Deploying a Wireless Sensor Network on an Active Volcano

Augmenting heavy and power-hungry data collection equipment with lighter, smaller wireless sensor network nodes leads to faster, larger deployments. Arrays comprising dozens of wireless sensor nodes are now possible, allowing scientific studies that aren't feasible with traditional instrumentation. Designing sensor networks to support volcanic studies requires addressing the high data rates and high data fidelity these studies demand. The authors' sensor-network application for volcanic data collection relies on triggered event detection and reliable data retrieval to meet bandwidth and data-quality demands.

**Geoffrey Werner-Allen,
Konrad Lorincz,
and Matt Welsh**
Harvard University

**Omar Marcillo
and Jeff Johnson**
University of New Hampshire

**Mario Ruiz
and Jonathan Lees**
University of North Carolina

Wireless sensor networks — in which numerous resource-limited nodes are linked via low-bandwidth wireless radios — have been the focus of intense research during the past few years. Since their conception, they've excited a range of scientific communities because of their potential to facilitate data acquisition and scientific studies. Collaborations between computer scientists and other domain scientists have produced networks that can record data at a scale and resolution not previously possible. Taking this progress one step further, wireless sensor networks can potentially advance the pursuit of geophysical studies of volcanic activity.

Two years ago, our team of computer scientists at Harvard University began collaborating with volcanologists at the University of North Carolina, the University of New Hampshire, and the Instituto Geofísico in Ecuador. Studying active volcanoes typically involves sensor arrays built to collect seismic and infrasonic (low-frequency acoustic) signals. Our group is among the first to study the use of tiny, low-power wireless sensor nodes for geophysical studies. In 2004, we deployed a small wireless sensor network on Volcán Tungurahua in central Ecuador as a proof of concept.[1] For three days, three nodes equipped with microphones collected continuous data from the erupting volcano.

In August 2005, we deployed a larger, more capable network on Volcán Reventador in northern Ecuador. The array consisted of 16 nodes equipped with seismoacoustic sensors deployed over 3 km.

The system routed the collected data through a multihop network and over a long-distance radio link to an observatory, where a laptop logged the collected data. Over three weeks, our network captured 230 volcanic events, producing useful data and letting us evaluate the performance of large-scale sensor networks for collecting high-resolution volcanic data.

In contrast with existing volcanic data-acquisition equipment, our nodes are smaller, lighter, and consume less power. The resulting spatial distribution greatly facilitates scientific studies of wave propagation phenomena and volcanic source mechanisms. Additionally — and extremely important for successful collaboration — this volcanic data-collection application presents numerous challenging computer science problems. Studying active volcanoes necessitates high data rates, high data fidelity, and sparse arrays with high spatial separation between nodes. The intersection of these scientific requirements with wireless sensor network nodes' current capabilities creates difficult computer science problems that require research and novel engineering.

Sensor Networks for Volcanic Monitoring

Wireless sensor networks can greatly assist the geophysics community. The increased scale promised by lighter, faster-to-deploy equipment will help address scientific questions beyond current equipment's' practical reach. Today's typical volcanic data-collection station consists of a group of bulky, heavy, power-hungry components that are difficult to move and require car batteries for power. Remote deployments often require vehicle or helicopter assistance for equipment installation and maintenance. Local storage is also a limiting factor — stations typically log data to a Compact Flash card or hard drive, which researchers must periodically retrieve, requiring them to regularly return to each station. Although these limitations make it difficult to deploy large networks of existing equipment, such large-scale experiments could help us achieve important insights into volcanoes' inner workings. Volcanic tomography,[2] for example, is one approach to the study of volcanoes' interior structure; collecting and analyzing signals from multiple stations can produce precise mappings of the volcanic edifice. In general, such mappings' precision and accuracy increases as stations are added to the data-collection network. Studies such as these could help

resolve debates over the physical processes at work within a volcano's interior.

The geophysics community has well-established tools and techniques it uses to process signals extracted by volcanic data-collection networks. These analytical methods require that our wireless sensor networks provide data of extremely high fidelity — a single missed or corrupted sample can invalidate an entire record. Small differences in sampling rates between two nodes can also frustrate analysis, so samples must be accurately time stamped to allow comparisons between nodes and between networks.

An important feature of volcanic signals is that much of the data analysis focuses on discrete events, such as eruptions, earthquakes, or tremor activity. Although volcanoes differ significantly in the nature of their activity, during our deployment,

Studying active volcanoes necessitates high data rates, high data fidelity, and sparse arrays with high spatial separation between nodes.

many interesting signals at Reventador spanned less than 60 seconds and occurred several dozen times per day. This let us design the network to capture time-limited events, rather than continuous signals.

