Vineyard Computing: Sensor Networks in Agricultural Production

Using ethnographic research methods, the authors studied the structure of the needs and priorities of people working in a vineyard to gain a better understanding of the potential for sensor networks in agriculture.

Jenna Burrell, Tim Brooke, and Richard Beckwith
Intel Research

Mobile and pervasive computing technologies provide us with some of the first opportunities to explore computing outside climate-controlled building environments. With this freedom comes an endless variety of environments that the research community has just begun to explore as potential sites for technology use. The original pervasive computing systems used office spaces and office mobility as a jumping-off point for concept explorations.[1] We pursued a different approach by looking at work environments outside the office, including medical clinics, manufacturing plants, and farms.

This article discusses an extended study of vineyard workers and their work practices to assess the potential for sensor network systems to aid work in this environment. The study's larger purpose is to find new directions and new topics that pervasive computing and sensor networks might address in designing technologies to support a broader range of users and activities. We expect that much of what we uncovered in this research will be useful to technology design for outdoor environments, other types of agriculture, and mobile work environments in general.

Previous research on sensor network applications has frequently focused on partnerships between technologists providing the sensor networks and biological and environmental researchers studying habitats and endangered species.[2-4] As a potential user group, agriculturalists are distinct from scientists doing habitat research. They focus on production rather than exploratory research, so they're not interested in spending time interpreting data. They want data that recommends a course of action, something that will save them time rather than create additional work. Also, agriculturalists aren't working in remote or fragile environments. They interact closely and physically with crops, touching and examining them each day. They know they can't farm remotely.

These two primary differences in work activities and priorities between agriculturalists and biologists indicates why our study is important in the discussion of sensor network applications. The sensor network application requirements for biological researchers aren't the same as those for agriculturalists and others working on vineyards, farms, or other sites of agricultural production.

In addition to looking at a new category of users, our study is also distinguished by our human-centered research approach. We used ethnographic methods including interviews, site tours, and observational work to broadly understand the work activities and priorities of the various roles working in a vineyard. This rigorous and holistic approach to what software developers might describe as requirements gathering was particularly important because we were studying a population with work activities very different from our own. In contrast to previous sensor network implementation projects, our target users weren't researchers, nor were they approaching their work from a research per-

 Published by the IEEE CS and IEEE ComSoc ■ 1536-1268/04/$20.00

spective. Their mindset was one of production and optimization.

Ethnographic research methods

Our group uses a general research approach that focuses on studying people and practices before technology interventions are designed and put into place. We employ ethnographic methods as a way to gather rich data about the people who inhabit environments that aren't well understood as sites for technology use. In this particular study of vineyards, we looked at people's roles across the entire value chain of wine production, with the belief that each role represents a different relationship with the vineyard and winery and different information and interaction needs.

We conducted semistructured interviews with vineyard managers, vineyard owners, winemakers, vineyard marketing people, and wine sellers. We also conducted site tours and photographed vineyards, wineries, and wine shops guided by our interview subjects. During the busy season, some members of our team became participant observers by joining work parties to help out during harvest and to put up nets to protect the grapes from migratory birds.

After studying the vineyard as a potential site for technology use, we moved into a second phase of the project to develop technology concepts and implement a working sensor network. We created a series of interface designs and technology-interaction concepts that would fit into this work environment on the basis of our analysis of observations and interviews. These concepts were presented to vineyard managers, wine makers, and agricultural researchers for further refinement and development, although they were not deployed as operational user interfaces. A second phase of research involved the limited deployment of a working sensor network in a local Oregon vineyard. The trial installation involved the deployment of approximately eighteen motes for a period of several weeks, during the late summer of 2002. This installation let us come face to face with the challenges of installing computing technology and working with sensor networks outdoors. A third phase of research, not described in this paper involved a much more ambitious sensor network deployment at a vineyard site in British Columbia.

