Low-Infrastructure Methods to Improve Internet Access for Mobile Users in Emerging Regions

Sibren Isaacman
Dept. of Electrical Engineering
Princeton University
isaacman@princeton.edu

Margaret Martonosi
Dept. of Computer Science
Princeton University
mrm@princeton.edu

ABSTRACT

As information technology supports more aspects of modern life, digital access has become an important tool for developing regions to lift themselves from poverty. Though broadband internet connectivity will not be universally available in the short-term, widely-employed mobile devices coupled with novel delay-tolerant networking do allow limited forms of connectivity. This paper explores the design space for internet access systems operating with constrained connectivity. Our starting point is C-LINK, a collaborative caching system that enhances the performance of interactive web access over DTN and cellular connectivity. We discuss our experiences and results from deploying C-LINK in Nicaragua, before moving on to a broader design study of other issues that further influence operation. We consider the impact of (i) storing web content collaboratively cached across all user nodes, (ii) hybrid transport layers exploiting the best attributes of limited cellular and DTN-style connectivity. We also explore the behavior of future systems under a range of usage and mobility scenarios. Even under adverse conditions, our techniques can improve average service latency for page requests by a factor of 2X. Our results point to the considerable power of leveraging user mobility and collaboration in providing very-low-infrastructure internet access to developing regions.

Categories and Subject Descriptors

C.2.1 [**Computer-Communications Networks**]: Network Architecture and Design; H.3.4 [**Information Systems**]: Information Storage And Retrieval—*World Wide Web (WWW)*

General Terms

Performance, Evaluation

Keywords

Caching, Delay Tolerant Networking, Simulation, Mobility

1. INTRODUCTION

The internet is a vital communication pathway throughout the developed world. While in North America, 80% of adults have internet access (either at home, work, or in public areas), the world averages roughly 20% internet access,

and some countries are as low as 5% [14]. Closing this "digital divide" is important for developing regions to become competitive on the world stage [15]. Bringing technology to developing regions has even been named one of the "Millennium Development Goals" by the United Nations [22].

Difficulties abound, however, in bringing internet connectivity to developing regions. Wired solutions are costly and impractical to install [3] and wireless solutions are often plagued by difficulties in erecting towers, licensing spectum, and other hidden costs to deployment. Nevertheless, delay-tolerant networks (DTNs) and low-bandwidth wireless approaches to connectivity are gaining traction even in largely-disconnected areas. While such approaches clearly work well for applications that are themselves delay-tolerant, interactive applications such as web-browsing are not yet well supported.

Our work is exploring methods that harness collaborative caching and hybrid network strategies to decrease the latency of interactive web access experienced by users in poorly-connected areas. By allowing peer nodes within a region to share cached web pages with one another, we have a low-infrastructure method to reduce the data transfers into the region and improve latency. In a previous deployment in rural Nicaragua, we demonstrated that our prototype C-LINK system—agnostic to both the browser type and the network connectivity—greatly improved internet search experience [8, 9].

Building on our past experiences, this paper explores broader issues that influence the design of a truly-robust next-generation C-LINK. In particular, we simulate different user request traces with different node mobility models to see how resilient C-LINK is to collaborative nodes moving into and out of range of each other. In general, nodes may be out of range either because of mobility of the users or intermittent power or usage of the device, however, the net result is the same: inability to access data. In addition, we explore the value of hybrid connectivity that combines high bandwidth but high latency delay-tolerant network (DTNs) with low-bandwidth, low-latency cellular connectivity.

We examine two important sources of mobility. First, the mobility of the mechanical backhaul must be considered. Vehicles may carry data with varying frequencies and with varying round trip times. Adding cellphones into the network allows for cases of nearly-instantaneous transport in one or both directions. Second, the increase in mobile user nodes (cellphones and laptops) in the region means that the devices participating in the collaborative cache may themselves not be stationary and, as such, may move out of range and therefore no longer be able to participate in the cache.

In exploring mobility scenarios, this paper shows that collaborative caching strategies perform well even with highly mobile nodes. With collaborative caching, the caching space available can gracefully and naturally increase as more nodes appear in the system, and can gracefully decline if nodes leave the area or are forced to power down. Selective shutdown and gradual deployment offer economic advantages that are particularly important in the environment we target.

Through the design and testing of our system, we have:

- Demonstrated the efficacy of the C-LINK collaborative caching system though both real deployment and simulation. The collaborative cache allows up to 85% lower miss rate in the cache over traditional cache strategies, allowing users to experience lower average web access latencies and greatly reducing non-local traffic requirements. For stationary users, collaboration drops effective miss latencies by 10X.

- Explored, through simulation, the impact of node mobility and related connect/disconnect patterns on collaborative caching. Even in such scenarios, collaboration offers 35-89% performance improvements over non-collaborative approaches.

- Quantified the benefits of a hybrid network in which small data packets and search requests are sent over a cellular network, while larger responses return over a DTN. In simulation, the hybrid network offers a 2X reduction in page access latency compared to the DTN-only network while transmitting only 16% of the data over the cellular connection.

- Shown the utility of the system by simulating traces gathered from a region of interest during a real deployment. Collaborative caching allows an improvement of 1.5X in hit rate over non-collaborative approaches.

The rest of the paper is organized as follows. Section 2 provides a motivation for collaborative caching and discusses our early experiences with a collaborative caching solution. Section 3 provides the experimental methodology used in examining a broader context for collaborative caching. We then explore in detail two factors affecting collaborative caching's behavior. In Section 4 we explore the mobility of the nodes, while in Section 5 we propose and evaluate a hybrid network solution for data transport. Section 6 presents an analysis of the behavior of our real world deployment under different mobility and network conditions. Section 7 presents related work, and in Section 8 we draw conclusions about the system and suggest future improvements.

2. COLLABORATIVELY-CACHED INTERNET ACCESS

In this section, we explore the general idea of collaborative caching and our past experiences with it. Using this as motivation, Section 2.4 poses further questions for the remainder of the paper to answer.