Of course, recording individual events doesn't adequately answer all the scientific questions that volcanologists pose. Indeed, understanding long-term trends requires complete waveforms spanning long time intervals. However, wireless sensor nodes' low radio bandwidth makes them inappropriate for such studies; thus, we focused on triggered event collection when designing our network.

Volcanic studies also require large internode separations to obtain widely separated views of seismic and infrasonic signals as they propagate. Array configurations often comprise one or more possibly intersecting lines of sensors, and the resulting topologies raise new challenges for sensor-network design, given that much previous work has focused on dense networks in which each node has several neighbors. Linear configurations can also affect achievable network bandwidth, which degrades when data must be
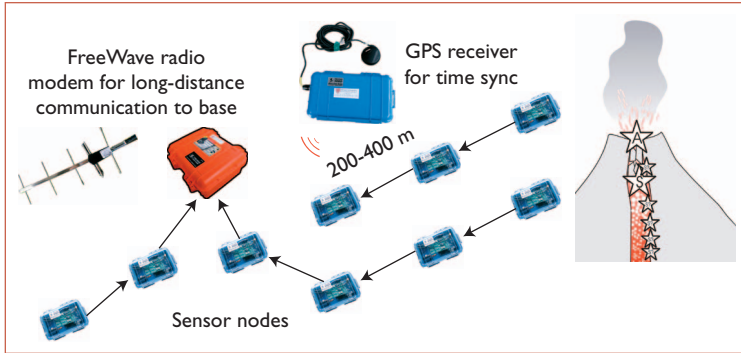
Figure 1. The volcano monitoring sensor-network architecture. The network consists of 16 sensor nodes, each with a microphone and siesmometer, collecting seismic and acoustic data on volcanic activity. Nodes relay data via a multihop network to a gateway node connected to a long-distance FreeWave modem, providing radio connectivity with a laptop at the observatory. A GPS receiver is used along with a multihop time-synchronization protocol to establish a network-wide timebase.

transmitted over multiple hops. Node failure poses a serious problem in sparse networks because a single failure can obscure a large portion of the network.

Sensor-Network Application Design

Given wireless sensor network nodes' current capabilities, we set out to design a data-collection network that would meet the scientific requirements we outlined in the previous section. Before describing our design in detail, let's take a high-level view of our sensor node hardware and overview the network's operation. Figure 1 shows our sensor network architecture.

Network Hardware

Our sensor network on Reventador comprised 16 stations equipped with seismic and acoustic sensors. Each station consisted of a Moteiv TMote Sky wireless sensor network node (www.moteiv. com) , an 8-dBi 2.4-GHz external omnidirectional antenna, a seismometer, a microphone, and a custom hardware interface board. We fitted each of 14 nodes with a Geospace Industrial GS-11 geophone — a single-axis seismometer with a corner frequency of 4.5 Hz — oriented vertically. We equipped the two remaining nodes with triaxial Geospace Industries GS-1 seismometers with corner frequencies of 1 Hz, yielding separate signals in each of the three axes.

The TMote Sky is a descendant of the Univeristy of California, Berkeley's Mica "mote" sen-

sor node. It features a Texas Instruments MSP430 microcontroller, 48 Kbytes of program memory, 10 Kbytes of static RAM, 1 Mbyte of external flash memory, and a 2.4-GHz Chipcon CC2420 IEEE 802.15.4 radio. The TMote Sky was designed to run TinyOS,[3] and all of our software development used this environment. We chose the TMote Sky because the MSP430 microprocessor provides several configurable ports that easily support external devices, and the large amount of flash memory was useful for buffering collected data, as we describe later.

We built a custom hardware board to integrate the TMote Sky with the seismoacoustic sensors. The board features up to four Texas Instruments AD7710 analog-to-digital converters (ADCs), providing resolution of up to 24 bits per channel. The MSP430 microcontroller provides on-board ADCs, but they're unsuitable for our application. First, they provide only 16 bits of resolution, whereas we required at least 20 bits. Second, seismoacoustic signals require an aggressive filter centered around 50 Hz. Because implementing such a filter using analog components isn't feasible, it's usually approximated digitally, which requires several factors of oversampling. To perform this filtering, the AD7710 samples at more than 30 kHz, while presenting a programmable output word rate of 100 Hz. The high sample rate and computation that digital filtering requires are best delegated to a specialized device.