Ethnographic research has proven in the past to be a particularly successful way of inspiring innovative technology concepts that directly address users' needs.[5,6] We also found in our study of vineyards that understanding the potential users' needs and work activities can provide feedback on how existing sensor network hardware and software and other pervasive computing technologies should be configured and redesigned. Our primary goal is to uncover the implications for sensor network design and research arising from user needs and the structure of work activities in agricultural-production environments.

All this new digital data

Pervasive computing technologies—such as sensor network systems—give us new capabilities for sensing and gathering data about an environment and new ways to manage this data digitally. We can gain information about temperature, lighting levels, humidity, the movement and presence of people, and many other aspects of the environment. However, these capabilities pose several questions in the application space. What data should we gather and how often? What level of computational interpretation should we apply to the data? How should we present data to the user? When should the system act on data and when should action be left up to the user? Our interviews and site visits gave us concrete examples of the kinds of sensor network applications that would be appropriate and beneficial in an agricultural environment.

What data should we gather and how often? What level of computational interpretation should we apply to the data? How should we present data to the user?

A combination of three factors provided some answers to these questions: equipment capabilities, environmental conditions, and user needs. Equipment capabilities include battery life limits, processor power, types of available sensors, memory space, sensor accuracy, and radio frequency (RF) transmission range. These factors can make certain potentially useful applications realistically impossible. For example, some researchers have described GPS localization as too power hungry to be realistic in a sensor network. The environment itself also provides answers to questions about data gathering by providing variation within a finite range along certain measurable axes.

In our implementation work, we discovered that great variability across the vineyard during the daytime but less variation at night. For this reason, sensor readings (a function that consumes a significant portion of the battery power) could be taken less frequently during night hours. Similarly, we discovered that variability of conditions across a vineyard might be of greater concern than RF transmission range in determining the density of sensor placement in a network.[7]

Although many environments have a seemingly infinite variety of measurable characteristics, user needs provide another limit on what actually should be measured and how often. For example, we learned from interviews that during the winter, there's a risk of frost damage to vines, so the vineyard needs a system to gather frequent temperature readings at night and alert the manager when the temperature is low. However, this need is seasonal, so frequent night temperature readings combined with a real-time alert system are only necessary in the winter.

Once we know what data will be useful and relevant, the question becomes what kind of computational interpretation do we need and what should we do once the data is interpreted. At one end of the spectrum, the data can simply be delivered raw. This approach has some obvious shortcomings for vineyard workers: raw data might not suggest any course of action, or it might require significant effort to draw useful conclusions. In a production environment, this extra interpretive work can be a significant time burden. At the other end of the spectrum, we might be able to thoroughly interpret the data and perform an action on the user's behalf. Proactive computing recommends this approach to remove the user from direct interaction with the system.[8] The benefit of being proactive is that users aren't overburdened by system demands because they don't interact with it directly.

In our interviews and site visits, vineyard managers indicated what level of data interpretation would provide value to them. These findings illuminate some characteristics of circumstances that recommend proactive computing versus the alternative, which is providing interpreted information without completing any sort of action. In any case, the data must be *actionable*, a term used repeatedly by one of the vineyard managers we interviewed. He wanted the data to suggest a tangible next step, so in our interface design work, we explored several forms of actionable data.

Once we know what data will be useful and relevant, the question becomes what kind of computational interpretation do we need and what should we do once the data is interpreted.

The first was a map of powdery-mildew risk that could be calculated from temperature data readings gathered throughout the vineyard over a period of time. A map generated in this way could easily demonstrate what areas of the vineyard were at the highest risk for powdery mildew and would let the vineyard manager spray pesticides on the specific at-risk area to avoid problems. Unanalyzed temperature data would have been insufficient for this purpose because you calculate powdery-mildew risk using one of a number of complex models that take temperature data gathered over time as input. Temperature data could also be used to make heat unit calculations that vineyard managers use to get a sense of the grapes' ripeness, which is a factor in deciding when to harvest.