2.1 Idea and Motivation

As the number of networking-capable devices increases in developing regions, solutions must be developed to connect the devices to the broader network. Currently, many of these solutions are very low-bandwidth, very high-cost, very high-latency, or a combination of the three. Thus, it is attractive
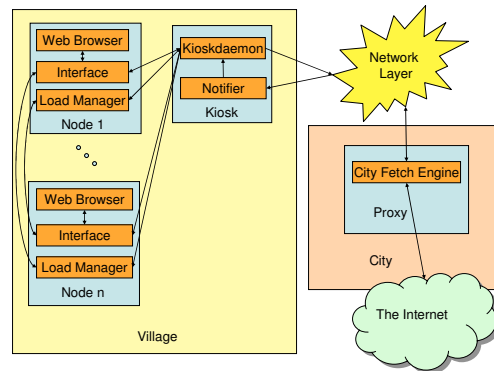

Figure 1: Diagram of the system. Nodes in the village are well connected (when the connection is functioning). The kiosk, in turn, is connected over a high latency (or high cost) link to the city, which in turn is connected to the internet.

to cache as much data as possible in an effort to prevent trips over whatever network link is available.

It is not clear, however, that a large, monolithic cache at some central point is the best solution. Other solutions are preferable for two reasons. First, there are reliability concerns that need to be addressed. In regions of the world where power is unstable, trusting a single machine to be up at all times is risky. If the cache fails, the users are left with nothing. Second, the devices in the regions may be highly mobile. Cellular phones are the most prevalent computing device in these areas and many other attempts to bring computing power to these regions focus on some form of ruggedized laptop. As users take their devices from region to region, a centralized cache would no longer be accessible.

Collaborative caching has the potential to overcome some of the issues posed by monolithic approaches. In such a cache, each user is responsible for a portion of the cache and is able to access content directly from other users. With collaborative caching, if a single node fails, the rest are still able to access the majority of the data. Similarly, as nodes move in and out of range of one another, they are always able to access at least some fraction of the data (and can always access their own portion of the cache). In fact, a collaborative solution offers two secondary benefits. First, the size of the cache will scale gracefully. As nodes come and go, the cache expands and contracts, giving more capacity as more is needed and less when less suffices. Second, allowing users to keep data locally offers a secondary source of content dispersal to supplement whatever other network layer is available.

2.2 C-LINK Prototype

To begin to demonstrate the functionality of the collaborative caching approach, we developed a system, C-LINK, making use of collaborative caching to provide generic web access to a rural areas.

Figure 1 is an abstract view of our system. The *village* is composed of devices (*nodes*) well-connected to a router (*kiosk*) which is in turn connected via a high latency to the *city* in which there is a proxy to the internet. The name C-LINK is derived from the five programs which make up the C-LINK system. These programs are: *City Fetch Engine*, *Load Manager*, *Interface*, *Notifier*, *Kioskdaemon*.

From the user's point of view, the system operates transparently. All that must be done is to configure the web browser on the node to use the Interface as a proxy. When a page is requested, the Interface queries the kiosk for the location of the page in the collaborative cache. If the page is available, it is served directly from the node containing the page. A kiosk node must always be available. If one is not, a new kiosk will immediately be spawned and a new index created by polling all available nodes.

In the event that the page is not available, users receive a message informing them that the page will be available at a later time. In the background, the kiosk places a request for the web page onto whatever network the system is configured to use (e.g., cellular, DTN, wired, etc.). This request eventually passes to the well-connected city, where the City Fetch Engine acts as a proxy for the node in the village. The City Fetch Engine may also be configured to prefetch data, if appropriate.

Once the request (and any prefetched data) returns to the village, the data is placed on the node that originally requested the data (now the *owner* of the data). Future requests for the data are served by the owner. Further details on the design of the system can be found in our previous workshop paper [9].

2.3 C-LINK Deployment Experiences

We deployed C-LINK across 5 machines in a library in a small town in western Nicaragua named Cinco Pinos. The well-connected "city" in our deployment was Somotillo, Nicaragua, roughly an hour's drive away. Data was transferred once a day when we made the drive from Cinco Pinos to Somotillo and back.

In our deployment, the City Fetch Engine was configured to perform aggressive prefetching of content, because the DTN is low-cost, high-bandwidth and high-latency. The low cost and high bandwidth make it worthwhile to send as much data to the village as possible in the hope it will be used. This helps mitigate the high latency. However, other configurations may opt to prefetch more judiciously.

Thirty-one teenagers and young adults visited the library during the 5 days of our deployment. Users were encouraged to browse and search the web on a variety of topics including health and medical issues and small business opportunities (particularly related to cooking and sewing). In addition, users also browsed on topics of their own choosing (with a heavy bias toward song lyrics!). Of the top-20 sites visited, 10% of them represented pages on assigned topics, while the remaining 90% were either casual browsing, ad sites, or pages we cannot easily categorize. Response from users was extremely positive, with most users requesting more time on the machines than we had slots to give them.

2.4 Questions Raised and Next Steps

This section synthesizes the lessons learned from C-LINK thus far, and poses remaining research questions that subsequent sections of the paper will address.

Collaborative caching is specifically designed with disconnections and mobility in mind. This makes C-LINK a natural candidate for use with mobile nodes. However, in an effort to maintain a controlled experiment, our original Nicaragua deployment consisted of purely stationary nodes. Although we were able to observe that the system functioned despite node failures and power outages, we were not able to explore node movement in any systematic way. What effect would mobility have on a collaborative cache?

Also, for simplicity in our initial deployment, we personally drove the route from Cinco Pinos to Somotillo and back once a day. However, there are five buses a day between the locations, so ongoing C-LINK service should use such resources instead. How, then, might these more-frequent buses have affected service provided by a collaborative cache?

In addition, like an increasing number of rural villages, Cinco Pinos is well-covered by a cellular tower in the middle of town. Although it might be prohibitively expensive to rely solely on cellular connectivity for high-data transfers, it is interesting to consider how cellular can be used in hybrid strategies. If we were to send requests out using SMS and await the reply on the buses, how much could be gained?

During the course of our deployment, we collected a trace of pages requested by our users. Such a trace is valuable for several reasons. First, it provides a look at real access patterns from a region of interest. Second, it offers a trace of generic internet usage as opposed to a specific application. This trace is one of the workloads we use to explore design decisions and answer the questions about next steps that we are posing.

3. METHODOLOGY

In order to answer broader questions about collaborative caching beyond those raised by our first deployment in Nicaragua, we use a simulator that allows us to "modify the deployment" before we return to the field. After Section 3.1 gives an overview of the simulator designed for this task, Section 3.2 explains its policies for cache management. Section 3.3 discusses modeling the underlying connectivity and mobility of the nodes. The traces used are described in Section 3.4.

