi and Spens, Karen M. 2007. Humanitarian Logistics in Disaster Relief Operations. *International Journal of Physical Distribution & Logistics Management*, Vol. 37 Issue: 2, 99-114.

Ritchie, H. and Roser, M. 2019. Natural Disasters. [online] Our World in Data. Available at: https://ourworldindata.org/natural-disasters.

Statista. 2019. Mobile OS market share 2018 | Statista. [online] Available at: https://www.statista.com/statistics/266136/global-market-share-held-by-smartphone-operating-systems/.

Appendix A



Messages

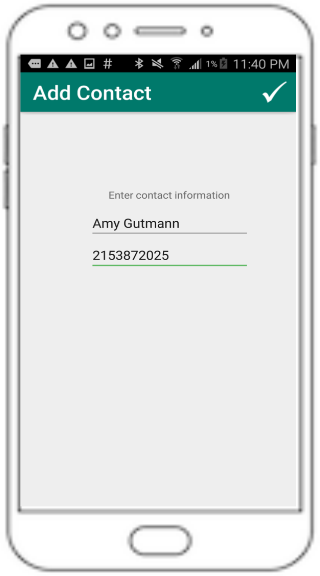
Ramya 11:39
I'm alright, let me know if you need help!

Simran 11:38
I'm fine, are you okay?

David 10:43
David! It's great to hear you're okay.

Ben 10:26
Our area has no power, but we're fine.

Jenna 9:19
I'm fine, how are you???

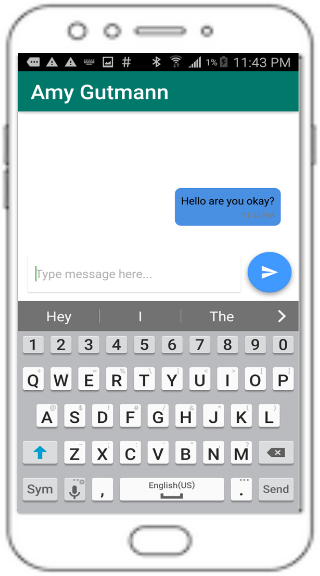


Add Contact

Enter contact information

Amy Gutmann

2153872025



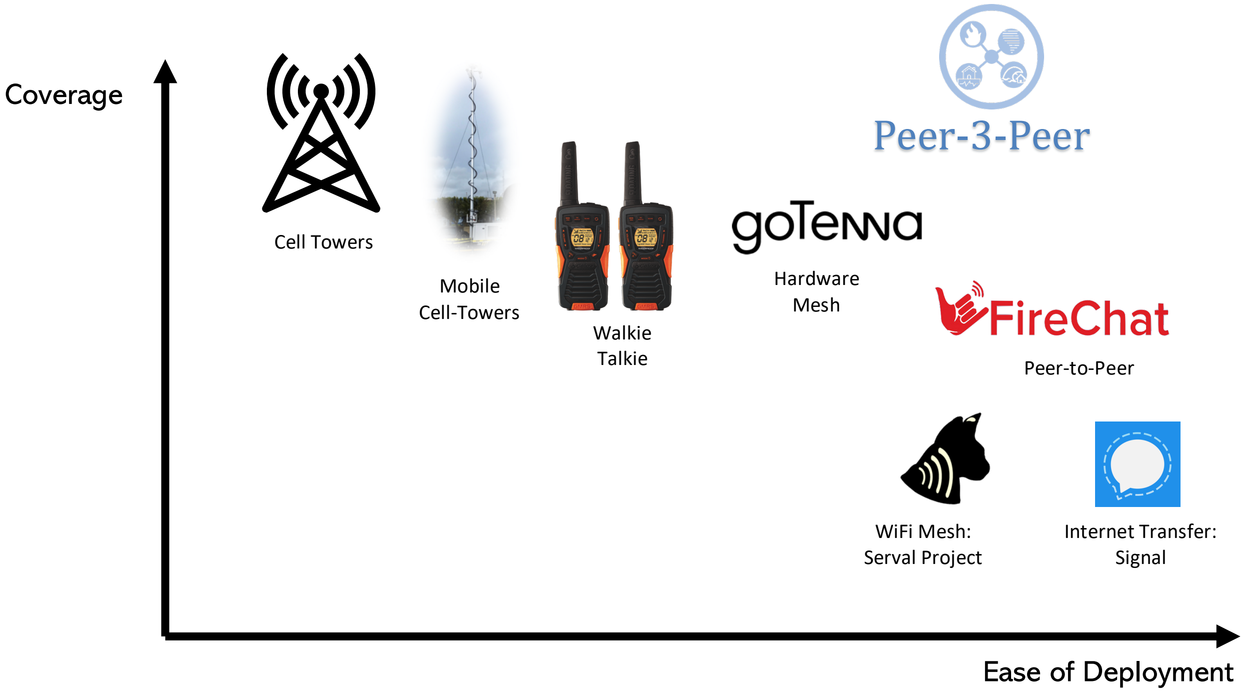
Amy Gutmann

Hello are you okay?

Type message here...

The above captures the user interface for different functions of the application. The screen on the left is what the user sees when opening the address book. When adding a new contact, the user will encounter the middle screen. The screen on the right captures the look of Peer-3-Peer's chat interface when reading messages between the user and a particular friend.

Appendix B



We observe our competitive landscape on the two dimensions of coverage and ease of deployment. The high coverage solutions are hardware intensive and require extensive setup beforehand. Even walkie talkies need to be purchased and distributed widely before a disaster. Peer-to-peer applications on the right are easy to deploy but have low coverage because of large gaps between clusters. Peer-3-Peer has high coverage because of the hardware nodes but can be easily deployed on the scene of a disaster.

Appendix C

Backbone Coverage	
WiFi Direct Range (mi)	0.125
Tree Depth	3
Backbone	1
Radius/Backbone (mi)	0.5
Area/Backbone (sq. mi.)	0.8
Disaster Region (sq. mi.)	1872
Backbone/Disaster	3743

Revenues	
Cost/Backbone	$33
Profit Margin	50%
Target Price/Backbone	$49.50
Revenue/Disaster	$185,295.00
Disasters/Year	325
Revenue/Year	$60,220,875.00

The above analysis determines the revenue per year calculated for Peer-3-Peer. First the technical metrics of our system, including the distance at which two phones can communicate, are used to determine the disaster area covered by a particular backbone node. Our protocol generates a tree structure from the Bluetooth mesh, and although the protocol can support routing and messaging for a tree of any length that forms, we conservatively estimated a tree of depth three in our calculation. Next, we considered the average area affected by disasters for a proportional mix of suburban and urban, localized and widespread disasters in 2017-2018, determining an estimate of 1,872 square miles affected. Using this value, we arrived at an estimate for the number of backbones required per disaster. Finally, to calculate the revenues, we seek to operate as a nonprofit organization. Considering the cost to produce our backbone nodes and their weatherproof casing, we apply a conservative 50% profit margin to arrive at a price of $49.50/node. We use these values to arrive at an annual revenue estimate to sustain growth and high-quality operations.