

# Habitat Monitoring: Application Driver for Wireless Communications Technology

Alberto Cerpa, Jeremy Elson,  
Deborah Estrin\*, Lewis Girod  
UCLA Computer Science Department  
Los Angeles, California 90095-1596  
{cerpa,jelson,destrin,girod}@  
lecs.cs.ucla.edu

Michael Hamilton  
James San Jacinto Mountains Reserve  
Box 1775  
Idyllwild, California 92549  
director@jamesreserve.edu

Jerry Zhao  
USC Information Sciences Institute  
4676 Admiralty Way  
Marina del Rey, California 90292  
zhaoy@isi.edu

## ABSTRACT

As new fabrication and integration technologies reduce the cost and size of micro-sensors and wireless interfaces, it becomes feasible to deploy densely distributed wireless networks of sensors and actuators. These systems promise to revolutionize biological, earth, and environmental monitoring applications, providing data at granularities unrealizable by other means. In addition to the challenges of miniaturization, new system architectures and new network algorithms must be developed to transform the vast quantity of raw sensor data into a manageable stream of high-level data. To address this, we propose a tiered system architecture in which data collected at numerous, inexpensive sensor nodes is filtered by local processing on its way through to larger, more capable and more expensive nodes.

We briefly describe Habitat monitoring as our motivating application and introduce initial system building blocks designed to support this application. The remainder of the paper presents details of our experimental platform.

**Keywords:** low-power wireless, sensor networks, testbeds, applications

## 1. INTRODUCTION

During the last decade, networking technologies have revolutionized the ways individuals and organizations exchange information and coordinate their activities. In this decade we will witness another revolution; this time one that involves observation and control of the physical world. The availability of micro-sensors and low-power wireless communications will enable the deployment of densely distributed sensor/actuator networks for a wide range of biological, earth and environmental monitoring applications in marine, soil, and atmospheric contexts. This technology has particular relevance in many Latin

American countries because of its applicability to environmental monitoring of the diverse and unique ecosystems.

To achieve scalability, robustness, and long-lived operation, sensor nodes themselves will execute significant signal processing, correlation, and network self-configuration inside the network. In this way these systems will emerge as the largest distributed systems ever deployed. These requirements raise fascinating challenges for Information Technology and communication research, as well as for their application domains. One of the novel issues for network design is the shift from manipulation and presentation of symbolic and numeric data to the interaction with the dynamic physical world through sensors and actuators. This raises the need for good physical models, which requires extensive data analysis of monitored data. A second challenge arises from the greatly increased level of environmental dynamics. While all good distributed systems are designed with reliability in mind, these new target applications present a level of ongoing dynamics that far exceeds the norm. Perhaps the most pervasive technical challenge arises from the energy constraints imposed by unattended systems. These systems must be long-lived and vigilant and operate unattended. Unlike traditional Internet systems the energy constraints on un-tethered nodes present enormous design challenges. Finally, as with the Internet, there are scaling challenges. However, given the other characteristics of the problem space, the traditional techniques are not directly applicable, and alternative techniques must be developed.

This paper focuses on a particular application of embedded wireless sensing technology. The habitat sensing array for biocomplexity mapping emphasizes the need for continual automatic self-configuration of the network to adapt to environmental dynamics, and the use of coordinated actuation in the form of programmed triggering of sensing and actuation to enable identification, recording and analysis of interesting events.

We introduce the key architectural principle for constructing long-lived wireless sensor networks, adaptive self-configuration, and then describe its applicability to Habitat monitoring. In the subsequent section we describe our tiered architecture, time synchronization techniques, and experimental platform developed to support this and other applications.

\*Correspondence author.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. SIGCOMM - Latin America & Caribbean 4/01 San Jose, Costa Rica © 2001 ACM ISBN 1-58113-354-5/01/0004...\$5.00

# Monitoreo del Hábitat: Controlador de aplicación para Tecnología de Comunicación Inalámbrica

Alberto Cerpa, Jeremy Elson,  
Deborah Estrin, Lewis Girod  
UCLA Computer Science Department  
Los Angeles, California 90095-1596  
{cerpa,jelson,destrin,girod}@  
lecs.cs.ucla.edu

Michael Hamilton  
James San Jacinto Mountains Reserve  
Box 1775  
Idylwild, California 92549  
[director@jamesreserve.edu](mailto:director@jamesreserve.edu)

Jerry Zhao  
USC Information Sciences Institute  
4676 Admiralty Way  
Marina del Rey, California 90292  
[zhaoy@isi.edu](mailto:zhaoy@isi.edu)

## RESUMEN

A medida que las tecnologías de fabricación e integración reducen el costo y el tamaño de los microsensores e interfases inalámbricas, se torna factible desplegar redes inalámbricas de sensores y actuadores densamente distribuidas. Estos sistemas prometen revolucionar las aplicaciones de monitoreo biológico, terrestre, y ambiental, suministrando datos en granularidades irrealizables por otros medios. Además de los desafíos de la miniaturización, deben desarrollarse las arquitecturas de los nuevos sistemas y los nuevos algoritmos de red para transformar la gran cantidad de datos sensores en bruto en un flujo manejable de datos de alto nivel. Para obtener esto, proponemos una arquitectura de sistema en hilera, en la cual los datos recogidos en numerosos nodos sensores de bajo costo son filtrados mediante un procesamiento local en su pasaje hacia nodos más grandes, más capaces y de mayor costo.

Describimos brevemente el monitoreo del Hábitat como nuestra aplicación motivante, y presentamos los bloques de construcción del sistema inicial diseñados para soportar esta aplicación. El resto del artículo presenta detalles de nuestra plataforma experimental.

**Palabras clave:** inalámbrico de baja potencia, redes sensoras, lechos de prueba, aplicaciones.

## 1. INTRODUCCIÓN

Durante la última década, las tecnologías de red han revolucionado las formas en que los individuos y organizaciones intercambian información y coordinan sus actividades. En esta década presenciaremos otra revolución; esta vez una que abarca la observación y

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.  
SIGCOMM - Latin America & Caribbean 4/01 San José, Costa Rica  
© 2001 ACM ISBN 1-58113-354-5/01/0004...\$5.00

control del mundo físico. La disponibilidad de microsensores y comunicaciones inalámbricas de baja potencia permitirán el despliegue de redes de sensores/actuadores densamente distribuidas para una amplia gama de aplicaciones de monitoreo biológicas, terrestres y ambientales, en contextos marinos, de suelos, y atmosféricos. Esta tecnología tiene una relevancia particular en muchos países de Latinoamérica debido a su aplicabilidad para el monitoreo ambiental de los diversos y únicos ecosistemas.

Para lograr escalabilidad, obtener robustez y alcanzar una operación duradera, los propios nodos sensores ejecutarán dentro de la red un significativo procesamiento de señales, correlación, y autoconfiguración de red. Así, estos sistemas surgirán como los mayores sistemas distribuidos que jamás se hayan desplegado. Estos requerimientos generan desafíos fascinantes para la Tecnología de la Información e investigación en comunicaciones, así como para sus dominios de aplicación. Una de las nuevas fuentes para el diseño de redes es el cambio de la manipulación y presentación de datos simbólicos y numéricos a la interacción con el dinámico mundo físico a través de sensores y actuadores. Esto aumenta la necesidad de buenos modelos físicos, los cuales requieren análisis extensivos de datos monitoreados. Un segundo desafío surge del muy aumentado nivel de dinámica ambiental. Mientras que todos los buenos sistemas distribuidos son diseñados con la mente puesta en la confiabilidad, estas nuevas aplicaciones de objetivo presentan un nivel ininterrumpido de dinámica que excede por lejos a la norma. Quizás el desafío técnico más penetrante surja de las limitaciones de energía impuestas por los sistemas sin vigilancia. Estos sistemas deben ser de larga vida y vigilantes, y operar solos. Al contrario de los sistemas tradicionales de Internet, las limitaciones de energía sobre los nodos desatados presentan enormes desafíos de diseño. Finalmente, al igual que con la Internet, existen desafíos de escala. Sin embargo, dadas las otras características del espacio del problema, las técnicas tradicionales no son aplicables directamente, y deben desarrollarse técnicas alternativas.

## 2. ADAPTIVE SELF-CONFIGURING SYSTEMS

The sheer number of distributed elements in these systems precludes dependence on manual configuration. Furthermore, the environmental dynamics to which these elements must adapt prevents design-time pre-configuration of these systems. Thus, realistic deployments of these unattended networks must self-reconfigure in response to node failure or incremental addition of nodes, and must adapt to changing environmental conditions. If we are to exploit the power of densely distributed sensing, these techniques for adaptation and self-configuration must scale to the anticipated sizes of these deployments. In recent years, some work has begun to allow networks of wireless nodes to discover their neighbors, acquire synchronism, and form efficient routes [Pottie-Kaiser00]. However, this nascent research has not yet addressed many fundamental issues in adaptively self-configuring the more complex sensing and actuation systems described here, particularly those arising from deploying embedded systems in real-world, environmentally-challenging contexts [Estrin-et.al.99]

Driven by our experimental domains, we are using this experimental platform to develop techniques for self-configuration:

- Integrated techniques for self-assembly and self-healing in these deeply distributed systems. These methods should enable self-configuration—both at the lower-level communication layers in addition to higher levels such as distributed name spaces.
- Simple localized algorithms that effect coordinated data collection and processing to achieve measurement aggregation or higher-level alert generation [Abelson99]. Preliminary research indicates that a particular paradigm for network organization, directed diffusion [Intanago-et.al.00], can efficiently achieve such coordination and resource allocation needs, but considerable experimentation and modeling work is still required.
- Protocol and system level techniques that enable energy-efficiency beyond what is feasible with low-power component design alone. Such techniques, designed for robust operation, can achieve system longevity without sacrificing vigilance.
- Techniques for time synchronization and localization in support of coordinated monitoring. At the target node scales, relying on global positioning systems alone may not be appropriate.
- In some contexts the ability of the node to move itself (or selected appendages), or to otherwise influence its location will be critical. Distributed robotics [Mataric95] in a constrained context will greatly extend the capabilities of these systems. Benefits of including self-mobilizing elements [Sukhatme99] are: self-configuring systems to adapt to realities of an inaccessible terrain, developing a robotic ecology for delivering energy sources to other system elements, and obtaining coverage of a larger area.

