

WiLSoN: The Wirelessly Linked Seismological Network and Its Application in the Middle American Subduction Experiment

Allen Husker, Igor Stubailo, Martin Lukac, Vinayak Naik, Richard Guy, Paul Davis, and Deborah Estrin
University of California, Los Angeles

INTRODUCTION

The state of the art in temporary seismology installations for years has been stand-alone sites. In such an installation, sites are typically visited once a month to collect data and check station health. The benefit is that stand-alone stations are very quick to permit and install, and sites are not limited by local infrastructure (*i.e.*, telephone or Internet) to collect data. Moreover, the technology for such installations is well-tested and mature. The downside is that the state of health of the system is unknown between data collection intervals, and when an interesting earthquake occurs researchers must wait for the data. In places where the local networks are sparse, the time to acquire data from temporary networks can be of great importance to the local population.

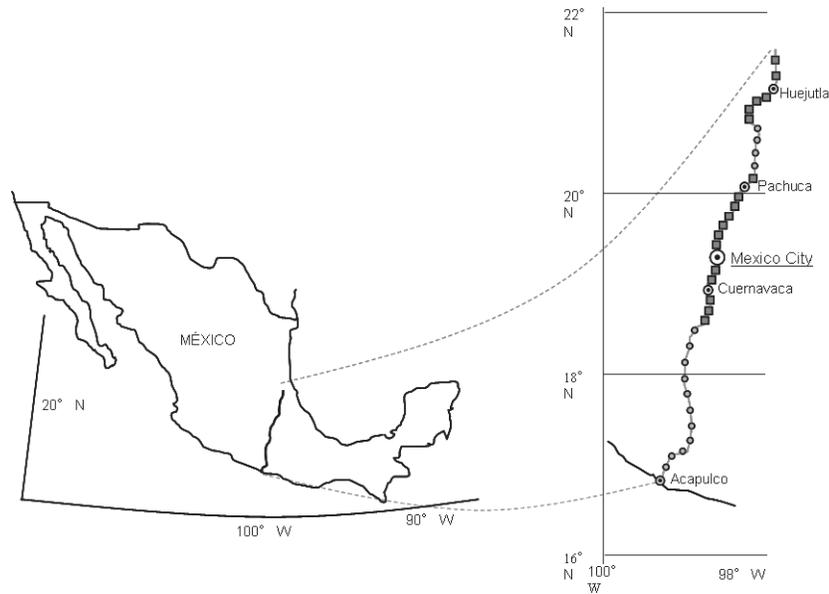
The technology to transport and share large amounts of data rapidly has increased dramatically over the past 10 years with the development of the Internet. Tools and protocols have been developed to link millions of users to millions of Web sites and data repositories. The Center for Embedded Networked Sensing (CENS) developed the Wirelessly Linked Seismological Network (WiLSoN) to extend the Internet into a seismological network. The seismology community already has deployed radios in both temporary (*e.g.*, Werner-Allen *et al.* 2006) and permanent seismic networks (*e.g.*, the High Performance Wireless Research and Education Network, the Southern California Seismic Network, USArray, the U.S. National Seismic Network, the Mexican Servicio Sismológico Nacional, and the Global Seismic Network) in the past. The goal of extending the Internet into a seismological network is to be able to dramatically increase scalability, to improve monitoring of the state of the network, and to improve ease of deployment. A freshly deployed seismic station within WiLSoN is able to join the network much like a laptop in a coffee shop. Then, much like the Internet, the data and station health information makes its way to a data repository or a user on the network without the user having to input or even know the route.