A pair of alkaline D cell batteries powered each sensor node — our network's remote location made it important to choose batteries maximizing node lifetime while keeping cost and weight low. D cells provided the best combination of low cost and high capacity, and they can power a node for more than a week. Roughly 75 percent of the power each node draws is consumed by the sensor interface board, primarily due to the ADCs' high power consumption. During our three-week deployment, we changed batteries on the entire sensor array only twice.

The network is monitored and controlled by a laptop base station, located at a makeshift volcano observatory roughly 4 km from the sensor network itself. FreeWave radio modems using 9-dBi directional Yagi antennae were used to establish a long-distance radio link between the sensor network and the observatory.

Typical Network Operation

Each node samples two or four channels of seismoacoustic data at 100 Hz, storing the data in

local flash memory. Nodes also transmit periodic status messages and perform time synchronization, as described later. When a node detects an interesting event, it routes a message to the base station laptop. If enough nodes report an event within a short time interval, the laptop initiates data collection, which proceeds in a round-robin fashion. The laptop downloads between 30 and 60 seconds of data from each node using a reliable data-collection protocol, ensuring that the system retrieves all buffered data from the event. When data collection completes, nodes return to sampling and storing sensor data.

Overcoming High Data Rates: Event Detection and Buffering

When designing high-data-rate sensing applications, we must remember an important limitation of current sensor-network nodes: low radio bandwidth. IEEE 802.15.4 radios, such as the Chipcon CC2420, have raw data rates of roughly 30 Kbytes per second. However, overheads caused by packet framing, medium access control (MAC), and multihop routing reduce the achievable data rate to less than 10 Kbytes per second, even in a single-hop network.

Consequently, nodes can acquire data faster than they can transmit it. Simply logging data to local storage for later retrieval is also infeasible for these applications. The TMote Sky's flash memory fills in roughly 20 minutes when recording two channels of data at 100 Hz. Fortunately, many interesting volcanic events will fit in this buffer. For a typical earthquake or explosion at Reventador, 60 seconds of data from each node is adequate.

Each sensor node stores sampled data in its local flash memory, which we treat as a circular buffer. Each block of data is time stamped using the local node time, which is later mapped to a global network time, as we explain in the next section. Each node runs an event detector on locally sampled data. Good event-detection algorithms produce high detection rates while maintaining small false-positive rates. The detection algorithm's sensitivity links these two metrics — a more sensitive detector correctly identifies more events at the expense of producing more false positives.

The data set that our previous deployment at Volcán Tungurahua[1] produced aided us in designing the event detector. We implemented a short-term average/long-term average threshold detector, which computes two exponentially weighted moving averages (EWMAs) with different gain constants. When the ratio between the short-term average and the long-term average exceeds a fixed threshold, the detector fires. The detector threshold lets nodes distinguish between low-amplitude signals, perhaps from distant earthquakes, and high-amplitude signals from nearby volcanic activity.

When the event detector on a node fires, it routes a small message to the base-station laptop. If enough nodes report events within a certain time window, the laptop initiates data collection from the entire network (including nodes that didn't report the event). This global filtering prevents spurious event detections from triggering a data-collection cycle. Fetching 60 seconds of data from all 16 nodes in the network takes roughly one hour. Because nodes can only buffer 20 minutes of eruption data locally, each node pauses sampling and reporting events until it has uploaded its data. Given that the latency associated with data collection prevents our network from capturing all events, optimizing the data-collection process is a focus of future work.

Reliable Data Transmission and Time Synchronization

Extracting high-fidelity data from a wireless sensor network is challenging for two primary reasons. First, the radio links are lossy and frequently asymmetrical. Second, the low-cost crystal oscillators on these nodes have low tolerances, causing clock rates to vary across the network. Much prior research has focused on addressing these challenges.[4,5]

We developed a reliable data-collection protocol, called Fetch, to retrieve buffered data from each node over a multihop network. Samples are buffered locally in blocks of 256 bytes, then tagged with sequence numbers and time stamps. During transmission, a sensor node fragments each requested block into several chunks, each of which is sent in a single radio message. The base-station laptop retrieves a block by flooding a request to the network using Drip, a variant of the TinyOS Trickle[6] data-dissemination protocol. The request contains the target node ID, the block sequence number, and a bitmap identifying missing chunks in the block.