## 3. HABITAT SENSING ARRAY FOR BIOMOLECULAR MAPPING

The challenge of understanding biocomplexity in the environment requires sophisticated and creative approaches that integrate information across temporal and spatial scales, consider multiple levels of organization and cross-conceptual boundaries [Walker-Steffen97, Gell-Mann95]. Long-term data-collection for systematic and ecological field studies and continuous environmental monitoring are the domain of Biological Field Stations, and offer opportunities to establish cross-cutting and integrated investigations that facilitate studies of biocomplexity [Michener-et.al.98, Lohr-et.al.95]. Over the past two decades we have seen extraordinary developments in the field of remote sensing and automated data collection, resulting in dramatic increases in spatial, spectral and temporal resolution at a geometrically declining cost per unit area [Colwell98]. Multi-purpose data analysis and visualization software provides tools to study large and complex data sets. The Internet facilitates global data access, distributed data processing, collaborative studies, virtual proximity and tele-robotic operation.

Remote sensing from satellite and airborne sensors has proved to be a tremendous tool for studying “large” biodiversity (e.g. spatial complexity of dominant plant species). While many scientists and land managers attempt to study biodiversity using top down remote sensing tools, the fact is that the vast majority of the biodiversity, and resulting biocomplexity, within an ecosystem exists at very small scales, and is not readily observable with even the best airborne and satellite based sensors [Keitt-Milne97]. To get down to where the complexity is, so to speak, sensing and monitoring needs to become ground based [Hamilton92, Hamilton00]. Breakthroughs in VLSI digital signal processing, miniature sensors, low-power micro-controllers and wireless digital networks will make possible the development of cheap and nearly ubiquitous ground-based monitoring systems for outdoor field. Fresh opportunities afforded by these technologies allow us to rethink how Biological Field Stations can participate in the global effort to answer the big questions posed by biocomplexity.

Observation techniques involving cameras and microphones are in increasingly widespread use, however they involve small numbers of devices and require continuous human observation, greatly constraining their capabilities in natural environments. Unattended, heterogeneous sensors/actuators will enable a vast range of new habitat studies via continuous monitoring techniques. The data from such a network will need to be filtered and partially analyzed within the network e.g. seismic sensors could trigger data-intensive assets such as cameras. The proposed technology offers the chance for programmed observation, triggered response with specified patterns, and automatically recorded and reported responses. Such capabilities require the development of robust, adaptive techniques for coordinating across distributed and heterogeneous sensor/actuator nodes, many of which may be wireless and energy-limited.

Fundamental technological advances are needed to enable adaptive, programmable multi-modal networks to identify indicators of interest and use those to trigger analysis, correlation, and recording of events. Moreover, current techniques will not scale to very large numbers of wireless nodes and do not make effective use of multiple sensor modalities. To realize this goal we are developing and planning to deploy unique and innovative capabilities at the James Reserve in Southern California. Three, multi-node monitoring grids (25-100 nodes per grid) will

Este artículo se enfoca en una aplicación particular de tecnología de percepción inalámbrica embebida. El arreglo sensor del hábitat para el trazado de biocomplejidad enfatiza la necesidad de una auto-configuración automática continua de la red para adaptarse a la dinámica ambiental, y el uso de una actuación coordinada en la forma de una activación programada de percepción y actuación para permitir la identificación, registro y análisis de eventos interesantes.

Introducimos el principio arquitectónico clave para la construcción de redes sensoras inalámbricas de larga vida, de auto-configuración adaptiva, y luego describimos su aplicabilidad para el monitoreo del Hábitat. En la siguiente sección describimos nuestra arquitectura en hileras, las técnicas de sincronización del tiempo, y la plataforma experimental desarrollada para soportar ésta y otras aplicaciones.

## 2 . SISTEMAS AUTO-CONFIGURANTES ADAPTIVOS

El mero número de elementos distribuidos en estos sistemas imposibilita la dependencia en la configuración manual. Además, la dinámica ambiental a la cual estos elementos deben adaptarse, impide la pre-configuración diseño-tiempo de estos sistemas. Así, los despliegues realistas de estas redes sin vigilancia deben auto-reconfigurarse en respuesta a una falla del nodo o a una adición incremental de nodos, y deben adaptarse a condiciones ambientales cambiantes. Si deseamos aprovechar la capacidad de percepción densamente distribuida, estas técnicas para adaptación y auto-configuración deben estar a escala con los tamaños anticipados de estos despliegues. En los últimos años, algún trabajo ha comenzado a permitir que las redes de nodos inalámbricos descubran a sus vecinos, adquieran sincronismo, y formen rutas eficientes [Pottie-Kaiser00]. Sin embargo, esta investigación naciente no ha tratado aún muchos asuntos fundamentales en la auto-configuración adaptiva de los sistemas de actuación y percepción más complejos aquí descritos, particularmente aquellos que surgen del despliegue de sistemas embebidos, en contextos ambientalmente desafiantes del mundo real [Estrin et al.99].

Llevados por nuestros dominios experimentales, estamos usando esta plataforma experimental para desarrollar técnicas de auto-configuración:

- Técnicas integradas para el auto-ensamble y auto-reparación en estos sistemas profundamente distribuidos. Estos métodos deberían permitir la auto-configuración, tanto en las capas de comunicación de niveles más

bajos, como en los niveles más altos tales como los espacios de nombre distribuidos.

- Algoritmos localizados simples que realizan una recolección y procesamiento de datos coordinados para obtener un conjunto de medidas o la generación de alertas de nivel más alto [Abelson99]. Las investigaciones preliminares indican que un paradigma particular para una organización de red de difusión dirigida [Intanago et al.00], puede alcanzar de modo eficiente tal coordinación y necesidades de asignación de recursos, pero todavía se requiere de mucha experimentación y modelaje.
- Protocolos y técnicas de nivel de sistema que permitan eficiencia en el uso de energía más allá de lo que es factible usando solamente diseño de componentes de baja potencia. Tales técnicas, diseñadas para operaciones sólidas, pueden alcanzar la vida útil del sistema sin sacrificar la vigilancia.
- Técnicas para la sincronización del tiempo y la localización en el soporte del monitoreo coordinado. A escala del nodo objetivo, confiar solamente en los sistemas de posicionamiento global puede no ser apropiado.
- En algunos contextos, la capacidad de moverse del nodo (o de apéndices seleccionados), o de influir de alguna otra manera en su ubicación, será crítica. La robótica distribuida [Mataric95] en un contexto restringido extenderá ampliamente las capacidades de estos sistemas. Los beneficios de incluir elementos automovilizantes [Sukhatme99] son: sistemas auto-configurantes para adaptarse a las realidades de un terreno inaccesible, desarrollando una ecología robótica para la distribución de fuentes de energía a otros elementos del sistema, y obteniendo cobertura de un área mayor.

## 3. ARREGLO DE SENsoRES EN EL HÁBITAT PARA EL TRAZADO DE BIOCOMPLEJIDAD

El desafío de entender la biocomplejidad en el ambiente requiere aproximaciones sofisticadas y creativas que integren información por sobre las escalas temporales y espaciales, consideren múltiples niveles de organización y crucen límites conceptuales [Walker-Steffen97, Gell-Mann95]. La recolección de datos a largo plazo para estudios de campo sistemáticos y ecológicos, y para el

be implemented for fixed view multimedia and environmental sensor data loggers (using wireless technologies and solar power, and ultimately capable of limited mobility, unique observation scales, proximity detection, and environmental ruggedness). We will develop and implement coordinated actuation to support experiments such as triggered emission and recording of acoustic signals from target species. Multiple perspective monitoring will be integrated through the addition of tower-based video cameras for coordinated hyper-stereoptical mapping (3D) of canopy topology and volume, monitoring of seasonal phenology of overstory tree species, and mid-ground level vegetation within the monitoring grids. Mobile nodes will also be integrated, such as an all-terrain robot for remote viewing, high resolution dimensional imaging, and “gap filling data collection” within each monitoring grid. In the long term we will incorporate tagged-animals into the system through the use of micro-RFID tags. All of these capabilities will require application of self-configuring and energy-conserving algorithms and protocols to achieve ad hoc, wireless system deployment and operation in uncontrollable environmental conditions.