The backbone of WiLSoN is the CENS Data Communication Controller (CDCC). The CDCC consists of a low-power 1-watt Intel XScale microcomputer running Linux (<http://platformx.sourceforge.net/Links/resource.html#XScale>), with a long-range SMC2532W-B 200mW 802.11b radio from SMC (<http://www.smc.com>) and a flash disk enclosed in a water-tight fiberglass box. The software has been tested on other computers and can be used with other 802.11 radios. The microcomputer is chosen because its small form-factor creates a lightweight system that allows for easy deployment. The 802.11b radio allows us to take advantage of standard Internet protocols. The flash disk provides a method to turn any station into a standard stand-alone site should wireless communication not be possible or lost for some reason. WiLSoN was designed to be able to transfer 24-bit, 100-sample/s data across at least a 25-node hop-by-hop network to a base station. We also sought to minimize overhead for site construction, installation time, and power requirements to a typical stand-alone site with the addition of the CDCC, antennas, and cabling.

Various groups have worked to develop similar smart wireless networks for temporary deployments. The only one known to the authors that was also applied directly to seismic applications was done by Werner-Allen *et al.* (2006) at Harvard. They created a network to monitor volcano activity. Their goals were much shorter in time frame and smaller in distance. They deployed 16 nodes over a total of 3 km during a 19-day period. Hence, they used a much weaker, lower bandwidth radio to conserve power. They also sampled at 100 samples/s, but they used an in-network event detect to send only interesting data. Their network could not handle the continuous data transfer of WiLSoN.

WiLSoN collected data in Mexico for more than a year and a half as part of the Middle American Subduction Experiment (<http://www.tectonics.caltech.edu/mase/>). The seismic array consisted of a line of 100 seismometers that ran more or less perpendicular from the Middle American trench through Mexico (figure 1). Fifty of the sites were the typical stand-alone stations that have been used most frequently in temporary installations. This included the seismometer, data-logger (Reftek in this case, <http://www.reftek.com/>), battery, and solar panel. The other 50 sites incorporated WiLSoN, which used the CDCC in place of a stand-alone data-logger. The CDCC does not have an analog to digital converter (ADC), so the Quanterra Q330 (<http://www.q330.com/>) was used. At the time, it was more suitable for our needs than the Reftek digitizer.

The seismometer node spacing was 5 km. However, the permit process, topography, vegetation, and man-made structures did not allow for a simple linear node-to-node transfer of data. Data



▲ **Figure 1.** Seismic installation. Dots represent sections of the line that had standard standalone stations. Squares represent where WiLSoN was located.

instead was routed by zigzagging across the seismic array, sometimes through extra repeater stations, until it finally reached a base station. Each base station was a RAID-1 configured desktop computer providing fault-tolerant data storage as well as processing power. These were connected to the Internet, which allowed us to receive the data from the field stations, transfer it to a big data storage array at the University of California, Los Angeles (UCLA), and remotely monitor stations in real time. Due to time, money, and sometimes physical constraints (*e.g.*, a mountain between two stations) not all of the 50 CENS seismic sites were routed to one of the base stations. Ten of the stations were either stand-alone, directly connected to the Internet themselves, or a hybrid of networked and stand-alone sites where data from linked sites was stored at one site until it could be collected.

Links of greater than 40 km and cases of up to five nodes all linked to a central node were sometimes necessary to complete the network. The result was greatly reduced radio bandwidth in many cases and a “challenged network.” CENS developed the Disruption Tolerant Shell (DTS) software to manage this type of network (Lukac *et al.* 2006). Lukac *et al.* (2006) describe the DTS computer science theory in detail, and Lukac *et al.* (2007) give a computer science study of the wireless connections from the experiment.