Proactive computing

Proactive computing would suggest that we design systems that interpret actionable data and then automatically act on it. Examples in this study's context are

- A vineyard equipped to spray itself in the appropriate area when there's a risk of powdery mildew
- An irrigation system that optimally rations limited ground water
- An automated call to the workers to come in and pick the grapes when they're ripe

In fact, vineyards in New Zealand and Australia mechanically (although not automatically) harvest their grapes because there's no labor pool to draw workers from. In the US, grape-picking teams primarily made up of migrant laborers make more sense economically for a farmer than investing in harvesting equipment.

However, automating the decision to harvest would be less than ideal, mainly because this is often a subjective and social decision based on incomplete information. This is precisely the kind of problem that humans are quite skilled at solving. The winemaker plays the primary role in deciding when to harvest and bases these decisions on the kind of wine the vineyard intends to create. A vineyard manager plays a role in the harvest decision by monitoring weather reports for the threat of rain. Rain can ruin ripe grapes by diluting the potent flavor of each grape or even causing them to burst. If rain is in the near-term forecast, it will often lead to picking the grapes before they're perfectly ripe.

Because weather is so unpredictable, the decision to harvest is always a judgment call. Because many vineyards are located in the same area, there is also the challenge of scheduling the local crew of workers to harvest the plants because all proximate vineyards typically decide to pick at around the same time. There's social pressure and competitiveness among local vineyards. We talked to one vineyard manager of several vineyards who used this to his advantage. To influence the decision about when to harvest, he'd mention to the winemaker or vineyard owner that another vineyard had decided to harvest—thus pressuring the winemaker or owner to follow suit. It was a subtle form of manipulation on the manager's part. So, the decision to harvest isn't well suited to a proactive-computing approach because it results from social factors and incomplete data and is not too difficult for humans to do on their own.

In contrast, other vineyard processes do lend themselves to a proactive-computing model. For example, irrigation is a major issue in many vineyards. An ideal proactive system would optimize water needs in different areas of the vineyard with available water—particularly because water is a limited, shared resource. Being able to water plants more selectively and precisely on the basis of individual plant needs and available water would save water. This type of precision would be time-consuming for a vineyard manager or worker, so a proactive system that does it on the manager's behalf makes sense.

Similarly, dealing with pests is another opportunity for proactive computing. It wouldn't make sense, for example, to detect the presence of birds and alert the manager about the problem. This could happen many, many times throughout the day, and birds require a more immediate reaction than the manager can provide. It only takes a minute or two for a flock of birds to do serious damage to a grape crop. A proactive approach would detect and respond to the bird presence, perhaps by shooting off a loud cannon. We were told in interviews that shooting off a loud cannon periodically is one approach to dealing with bird threats. However, birds often get accustomed to the same loud sound and continue to eat grapes in spite of the cannons.

In these examples, proactive computing plays an important role in dealing with problems with two characteristics: those that require more immediate reaction than human capabilities can fulfill and those that require time-consuming activities that would overburden vineyard workers. In the case of irrigation, there's a sophisticated level of optimization and computational work involved that computing power can help address. The financial investment involved in equipping a vineyard with proactive systems will be an important consideration. Some work that is repetitive and time-consuming, such as pruning, will continue to be done by workers because, compared to the cost of labor, equipment to do the task is too expensive or too complex to automate.

Our findings about the need for actionable data also led us to conclude that pervasive computing systems would need to be designed with domain experts' involvement. For example, the models one might use to illustrate powdery-mildew risks in our interfaces were developed at the University of California at Davis as part of the viticulture research program.[9] It will most likely be agricultural researchers who take the capabilities provided by ubiquitous-computing technologies and connect them with applied uses in the vineyard. Sensor-net equipment will also play a role in domain-specific research by enabling researchers to gather new data that could lead to new knowledge about growing grapes and other types of crops. Perhaps not surprisingly, the researchers we've been in contact with have already shown great interest in using these technologies for research purposes.