This network will allow us to develop scalable techniques for non-destructive, multi-scale spatio-temporal sampling and biocomplexity data visualizations, thus enabling the rapid and low-cost mapping of new dynamic scales of species diversity, ecosystem structure, and environmental change. The facilities will provide on-site and Internet-based opportunities for graduate students and faculty to utilize these new tools, including training and research application consulting. This technology promises great opportunities in education and research alike.

In the following sections we describe two critical system building blocks needed to realize long-lived wireless sensor networks, and then present the details of our experimental platform.

## 4. TWO SYSTEM BUILDING BLOCKS FOR LOW DUTY-CYCLE OPERATION

While many aspects of deployed sensor networks will be specific to particular applications, we have begun to identify key building blocks that will enable a wide range of applications. This section discusses a “Frisbee” model for low-duty-cycle operation and techniques for time synchronization.

### 4.1 The “Frisbee” Model

One of the most important design goals of sensor network applications is minimizing the power used. Energy is a precious resource in sensor networks because there is a finite amount available; every use of it directly affects the lifetime of the network [Kaiser-Pottie00]. This makes energy-efficiency critical.

In networks where “interesting” events happen infrequently, an effective way of saving power is to turn nodes off when they are not needed. In an ideal but unattainable world, a sensor network is completely asleep during the long lulls in activity. When something does happen, only a limited zone of the network that is close to the event is kept in its fully active state. The active zone should be centered at the current location of a target phenomenon that is being tracked; and, of course, the zone should move through the network along with the target. Nodes that are not within sensing range of the event are outside of the zone, and therefore do not waste energy on data acquisition or

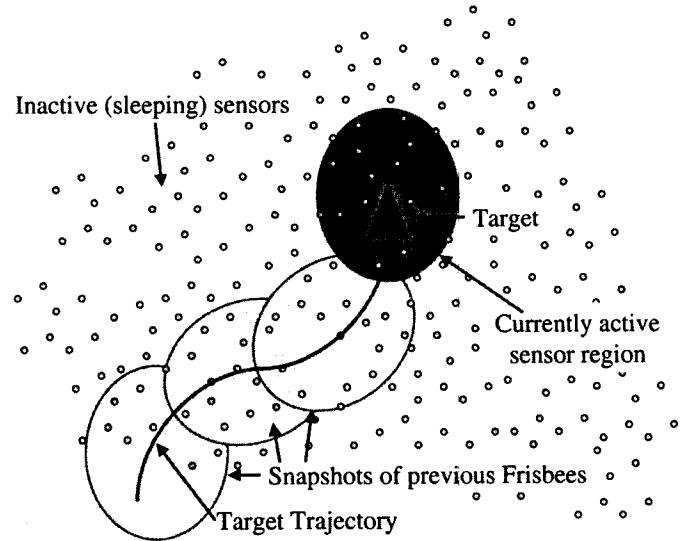


Figure 1: The “Frisbee” model. The triangle represents the target being tracked by the network. The shaded regions are the sets of sensors awakened over time.

housekeeping tasks such as maintaining time synchronization. Optimally, the zone should move such that a phenomenon of interest is always kept inside of the zone. The zone might be circular, with its radius proportional to the speed of the object. We call this model of a constantly moving circular zone the “Frisbee” model.

To implement an approximation of the optimal Frisbee, two essential components required: (1) A low-power operating mode with wakeup: some sensors must always be vigilant; others must have a way of saving power such that they can be awakened by an external stimulus; and (2) Definition of the Frisbee boundary: sensors must use localized algorithms so that nodes can autonomously decide whether or not they are part of the Frisbee; i.e., entering or leaving it.

#### 4.1.1 Power Savings with Wakeup

At the core of our idea is the fact that sensor nodes can be designed with a “power-saving” mode in which they are asleep. This sleep mode must require less power than a mode in which nodes are maintaining active vigilance. This mode is distinguished from simply being off because it must be possible for an external stimulus to “wake up” the nodes, bringing them from their low-power mode into a vigilant mode. The nodes should be able to generate this stimulus so that they can awaken their peers.

Most nodes should be asleep most of the time. Some nodes will remain awake all the time to serve as “sentries” – performing tasks such as accepting tasking instructions from users and looking for a potential event of interest. When a sentry senses a nearby event, it wakes up nodes in its vicinity, allowing the group to acquire data with finer granularity or a wider range of modalities than would be possible with a single (sentry) sensor. As the target moves, nodes will send further wakeup signals to other nodes in the target’s direction of motion. We envi-

monitoreo ambiental continuo, son dominio de las Estaciones de Campo Biológicas, y ofrecen oportunidades para establecer intercalados e investigaciones integradas que faciliten los estudios de biocomplejidad [Michener et al.98, Lohr-et.al.95]. Durante las pasadas dos décadas hemos visto desarrollos extraordinarios en el campo de la percepción remota, y de la recolección de datos automatizada, resultando en aumentos muy importantes en la resolución espacial, espectral y temporal a un costo por unidad de área, geométricamente declinante [Colwell98]. Los análisis de datos multi-propósito y la visualización del software suministran herramientas para estudiar grandes y complejos conjuntos de datos. La Internet facilita el acceso global a los datos, el procesamiento de datos distribuidos, los estudios en colaboración, la proximidad virtual, y la operación tele-robótica.

La percepción remota desde satélites y sensores de transmisión aérea ha demostrado ser una herramienta muy útil para el estudio de la "gran" biodiversidad (por ej. complejidad espacial de las especies de plantas dominantes). Mientras que muchos científicos y terratenientes intentan estudiar la biodiversidad utilizando herramientas de percepción remota desde la altura hacia abajo, el hecho es que la amplia mayoría de la biodiversidad, y de la biocomplejidad resultante, existen dentro de un ecosistema a escalas muy pequeñas, y no son fácilmente observables aún con los mejores sensores de transmisión aérea o de base satelital [Keitt-Milne97]. Para llegar a donde se encuentra la complejidad, por así decirlo, la percepción y el monitoreo necesitan tener su base en el terreno [Hamilton92, Hamilton00]. Los avances en el procesamiento de señales digitales VLSI, los sensores en miniatura, los micro-controladores de baja potencia y las redes digitales inalámbricas harán posible el desarrollo de sistemas de monitoreo de base terrenal, para el campo exterior baratos y casi ubicuos. Las nuevas oportunidades que brindan estas tecnologías nos permiten repensar de qué modo las Estaciones de Campo Biológicas pueden participar en el esfuerzo global para responder a los grandes interrogantes planteados por la biocomplejidad.

Las técnicas de observación que incluyen cámaras y micrófonos están teniendo un uso cada vez mayor, sin embargo, incluyen pequeñas cantidades de dispositivos y requieren de una observación humana continua, restringiendo en gran forma sus capacidades en ambientes naturales. Los sensores/actuadores heterogéneos y sin vigilancia permitirán una amplia gama de nuevos estudios de hábitat a través de técnicas de monitoreo continuo. Los datos de tal red necesitarán ser filtrados y parcialmente analizados dentro de la red; por ej. los sensores sísmicos podrían activar recursos de

datos intensivos tales como cámaras. La tecnología propuesta ofrece la posibilidad de observación programada, respuesta activada con patrones especificados, y respuestas automáticamente registradas e informadas. Tales capacidades requieren el desarrollo de técnicas sólidas y adaptivas para coordinar por sobre nodos sensores/actuadores heterogéneos, muchos de los cuales pueden ser inalámbricos y de energía limitada.

Son necesarios avances tecnológicos fundamentales para permitir que redes multimodales programables y adaptivas identifiquen indicadores de interés, y sean usadas para activar el análisis, correlación y registro de eventos. Mas aún, las técnicas actuales no llegarán a números muy altos de nodos inalámbricos y no harán uso efectivo de las modalidades de sensores múltiples. Para alcanzar esta meta estamos desarrollando y planeando desplegar capacidades únicas e innovativas en la Reserva *James* de California del Sur. Tres rejillas de monitoreo multi-modal (25-100 nodos por rejilla) serán implementadas para multimedia de visión fija y compiladores de datos de sensores ambientales (utilizando tecnologías inalámbricas y energía solar, y finalmente capaces de movilidad limitada, escalas de observación únicas, detección de proximidad y ruedas ambientales). Desarrolla-remos e implementaremos una acción coordinada para apoyar experimentos tales como emisión activada y registro objetivo de señales acústicas de especies. Se integrará un monitoreo multi-perspectiva a través de la adición de video cámaras con base en las torres para el trazado hiper-estéreo-óptico coordinado (3D) de la topología y volumen de la bóveda arbórea, el monitoreo de la fenomenología estacional de las especies arbóreas de follaje más alto, y de la vegetación a nivel medio del terreno, dentro de las rejillas de monitoreo. Los nodos móviles serán también integrados, tal como un robot todo-terreno, para visión remota, imagenología dimensional de alta resolución, y "recolección de datos para cubrir faltantes" dentro de cada rejilla de monitoreo. A largo plazo, incorporaremos animales etiquetados dentro del sistema a través del uso de micro etiquetas RFID. La totalidad de estas capacidades requerirán la aplicación de algoritmos auto-configurentes y conservadores de energía, y de protocolos para obtener en forma *ad hoc* un despliegue y operación del sistema inalámbrico en condiciones ambientales incontrolables.