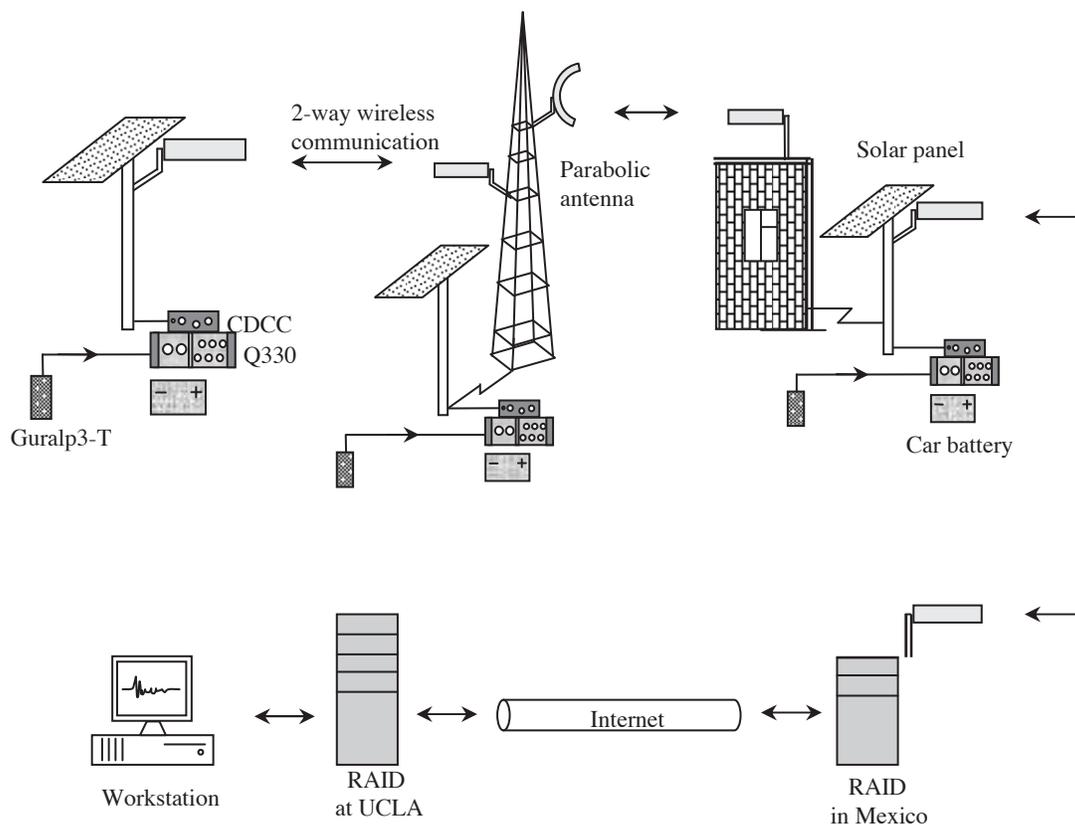
SITE SELECTION

WiLSoN significantly increases the time and manpower necessary for site selection. Stand-alone site selection requires at minimum one trained person and a vehicle. The minimum for WiLSoN is two people and two vehicles, because radio connections between sites must be tested. The 802.11b radios use 2.4 GHz providing high throughput, but they also make line-of-sight absolutely necessary for long distances. Connections are not possible from behind most trees because the extremely

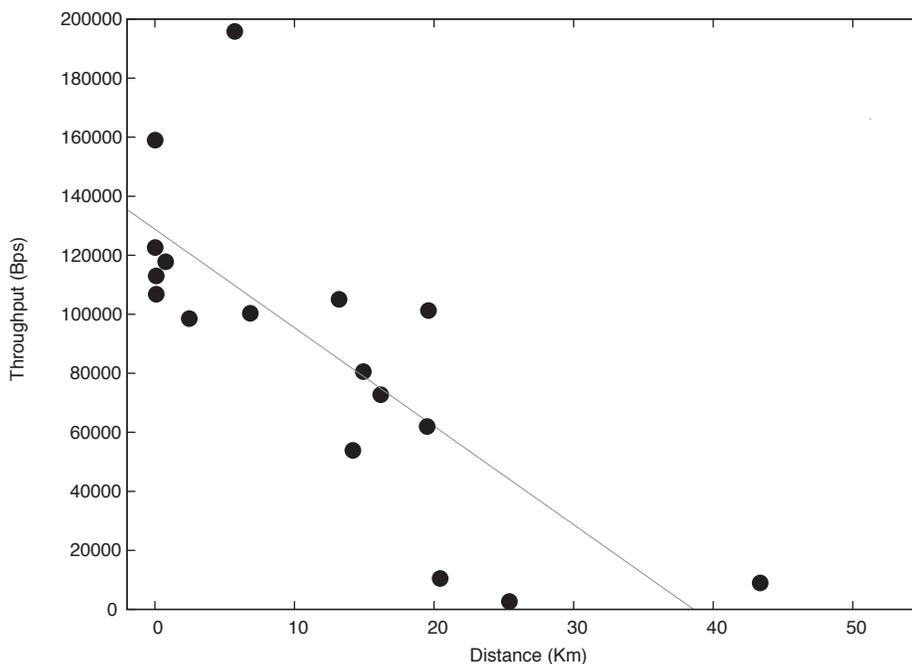
small wavelength can be attenuated by just a few leaves. Thus, in rural environments trees provide the largest obstacle. In addition, no land is truly flat. Over the 5-km node spacing, elevation can easily change by a few meters. Installing a wind-resistant, many-meter-tall mast is expensive and time consuming, so instead it is necessary to search for hills for each site. In urban environments, buildings and manmade structures provide the largest obstacle to obtaining line of sight. Figure 2 shows various site setups that we used in WiLSoN.