3.1 Simulator Description

The simulator is a trace driven model of the system described in Section 2.2. Each page access in the trace is considered to be a request from a user for a page. For each entry in the trace we have the requested URL, the ID of the node making the request, and timestamp information. For each trace entry we must consider both the caching situation of the requested data (is it present in the village or not, and if present, what node holds the data?) and the connectivity situation of the relevant nodes (are the relevant nodes currently within communication range of the requesting node?).

Since we assume no pre-loaded data, the first request of a particular URL in the trace is presumed to be a "miss". Upon the first appearance in the trace, the request is placed into a waiting queue. The latency to service the request depends on the connectivity being simulated, as described in Section 3.3, below. Subsequent requests to the pages result in misses if the waiting queue has not been emptied. When the waiting queue is emptied, the current time and recorded request times are used to calculate the latency of page retrieval. The page is then placed in the cache of the original requester and the cache utilization of that node is updated. Subsequent accesses to the page result in a hit being recorded along with an access latency of 0 (since deployment experiences show latency to be less than a second and even at scale would be far less than a minute, our smallest granularity).

The simulator tracks not only pages, but also the nodes involved. Nodes are considered to appear in the village upon

their first appearance in the trace. This is a reasonable model, as the kiosk will not know that a node is willing to participate in the collaborative cache until a request has been issued. To account for mobility of users, future encounters of the node in the trace are considered to be connected or disconnected (available or unavailable) based on the connectivity model given in Section 3.3.

3.2 Collaborative Cache Parameters

Our simulator allows us to examine the effects of cache size and management on the performance of the collaborative cache. However, in this study, we focus mostly on mobility. Thus, at the beginning of the simulation, a fixed size is given as a cap on the amount of storage available at each node.

In addition to cache space on each village machine, we can independently control the cache space available to the kiosk, thereby allowing us to determine the influence of pages that are cached on the kiosk. Because we are focusing on low-infrastructure scenarios, we envision the kiosk to be much like one of the nodes. For simulations in this paper, we fix storage space on the kiosk to be slightly greater than the storage of a "normal" node. It is not intended to be a large central data server because we are interested in exploring connectivity options that do not require such a single well-endowed machine. In a real deployment, the machine on which the Kioskdaemon runs is simply a machine that is more likely to be "always on." This machine runs the special software, but is in no other way specially configured. If the kioskdaemon's machine fails at any point, the other peer nodes detect this and dynamically elect a new collaborating machine to begin running the kioskdaemon.

Finally, our simulator allows us to turn the collaborative cache off. When collaborative caching is turned off, a node is unable to access data stored on a different node. This includes cases in which a local cache is full, meaning the page must simply be deleted. Further, requests made by a node for a page in the waiting queue result in a new entry in the queue. Thus, a node is not able to piggyback off a request made by another node in any way.

Eviction in the simulator is done via a least recently used (LRU) policy. Each time a page is accessed at a node it is moved to the end of the LRU queue. When the cache limit is reached, the pages at the front of the LRU queue are placed onto the emptiest node in the pool. If all nodes are full, the page is deleted. When a request is made for a page that has been deleted, a new request is made and a miss is recorded.

3.3 Modeling Connectivity

Because the simulator models DTNs and hybrid networks as well as node disconnections, additional functionality must be added. If a request is made from a device that is currently out of range of the kiosk, the request fails. In this case, the request is marked as a hit only if the page is cached locally on the node making the request. Conversely, if a request is made by a node within range but the owner is out of range, the request is marked as a miss and the page is re-requested from the city. (This also happens in the real system in case of power failure.) Finally, if a reply is received but the requestor is out of range, the page is stored at the kiosk until the requester moves back in range. Other nodes attempting to access this data can have it served from the kiosk unless the kiosk itself runs out of storage space and must evict the page or the kiosk fails.

The latency at which requests are handled also depends on the connectivity. For DTNs, we simulate a system in which the request queue is emptied on some parameterized interval (e.g., the default of once per hour). Once the queue is emptied, it takes some time for the response to return. This response latency is another configurable parameter, again with a default of one hour. Thus, for the default DTN, the length of time a miss takes to be serviced is bounded on the bottom by 60 minutes (if the request is made immediately before the queue is emptied) and on the top by 120 minutes (if a request is made immediately after the queue is emptied). These parameters are set to abstractly reflect the latency of using vehicular transport to move data between an internet-connected city and a rural user device.

We model mobility and node connectivity using a probabilistic approach. Once a node appears, it remains "in range" of the village (and its kiosk) based on a time selected from a given probability distribution. The node is then considered to be "out of range" for a time selected from a separate, independent distribution. These two distributions can be chosen to match different characteristics. The node continues cycling in and out of range for the duration of the trace, each instance selecting times from the appropriate distribution. Nodes that are able to communicate to the kiosk are assumed to be able to communicate with one another. This abstraction seems reasonable for the small villages we target.

Section 4.1 describes the particular connectivity and mobility model statistics that we explore for this paper.

3.4 Description Of Traces

The traces we used are from two distinct workloads: a request trace from Princeton's educational courseware servers, and a request trace from our C-LINK deployment in Nicaragua. The information available to use in both trace files was: anonymized requester IP address, anonymized URL requested, timestamp, and number of bytes returned from server. Section 3.4.1 describes the large-scale trace from Princeton and Section 3.4.2 contains information regarding the trace taken from our Nicaraguan deployment.

3.4.1 Blackboard

Our first workload is a trace taken from Princeton University's Blackboard servers, which represents a typical education application. The traces we analyzed from Princeton's servers were collected over a 24 hour period in March 2008. All entries were fully anonymized before we received them. Over the course of 24 hours, 4,470 unique nodes appear in the trace file. The heaviest user requested just over 127MB of unique data. In this trace, 700,000 URLs are requested, but only 200,000 of these are unique URLs, implying very high repetition rates and potential for caching. This educational usage fits well with a vision for rural internet drivers. Because of the size and continuity of this trace as compared to the smaller trace from the real system, we use this trace in Sections 4 and 5 to explore the general characteristics of a collaborative caching scheme.