Esta red nos permitirá desarrollar técnicas escalables para muestreos espacio temporal multi-escala, no destructivos, y visualizaciones de datos de biocomplejidad, permitiendo así el trazado rápido y a bajo costo de nuevas escalas dinámicas de diversidad de especies, estructura del ecosistema, y cambio ambiental. Las instalaciones brindarán oportunidades en el sitio y basadas en Internet para estudiantes graduados, y la

sion a “wakeup waveform” that precedes the phenomenon being tracked. As the target moves beyond the useful range of sensor nodes, those nodes go back to sleep.

An interesting observation is that by increasing the density of nodes we may be able to guarantee that enough nodes will be listening soon enough to achieve low latency wakeup, even with relatively low duty cycle wakeup at the level of individual nodes. On detection we might wake up every node that comes on cycle, so that over the course of an entire cycle, gradually all the nodes in the vicinity would be wakened. In a sense we are exploiting parallelism, by using larger numbers of low duty cycle (low power) nodes, instead of smaller numbers of higher power nodes.

#### 4.1.2 Localized Algorithms for Defining the Frisbee Boundary

A central theme pervading the design of all sensor networks is a fully distributed, decentralized design. This naturally extends to the algorithms that we plan to develop for defining the Frisbee boundary. That is to say, each node will autonomously decide whether or not it is “in the Frisbee” – there is no central controller that doles out instructions to nodes.

One example of a simple distributed algorithm for this is as follows. Nodes that have recently detected the target wake up all other nodes within a fixed radius. If possible, neighboring nodes that are already awake should be queried so that the target’s speed and direction can be determined, in order to better shape and direct the “wakeup waveform.” After a certain amount of time without any detected activity, the nodes time out and return to their previous sleep state.

The design of this algorithm will also incorporate the idea of **adaptive fidelity**. In motion tracking, ever-denser distributions of sensors will usually lead to increasingly precise tracking results. However, if a particular application considers network lifetime to be more important than tracking precision, it is possible to adjust this by only waking up, say, one-half or one-quarter of the nodes that fall physically within the Frisbee. Another potential variation is that sensors closer to the center of the Frisbee are awakened with a higher density than nodes near the edges.

With a framework such as the above in place, each application will be able to tune its desired precision and commensurate energy expenditure. Applications can define what percentage of the sensor nodes within a Frisbee should be awakened every time the Frisbee moves into a new region. The decision of which exact nodes should actually be awakened is, again, a decentralized decision, likely to be made through randomization or some other technique.

## 4.2 Time Synchronization

Time synchronization is a building block critical to many aspects of a distributed sensor system. For example, it is critical to a common sensor network feature: data aggregation. Due to the high energy cost of communication compared to computation [Kaiser-Pottie00], local processing, summarization, and aggregation of data is often employed in order to minimize the size and frequency of transmissions. Suppression of duplicate notifications of the same event from a group of nearby sensors

can result in significant energy savings [Intanago00]. To recognize duplicates, events must be timestamped with a precision on the same order as the event frequency. Correct time synchronization can also lead to energy savings in systems that use TDMA or other types of scheduled wake-ups; the size of the guard bands used to ensure rendezvous in the face of clock skew is inversely proportional to the synchronization precision.

For an application such as ours, which involves the detection of the speed and direction of phenomena such as tagged animals, time synchronization is critical — a set of distributed sensors must share a common time coordinate in order to integrate a series of proximity detections into a determination of speed and direction. Important variables that will vary with the application are the required precision of the time synchronization, and (closely related) the required frequency with which the sensors must be able to determine a “fix” on the tracked object. Both of these are informed by the expected nominal speed of tracked objects. Another factor affecting these parameters is the geographic density of sensors relative to both the tracked object’s speed and the effective range of the sensors.

Maintaining synchronized time *energy-efficiently* among a large cluster of nodes that may often be off is a challenge not undertaken by conventional wired networks. The needs of a time coordinate and constraints on a synchronization system in a distributed sensor network vary along numerous axes: precision, scope, lifetime, cost, form factor, and so on. A variety of methods methods for establishing time synchronization are available across these same axes.

The simplest is the distinction between global and local time. A global timebase is one that is shared by every node in the network, or that exists completely independent of the sensor network. By some metrics, these are the best solutions because they eliminate accumulated error that invariably comes from distributing time hop-by-hop within a network. A single time signal broadcast to all nodes can go a long way towards reducing time-sync error. Existing time distribution infrastructures such as the WWVB time signal or GPS-time may be used for this purpose. We are investigating the use of various low-power devices that receive these time signals and can be integrated with our sensor node hardware.

While the idea of using universal time is attractive, it also does have a number of drawbacks. The hardware required to receive special time signals consumes energy and physical space, both of which are quite limited in tags. Adding a new piece of specialized hardware will always be more resource-intensive than a solution that uses existing communication capabilities of a sensor node. Moreover, universal time by definition depends on an existing time distribution infrastructure, which is often unavailable. WWVB works only in the continental US; no infrastructure is available underwater, inside of many structures, or on Mars.

For these reasons, it is also wise to consider a peer-to-peer time distribution strategy. One example is that employed by NTP, the Network Time Protocol [Mills96]. NTP can be used to establish a federation of synchronized nodes, even if the federation has no external time source. An NTP-like algorithm operates over the existing communications infrastructure of nodes; peers repeatedly poll each other for their current time in an effort

facultad de utilizar estas nuevas herramientas, incluyendo entrenamiento y consultoría en aplicaciones de investigación. Esta tecnología promete grandes oportunidades tanto en educación como en investigación.

En las siguientes secciones describimos dos bloques críticos de construcción de sistema necesarios para realizar redes sensoras inalámbricas de larga vida, y luego presentamos los detalles de nuestra plataforma experimental.

#### 4. DOS BLOQUES DE CONSTRUCCIÓN DE SISTEMA PARA OPERACIÓN EN CICLO DE TAREA B AJO

Mientras que muchos aspectos de las redes de sensores desplegadas serán específicos para aplicaciones particulares, hemos comenzado a identificar bloques de construcción claves que permitirán una amplia gama de aplicaciones. Esta sección trata un modelo "Frisbee" para operación en ciclo de tarea (duty cycle) bajo y técnicas para la sincronización del tiempo.

##### 4.1 El Modelo "Frisbee"

Una de las metas más importantes en el diseño de las aplicaciones de red de sensores es minimizar la energía utilizada. La energía es un recurso precioso en las redes sensoras ya que existe una cantidad finita disponible; todos los usos de ella afectan directamente el tiempo de vida útil de la red [Kaiser-Pottie00]. Esto hace de la eficiencia en el uso de la energía algo crítico.

En redes donde eventos "interesantes" suceden en forma infrecuente, un modo efectivo de ahorrar energía es apagar los nodos cuando no se los necesita. En un mundo ideal pero imposible, una red de sensores está completamente dormida durante los largos períodos de inactividad. Cuando algo ocurre, solamente una limitada zona de la red que está cerca del evento se mantiene en un estado totalmente activo. La zona activa debería estar centrada en la ubicación actual del fenómeno objetivo que se está rastreando; y, por supuesto, la zona debería moverse a través de la red junto con el objetivo. Los nodos que no están dentro del radio de percepción del evento están fuera de la zona, y por lo tanto no desperdician energía en la adquisición de datos o en tareas domésticas tales como el mantenimiento de la sincronización del tiempo.

(INSERTAR FIGURA)

Sensores inactivos (dormidos) - Objetivo

- Región del sensor en actividad

- Instantáneas de Frisbees previos
- Trayectoria del objetivo

**Figura 1:** El modelo "Frisbee". El triángulo representa el objetivo que está siendo rastreado por la red. Las regiones sombreadas son los juegos de sensores despertados en el transcurrir del tiempo.

En forma óptima, la zona debería moverse en forma tal que un fenómeno de interés se mantenga siempre dentro de la zona. La zona podría ser circular, con un radio proporcional a la velocidad del objeto. Llamamos a este modelo que consta de una zona circular en movimiento constante, el modelo "Frisbee".

Para implementar una aproximación del Frisbee óptimo, se requieren dos componentes esenciales: (1) Un modo de operación de baja energía con despertador: algunos sensores deben estar siempre vigilantes; otros deben tener un modo de ahorrar energía de forma tal que puedan ser despertados por un estímulo externo; y (2) Una definición del límite del Frisbee: los sensores deben utilizar algoritmos localizados de modo que los nodos puedan decidir en forma autónoma si son parte del Frisbee; i.e., ingresando o dejándolo.

##### 4.1.1 Ahorros de energía con despertador

En el núcleo de nuestra idea está el hecho de que los nodos sensores pueden ser diseñados con un modo "ahorro de energía" en el cual permanecen dormidos. Este modo "dormido" debe requerir menos energía que un modo en el cual los nodos estén manteniendo una vigilancia activa. Este modo se distingue del modo "apagado" (off) en que debe ser posible para un estímulo externo "despertar" los nodos, llevándolos de un modo de baja energía a un modo alerta. Los nodos deberían ser capaces de generar este estímulo de forma que pudieran despertar a sus pares.

La mayoría de los nodos deberían estar dormidos la mayor parte del tiempo. Algunos nodos permanecerán despiertos todo el tiempo para servir de "centinelas" - realizando tareas tales como la aceptación de instrucciones de tareas de parte de los usuarios, y buscando potenciales eventos de interés. Cuando un centinela percibe un evento cercano, despierta a los nodos de su vecindad, permitiendo que el grupo recoja datos con granularidad más fina, o una gama más amplia

to establish their relative bias and skew. Of course, if some NTP nodes have access to an external time source, a hybrid distribution algorithm is possible.