We tested radio connections between potential sites using the 802.11b radio in a handheld computer and antennas we would eventually employ in the experiment. We tested communications almost entirely with the standard networking tool, “ping.” Ping reports the amount of time to send and receive a 64-byte packet in real time. In addition, it reports the percentage of lost packets. The percentage of lost packets was much more important than round-trip time travel of the packet. A ping time of 4 ms often became 500 ms when another site was transferring data through the site being pinged. Eventually the ping time would return to 4 ms, but a loss of just 20% was enough to stop all data transfer. Losses in locations with obvious line of sight told us that antennas were misaligned.

We started by searching only for the linear 5-km hop-by-hop connections. However, we learned that 10- to 20-km connections are possible with the aid of a hill. Throughput falls off with distance between nodes, but it was sufficient for our experiment (figure 3). We found that it was much faster to permit if the wireless routes did not always follow the geographic topology. In an attempt to speed site selections and permissions, we also did not test splitters or bandwidth requirements. The result was a challenged network, as mentioned. Despite all the shortcuts taken to speed up installation, we found that it took about three months to permit each of the 50 stand-alone sites and about six months to permit the 50 WiLSoN sites.



▲ **Figure 2.** Examples of the data propagation within WiSoN. Starting at the top left and rotating clockwise: The most standard setup was the antenna connected directly to the solar panel mast. Eight pre-existing radio towers were used within the experiment. Parabolic antennas provided extra distance and were used at times with all types of setups. The roof installations required a building somewhat close to the ground installation. There were 20 building installations in our experiment. Finally we had four RAIDs at various universities within Mexico to collect the data and provide Internet access to send it to UCLA.



▲ **Figure 3.** Throughput as a function of distance. Throughput is measured as the bits per second transfer rate during file transfers. Data taken from Lukac *et al.* (2007).

SOFTWARE DEVELOPED BY CENS

Duiker

We found existing commercial software did not meet our purposes. In particular, we wanted flexibility for an ad hoc node configuration and a minimum use of computer resources at the node to be able to add future features. To achieve these goals, Irvine (2006) developed tools to acquire Q330 data at a node in the field and transfer it hop-by-hop all the way to a RAID array at UCLA. The data acquisition component is called Duiker, and it resides on the CDCC collecting raw data packets from the digitizer. One hour of data is about 1–2 MB depending on the amount of noise or signal. Low noise and little to no seismic waveform make for smaller data packets because they can be more compressed. The data is buffered to the flash disk for an hour and then bundled to be transferred. The data delivery part transfers the data with existing Internet protocol to the next hop. If the next hop is not reachable, it will store the data until it is available. The data can be retrieved physically from the CDCC by changing the flash disk if the wireless connection is permanently lost.

The tools use very little system resources (640 kB memory and < 1% CPU on a 400-MHz XScale processor) and can be configured to properly name packets based on standard conventions. Duiker was developed for systems where a computer connects directly to a Q330 digitizer. Duiker runs very quickly with little configuration. Only the internal serial number and IP of the Q330 are needed for the data collection. The program can be downloaded along with detailed user instructions at <http://research.cens.ucla.edu/areas/2007/seismic/duiker/>.

