WiRE: A New Rural Connectivity Paradigm

Aditya Dhananjay, Matt Tierney, Jinyang Li, and Lakshminarayanan Subramanian
New York University
{aditya, tierney, jinyang, lakshmi}@cs.nyu.edu

ABSTRACT

Many rural areas in developing regions remain largely disconnected from the rest of the world due to low purchasing power and the exorbitant cost of existing connectivity solutions. Wireless Rural Extensions (WiRE) is a low-power rural wireless network architecture that provides inexpensive, self-sustainable, and high-bandwidth connectivity. WiRE relies on a high-bandwidth directional wireless backbone with local distribution networks to provide focused IP coverage. WiRE also provides cellular connectivity using OpenBTS-based GSM microcells. It supports a naming and addressing framework that inter-operates with traditional telecom networks and enables a wide range of mobile services on a common IP framework. The entire WiRE network can be built by integrating a range of off-the-shelf components and existing open source tools.

Categories and Subject Descriptors

C.2.1 [**Network Architecture and Design**]: Wireless communications

General Terms

Design, Economics, Reliability

1. INTRODUCTION

The existing cellular connectivity model is not economically viable in rural settings due to two fundamental challenges: a) rural regions lack stable grid-based power sources, and b) low density and purchasing power of the users leads to insufficient revenue to cover the high capital and operationsl expenses.

Wireless Rural Extensions (WiRE) is a low-cost and low-power rural network architecture that extends data and voice connectivity from the closest city/town to nearby rural regions. Unlike cellular networks which provide universal coverage at high power consumption, WiRE provides focused coverage at low power consumption using a highly directional wireless backhaul network to efficiently reach out to sparsely spread out rural regions. WiRE uses OpenBTS-based GSM microcells (1–2km radius) and small-scale wireless mesh networks to provide cellular and Internet services within each rural region. The low-power consumption allows the entire WiRE network to be completely solar-powered with no dependence on the grid or oil. WiRE leverages and integrates a large-body of prior work into a unified rural network connectivity solution for extending cellular and Internet services. Achieving this objective requires us to address several non-trivial challenges, the following of which are illustrated in this demo: a) builing a back-haul network using low-cost and low-power hardware, b) using the naming and addressing layer to manage end-user identities in the face of network partition and churn, c) enabling low-power cellular base stations using OpenBTS, and d) end-to-end system integration.

SIGCOMM'11, August 15–19, 2011, Toronto, Ontario, Canada.
ACM 978-1-4503-0797-0/11/08.

2. WiRE NETWORK

There are two main components of a WiRE network: a) local distribution mesh networks that cover small regions (ROMA [1]), and b) long distance point-to-point links (WiLD-Net [3]) that form the backbone that connects the regions together. Each region has an *egress point* through which it connects to the back-haul. A region may also leverage OpenBTS, a software-based GSM base station that operates over USRP radio boards. It allows GSM phones to make calls and terminates calls on the same box, and forwards the voice data to the open-source Asterisk PBX system via Inter-Asterisk eXchange (IAX). The advantages are: a) end-users can use their existing cellular phones, b) it can be made to inter-operate with existing telephony networks (§3.2), and c) the power required to run the entire setup is less than 100W, which is more than an order of magnitude lower than traditional GSM base stations, though the coverage area is restricted to $1-2$ kms in outdoor settings. In order to enable telephony, there are PBX servers at the egress point of each region and at the main gateway to the city, where the WiRE network connects with the outside world.

3. NAMING AND ADDRESSING

End-users have unique identities; we need to discover and locate identities, for telephony and other applications. Let us say that the network is trying to route a call to an identity *John*, and therefore first needs to *find* John.

When a client m associates with an infrastructure node p (the publisher), the information pertaining this client needs to be disseminated. Inspired by SEATTLE[2], address resolution in WiRE works through a DHT mechanism, with consistent hashing. The underlying routing layer provides the address resolution layer with the current list of infrastructure nodes K that are alive and reachable in the network. For every node $k \in K$, the publisher calculates $H(k)$, where

H is a standard hash function. It then calculates $H(m)$, and the resolver $\hat{k}$ is chosen as that node k that minimizes $H(k) - H(m)$. The publisher p then sends a message (over TCP) to the resolver $\hat{k}$, telling it that m is now reachable through publisher p. The querier uses the identical hashing mechanism to find m. If the publisher p and the querier x are in the same connected component, we are guaranteed that they will hit the same resolver node.