## 5. TIERED ARCHITECTURE

Although Moore's law predicts that hardware for sensor networks will inexorably become smaller, cheaper, and more powerful, technological advances will never prevent the need to make tradeoffs. Even as our notions of metrics such as "fast" and "small" evolve, there will always be compromises: nodes will need to be faster *or* more energy-efficient, smaller *or* more capable, cheaper *or* more durable.

Instead of choosing a single hardware platform that makes a particular set of compromises, we believe an effective design is one that uses a tiered platform consisting of a heterogeneous collection of hardware. Larger, faster, and more expensive hardware ("sensors") can be used more effectively by also using smaller, cheaper, and more limited nodes ("tags"). An analogy can be made to the memory hierarchy commonly found in desktop computer systems. CPUs typically have extremely expensive, fast on-chip cache, backed by slower but larger L2 cache, main memory, and ultimately on-disk swap space. This organization, combined with a tendency in computation for locality of reference, results in a memory system that appears to be as large and as cheap (per-byte) as the swap space, but as fast as the on-chip cache memory. In sensor networks, where localized algorithms are a primary design goal, similar benefits can be realized by creating the network from a spectrum of hardware ranging from small, cheap, and numerous, to large, expensive, and powerful.

The smaller "tag" devices will trade functionality and flexibility for smaller form factor and power. Alone, they would not be adequate to support our sensor network application. However, in conjunction with more endowed nodes, they significantly enhance the network's capabilities. There are many possible advantages to augmenting sensor nodes with small-form-factor tags, including: (1) Density: Tags, by definition, can be significantly lower cost and therefore can be deployed in larger numbers, more densely, than larger, higher capacity sensor nodes, (2) Longevity: Tags can be significantly lower power and therefore can be deployed for longer periods of time, or at higher duty cycles, than larger, higher capacity sensor nodes, particularly if we are able to exploit higher density, and (3) Form factor: Tags are smaller and therefore can be (a) more easily and unobtrusively attached to a wider variety of targets (e.g., for tracking, condition based maintenance, and other logging applications), and (b) deployed with high density.

## 6. EXPERIMENTAL PLATFORMS

Our initial testbed consisted of 5 Toshiba Libretto 50CT laptops running RedHat 6.0 with radiometrix transceivers [RPC]. After our initial experiences with those laptops, we gradually clarified our requirements for the testbed:

- Small in Dimensions.
- Low-Power (long lifetime).
- Moderate Computing Resources
- Flexible I/O capability.

- Well-supported Operating System.

At the same time we realized that one size did not fit all, and that we would need to work in an heterogeneous environment with nodes of different capabilities, as described in the previous section.

In this section we describe the hardware and software developed for our experimental platforms.

### 6.1 Hardware platform

#### 6.1.1 PC104

The PC104 nodes are our "high end" nodes of our tiered sensor architecture. We chose to use off-shelf PC-104-based products to replace the Librettos. PC-104 is a well supported standard [104] and compatible with desktop PCs. They can be designed for low power applications and equipped with processors ranging from 386 to Pentium II and from memory ranging from 0MB RAM to 64MB and more. There is a full spectrum of PC-104 peripheral devices including digital I/O, sensors, actuators, most of which are compatible. Before settling on the PC-104, we also considered the following alternatives:

- Laptops. Those real PCs are expensive, and have unnecessary devices for our testbed such as LCD display, IDE Hard drive, etc., each of which consume significant power.
- Palm [PAL] and other PDAs. Even though they have a compact form factor, they have inadequate computing power for our applications. Another drawback is that they usually use vendor-specific operating systems, applications and devices, and are thus difficult to extend with new hardware and software capabilities. They also have often unnecessary and power-hungry displays.
- TINI [TIN] and ucsimm [UCS]. These small devices are promising platforms, small and cheap with support for open source OS (UCLinux for ucsimm) or standardized virtual machine (Java for TINI). However, they are still in their preliminary stage and do not provide the full functionality of the Linux development environment.

The final factor that led us to PC-104 products is their ability to support almost all PC software. We chose Linux as our operating system because it is one of the most widely supported open source operating systems. We spent time evaluating different distributions or distribution building tools such as LEX, and CCLinux [CCL]. However, we decided to use our homebrew distribution based on a 2.2.12 Kernel and glibc2.1. It also contains some utility programs for management and configuration. It is binary-compatible to RedHat 6.1, thus we can develop and debug our applications on desktops/laptops, and later move to our PC-104 based testbed easily.

We are currently using an Advanced Digital Logic MSM486SN16 (Figure 2). Its features include: (1) AMD ElanSC400, 3.3V, 66MHz CPU, (2) 16MB RAM, 8kBytes (1st Level) Cache, (3) Hard Disk (E-IDE), (4) Floppy Disk Interface (3.5"), (5) COM1(optio SIR IrDA mode), COM2, COM3 and LPT1, LCD-Character Interface, Key Matrix Scanner, (6) RTC with Battery Back-up, Watchdog with Power Failure Detection, Power Management

de modalidades de la que sería posible con un sólo sensor (centinela). A medida que el objetivo se desplaza, los nodos enviarán ulteriores señales despertadoras a otros nodos en la dirección de movimiento del objetivo. Imaginamos un "frente de onda despertador" que precede al fenómeno que está siendo rastreado. A medida que el objetivo se mueve más allá del alcance útil de los nodos sensores, dichos nodos vuelven a dormir.

Una observación interesante es que al incrementar la densidad de los nodos podremos ser capaces de garantizar que una cantidad de suficiente de nodos estará escuchando lo suficientemente pronto como para alcanzar un despertar de estado latente bajo, aún con un despertar de ciclo de tarea relativamente bajo a nivel de nodos individuales. Al momento de la detección podremos despertar a todos los nodos que entren en ciclo, de modo que durante el curso de un ciclo entero, todos los nodos en la vecindad serán despertados gradualmente. En cierto sentido estamos explotando el paralelismo, mediante el uso de grandes cantidades de nodos de ciclo de tarea bajo (baja potencia), en lugar de pequeñas cantidades de nodos de poder más alto.

#### 4.1.2 *Algoritmos localizados para definir el límite del Frisbee*

Un tema fundamental que interesa al diseño de todas las redes sensoras es lograr un diseño descentralizado, totalmente distribuido. Esto se extiende naturalmente a los algoritmos que planeamos desarrollar para definir el límite del Frisbee. Es decir, que cada nodo decidirá automáticamente si está o no "en el Frisbee" - no hay un controlador central que distribuya instrucciones a los nodos.

Un ejemplo de algoritmo de distribución simple para esto es el siguiente: nodos que recientemente han detectado al objetivo despertan a todos los demás nodos dentro de un radio fijado. Si fuera posible, los nodos vecinos que ya están despiertos deberían ser indagados, para que la velocidad del objetivo y la dirección puedan ser determinadas de modo de dar mejor forma y dirigir el "frente de onda despertador". Luego de un cierto período sin detectar actividad, el tiempo de los nodos se agota y vuelven a su estado previo de dormidos.

El diseño de este algoritmo incorporará también la idea de **fidelidad adaptiva**. En el rastreo de movimiento, la distribución cada vez más densa de sensores generalmente conducirá a resultados de rastreo cada vez más precisos. Sin embargo, si una aplicación particular considera que el tiempo de vida de la red es más importante que la precisión del rastreo, es posible ajustar esto con tan sólo despertar, digamos, la mitad o un cuarto de los nodos que físicamente caen dentro del

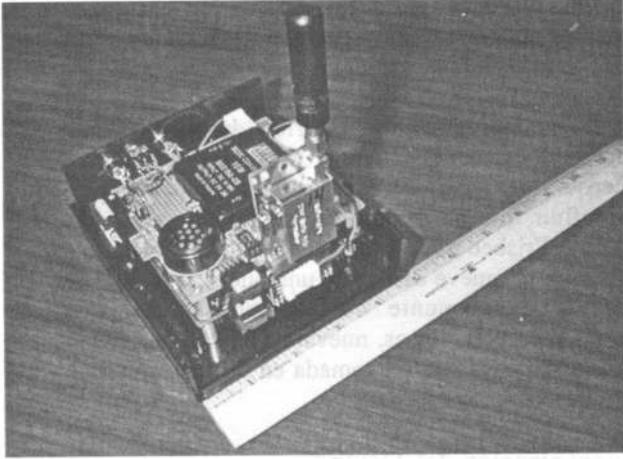
Frisbee. Otra variación potencial es que los sensores más cercanos al centro del Frisbee son despertados con una densidad más alta que los nodos cerca de los bordes.

Dentro de un marco como el arriba descrito, cada aplicación será capaz de sintonizar la precisión deseada y compensar el gasto de energía. Las aplicaciones pueden definir qué porcentaje de los nodos sensores dentro de un Frisbee deberían ser despertados toda vez que el Frisbee se mueve dentro de una nueva región. La decisión de exactamente cuáles nodos deberían realmente ser despertados es, nuevamente, una decisión descentralizada, capaz de ser tomada en forma aleatoria o a través de alguna otra técnica.