After the data is delivered to the final data repository, the conversion tool converts it to the mseed format while still backing up the original raw files. Keeping the raw files allows us to reprocess them in case a bug is discovered in the conversion process from the Q330 data packet format to mseed. Waveforms acquired with Duiker correlated with those acquired by the stand-alone sites, so the integrity of the data has already been shown.

The final component of the suite of tools created by Irvine (2006) is the data delivery tool. The tool used a predefined static route through the network. The static routing was designed for a pure hop-by-hop network with no breaks in connectivity. However, during the deployment there was no pure hop-by-hop network, and there were many breaks in the connectivity. Thus, DTS was developed during the experiment to account for the static routing limitations and deployed on a portion of WiLSoN toward the final stages of the experiment.

Disruption Tolerant Shell (DTS)

The sites were often placed at the limits of the 802.11b radios due to constraints such as obtaining permissions for sites, line of sight, placing a station every 5 km, safety of the site, and, of course, preferred seismological conditions. Link tests were only done between two nodes before installation to make sure that some link was possible. However, after installation, up to four additional nodes were assigned to connect to a single site. The static network under such conditions became a challenged net-

work (Lukac *et al.* 2006) and reacted dynamically. Throughout the course of a day, strength of connections between sites varied due to weather conditions, changing ambient radio noise, and file transfers within the network itself. Intelligent networking was required to maintain the network and lower the cost of labor.

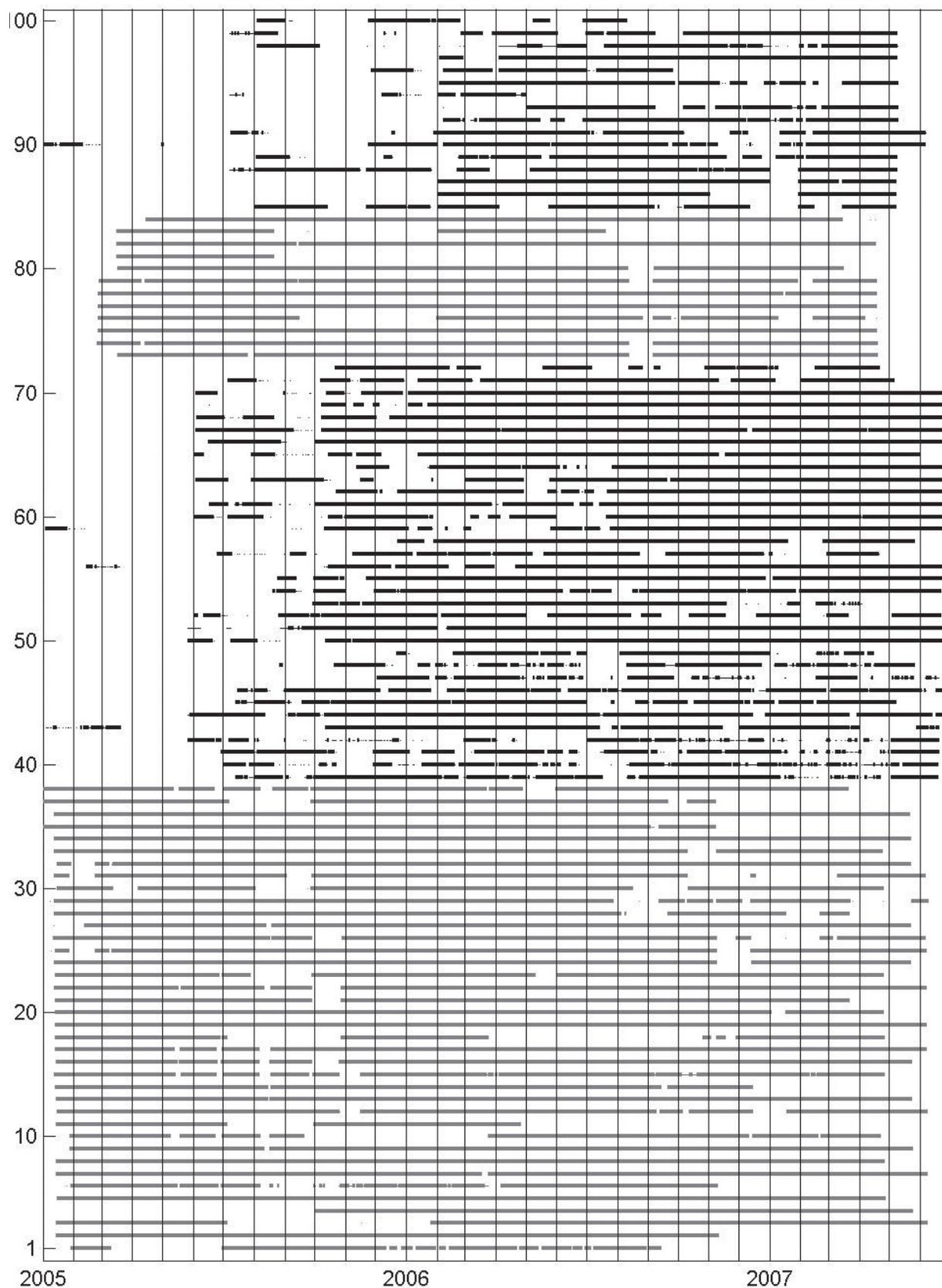
Delay-Tolerant Networking techniques (Fall 2006) were applied to the Cuernavaca portion of the network (figure 1) in the winter of 2006 with the installation of the DTS (Lukac *et al.* 2006). The 1–2.5-MB file created each hour by Duiker was sent through the system until it reached a sink node with a connection to the Internet where it could be transferred to UCLA. Rather than requiring a predefined connection through all nodes to the sink node, DTS transferred the data hop-by-hop until the data reached the sink. The files were transferred hop-by-hop over the dynamically determined path using Internet protocols. The route of file transfer was determined and updated using existing link metrics (De Couto *et al.* 2003). In our deployment DTS was set to send data aggressively through the system using the best available path. If file transfer stalled for more than 45 seconds, the transfer was stopped and reinitiated. The next time it started, it could potentially use a different route.