The target node replies by sending the requested chunks over a multihop path to the base station. Our system constructs the routing tree using Multi-HopLQI, a variant of the TinyOS MintRoute[4] routing protocol modified to select routes based on the

CC2420 link quality indicator (LQI) metric. Link-layer acknowledgments and retransmissions at each hop improve reliability. Retrieving one minute of stored data from a two-channel sensor node requires fetching 206 blocks and can take several minutes to complete, depending on the multihop path's quality and the node's depth in the routing tree.

Scientific volcano studies require sampled data to be accurately time stamped; in our case, a global clock accuracy of ten milliseconds was sufficient. We chose to use the Flooding Time Synchronization Protocol (FTSP),[5] developed at Vanderbilt University, to establish a global clock across our network. FTSP's published accuracy is very high, and the TinyOS code was straightforward to integrate into our application. One of the nodes used a Garmin GPS receiver to map the FTSP global time to GMT. Unfortunately, FTSP occasionally exhibited unexpected behavior, in which nodes would report inaccurate global times, preventing some data from being correctly time stamped. We're currently developing techniques to correct our data set's time stamps based on the large amount of status messages logged from each node, which provide a mapping from the local clock to the FTSP global time.

Command and Control
A feature missing from most traditional volcanic data-acquisition equipment is real-time network control and monitoring. The long-distance radio link between the observatory and the sensor network lets our laptop monitor and control the network's activity. We developed a Java-based GUI for monitoring the network's behavior and manually setting parameters, such as sampling rates and event-detection thresholds. In addition, the GUI was responsible for controlling data collection following a triggered event, moving significant complexity out of the sensor network. The laptop logged all packets received from the sensor network, facilitating later analysis of the network's operation.

The GUI also displayed a table summarizing network state, based on the periodic status messages that each node transmitted. Each table entry included the node ID; local and global time stamps; various status flags; the amount of locally stored data; depth, parent, and radio link quality in the routing tree; and the node's temperature and battery voltage. This functionality greatly aided sensor deployment by letting a team member rapidly determine whether a new node had

joined the network as well as the quality of its radio connectivity.

Deploying on Volcán Reventador
Volcán Reventador is located in northern Ecuador, roughly three hours from the capital, Quito. Long dormant, Reventador reawakened suddenly in 2002, erupting with massive force. Ash thrown into the air blanketed Quito's streets 100 kilometers to the east, closing schools and the airport. Pyroclastic flows raced down the mountain, flattening forests, displacing an oil pipeline, and severing a major highway. After 18 months of quiescence, renewed activity began in November 2004. During our deployment, Reventador's activity consisted of discrete, relatively small explosive events that ejected incandescent blocks, gas, and ash several times a day. Corresponding seismic activity included explosion earthquakes, extended-duration shaking (tremors), and shallow-rock-fracturing earthquakes that might have been associated with magma migration within the volcano.

Several features of Volcán Reventador made it ideal for our experiment. Reaching 3,500 meters at its peak, Reventador sits at a low elevation compared to other Ecuadorean volcanoes, making deployment less strenuous. Its climate is moderate, with temperatures ranging between 10 and 30 degrees Celsius. The 2002 explosion produced pyroclastic flows that left large parts of the flanks denuded of vegetation. Given that our radio antennas' effectiveness can be severely degraded by obstacles to line-of-sight, the lack of vegetation greatly simplified sensor-node positioning.

Our base while working at Reventador was the Hosteria El Reventador, a small hotel located nearby on the highway from Quito to Lago Agria. The hotel provided us with space to set up our equipment and ran an electric generator that powered our laptops and other equipment at the makeshift observatory.

Sensor-Network Device Enclosures and Physical Setup
A single sensor network node, interface board, and battery holder were all housed inside a small weatherproof and watertight Pelican case, as Figure 2 shows. We installed environmental connectors through the case, letting us attach cables to external sensors and antennae without opening the case and disturbing the equipment inside. For working in wet and gritty conditions, these external connectors became a tremendous asset.

Installing a station involved covering the Pelican case with rocks to anchor it and shield the contents from direct sunlight. We elevated the antennae on 1.5-meter lengths of PVC piping to minimize ground effects, which can reduce radio range. We buried the seismometers nearby, but far enough away that they remained undisturbed by any wind-induced shaking of the antenna pole. Typically, we mounted the microphone on the antenna pole and shielded it from the wind and elements with plastic tape. Installation took several minutes per node, and the equipment was sufficiently light and small that an individual could carry six stations in a large pack. The PVC poles were light but bulky and proved the most awkward part of each station to cart around.