## 4.2 Sincronización del Tiempo

La sincronización del tiempo es un bloque de construcción crítico para muchos aspectos de un sistema sensor distribuido. Por ejemplo, la acumulación de datos es crítica para una característica común de las redes sensoras.. Debido al alto costo energético de las comunicaciones comparado con la computación [Kaiser-Pottie00], el procesamiento local, el proceso de resumen, y el agregado de datos son a menudo empleados para minimizar el tamaño y las frecuencias de las transmisiones. La supresión de notificaciones duplicadas del mismo evento desde un grupo de sensores cercanos puede resultar en un significativo ahorro de energía [Intanago00]. Para reconocer a los duplicados, los eventos deben contar con una marca horaria de una precisión similar a la frecuencia del evento. La correcta sincronización del tiempo puede llevar también al ahorro de energía en sistemas que usen TDMA u otros tipos de despertadores programados; el tamaño de las bandas de protección utilizadas para asegurar las reuniones en la cara del esviaje del reloj es inversamente proporcional a la precisión de la sincronización.

Para una aplicación tal como la nuestra, la cual abarca la detección de la velocidad y dirección de fenómenos tales como animales etiquetados, la sincronización del tiempo es crítica - un conjunto de sensores distribuidos debe compartir una coordenada de tiempo común para poder integrar una serie de detecciones de proximidad dentro de una determinación de velocidad y dirección. Las variables importantes que cambiarán con la aplicación son, la precisión requerida en la sincronización del tiempo, y (muy relacionada) la frecuencia requerida con la cual los sensores deben ser capaces de determinar un "arreglo" en el objeto rastreado. Ambas son informadas por la velocidad nominal esperada de los objetos rastreados. Otro factor que afecta a estos parámetros es la densidad geográfica de los sensores relativa tanto a la



**Figure 2: Our prototype PC104-based sensor node.**

(Suspend/Sleep), (7) Form Factor: PC/104 (3.6" x 3.8" x 0.6"), (8) BUS: PC/104 (ISA), (9) MSFLASH16 from ADL. 16MB IDE Flash Disk, and (10) Power: 5VDC only, 600mA/3W.

Most of PC104 devices require a strict 5V power supply (as little as 5.25V will cause damage to the MSM486SN16). We therefore designed a PCB power board for MSM486SN16, including: (1) Power Supply, 9-18V to 5V converter, (2) Infrared transceiver for IrDA or Optical Serial Asynchronous Communication, (3) Speaker Amplifier, and (4) Prototype Area. The size of the board and its standoff configuration are the same as PC104 boards for easy stacking.

### 6.1.2 Radio Subsystem

We are currently using a Radio Packet Controller [RPC] at 418Mhz. We have a small adaptor designed to attach the RPC module to the PC104 system. We wrote a kernel mode driver to use the RPC via the parallel port (see software section). The next generation of this hardware will use an RF Monolithics radio instead and will provide variable power transmission and signal strength measurement.

### 6.1.3 Tags

Tags are untethered devices that have a small enough form factor to be easily attached to objects that are of human scale. We term this form-factor "velcroable," meaning that the device is small enough and light enough to be attached to another object using Velcro brand fasteners.

The scale of tags puts them into an approximate one-to-one correspondence with human-scale objects, such as furniture, shipping packages, laptop computers, and specific regions in a room on a scale of humans, e.g. a 2 square meter patch of the upper left wall. The tags are untethered so that they can recede into the background, performing their tasks with minimal disruption of the environment and minimal infrastructure requirements. This untethered requirement bounds the capability of a tag: the energy available to it is limited by its form factor while its capabilities are bounded by the energy cost of its task.

In our system, tags are implemented with a modular architecture. This is desirable for development purposes, and it leaves

room to improve efficiency through integration. The modules that compose a tag are a loosely coupled system connected by a bus. Each module has local computational capability, implemented by a microcontroller that is interfaced to local resources such as sensors, actuators, communications hardware, or memory. In many respects, each module can operate standalone, only waking up another module when the resources that module controls are needed for its task.

The requirements of the bus are relaxed to accommodate long duty cycles. In our architecture, a master module stays on all the time at very low clock rates and acts as a central point of coordination. Other modules may operate on fixed or variable duty cycles. For example, a sensor module might be operating in a low-power vigilant mode, and when a particular condition is detected, it wakes up the RF board and reports the event. The RF board may then test the channel to see if any other tags are reporting, before reporting an event itself.

We are currently developing several components of the tag platform:

- The master module. This module hosts a number of services that most tag systems will need, including a real time clock, an EEPROM memory, and three UARTs for communication to external devices such as a larger PC-104 based system or an external sensor module. The master module can operate at a very slow clock rate for power savings, or at a high rate when precise coordination is required. It also has a pair of interrupt lines connecting to each of the other modules in the system. During times of high power expenditure, these interrupt lines are used for precise timing and synchronization, while during times of minimal power expenditure, they are used to wake up a module so that it can prepare to receive a message on the serial data bus.
- Radio controller module. This module provides a radio network interface. An RFM baseband radio is incorporated into this module driven by a micro-controller. The micro-controller implements a MAC layer as well as other protocol components. A large (32K) SRAM is also included on the radio module, enabling the storage of state pertaining to higher layer protocols. The simplest applications envisioned, such as a simple beaconing or sensor reporting system, may be implemented directly in the radio controller, eliminating the need for other external components.
- Sensor interface module. This module has a variety of on-board sensors and several actuators, interfaced to a micro-controller. The micro-controller can perform simple analysis of the sensor data as it streams into the system, and can watch for certain characteristic events. It can also record time series, possibly sending them over the bus to the SRAM in the radio module or storing them into the master module EEPROM. The sensor module will also feature a remote low-power wakeup device that can wake up the processor when a certain frequency of sound is detected. This enables most of the devices to be completely off, waiting to be woken up by one of the few left on when an event occurs.

velocidad del objeto rastreado como al alcance efectivo de los sensores.

Mantener el tiempo sincronizado en forma *eficiente con la energía*, en medio de un gran racimo de nodos que a menudo podrá estar inactivo, es un desafío no acometido por las redes de cable convencionales. Las necesidades de una coordenada de tiempo y las limitaciones sobre un sistema de sincronización en una red de sensores distribuidos varían junto con los numerosos ejes: precisión, alcance, tiempo de vida, costo, factor de forma, etc. A través de esos mismos ejes está disponible una variedad de métodos para establecer la sincronización del tiempo.

La más simple es la distinción entre tiempo global y local. Una base de tiempo global es aquella que es compartida por todos los nodos en la red, o que existe completamente independiente de la red sensora. Junto a algunas métricas, estas son las mejores soluciones ya que eliminan el error acumulado que invariablemente surge de distribuir el tiempo salto por salto dentro de una red. Una sola transmisión de señal a todos los nodos puede recorrer un largo camino hacia la reducción del error de sincronización del tiempo. Las infraestructuras de distribución del tiempo existentes, tales como la señal del tiempo WWVB o tiempo GPS podrán ser usadas para este propósito. Estamos investigando el uso de varios dispositivos de baja potencia que reciben estas señales de tiempo y que pueden ser integrados con nuestro hardware de nodos sensores.

Mientras que la idea de usar el tiempo universal es atractiva, tiene también una cantidad de inconvenientes. El hardware que se requiere para recibir señales de tiempo especiales consume energía y espacio físico, siendo que ambos están bastante limitados en las etiquetas. El agregar una nueva pieza de hardware especializado será siempre más intensivo, con relación al recurso, que una solución que utilice las capacidades de comunicación existentes en un nodo sensor. Por otra parte, el tiempo universal por definición depende de una infraestructura de distribución de tiempo existente, la cual por lo general no está disponible. WWVB opera solamente en la parte continental de EE.UU.; no existe una infraestructura disponible debajo del agua, dentro de muchas estructuras, o en Marte.

Por estas razones, es también sabio considerar una estrategia de distribución del tiempo par-a-par. Un ejemplo es aquel empleado por NTP, el Protocolo de Tiempo de Red [Mills96]. El NTP puede ser usado para establecer una federación de nodos sincronizados, aún si la federación no tiene una fuente de tiempo externa. Un algoritmo del tipo NTP opera sobre la infraestructura de nodos de comunicaciones existente; los pares se

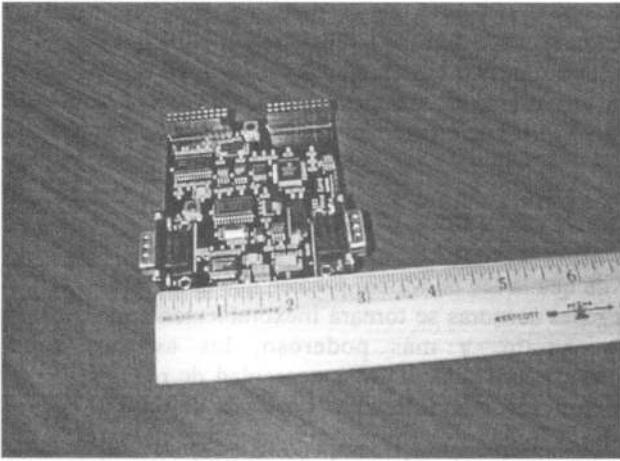
consultan entre sí acerca de su tiempo actual, en un esfuerzo por establecer sus tensiones y oblicuidad relativas. Por supuesto que si algunos nodos NTP tienen acceso a una fuente de tiempo externa, es posible un algoritmo de distribución híbrida.