> Our findings about the need for actionable data led us to conclude that pervasive computing systems would need to be designed with domain experts' involvement.

Human touchpoints

The concept of *human touchpoints* can be a useful way to think about user interaction with pervasive computing systems. We define a human touchpoint as a portal that connects an individual with the underlying system infrastructure—in this case a sensor network—either by supplying representations of data gathered by the infrastructure or by placing the individual in the role of providing input. What is characteristic of pervasive computing systems is that a single system can have multiple human touchpoints of various types. In our study of people in the vineyard and winemaking industry, we found that providing a variety of human touchpoints was important to address the different roles and responsibilities of a heterogeneous population of potential users that included vineyard managers, hired temporary labor, winemakers, and vineyard owners.

How should data be presented to the user? In what ways can users input data into the system? In our interviews, we uncovered divergent sets of priorities and tasks associated with different roles. The vineyard manager is an agriculturalist who knows about pests, irrigation needs, and all the information associated with successfully growing high-quality grapes. The manager also does business and personnel management work and handles timecards, budgets, and work delega-

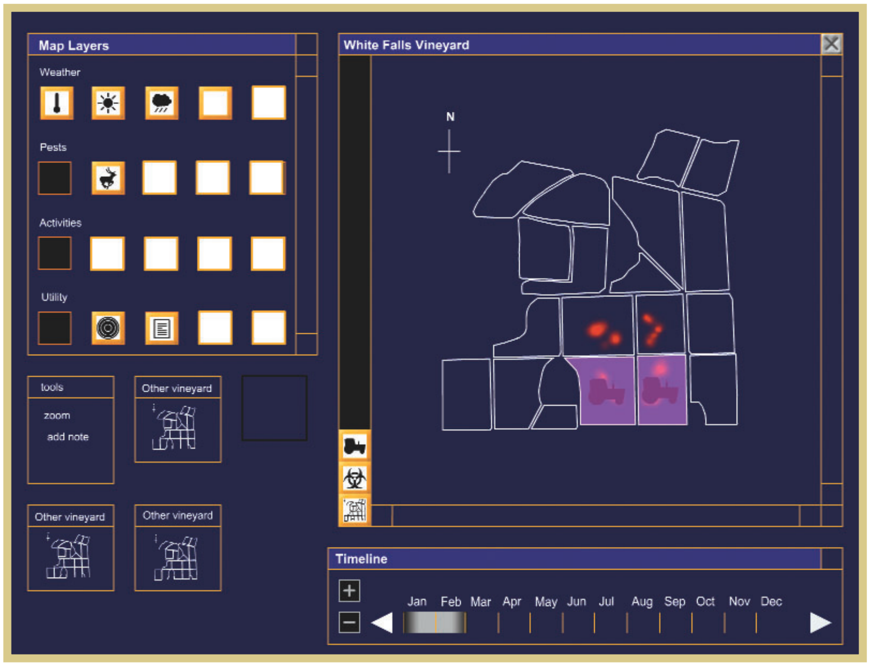

Figure 1. A vineyard manager's interface shows a map of grapes in the vineyard, patches with high powdery-mildew risk, and areas that have been sprayed with pesticides via tractor.

tion. The winemaker is an artisan as well as a scientist who uses both chemistry and good taste to transform grapes into fine wine and blend various wines into something interesting, complex, or marketable. Vineyard workers in our area of the country are migrant workers and often speak only Spanish. They work in teams during the harvest and are paid according to how much they pick. The weight of picked grapes is tracked and associated with each worker. From these examples, it's apparent that

- The vineyard manager has management responsibilities that the winemaker and vineyard workers do not.
- Winemakers have a subjective element in their work process that vineyard managers and vineyard workers do not.
- Vineyard workers have a significant manual-labor component in their activities and often don't even speak the same language as the vineyard manager and winemakers.