3.4.2 Real System

Our second set of traces come from our deployment of the system in Cinco Pinos, Nicaragua. These traces were collected over the course of a week in May 2009. The trace consists of 5 nodes, though dozens of individuals used the machines. In all, nearly 30,000 URLs are requested of which over 7000 are unique. In total, 104MB of web data was transferred from the city to the village, including prefetched
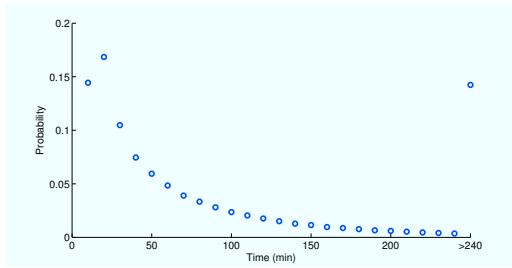
Figure 2: Distribution of time spent in range, "Quebec-Real". The tail continues to 26243 minutes but the graph stops at 4 hours for clarity. 14% of all connections last longer than 4 hours.
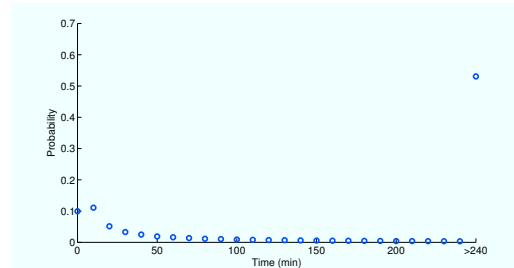

Figure 3: Distribution of time spent out of range, "Quebec-Real". The tail continues to 1377512 minutes but the graph stops at 4 hours for clarity. 53% of the disconnections are longer than 4 hours.

data. Section 6 uses this trace to explore design decisions in the specific context of a likely deployment.

4. NODE MOBILITY

Many of the computing devices in developing regions, from cellular phones to ruggedized laptops, will be mobile, and most connectivity models—wifi, cellular, and others—will allow for mobility. Thus, designs for rural connectivity should account for mobility and where possible, even exploit it as we show here. Section 4.1 discusses the mobility models for which we explore our approach. Section 4.2 then demonstrates the effects mobility has on the system and the gains that can be made when collaborative caching is used. The simulations presented in this section are all done using the Princeton trace because of its scale.

4.1 Incorporating User Mobility

Though the original deployment involved purely stationary machines, real systems must include mobile devices that are only intermittently available. To explore these issues, we consider three movement models: "Stationary", "Quebec-Real," and "Quebec-Uniform."

4.1.1 Stationary Model

In the "stationary" movement model, all nodes are always in range. They enter the village as soon as they appear in the trace file and never leave. Under this movement model, requesting nodes are always available, never disconnected. As such, once requested data arrives in a village, it is only transiently stored on the kiosk until it can be transferred to the user device. In the simulation, this happens quickly enough that the functional appearance is that data is never found to be stored on the kiosk.

4.1.2 Quebec-Real

As in the all models, when a node first appears in the trace under the "Quebec-Real" node mobility model, it is defined to be in range of the kiosk. Henceforth, the node remains in range for a duration selected from one probability distribution and then is out of range for a time selected from a separate, independent distribution.

For the "Quebec-Real" mobility model, we derive our in-range and out-of-range probability distributions from real-world wireless connectivity data collected in Quebec, Canada and archived at the CRAWDAD repository [13]. This data set provides a list of connection lengths between mobile devices and access points which were used as the "in range" times for the "Quebec-Real" movement model. Using the same data set, we were able to create the "out of range"

times by looking at the time between successive connections. In 10% of the cases, this time was negative (indicating overlapping connection) in which case we used 0 as the length of disconnection, effectively modeling two back-to-back connections. The distributions for connection and disconnection times are presented in Figures 2 and 3, respectively. The average connection time is 6544 minutes. The average disconnection time is 362 minutes.

4.1.3 Quebec-Uniform

The problems with the "Quebec-Real" data are twofold. First, it is too long; the data cover a time span much longer than anything considered in either of our two traces. Second, the trace represents a fundamentally different mode of travel that exists in a rural environment. In an urban environment there is a lot more mobility possible than in a rural environment. Thus, at the extremes it is possible to be out of range for much longer durations than our experience in developing regions shows is common. In fact, the extremely long tail of the disconnection distribution shows that long disconnections happen frequently.

"Quebec-Uniform" represents a second point in the mobility spectrum, without the above problems. The most compelling point of the urban trace was that in general, connections lasted longer than disconnections. Thus, we used the mean value of connection and disconnection time (6544 and 362 minutes, respectively) to create a distribution that had smaller extremes than the urban trace and more evenly spaced contacts.

In "Quebec-Uniform," the C++ pseudo-random number generator is used to generate a random length of time for each connection and disconnection. The model selects randomly from the uniform distribution centered around 6544 minutes (i.e., uniformly from 0 to 13088) for connections and 362 minutes (i.e., uniformly from 0 to 724) for disconnections. Of course, "Quebec-Uniform" is not a perfect representation of rural movement either; it simply represents a markedly different point on the mobility continuum and is thus worth exploring more fully. Future work into real mobility in rural regions can help determine where between "Quebec-Real" and "Quebec-Uniform" users lie.

4.2 Effects of Mobility

We examine the effects of mobility by looking at two metrics of interest. First, in Section 4.2.1, we examine the miss rate seen by users. This gives a feel for how frequently user are able to access content they are interested in. Second, in Section 4.2.2, we examine latency. Here we examine how long, on average, a user would have to wait for a page.
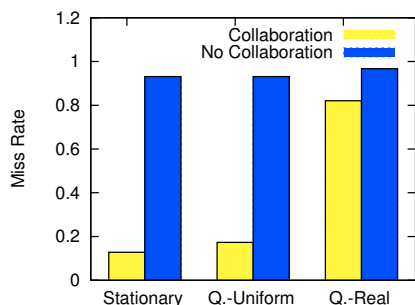
Figure 4: Effect of node movement model on hit rate. Quebec-Uniform and Stationary nodes see 5.3X-7.1X improvements in miss rate. Even the 15% gains of "Quebec-Real" represent significant gains in functionality.

4.2.1 Miss rate

Figure 4 shows the decrease in miss rate, as compared to caching at individual nodes, offered by a collaborative cache under the movement patterns examined. Collaborative caching improves miss rates anywhere from a 15% to a 7.1X improvement. Although miss rates for "Quebec-Real" movement remain fairly high, the improvement is significant enough to greatly increase the usability of the system as a whole. Miss rates for the "Stationary" and "Quebec-Real" movement models drop to 12% and 17% respecively.