3.1 Partitions and Recovery

Say an identity m published by p has been stored at resolver r. At some point of time, the network becomes partitioned into P_1 and P_2, such that $p \in P_1$ and $r \in P_2$. If any node in P_1 now tries to find m, the query will fail because the querier will choose a different resolver s. This resolver s knows nothing about m because p has not registered m with it. To overcome this issue, each publisher keeps a local list of identities that it is the publisher of, along with the resolver at which this entry is stored. The publisher p periodically monitors the network topology for changes. Suppose it is the publisher for identity m and the entry was stored at resolver r. If r is unreachable from p, it immediately re-hashes m by choosing a new resolver s (from its present partition) and stores the entry there. The old resolver r similarly ensures that it does not store values published by nodes that are not in its present partition. Finally, when two or more partitions become reconnected, re-hashing takes place to maintain consistency.

3.2 Voice Calls and Interoperability

We now describe how WiRE enables telephony and integrates with existing telephony providers.

WiRE to WiRE: Assume that an identity m is trying to call another identity n. Suppose the PBX servers (publishers) of the region containing m and n are $PBX(m)$ and $PBX(n)$ respectively. First, m contacts $PBX(m)$ stating its intention to call n. $PBX(m)$ then uses the address resolution layer and queries for n, whereupon it learns that it needs to tunnel the call to $PBX(n)$. Finally at $PBX(n)$, the call is again tunnelled through to n.

Outside World to WiRE: An existing cellular network user who wants to also use the WiRE network sets up call forwarding to a WiRE specific phone number $N_{gateway}$, which is associated with the PBX gateway. When this user receives a call on the cellular network but is not reachable, the cellular network forwards to $N_{gateway}$. Once the call reaches the WiRE PBX gateway, the destination phone number is resolved to the particular WiRE identifier (say, m). This registration (phone number to IMSI number pairing) is done one-time, when the user signs up for WiRE service. The network then needs to find the location of m, which is done using the techniques discussed above.

WiRE to Outside World: When a WiRE user wants to make a call to the outside world, the call is simply forwarded to the PBX gateway at the main gateway of the WiRE network. This gateway then routes the call through to the existing telephone service provider.

4. EVALUATION

We expect a WiRE deployment to experience churn; end-users will enter and exit the network. On our 12-node testbed, we measure how long it takes for a user identity to be discoverable by all other users in the network, after it has joined the network. Figure 1 is a box-and-whisker plot of the convergence times, as a function of the number of identities at each of the 12 publishers.

		Num. of Identities		
		100	300	500
Num. of Failures	1	2	3	4
	3	4	4	21
	5	7	22	24

Table 1: Recovery (reconvergence) time measured in seconds, as a function of the number of node failures, and the number of identities at each publisher.

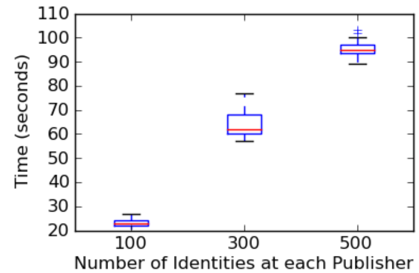

Figure 1: Convergence time varies linearly with the number of identities published at each node.

Due to many failure scenarios, we would like the network to be partition tolerant. We artificially kill nodes to create partitions, and then measure how long it takes the naming and addressing layer to re-hash and recover. From Table 1, we see that in all cases, the convergence time is within acceptable limits.

In order to study the quality of voice calls, we plot the PESQ scores over different backbone path lengths in Figure 2. A reference point is given by hop-count 0, which shows the PESQ score through a wired ethernet connection is 2.95. Even over long multi-hop wireless paths, the PESQ scores remain comparable with that of a wired connection.

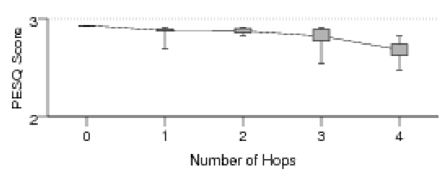

Figure 2: PESQ scores indicate that the speech quality is good, even with longer path lengths.

5. REFERENCES

[1] A. Dhananjay, H. Zhang, J. Li, and L. Subramanian. Practical, distributed channel assignment and routing in dual-radio mesh networks. In *SIGCOMM '09*.

[2] C. Kim, M. Caesar, and J. Rexford. Floodless in SEATTLE: A Scalable Ethernet Architecture for Large Enterprises. In *SIGCOMM '08*.

[3] R. Patra, S. Nedevschi, S. Surana, A. Sheth, L. Subramanian, and E. Brewer. WiLDNet: Design and Implementation of High Performance WiFi Based Long Distance Networks. *Proceedings of NSDI 2007*, 2007.