Our interest in the roles engaging in collaborative work suggests the relevance of research in the domain of computer-supported cooperative work (CSCW). Pervasive computing systems often have some of the same characteristics as networked groupware applications that CSCW researchers study. Both fields must address the needs of heterogeneous user populations working collaboratively. However, pervasive computing systems differ from traditional CSCW applications and technologies because of their strong tie to physical environments and physical activities that involve and emphasize tool use and the location of activities, rather than information management and knowledge work as is typical in office environments. The needs of different roles in the vineyard go far beyond providing access to different kinds of information; these roles represent completely different work paradigms. Human touchpoints in pervasive computing systems must negotiate between these paradigms.

For example, an interface that negotiates between these roles would provide multiple interdependent interfaces suited to each role; it might address the vineyard manager's job of managing and coordinating activities and paperwork. This task falls outside the weather- and environment-monitoring capabilities we generally assume sensor network systems are good for. Through interviews, we discovered that vineyard managers are interested in ways that technology can help them with business management tasks, which often involve a lot of time-consuming data entry. A sensor network could support management needs by tracking activities, personnel, and equipment through the vineyard and incorporating this data automatically into budgets and timecards.

For the system to work, it would require multiple human touchpoints. One touchpoint would allow the manager to call up vineyard activity data and view it. A second touchpoint would allow vineyard workers to enter input about their activities into the system. Because the workers are primarily manual laborers, a system requiring them to type or explicitly enter data would interfere with their primary work activities. To resolve this issue, we developed the concept of tagged tools as a way to help gather data for budgeting and activity tracking.

For example, the manager might want to know when and where the vineyard was sprayed with pesticides to assess the risk of a powdery-mildew outbreak (see Figure 1). If we instrument the vineyard with a static sensor network, a pesticide sprayer tagged with an RF identification tag or sensor network mote could be operated by a vineyard worker and tracked as it sprayed areas of the vineyard. The pesticide sprayer moving through the vineyard would then be the worker's human touchpoint to serve as the input device into the sensor network system. This input device would operate within the vineyard worker's work paradigm while still providing for the vineyard manager's information needs.

Similarly, pruning shears, shovels, and picking boxes could also be given unique RF identification tags to track the loca-

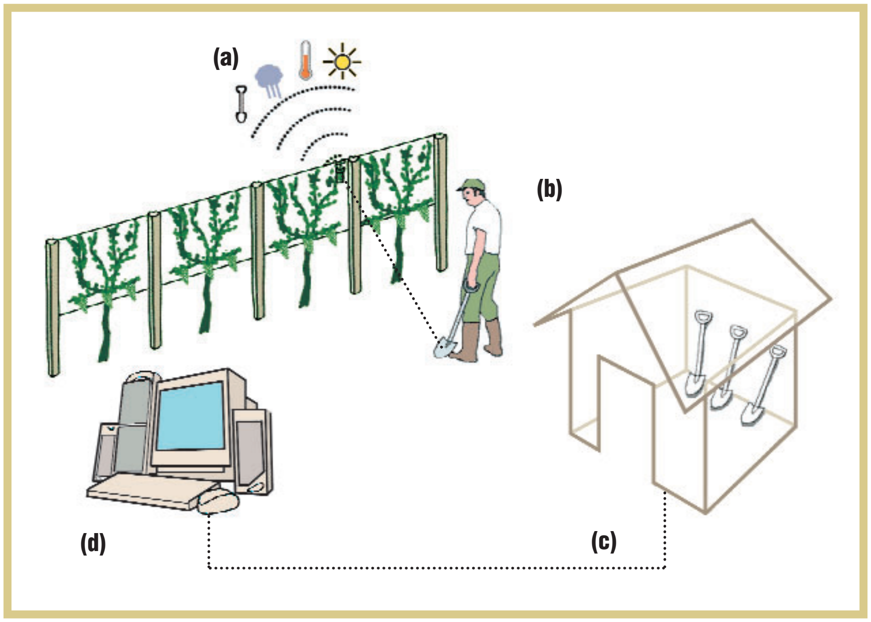

Figure 2. Data mule system architecture in the vineyard. (a) The motes record environmental data and vineyard activities. (b) In the course of daily activities, the worker collects more data onto the shovel. (c) The worker takes the tool back to the shed. (d) Back in the tool shed, the shovels upload their data to the central database.