Differences in miss rate performance can be understood by examining the mobility models. Although the two models derived from the Quebec data have the same mean, the very long tail in the "Quebec-Real" model greatly influences the miss rate. Because our web access trace covers a shorter period of time than the mobility distribution trace, the result of the tail is that in many cases, a node that leaves the system never returns. In contrast, the "Quebec-Uniform" model is better suited to our usage scenarios, with nodes less likely to leave permanently.

Even under the extreme conditions of "Quebec-Real", collaborative caching is a viable way to reduce the miss rates for web browsing in low-infrastructure environments. We validate that collaboration is beneficial even in mobile situations in which a traditional cache could not operate.

It is important to note that despite the successful performance, the model simulated is a pessimistic scenario. In a real system, two nodes that are both out of range of the kiosk but still within range of one another are still able to exchange data, though this does not occur in the simulator.

Future deployments of a collaborative cache could further exploit mobility. Cache management in the simulator (and prototype system) is simply LRU. However, if nodes are able to sense that they are leaving their current locations, intelligent redistribution of the cached data could also serve to increase hits in the cache. Also, the simulator accounts for only one village. However, nodes moving between villages can serve as data relays that would increase the amount of data available in the cache and thus decrease miss rate.

4.2.2 Latency

Nodes moving in and out of the system have a significant impact on the average latency to access a page. Figure 5 demonstrates that here, too, significant gains are possible. Even under the challenging "Quebec-Real" mobility model, latencies are reduced more than 35% when comparing collaborative caching against caching at individual nodes. The
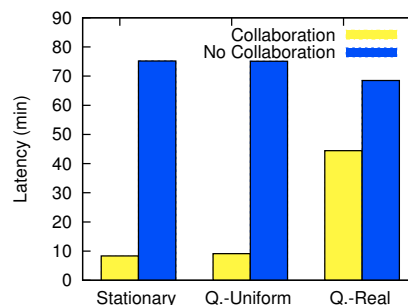

Figure 5: Effect of node movement model on latency. Reductions in latencies of 35%-89% are achievable, despite nodes frequently moving out of range.

other movement models with fewer permanent node departures allow latency reductions of over 8X. With no collaboration in place (i.e., caching only at individual nodes), a user would have to wait an average of 68 minutes or more to access the data they had requested. However, with the benefit of a collaborative cache, a stationary node would only have to wait an average of 8.4 minutes and even the frequently disconnected "Quebec-Real" nodes need to wait an average of only 44 minutes.

A collaborative cache is designed to gracefully scale both up and down as nodes enter and leave the area. We see here that the system allows this while maintaining a reasonable latency, given the alternatives. Although a 9-44 minute delay seems excessive in the fully connected world, it is a marked improvement over non-cached access. By reducing the wait by a half or more, the system becomes a viable way for a user to interact with remote content.

Interestingly, collaborative caching provides greater leverage in reducing latency than in reducing miss rate. This occurs regardless of the mobility of the nodes. In addition, the relationship between latency and miss rate is not linear. Thus, it is not sufficient to measure miss rate or latency alone as each provides insight into the user experience.

5. HYBRID NETWORKS

The rapid growth of cellular networks in developing regions offers the opportunity to consider hybrid networks that use both DTN and cellular communication. Interestingly, cellular networks offer very low latency connectivity, but also are likely to either be very bandwidth-constrained, or very expensive per-byte, or both. In contrast, DTNs are typically very low-cost and high-bandwidth (lots of data fits on a public bus), but with the downside of high latency. Our goal in this section is to explore hybrids that make the best use of both. Ultimately, we want to reduce effective data latency while minimizing the number of bytes transferred over the relatively expensive cellular link.

We focus here on a hybrid network in which page requests are sent via the cellular link, while replies return via the DTN. This is because typically the initial request for a page is only a few bytes but the reply is more substantial. A request for a page via HTTP contains no body, simply a short header containing the server and page name. Thus, the outgoing message is likely to be no more than a few hundred bytes but the return may be many kilobytes if the response is an image and many megabytes if video is sent. Even the HTML on an average web page is tens of kilobytes [16]. For example, if we consider our Nicaraguan deployment

trace and envision what would have happened with a hybrid network, only 16% of the total bytes transferred would have used the cellular link, keeping costs low.

In the section, we investigate the effect a hybrid network would have on the performance of a collaborative cache. We first simulate this with the Princeton trace because its scale and pattern of continuous access more fully test the functioning of the cache. The following Section then explores the Nicaragua trace in detail.

5.1 Modeling Hybrid Connectivity

In an effort to examine the effects of the underlying network on our collaborative caching scheme, we have created a model in which we are able to manipulate two properties of the round trip time of a request: the *leave time* and the *length of trip*. As explained in Section 3.1, when a request is not found anywhere in the village, it is placed into a queue until the request can be transmitted to the city. Thus, a request will not be serviced immediately, but is subject to the frequency with which requests are gathered and transported. The *leave time* parameter is the granularity of time at which this waiting queue is emptied. Once the queue has been emptied, *length of trip* determines the time it takes for a reply to be received back at the village. Therefore, the lower bound on the amount of time that a request must wait to receive a reply is *length of trip* (if the request is made immediately before the queue is next emptied). The upper bound is the sum of *length of trip* and *leave time* (if the request is made immediately after the queue is emptied).

These two parameters give us the flexibility to model long round trip times that occur frequently, short round trips that occur rarely, and much in between. We use this flexibility to model the various possible underlying network layers. The most optimistic network layer assumes *instantaneous* connections. In such a connection, requests are sent out as soon as they enter the waiting queue and a reply is received immediately. We can model this in our simulator by setting both the *leave time* and the *length of trip* to 0. This network layer will provide (unachievable) best-case results for any caching solution. At the other extreme, we model the bus-based DTN as a bus that arrives once an hour. By setting both the *leave time* and *length of trip* to 60, we are left with the configuration previously considered, which we here term the "DTN" configuration.