## 5. ARQUITECTURA EN HILERAS

A pesar de que la ley de Moore predice que el hardware para las redes sensoras se tornará inexorablemente más pequeño, barato, y más poderoso, los avances tecnológicos nunca impedirán la necesidad de realizar intercambios. Así como nuestras nociones de métrica tales como "rápido" y "pequeño" evolucionan, siempre habrá que acomodar algo: los nodos necesitarán ser más rápidos o más eficientes con respecto a la energía, más pequeños o más capaces, más baratos o más durables.

En lugar de elegir una sola plataforma de hardware que realice una determinada serie de acuerdos, creemos que un diseño efectivo es uno que utiliza una plataforma en hilera consistente en una colección heterogénea de hardware. Un hardware (sensores) mayor, más rápido, y más caro, puede ser usado en forma más efectiva si al mismo tiempo se usan nodos ("etiquetas") más pequeños, más baratos, y más limitados. Se puede hacer una analogía con la jerarquía de memoria que se encuentra comúnmente en los sistemas de computadoras de escritorio. Las CPUs por lo general tienen un caché en el chip, extremadamente caro y veloz, respaldado por un caché L2 más lento pero mayor, memoria principal, y últimamente espacio de intercambio (swap) en el disco. Esta organización, combinada con una tendencia en computación por lugar de referencia, resulta en un sistema de memoria que parece ser tan grande y tan barato (por byte) como el espacio de intercambio (swap), pero tan rápido como la memoria caché en el chip. En las redes sensoras, donde los algoritmos localizados son un objetivo de diseño primario, se pueden percibir beneficios similares mediante la creación de la red a partir de un espectro de hardware que vaya desde pequeño, barato, y numeroso, a grande, caro, y poderoso.

Los dispositivos de "etiqueta" más pequeños cambiarán funcionalidad y flexibilidad, por factor de forma más pequeño y poder. Solos, no serían adecuados para soportar nuestra aplicación de red sensora. Sin embargo, en conjunción con nodos más dotados, realzan significativamente las capacidades de la red. Existen muchas posibles ventajas para aumentar los nodos sensores con etiquetas de factor de forma pequeña, incluyendo: (1) Densidad: Por definición, las etiquetas pueden ser de costo significativamente menor y por lo tanto pueden ser desplegadas en grandes cantidades, más



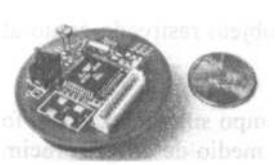
**Figure 3:** Prototype radio controller module for use with our “tag” platform.

- CPU or DSP module. Some applications may require more computational power and more memory when an event occurs. To handle these cases, a computational module can be developed. This module would be invoked when necessary to perform a complex calculation and then resume sleeping.

The tag implementation favors COTS, simplicity of design and flexibility, over optimization of the hardware design. Our tags require a modular architecture. Each module has an inexpensive processor and connects to other modules via a system bus. Modules may be powered down via software controlled switches on each board. None of the modules require complex design. However, modularity would allow for other types of sensors to be added as the outcome of other research projects now or in the future.

We envision that the following modules will be needed in order to get a useful experimental platform:

- Power supply/host interface module. This module provides a real time clock, regulated power to the bus and one or more serial interfaces for debugging/logging output, or for interfacing to a host processor such as a SENSTIT node or PC.
- Radio controller module. This module provides a radio network interface to other modules in the system. An RFM baseband radio is incorporated into this module with a micro-controller to drive it. The simplest applications envisioned may be implemented directly in the radio controller, eliminating the need for other external components.
- Sensor sampling and storage module. The simplest form of this module samples from an A/D converter and stores the resulting time series in a serial memory. After sampling, the data can be loaded over the bus into another module to process it. A more powerful version with a faster CPU will be developed as part of the development of an acoustic ranging sensor based on ultrasound.



**Figure 4:** Pister’s “COTS Mote”, developed at U.C. Berkeley [Pister99]

- Remote wakeup module. This module provides a way for a node to be “awakened” from a very low-power sleep state. For such a module to be useful, the wakeup module must consume far less energy than the node it is awakening. A trivial example of a wakeup module is an “on/off” button. We plan to leverage other existing technology, such as a module that can wake nodes up in response to high-energy RF pulses.

#### 6.1.4 Motes

Motes are the smallest components of our tiered sensor architecture. Where tags are on a scale comparable to human scale, motes are much smaller and much more numerous. Motes are envisioned to be small enough to float in the air [Pister99] or to be attached in large numbers to a surface [Abelson99]. The cost of motes is intended to be small enough that they are entirely disposable. The technology to produce motes is thought to come from the area of MEMS.

Motes communicate with macro-scale objects through low power RF and through visible light. Through extensive integration, low power RF can be included on the same silicon die as the mote itself, and the antenna may serve as a “tail” that enables it to float in an air current. Another possible communication technique uses steerable corner-cube reflectors to reflect laser light shined on a collection of motes. A third communication technique uses a steerable mirror to point a laser beam. All of these options rely on MEMS technology to provide this functionality in such a small package.

Although motes the size of dust motes are still a long way off, current research is progressing using very small and simple devices that integrate a microcontroller with sensors and a radio on a small board.

We are using the mote developed by Kris Pister at UC Berkeley [Pister99], shown in Figure 4.

## 6.2 Software

### 6.2.1 Radiometrix Device Drivers

This package [RDD] contains a Linux device driver for the RPC (Radio Packet Controller) model of radio manufactured by Radiometrix. The RPC is a fairly low power, self-contained, short-range, plug-on radio. It has been a critical part of our testbed infrastructure for implementation and validation of directed diffusion and other algorithms.

### 6.2.2 Emlog

Emlog [EML] is a Linux kernel module that makes it easy to access the most recent (and only the most recent) output from a process. It works just like “tail -f” on a log file, except that the storage required never grows. This is very important for

densamente que los nodos sensores más grandes y de mayor capacidad, (2) Longevidad: Las etiquetas pueden ser de una baja energía significativa y por lo tanto pueden ser desplegadas durante períodos de tiempo más largos, o en ciclos de tarea más altos, que los nodos sensores más grandes y de mayor capacidad, particularmente si somos capaces de explotar la densidad más alta, y (3) Factor de forma: Las etiquetas son más pequeñas y por lo tanto pueden ser (a) adjuntadas más fácilmente y sin obstrucciones a una mayor variedad de objetivos (ej. para el rastreo, mantenimiento basado en la condición, y otras aplicaciones de transporte [logging]), y (b) desplegadas con alta densidad.

## 6. PLATAFORMAS EXPERIMENTALES

Nuestro banco de prueba inicial consistió de 5 computadoras laptop Toshiba Libretto 50CT, corriendo el RedHat 6.0 con transmisores/receptores Radiometrix [RPC]. Luego de nuestra experiencia inicial con esos laptops, gradualmente clarificamos nuestros requerimientos para el banco de prueba:

- Dimensiones pequeñas
- Baja potencia (largo tiempo de vida)
- Recursos de computación moderados
- Capacidad de I/O flexible
- Buen soporte para el Sistema operativo.

Al mismo tiempo nos dimos cuenta que un solo tamaño no se ajustaba para todos, y que necesitaríamos trabajar en un ambiente heterogéneo con nodos de diferentes capacidades, tal como se describió en la sección previa.

En esta sección describimos el hardware y software desarrollado para nuestras plataformas experimentales.

### 6.1 Plataforma de Hardware

#### 6.1.1 PC104

Los nodos PC104 son los nodos más sofisticados en nuestra arquitectura de sensores en hilera. Elegimos usar productos fácilmente disponibles basados en el PC-104, para reemplazar los computadoras Libretto. El PC-104 es un estándar bien soportado [104] y compatible con los PCs de escritorio. Pueden ser diseñados para aplicaciones de baja potencia y equipados con procesadores que van desde 386 a Pentium II, y con memoria que va desde 0MB RAM a 64MB y más. Existe un completo espectro de dispositivos periféricos del PC 104, incluyendo I/O digital, sensores, actuadores, la mayoría de los cuales son compatibles. Antes de ajustar el PC-104, consideraremos también las siguientes alternativas:

- Computadoras Laptop. Estos PCs reales son caros, y tienen dispositivos innecesarios para nuestro banco de prueba tales como la pantalla de LCD, el disco duro IDE, etc., cada uno de los cuales consume una cantidad de energía significativa.
- Palm [PAL] y otros PDAs. A pesar de que cuentan con un factor de forma compacto, tienen un poder de computación inadecuado para nuestras aplicaciones. Otro inconveniente es que por lo general utilizan sistemas operativos, aplicaciones y dispositivos específicos del vendedor, y son así difíciles de ampliar con nuevas capacidades de hardware y software. Por lo general también tienen pantallas innecesarias y muy consumidoras de energía.
- TINI [TIN] y ucsimm [UCS]. Estos pequeños dispositivos son plataformas promisorias, pequeñas y baratas con soporte para fuente abierta -OS- (UCLinux para ucsimm) o máquina virtual normalizada (Java para TINI). Sin embargo, todavía están en su etapa preliminar y no brindan la funcionalidad total del ambiente de desarrollo Linux.