tion and type of activity. By selecting and using these tools, the vineyard worker would provide the necessary input into the system naturally and effortlessly. The concept of tracking workers' movement through space has already been suggested as a useful tool for generating billing reports and time studies in custodial environments such as hospitals.[10] We envision an instantiation of this idea with the added concept of tagged tools to provide an indication of workers' activities. The manager's need to track activities in the vineyard also suggests that focusing attention on developing localization algorithms for sensor networks—specifically tracking the location of tagged objects moving through a sensor network—is a research direction potentially useful for agricultural applications.

System architecture

Our study also suggested different types of interfaces that could be seamlessly incorporated into the vineyard, including the tagged tools described earlier. However, our understanding of the workflow also suggested some ways that the system infrastructure itself could be reorganized to optimize power management and equipment costs. Our efforts to create a working sensor network implementation in a local vineyard gave us some insight into the interplay between power management, equipment costs, system architecture, and user needs.

Power management is one of the primary issues in the design of sensor network systems intended to operate wirelessly.[11] An ideal system would be a sensor network made up of devices that have an extremely long battery life and are automatically rechargeable or are tiny, disposable, inexpensive, and easily replaced. The concepts of Smart Dust and Paintable Computers are two proposals of this ideal vision.[12,13] Because we believe that sensor networks are useful in the near term, we must realistically face power management issues to avoid the worst-case scenario where batteries must be frequently replaced in hundreds or thousands of individual devices. We have uncovered opportunities for a system wide approach to power management by designing the software and system architecture to optimize power management. However, our modest gains could be greatly improved if the hardware were redesigned with these system wide configurations in mind.

Self-organizing ad hoc sensor networks are generally considered the default system architecture, in part because they present more interesting computational problems for computer scientists to tackle. However, this architecture assumes RF connections, often using TDMA (time division multiple access, a technology for delivering digital wireless service) between each mote and its neighbors. This arrangement of system components requires enough equipment to cover a space with a fully connected network. It's an optimal architecture for some types of applications but is by no means the only one or always the ideal arrangement of the network. Specifically, the self-organizing multihop architecture that forwards data is the only architecture that makes much sense for sensor network applications in remote, inaccessible environments.

We discovered that other system architectures could be employed in vineyards because they are neither remote nor inaccessible. For example, one architecture used *data mules* to collect and transport data from sensor network motes distributed throughout the vineyard (see Figure 2).[14] From our interviews and observations, we learned that during the growing season, workers move up and down the rows a lot. In one vineyard, two family dogs also spent a lot of time going up and down the rows. Any of these moving bodies (even the dogs) could serve as a "data mule" by carrying a small device that simply and invisibly gathers data wirelessly from the static, distributed motes.

The data would be transmitted from the static mote to the data mule mote whenever the two motes are in physical proximity and there's new data to transmit. This configuration does not represent a distinct advantage in terms of power savings because static motes must remain in RF listening mode in order to communicate with the data mule mote. However, it does save on equipment costs because we can distribute motes sparsely throughout the vineyard since they don't need to communicate with neighboring motes. Several applications in the vineyard lend themselves to in-network data storage and processing. When combined with infrequent synchronization, this configuration has the potential for significant power savings and is amenable to a data mule solution. In particular, in-mote distributed processing could be used to calculate heat units that determine the appropriate time to harvest grapes. Because vineyard managers don't need to calculate heat units immediately—a latency of a few hours or a day is suitable—this application doesn't need a connected live-data sensor network. The need for application-specific data aggregation and in-network processing is a unique requirement of sensor networks. This requirement distinguishes sensor networks from traditional wireless networks.[15] We employed this strategy by using in-network data processing to reduce the quantity and frequency of RF transmission. To calculate heat units, we simply needed the daily high and low temperatures. Each mote gathered data once every 60 seconds and then compared each data reading to stored high and low temperature points for the day. A new low or new high would replace the old one.