In the hybrid network, we send the initial request over the much more expensive (but faster) cellular network, and wait for the reply to come back over the DTN. To model such a network, the reply comes back over the DTN so it incurs a one-way trip time that is half the DTN's *length of trip* or 30 minutes by default. Regarding wait time granularity, the simulator assumes no delay for the request to be sent, but the reply must still wait up to an hour for the return bus. Thus, for the hybrid network, we set *length of trip* to 30 and *leave time* to 60 to model total latency to receive data.

5.2 Effects of a Hybrid Network

As with mobility, we study the network layer by looking at two metrics of interest. Section 5.2.1 discusses miss rate, and Section 5.2.2 examines latency. To isolate the network layer from effects caused by node mobility, we investigate collaborative caching under our three mobility patterns.

5.2.1 Miss rate

Figure 6 shows the miss rate of collaborative caching for different network approaches. Across the board, the hybrid
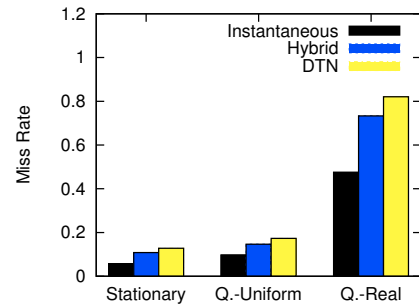

Figure 6: Effect of network type on hit rate in a collaborative cache. A hybrid network can save an extra 15% on miss rates compared to a DTN on top of the 15%-85% reductions afforded by the basic collaborative cache.

network outperforms the vanilla DTN. The hybrid network has hit rates 15% better than the DTN for Stationary or "Quebec-Uniform" nodes. Even in the extremely difficult case of "Quebec-Real," a hybrid network shows 11% improvements over a DTN. The reason that a hybrid network can improve *miss rate* rather than only latency has to do with cases where several requests to the same page happen close to each other in time. In such cases, if data is returned more quickly after the first request, there is a better chance for subsequent requesting nodes to experience a hit at times that would otherwise have been a "still-waiting" miss.

The hit rate improvements from a hybrid network come in addition to the improvement collaborative caching offers over non-collaborative techniques. For stationary nodes, improvements topped 6.7X. The instantaneous case with stationary nodes shows the best performance, decreasing miss rates from 70% to 6% of accesses. Even the worst case scenario, DTN with "Quebec-Real" movement, brought miss rates down from 97% to 82%.

In addition to an improved user experience, the decrease in miss rate due to collaborative caching and a hybrid network allows for drastic savings in the number of bytes that must be transported from city to village. When no collaborative caching is employed, the network connection must transport 2.5 GB of data. Under collaborative caching with a hybrid cellular/DTN network, only 600MB of data must be transported—bandwidth is cut by roughly 4X. This can either save money (due to reduced DTN storage or cellular air-time costs) or improve opportunities for data prefetching with "spare" bandwidth.

5.2.2 Latency

Figure 7 shows the average latency is reduced between 35% and 53% when collaborative caching is introduced into the system with "Quebec-Real" mobility and up to 90% if the nodes are stationary.

The previous section demonstrated the improvements in miss rate afforded by a hybrid network; here we show that its latency benefits are even greater. Hybrid networks cut transit times roughly in half for remote page requests, which significantly improves the average latency required to access a page. Even with no collaborative caching, a hybrid network performs 33% faster than a DTN for stationary nodes. With a collaborative cache in place, this improvement is raised to 40%. Improvements are even greater with "Quebec-Real", because of its challenging heavy-tailed mobility pattern. Under these conditions a hybrid network reduces la-
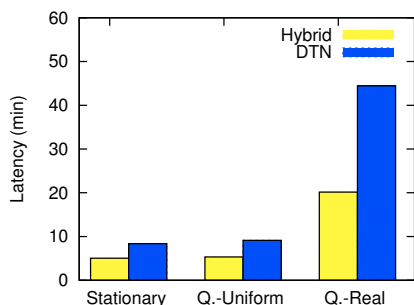
Figure 7: Effect of network type on latency. Collaborative caching improves average wait times by up to 90%. Using a hybrid network by itself can improve latency by up to 38%. Instantaneous communication incurs no latency and thus has no bars in the graph.

Figure 8: Latencies achieved using different possible network solutions. Using a hybrid network would results in a further 3.4X latency improvement over the previous deployment.

tency 38%-57%. A user with a "Quebec-Real" movement would wait an average of only 20-44 minutes to receive their data rather than the 42-68 minutes they would have had to wait without collaborative caching. A stationary user waits only 5-8 minutes on average.

The latency improvements demonstrated by the hybrid network, even in the absence of the collaborative cache argue that it is a useful technique in its own right. Only a small fraction of the traffic needs to travel over the relatively-expensive cellular connection to see vast improvements in system usability. The improvement in hit rate means that users are more likely to find the pages that they are looking for, while the improvement in average latency means that less time is spent by the user waiting for her pages to arrive.

Ideally, the hybrid network and collaborative cache will both be put in place to assist in web content delivery. Our work has demonstrated the significant gains they offer in hit rate, required bandwidth, and effective request latency. In future work, one could also explore further optimizations regarding when to use cellular versus DTN transfers. For example, one could send GET requests via SMS and save large POST requests for traditional DTN transfer.

6. DESIGN EXPLORATION FOR NICARAGUA USAGE SCENARIO

Previous sections used a large trace to generally quantify the performance advantages of collaborative caching and to consider the impact of node mobility and hybrid DTN/Cellular networks. This section now explores how the resulting design decisions work on the shorter but more realistic trace collected during our first Nicaragua deployment.

Section 6.1 begins by demonstrating the latency improvement attributable to collaborative caching. In Section 6.2 we answer our question about the network layer: "How would using the bus or a hybrid network have impacted our deployment?". Finally, in Section 6.3 we answer our question regarding the mobility of the users themselves.

6.1 Benefits of Collaborative Caching

In total, the people using the system made 28293 requests of which 17% were hits local to the machine they were currently using and another 45% were retrieved from other machines in the village. With collaborative caching, the observed miss rate was therefore 38%. By setting simulation parameters to mimic the Nicaragua deployment, we can con-
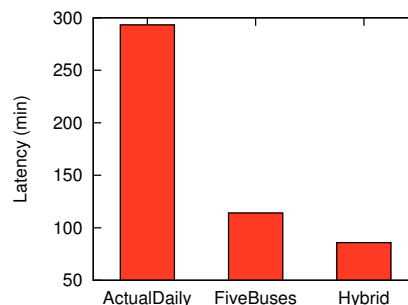
sider the non-collaborative case. Without collaboration, the miss rate is 60%. Since each miss potentially results in a large DTN miss penalty, this improvement is quite dramatic.