Lukac *et al.* (2006) developed DTS to manage the system in addition to transfer data. DTS gives users access to routing and data transfer information of the entire network at any node. In addition, users can issue any Linux shell commands on any node in the system and see the response from all nodes. The command and response are sent hop-by-hop until every node receives it. It also allows software updates in the same way. The advantage to having information from the entire network on each node is that when field engineers are visiting specific sites, they can have access to the entire network quickly.

DTS adds a 2-KB header to every data file transferred through the seismic stations to log information about each node along the path traveled by the data. This makes it possible to preemptively improve the system. Areas with low bandwidth or bottlenecks may have to be physically visited to install an amplifier, change antennas, or cut growing foliage. Eventually, this information will be the basis for a network-wide watchdog to warn users of unresponsive nodes. DTS and future updates can be found at <http://research.cens.ucla.edu/areas/2007/seismic/duiker/> along with detailed user instructions.

COMPARISON OF WiLSoN WITH THE STAND-ALONE SEISMIC SITES AND FUTURE WORK

The proof of concept of WiLSoN worked quite well. The management of WiLSoN was done almost entirely from the United States while being deployed in Mexico. We were able to determine most site problems from logging into the network and probing the sites. In addition, we collected data in near real time. We collected 2.5 GB of data daily. However, we had more data gaps than the standard method of stand-alone data collection (figure 4). It should be noted that the southern sites, where the stand-alone stations were, were given the highest priority due to the most interesting geophysical science. In addition, all of the stand-alone sites were installed before beginning the



▲ **Figure 4.** The times when data was recorded in the experiment during its installation. The station number is listed on the left. The light bars represent data from stand-alone sites. The darker bars are from WiSoN. The stations are in order according to latitude with those at the top the most northerly.