The only data that needed to be transmitted via RF was the absolute maximum and minimum temperature for the day, because this is all that was required to calculate heat units. It should be noted that we used Eeprom (electrically erasable programmable read-only memory) in our implementation to store data locally on the mote. To effect power savings, a more power efficient technology, such as flash, would be necessary. In fact, flash was built into the sensor network motes we used, but writing data to flash was not yet implemented in the TinyOS version we were using. Our ability to design a system that limited RF transmission and took advantage of in-network data processing rested on our understanding of vineyard work. We learned from talking to vineyard managers that heat-unit calculations were useful, actionable data that would impact harvest. And we learned what data was required to make these calculations. We determined that in-network processing was possible because the situation required only simple calculations.

Observing the constant movement of people and dogs in the vineyard led us to consider a system architecture that relied on data mules to reduce equipment costs. Vineyard managers' use of heat units to make harvest decisions led us to use power-efficient in-network data processing. In these examples, the vineyard work patterns directly influenced our ability to create a useful sensor network application and to optimize it to conserve power and save on equipment costs.

The way work is done in a vineyard has direct implications for designing and configuring these environments' sensor networks.

While exploring the potential for sensor networks in agriculture, we gained an understanding of the structure of vineyard work, the needs and priorities of the people who work there, and the interaction between various stakeholders and roles involved. We found that the way work is done in a vineyard has direct implications for designing and configuring these environments' sensor networks. Looking toward the future of sensor network research, we can recommend several areas where pervasive technology and sensor network researchers might focus their efforts to address the needs and priorities of people working in agricultural environments. One area of need is supporting alternative system architectures.

Because agricultural work involves daily movement through the farm, using data mules is a sensible approach to reduce equipment cost. We also need good localization algorithms to track equipment and people moving through the space. This capability would provide useful data for management needs, including budgets, timecards, and government-regulation paperwork. Agricultural environments also could use proactive-computing approaches that can act on the user's behalf for applications requiring a faster-than-human response time or that require precise, time-consuming optimization. Research on optimized networks to loop sensor data with actuators would provide for proactive applications. Irrigation, frost detection, and pest detection are all examples of applications in agriculture that would benefit from proactive approaches.

This article has not described a single, comprehensive solution for equipping agricultural environments but a variety of sensor network configurations and applications that can address different priorities in the vineyard. Some of the sensor network configurations and features we've described are compatible with each other and some aren't. For example, a sparse distribution of sensor network motes using data mules for data forwarding won't support localization algorithms that rely on triangulation. Different system configurations will vary by cost and capabilities. In practice, there will likely be a plurality of useful sensor network systems employed in agricultural environments to address different priorities.

For example, a simple sparse network employing data mules might be a useful, inexpensive entry-level system that can be upgraded later to include more motes and provide precise localization capabilities. Similarly, a data mule system architecture will not support proactive computing applications that require real-time response. However, agricultural work depends on seasons and time of day, so a sensor network that can self-configure according to temporal factors could combine some of these approaches. For example, a proactive system could monitor for frost during winter nights or for birds during bird migration. Other times of year, the system would use a data mule approach. These examples suggest the potential for several creative strategies for combining capabilities and system configurations.

Taking a high-level view, the interface design and implementation of human touchpoints in the sensor network infrastructure must take into account collaborative work environments and provide mediation between vineyard managers, owners, workers, and winemakers. Research in any of these areas will be useful in the eventual development of sensor network technologies as consumer products for agricultural monitoring.

REFERENCES

1. R. Want et al., *The Parctab Ubiquitous Computing Experiment*, tech. report, Xerox PARC, 1995.

2. A. Cerpa et al., "Habitat Monitoring: Application Driver for Wireless Communications Technology," *Proc. ACM SIGCOMM Workshop Data Communications in Latin America and the Caribbean*, ACM Press, pp. 20–41.