Of the 62% of access that were hits, 70% were on pages marked "non-cacheable". Without the benefit of a collaborative cache willing to cache them anyway (serving a stale page and fetching fresh content in the background), all of these pages would have resulted in cache misses, further increasing the miss rate and average latency to access pages that would have occurred.

The achieved miss rate is the best possible miss rate for this trace and transport layer. A monolithic cache could not have performed better, and in fact, a monolithic cache would have suffered due to power failures we experienced on our kiosk. However, because of the distributed nature of a collaborative cache, our collaborative system was still able to function.

6.2 Transport Layers

Although five buses per day travel between Somotillo (our "city") and Cinco Pinos (our "village"), we drove the route ourselves once per day to ease the deployment. Thus, our deployment offered a one-day turnaround time on web requests not serviced in the village. In this section, however, we use this trace to compare this previous deployment against a planned future one using public buses. Note that while the previous sections considered hourly buses, we focus here on a five-bus-per-day schedule that allows us to envision a real deployment as closely as possible.

Figure 8 shows the average latencies possible with other network configurations. "ActualDaily" refers to the once-a-day driving approach we actually used in our prior deployment. "FiveBuses" refers to a 5-bus-per-day schedule. In this model, a request experiencing maximum wait time would require 288 minutes of wait time between buses and 120 minutes of transit time for a total of nearly 7 hours of latency without caching. The third approach is a hybrid in which requests are sent out via cellular, but replies return via one of the five buses per day. Our results show that the once-a-day driving approach resulted in average latencies of nearly 6 hours. In contrast, the hybrid cell/bus network would return pages in under 90 minutes, despite the potential for replies to experience long wait times between buses.

Clearly, the 3.4X latency improvement offered by the hybrid network has the potential to make an already well-received system even better in future deployments. The interplay between latencies and miss rates is also important
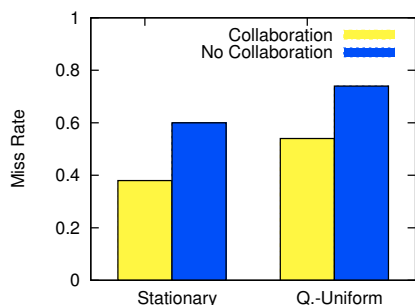
Figure 9: Miss rates given four possible configurations of the C-LINK deployment in Nicaragua.

to note. Had users been able to take advantage of the much shorter latencies provided by a hybrid network, they would have been able to make new requests for pages of interest sooner, resulting in reductions in miss rates that cannot be captured by looking at this trace.

6.3 Mobile Nodes

Our first deployment used netbooks as our user nodes, but for logistical reasons, we kept all the machines in the library, to which users were invited to come. Nevertheless, despite the lack of *actual* mobility, frequent power failures tested the "mobility" and disconnection measures that we had put into place. The trace and simulator allow us now to perform more complete studies of mobility, with users regularly entering and exiting the collaboration.

In simulating highly mobile users, we see the expected increase in miss rate. Figure 9 shows the miss rates given the four possible configurations: nodes that either move or are stationary, both with and without collaboration. Despite the mobility of the nodes, collaborative caching still offers a 22% reduction in miss rates. This indicates that though the system suffers due to the mobility of the nodes, collaborative caching retains its benefits.

Even if the users in our Nicaragua deployment had behaved in this way, we would have been able to provide a level of service significantly above what would have otherwise been possible. Access latencies would drop from 478 minutes to 336, a 30% improvement in the average time it would take a user to access a page.

7. RELATED WORK

Delay-Tolerant Networks Our exploration of collaborative caching touches on issues of delay-tolerant networks (DTNs) as well as hybrids of DTNs with cellular connectivity. In previous work, the Tetherless Computing Lab [1] and DakNet [18] have each built DTNs upon which more general internet traffic is may be carried. Additionally, some projects have explored custom hardware or protocols to forward specifically-collected data in a DTN manner [11, 17].

The Coda [12] and Rover [10] projects also considered disconnected operation, with both using local caches to hide network disconnections. However, because they required specialized file systems (and thereby code changes in applications of interest), they would be difficult to deploy widely in developing regions with heterogeneous machines. Further, these approaches do not make use of the well-connected nature of the network that may exist within the villages themselves. Our C-LINK and collaborative caching ideas

are browser-independent and can successfully exploit well-connected within-village networks when they exist.

In our first Nicaraguan deployment, C-LINK ran on top of KioskNet [1] but our ideas and our system are both agnostic to network layer. For example, Section 5 explored the possible benefits of hybrid cellular and DTN connectivity, which might employ Delay Tolerant Link State Routing [6].

Web Caching Our work clearly relates to web caching, but the classic web-caching model involves clients checking a local on-machine cache, and perhaps an upstream proxy cache. Only if not found in a proxy cache will the request be sent further upstream to the servers. Naturally, multiple levels of proxies may exist between a client and server and the proxies can be either explicitly-called or transparent [2]. For our disconnected village model, we do not assume the highly-capable proxy/kiosk that exists in the "classic" model, because we are interested in exploring whether collaborative peer-level caching can obviate the need for highly-capable large-storage proxies. In collaborative caching, caches make requests to one another for data [7, 19, 23]. If the data requested exists in a cache with which the current cache is collaborating, the data can be sent from one cache to the other, obviating the need to go further out.

Hierarchical collaborative caches such as Squid [23] are distinct from the type of collaborative caching that we propose and do not address directly the needs of the developing world, frequently breaking under disconnection. Squid, in particular, functions poorly even in the case of intermittent connections, and not at all in offline mode. As distinct from other hierarchical caches, ours assumes only a single proxy at a time, and further assumes that machine has no more storage than its peers. Except for having a weak internet connection, the proxy (kiosk) is similar to other participating nodes. Since such assumptions do not fit typical hierarchical web caching, we adopt a more collaborative, peer-to-peer approach [20]. In our model, client peers that have the requested data can serve the data rather than requiring a trip over a slower upstream link.