Network Location and Topology

We installed our stations in a roughly linear configuration that radiated away from the volcano's vent and produced an aperture of more than three kilometers. We attempted to position the stations as far apart as the radios on each node would allow. Although our antennae could maintain radio links of more than 400 meters, the geography at the deployment site occasionally required installing additional stations to maintain radio connectivity. Other times, we deployed a node expecting it to communicate with an immediate neighbor but later noticed that the node was bypassing its closest companion in favor of a node closer to the base station. Most nodes communicated with the base station over three or fewer hops, but a few were moving data over as many as six.

In addition to the sensor nodes, we used several other pieces of equipment. Three Freewave (www.freewave.com) radio modems provided a long-distance, reliable radio link between the sensor network and the observatory laptop. Each Freewave required a car battery for power, recharged by solar panels. A small number of Crossbow (www.xbow.com) MicaZ sensor network nodes served supporting roles. One interfaced between the network and the Freewave modem and another was attached to a GPS receiver to provide a global timebase.

Early Results

We deployed our sensor network at Volcán Reventador for more than three weeks, during which time we collected seismoacoustic signals from several hundred events. We've only just begun rigor-

Figure 2. A two-component station. The blue Pelican case contains the wireless sensor node and hardware interface board. The external antenna is mounted on the PVC pole to reduce ground effects. A microphone is taped to the PVC pole, and a single seismometer is buried nearby.

ously analyzing our system's performance, but we've made some early observations.

In general, we were pleased with our systems' performance. During the 19-day deployment, we retrieved data from the network 61 percent of the time. Many short outages occurred because — due to the volcano's remote location — powering the logging laptop around the clock was often impossible. By far the longest continuous network outage was due to a software component failure, which took the system offline for three days until researchers returned to the deployment site to reprogram nodes manually.

Our event-triggered model worked well. During the deployment, our network detected 230 eruptions and other volcanic events, and logged nearly 107 Mbytes of data. Figure 3 shows an
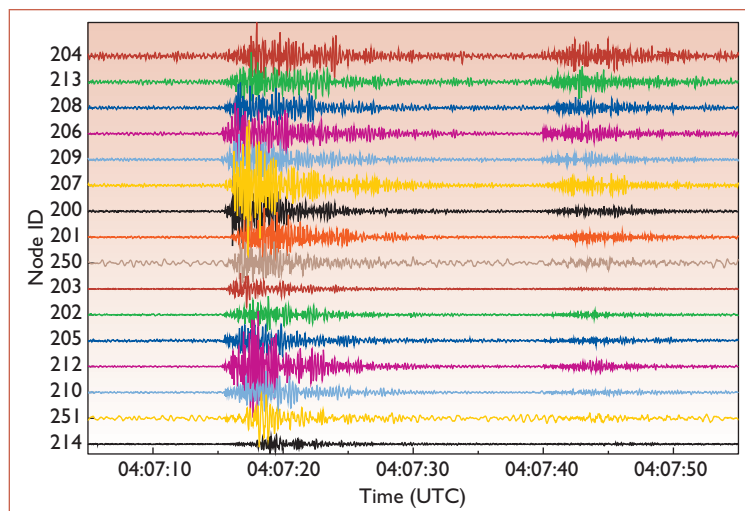
Figure 3. An event captured by our network. The event shown was a volcano tectonic (VT) event and had no interesting acoustic component. The data shown has undergone several rounds of postprocessing, including timing rectification. We show only seismic signals.

example of a typical earthquake our network recorded. By examining the data downloaded from the network, we verified that the local and global event detectors were functioning properly. As we described, we disabled sampling during data collection, implying that the system was unable to record two back-to-back events. In some instances, this meant that a small seismic event would trigger data collection, and we'd miss a large explosion shortly thereafter. We plan to revisit our approach to event detection and data collection to take this into account.

Our deployment raises many exciting directions for future work. We plan to continue improving our sensor-network design and pursuing additional deployments at active volcanoes. This work will focus on improving event detection and prioritization, as well as optimizing the data-collection path. We hope to deploy a much larger (100-node) array for several months, with continuous Internet connectivity via a satellite uplink. We're collaborating with the SensorWebs project at NASA and the Jet Propulsion Lab to allow our ground-based sensor network to trigger satellite imaging of the volcano after a large eruption. We assembled equipment required to test this idea at Reventador, but were unable to establish a reliable Internet connection at the deployment site via satellite.