3. A. Mainwaring et al., "Wireless Sensor Networks for Habitat Monitoring," *Proc. ACM Int'l Workshop Wireless Sensor Networks and Applications*, ACM Press, 2002, pp. 88–97.

4. E. Biagioni and K.W. Bridges, "The Application of Remote Sensor Technology to Assist the Recovery of Rare and Endangered Species," *Int'l J. High Performance Computing Applications*, vol. 16, no. 3, Aug. 2002, pp. 315–324.

5. T. Salvador, G. Bell, and K. Anderson, "Design Ethnography," *Design Management J.*, vol. 10, no. 4, Fall 1999, pp. 35–41.

6. S. Lewis et al., "Ethnographic Data for Product Development: A Collaborative Process," *Interactions*, vol. 3, no. 6, Nov./Dec. 1996, pp 52–69.

7. D. Estrin et al., "Instrumenting the World with Wireless Sensor Networks," *Proc. Int'l Conf. Acoustics, Speech, and Signal Processng* (ICASSP 2001), vol. 4, IEEE Press, 2001, pp. 2033–2036.

8. D. Tennenhouse, "Proactive Computing," *Comm. ACM*, vol. 43, no. 5, May 2000, pp 43–50.

9. "Crop: Grape Disease: Powdery Mildew," *Disease Model Database*, 2001, www.ipm.ucdavis.edu/DISEASE/DATABASE/grapepowderymildew.html.

10. G.T. Carroll et al., *Time and Accounting System*, US patent 4549264, Patent and Trademark Office, 1983.

11. J. Lifton, "Pushpin Computing System Overview: A Platform for Distributed, Embedded, Ubiquitous Sensor Networks," *Proc. 1st Int'l Pervasive Computing Conf. 2002, Lecture Notes in Computer Science, no. 2414*, Springer-Verlag, 2002, pp. 139–151.

12. W. Butera, *Programming a Paintable Computer*, doctoral dissertation, MIT Media Laboratory, 2002.

13. D. Culler et al., "A Network-Centric Approach to Embedded Software for Tiny Devices," *Proc. 1st ACM Int'l Conf. Embedded Software* (EMSOFT 01), ACM Press, 2001, pp. 114–130.

l.,

modeling a three-tier architecture for sparse sensor networks," 2003, IRS-TR-03-001; www.intel-research.net/Publications/Seattle/012220031206_114.pdf.

15. D. Estrin, "Next Century Challenges: Scalable Coordination on Sensor Networks," *Proc. 5th Ann. Int'l Conf. Mobile Computing and Networking* (MobiCom 99), ACM Press, 1999, pp. 263–270.

the AUTHORS

Jenna Burrell is a doctoral student in sociology at the London School of Economics. As an intern and later as a full-time employee with Intel's People and Practices Research Group, she did research on animators, college students, fab technicians, and vineyard workers. She received a BA in computer science from Cornell University. Contact her at 2111 NE 25th Ave. (JF3-377), Hillsboro, OR 97124-5961; jenna@jennamail.com.

Tim Brooke is an interaction designer with the People and Practices Research Group at Intel. His current research focuses on the kind of user experiences that ubiquitous computing can enable. He received an MA in computer-related design from the Royal College of Art in the UK and is a member of the IEE (UK). Contact him at 2111 NE 25th Ave. (JF3-377), Hillsboro, OR 97124-5961; timothy.l.brooke@intel.com.

Richard Beckwith is a research psychologist with Intel's Corporate Technology Group. He works in the People and Practices Research group, whose main focus is to find new uses for technology. He received a PhD in developmental psychology from Teachers College, Columbia University. Contact him at 2111 NE 25th Ave. (JF3-377), Hillsboro, OR 97124-5961; richard.beckwith@intel.com.