Though collaborative caching was found by Wolman et al. [24] to provide only limited benefits, Wolman concluded that smaller populations (<20K users) saw the most benefits. Our system is designed for classrooms and internet cafes in rural settings where user counts less than 100 per village are expected. In this space, we are the first to study the effects of collaborative caching.

User Applications in Developing Regions The ICT community has increasingly explored technologies tailored for use in developing regions. For example, the CAM project's suite of applications demonstrate a wide range of possible functionalities for rural and developing region users [17] .

More closely related to our own project, TEK [21] allows for disconnected Web Browsing via email. Similarly, RuralCafe [5] attempts to perform local query refinement to reduce the number of required trips. More recent work [4] offers a user study of how different web interaction models (RuralCafe and others) influence user behavior. Each of these approaches, however, uses a large local per-node store. By replacing these local stores with collaborative cache, we can either improve the effectiveness of the storage, or get the same effectiveness from a smaller store.

8. CONCLUSIONS

To summarize, this paper has explored the potential for collaborative web caching even in the face of challenges in-

volving high mobility and disconnection rates. By allowing participating machines to fetch webpages of interest from other collaborating peer nodes in the same village, C-LINK reduces average request latency by 2X-9X across a wide range of possible usage and mobility scenarios. C-LINK also greatly reduces the amount of traffic that travels over the constrained village-city network link. This opens up the opportunity for using predictive web prefetching to further improve interactivity for users in nearly-disconnected areas.

While traditional web-caching at a proxy or gateway continues to be useful, our work shows that collaborative caching by participating nodes offers significant benefits over kiosk storage alone. The particular values of kiosk and node storage that offer optimal performance vary with the workload, but even for workloads with only weakly-correlated accesses, collaborative node storage offers performance improvements.

As cellular connectivity becomes more widely available, hybrid networking techniques can further improve C-LINK's performance by allowing small requests to be sent over the low latency cellular network while responses return via DTN connectivity to reduce cost. Such approaches offer more than a 3X improvement in latency while requiring only a small fraction of the data to travel over the cellular network (as SMS messages). By extrapolating from real-deployment measurements into more forward-looking scenarios, this paper offers concrete results on the potential of hybrid networks and collaborative caching across a broad design space.

Overall, our prototyping results and scale-out simulations demonstrate the value of using systems like C-LINK in mostly-disconnected regions of the developing world. While DTNs alone have offered impressive improvements for users of email and low-interactivity applications, our ideas are important for improving the performance of interactive web-based software that in turn can lead to important educational and information advances.

9. ACKNOWLEDGMENTS

10. REFERENCES

[1] A.Seth et al. Low-cost communication for rural internet kiosks using mechanical backhaul. *12th Intl. Conf. on Mobile Comp. and Networking*, Sept. 2006.

[2] G. Barish and K. Obraczke. World Wide Web caching: Trends and techniques. *IEEE Communications*, 38(5), May 2000.

[3] E. Brewer et al. The case for technology in developing regions. *Computer*, 38(6), May 2005.

[4] J. Chen, A. Dhanajay, S. Amershi, and L. Subramanian. Comparing web interaction models in developing regions. *ACM DEV 2011*, Jan. 2011.

[5] J. Chen, L. Subramanian, and J. Li. RuralCafe: Web search in the rural developing world. *18th Intl. Conference on WWW*, April 2009.

[6] M. Demmer and K. Fall. DTLSR: Delay tolerant routing for developing regions. *ACM SIGCOMM Workshop on Networked Systems in Developing Regions*, Aug. 2007.

[7] L. Fan et al. Summary cache: A scalable wide-area web cache sharing protocol. *IEEE/ACM Transactions on Networking*, 8(3), June 2000.

[8] S. Isaacman and M. Martonosi. Potential for collaborative caching in largely-disconnected villages. *2008 ACM Workshop on Wireless Networks and Systems for Developing Regions*, September 2008.

[9] S. Isaacman and M. Martonosi. The C-LINK system for collaborative web usage: A real-world deployment in rural Nicaragua. *Workshop on Networked Systems for Developing Regions*, 2009.

[10] A. Joseph et al. Rover: a toolkit for mobile information access. *15th ACM Symp. on OS Principles*, 1995.

[11] P. Juang et al. Energy-efficient computing for wildlife tracking: Design tradeoffs and early experiences with ZebraNet. *Architectural Support for Programming Languages and Operating Systems*, Oct. 2002.

[12] J. J. Kistler and M. Satyanarayanan. Disconnected operation in the coda file system. *ACM Trans. on Comp. Systems*, 10, Feb. 1992.

[13] M. Lenczner, B. Grégoire, and F. Proulx. CRAWDAD trace ilesansfil/wifidog/session/04_07 (v. 2007-08-27). Downloaded from http://crawdad.cs.dartmouth.edu/ilesansfil/ wifidog/session/04_07, Aug. 2007.

[14] Miniwatts Marketing Group. World Internet Usage Statistics and World Population Stats, May 2008. `http://www.internetworldstats.com/stats.htm`.

[15] L. Osin. Computers in education in developing countries: Why and how? *Education and Tech. Series*, 3(1), 1998.

[16] A. King. The Average Web Page. 2008. `http://http://www.optimizationweek.com/ reviews/average-web-page/`.

[17] T. S. Parikh and E. D. Lazowska. Designing an architecture for delivering mobile information services to the rural developing world. *15th Intl. Conference on World Wide Web*, 2006.

[18] A. S. Pentland, R. Fletcher, and A. Hasson. DakNet: Rethinking connectivity in developing nations. *IEEE Computer 37*, 1, Jan. 2004.

[19] A. Rousskov and D. Wessels. Cache digests. *Computer Networks and ISDN Systems*, 30(22-23), Nov. 1998.

[20] U. Saif et al. Poor man's broadband: Peer-to-peer dialup networking. *ACM SIGCOMM Computer Comm. Rev.*, 37(5), Oct. 2007.

[21] W. Thies et al. Searching the world wide web in low-connectivity communities. *11th Intl. WWW Conf.*, May 2002.

[22] United Nations. Millennium Development Goals. 2008. `http://www.un.org/millenniumgoals/`.

[23] D. Wessels and K. Claffy. ICP and the Squid web cache. *IEEE Journal on Selected Areas in Communications*, 16(3), April 1998.

[24] A. Wolman et al. On the scale and performance of cooperative web proxy caching. *17th ACM Symp. on OS Principles*, 1999.