WiLSoN deployment. The southern section of WiLSoN had most stations installed by the middle of October 2005. The northern section had most installed by February 2006. We take from February 2006 to February 2007 as our comparison to the stand-alone sites and only include those 40 WiLSoN sites that were wirelessly linked to Internet sink nodes. We find that WiLSoN had data 78% of the time while the stand-alone sites gave data 86% of the time. The difference between the two largely was due to development of the WiLSoN system during installation and operation. Problems arising from vandalism or environmental damage specifically to WiLSoN components were very limited.

Development has already started on in-network timing for situations where GPS time is not available at all nodes. This situation arises in urban environments where large buildings can block satellite view at ground level where the seismometer is stationed. It can also be used for monitoring inside structures such as buildings. In-network time distributes GPS from those nodes that have a good satellite lock to those nodes that do not (Elson *et al.* 2002). In addition, we recommend including some system self-management such as automatic recentering of masses in seismometers with RMS readings far from 0, warnings for those sites that have not responded to centering, notification of broken links or stations unable to deliver data, and low-power notifications. ✉

ACKNOWLEDGMENTS

We would like to thank Xyoli Pérez-Campos, Mario Islas Herrera, Oscar Martínez Susano, Jorge Soto, Aída Quezada Reyes, Arturo Iglesias, Lizbeth Espejo, Luís Antonio Placencia Gómez, Luís Edgar Rodríguez, Fernando Greene, Alma Quezada, Steve Skinner, Irving Flores, Manolo Vega, Jesús Ramírez, and Daniel Molina for countless hours spent installing and maintaining the system. This research was supported by the Betty and Gordon Moore Foundation through the Tectonics Observatory at Caltech, a University of California Institute for

Mexico and the United States-Consejo Nacional de Ciencia y Tecnología (UC MEXUS-CONACYT) Collaborative Grant, and CENS at UCLA, a National Science Foundation center.

REFERENCES

- De Couto, D. S. J., D. Aguayo, J. Bicket, and R. Morris (2003). High-throughput path metric for multi-hop wireless routing. In *Proceedings of the 9th Association for Computing Machinery (ACM) International Conference on Mobile Computing and Networking (MobiCom '03)*. San Diego, CA: MobiCom, , 134–146..
- Elson, J., L. Girod, and D. Estrin (2002). Fine-grained network time synchronization using reference broadcasts. In *Proceedings of the Fifth Symposium on Operating Systems Design and Implementation (OSDI)*. Boston: OSDI, 141–163. .
- Fall, K. (2003). A delay-tolerant network architecture for challenged inter-nets. In *Special Interest Group on Data Communication (SIGCOMM '03): Proceedings of the 2003 conference on Applications, technologies, architectures, and protocols for computer communications*. Karlsruhe, Germany: Association for Computer Machinery (ACM) Press, 27–34. .
- Irvine, S. (2006). A data acquisition software suite for remotely deployed Q330 digitizers. <http://research.cens.ucla.edu/areas/2007/seismic/duiker/>.
- Lukac, M., L. Girod, and D. Estrin (2006). Disruption Tolerant Shell. In *Proceedings of the 2006 Special Interest Group on Data Communication (SIGCOMM) workshop on Challenged network*. Pisa, Italy: SIGCOMM, 189–196..
- Lukac, M., V. Naik, I. Stubailo, A. Husker, and D. Estrin (2007). In vivo characterization of a wide area 802.11b wireless seismic array, CENS Technical Report 74.
- Werner-Allen, G., K. Lorincz, J. Johnson, J. Lees, and M. Welsh (2006). Fidelity and yield in a volcano monitoring sensor network. In *Proceedings of the 7th USENIX Symposium on Operating Systems Design and Implementation (OSDI)*. Seattle: OSDI, 381–396..

*University of California at Los Angeles
595 Charles Young Drive East
3806 Geology Building
Los Angeles, California 90095 USA
uskerhay@moho.ess.ucla.edu
(A. H.)*