A more ambitious research goal involves sophisticated distributed data-processing within the sensor network itself. Sensor nodes, for example, can collaborate to perform calculations of energy release, signal correlation, source localization, and perhaps tomographic imaging of the volcanic edifice. By pushing this computation into the network, we can greatly reduce the radio bandwidth requirements and scale up to much larger arrays. We're excited by the opportunities that sensor networks have opened up for geophysical studies.

References

1. G. Werner-Allen et al., "Monitoring Volcanic Eruptions with a Wireless Sensor Network," *Proc. 2nd European Workshop Wireless Sensor Networks* (EWSN 05), IEEE Press, 2005; www.eecs.harvard.edu/~mdw/papers/volcano-ewsn05.pdf.
2. J.M. Lees, "The Magma System of Mount St. Helens: Nonlinear High Resolution P-Wave Tomography," *J. Volcanic and Geothermal Research*, vol. 53, nos. 1–4, 1992, pp. 103–116.
3. J. Hill et al., "System Architecture Directions for Networked Sensors," *Proc. 9th Int'l Conf. Architectural Support for Programming Languages and Operating Systems*, ACM Press, 2000, pp. 93–104.
4. A. Woo, T. Tong, and D. Culler, "Taming the Underlying Challenges of Reliable Multihop Routing in Sensor Networks," *Proc. 1st ACM Conf. Embedded Networked Sensor Systems* (SenSys 03), ACM Press, 2003, pp. 14–27.
5. M. Maroti et al., "The Flooding Time Synchronization Protocol," *Proc. 2nd ACM Conf. Embedded Networked Sensor Systems*, ACM Press, 2004, pp. 39–49.
6. P. Levis et al., "Trickle: A Self-Regulating Algorithm for Code Propagation and Maintenance in Wireless Sensor Networks," *Proc. 1st Usenix/ACM Symp. Networked Systems Design and Implementation* (NSDI 04), Usenix Assoc., 2004, pp. 15–28.

Geoffrey Werner-Allen is a second-year PhD candidate in computer science in the Division of Engineering and Applied Science at Harvard University. His research interests include wireless sensor networks, biologically inspired and distributed algorithms, tool development, and software engineering. Werner-Allen has an AB in physics from Harvard University. He is a student member of the ACM. Contact him at werner@eecs.harvard.edu; www.eecs.harvard.edu/~werner.

Konrad Lorincz is a fourth-year PhD candidate in computer science at Harvard University. His research interests include distributed systems, networks, wireless sensor networks, location tracking, and software engineering. Lorincz has

an MS in computer science from Harvard University. Contact him at konrad@eecs.harvard.edu; www.eecs.harvard.edu/~konrad.

Mario Ruiz is an associate professor at the Escuela Politecnica Nacional and a third-year PhD student at the University of North Carolina, Chapel Hill. His research field is volcano seismology. Ruiz has an MS in geophysics from the New Mexico Institute of Technology. Contact him at mruiz@email.unc.edu.

Omar Marcillo is a graduate student in the Department of Earth Sciences at the University of New Hampshire. His primary research interest is the development of instrumentation for volcano monitoring and geophysical studies. Contact him at oed4@unh.edu.

Jeff Johnson is a research assistant professor of geophysics and volcanology in the Department of Earth Sciences at the University of New Hampshire. His research interests include eruption dynamics and volcano monitoring. Johnson has an MS from Stanford and a PhD in geophysics from the University of Washington. He is an active member of the American Geophysical Union and Seismological Society of America and the International Association of Volcanology and Chemistry of the Earth's Interior. Contact him at jeff.johnson@unh.edu; http://earth.unh.edu/johnson/johnson.htm.

Jonathan Lees is an associate professor in the Department of Geological Sciences at the University of North Carolina, Chapel Hill, where he specializes in geophysics, seismology and volcanology. His research is directed toward understanding the dynamics of volcanic explosions as they relate to the shallow conduit system as well as the deep plumbing structure of the volcano edifice. Lees has a PhD in geophysics from the University of Washington. He is an active member of the American Geophysical Union, the Society of Exploration Geophysicists, and the Seismological Society of America. Contact him at jonathan_lees@unc.edu; www.unc.edu/~leesj.

Matt Welsh is an assistant professor of computer science at Harvard University. His research interests include operating system, network, and programming-language support for massive-scale distributed systems, including Internet services and sensor networks. Welsh has a PhD in computer science from the University of California, Berkeley. He is a member of the ACM and the IEEE. Contact him at mdw@eecs.harvard.edu; www.eecs.harvard.edu/~mdw.