El factor final que nos condujo a los productos PC-104 es su capacidad para soportar casi todo el software de PC. Escogimos al Linux como nuestro sistema operativo debido a que es uno de los sistemas operativos de fuente abierta más ampliamente soportados. Pasamos tiempo evaluando diferentes distribuciones o herramientas de construcción de distribución tales como LEX y CCLinux [CCL]. Sin embargo, decidimos usar nuestra distribución casera basada en un 2.2.12 Kernel y glibc2.1. Contiene también algunos programas utilitarios para administración y configuración. Es compatible con el sistema binario para el RedHat 6.1, así podemos desarrollar, y detectar y corregir (debug) nuestras aplicaciones en desktops/laptops, y más tarde movernos fácilmente a nuestro banco de prueba basado en el PC-104.

Actualmente estamos usando un Advanced Digital Logic MSM486SN16 (Figura 2). Sus características incluyen: (1) MAD ElanSC400, 3.3V, CPU 66MHz, (2) 16 MB RAM, 8kBytes (1er nivel) Caché, (3) Disco Duro (E-IDE), (4) Interfase de discos flexibles (3.5"), (5) COM1 (modo opcional SIR IrDA), COM2, COM3 y LPT1, Interfase de caracteres LCD, Scanner Key Matrix, (6) RTC con respaldo de batería. Vigía con detección de corte de energía, manejo de energía (Suspender/Dormir), (7) Factor de Forma: PC/104 (3.6" x 3.8" x 0.6"), (8) BUS: PC/104 (ISA), (9)

our logging and debugging facilities in embedded systems where there isn't enough memory or disk space for keeping complete log files, but the most recent debugging messages are sometimes needed (e.g., after an error is observed).

### 6.2.3 Parapin

Parapin [PAR] makes it easy to write C code under Linux that controls individual pins on a PC parallel port. This kind of control is very useful for electronics projects that use the PC's parallel port as a generic digital I/O interface. Parapin goes to great lengths to insulate the programmer from the somewhat complex parallel port programming interface provided by the PC hardware, making it easy to use the parallel port for digital I/O. By the same token, this abstraction also makes Parapin less useful in applications that need to actually use the parallel port as a parallel port (e.g., for talking to a printer).

## 7. CONCLUSIONS

We have described some of the system building blocks we are developing for distributed sensor networks, presented details of our experimental testbeds for wireless sensor network development, and described the habitat monitoring application for which they are being developed. Many of our system building blocks are being developed in parallel with the goal of eventual integration into a deployable system.

## 8. ACKNOWLEDGEMENTS

This work was supported by DARPA under grant No. DABT63-99-1-0011 as part of the SCADDS project, NSF grant ANI-9979457 as part of the SCOWR project, and was also made possible in part due to support from Cisco Systems.

## References

- [Abelson99] H. Abelson, D. Allen, D. Coore, C. Hanson, G. Homsy, T. Knight, R. Nagpal, E. Rauch, G. Sussman and R. Weiss, "Amorphous Computing", MIT AI Memo 1665, August 1999.
- [Colwell98] Colwell, Rita, 1998. Testimony of Dr. Rita Colwell, Director, National Science Foundation, Before the Basic Research Subcommittee, House Science Committee, Hearing on Remote Sensing as a Research and Management Tool. September 28, 1998.  
<http://www.nsf.gov/od/lpa/congress/rc80928.htm>
- [Estrin-etal.99] Deborah Estrin, Ramesh Govindan, John Heidemann and Satish Kumar "Next Century Challenges: Scalable Coordination in Sensor Networks", ACM MobiCom 99, August 99, Seattle, WA.
- [Gell-Mann95] Gell-Mann, Murray. 1995. What Is Complexity? Complexity, Vol. 1, no.1 John Wiley and Sons, Inc.
- [Hamilton00] Hamilton, Michael P. 2000. Hummercams, Robots, and the Virtual Reserve. Directors Notebook, James San Jacinto Mountains Reserve web site. February 6, 2000.  
<http://www.jamesreserve.edu/news.html>
- [Hamilton-Flaxman92] Hamilton, M.P. and M. Flaxman. 1992. Scientific data visualization and biological diversity: new tools for spatializing multimedia observations of species and ecosystems. *Landscape and Urban Planning*. 21:285-297.
- [Intanago-et.al.00] Intanagonwiwat, C., Govindan, R., Estrin, D. "Directed Diffusion: A Scalable and Robust Communication Paradigm for Sensor Networks", ACM MobiCom 2000, August 00, Boston, MA.
- [Keitt-etal97] Keitt, T.H., D.L. Urban, and B.T. Milne. 1997. Detecting critical scales in fragmented landscapes. *Conservation Ecology* [online]1(1): 4. Available from the Internet. <http://www.consecol.org/vol1/iss1/art4>
- [Lohr-etal95] Lohr, S. A., P. G. Connors, J. A. Stanford and J. S. Clegg. 1995. A New Horizon for Biological Field Stations and Marine Laboratories. *Miscellaneous Publication* 3, Rocky Mountain Biological Laboratory, Crested Butte, CO, 36 pp.
- [Mataric95] Maja J. Mataric, "Issues and Approaches in the Design of Collective Autonomous Agents", *Robotics and Autonomous Systems*, 16(2-4), Dec 1995, pp. 321-331
- [Michener-etal98] Michener, W. K., J. H. Porter and S. G. Stafford. 1998. Data and Information Management in the Ecological Sciences: A Resource Guide. LTER Network Office, University of New Mexico, Albuquerque, NM. 138 pp. ("DIMES report")  
<http://www.lternet.edu/ecoinformatics/guide/frame.htm>
- [Mills94] David L. Mills. Internet Time Synchronization: The Network Time Protocol. In Zhonghua Yang and T. Anthony Marsland, editors, *Global States and Time in Distributed Systems*. IEEE Computer Society Press, 1994.
- [Pister99] J. M. Kahn, R. H. Katz and K. S. J. Pister, "Mobile Networking for Smart Dust", ACM/IEEE Intl. Conf. on Mobile Computing and Networking (MobiCom 99), Seattle, WA, August 17-19, 1999.
- [Pottie-Kaiser00] G. Pottie and W. Kaiser, "Wireless Sensor Networks", *Communications of the ACM*, Vol. 43, No. 5, May 2000, pp. 51-58
- [Sukhatme99] Gaurav S. Sukhatme, James F. Montgomery, and Maja J. Mataric, "Design and Implementation of a Mechanically Heterogeneous Robot Group", *Proceedings of Mobile Robots XIV - SPIE 99*, Boston, MA
- [Sukhatme00] Gaurav S. Sukhatme and Maja J. Mataric, "Embedding Robots into the Internet", *Communications of the ACM*, May 2000, 43(5), pp. 67-73
- [Walker-Steffen97] Walker, B., and W. Steffen. 1997. An overview of the implications of global change for natural and managed terrestrial ecosystems. *Conservation Ecology* [online]1(2)  
<http://www.consecol.org/vol1/iss2/art2>

## Web Page References

- [RDD] Radiometrix device drivers,  
<http://www.circlemud.org/jelson/software/radiometrix>
- [EML] Emlog Software Package,  
<http://www.circlemud.org/jelson/software/emlog>
- [PAR] Parapin PC Pin Control Library  
<http://www.circlemud.org/jelson/software/parapin>
- [RPC] Radio Packet Controller, <http://www.radiometrix.com>
- [104] PC104 Consortium, <http://www.pc104.org>
- [PAL] 3COM Palm Pilot Devices, <http://www.palm.com>
- [TIN] TINI: Tiny InterNet Interface, Ibutton,  
<http://www.ibutton.com/TINI/index.html>
- [UCS] Embedded Linux/Microcontroller Project,  
<http://www.uclinux.com/>
- [CCL] CCLinux: The Minimal Linux Distribution,  
<http://www.cosmicchaos.com/CCLinux/index.shtml>

MSFLASH16 desde ADL, 16MB disco Flash IDE, y (10) Energía: 5VDC solamente, 600mA/3W.

(INSERTAR FIGURA)

**Figura 2: Nuestro prototipo de nodo sensor basado en el PC104.**

La mayoría de los dispositivos PC104 requieren un estricto suministro de energía de 5V (con tan sólo 5.25 V se causaría daño al MSM486SN16). Por lo tanto diseñamos un tablero de energía para el MSM486SN16, incluyendo: (1) Suministro de energía de 9-18V a convertidor de 5V, (2) Transmisor/Receptor infrarrojo para IrDA o Comunicación Asincrónica Serial Óptica, (3) Amplificador de parlante, y (4) Área prototipo. El tamaño del tablero y su configuración balanceada son los mismos que los de los tableros PC104 para un fácil apilamiento.

#### *6.1.2 Subsistema de Radio*

Actualmente estamos usando un Controlador de Paquete de Radio [RPC] a 418 Mhz. Tenemos un pequeño adaptador diseñado para adjuntar el módulo RPC al sistema PC104. Escribimos un driver de modo de núcleo (*kernel*) para usar el RPC a través del puerto paralelo (ver la sección de software). En su lugar, la siguiente generación de este hardware utilizará un radio RF Monolítico y brindará transmisión de energía y medición de potencia de la señal variable.

#### *6.1.3 Etiquetas (Tags)*

Las etiquetas son dispositivos desencadenados que tienen un factor de forma lo suficientemente pequeño como para ser fácilmente adjuntado a objetos que son de escala humana. Le dimos a este factor de forma la denominación de "ajustable mediante Velcro", significando que el dispositivo es lo suficientemente pequeño y ligero como para ser adjuntado a otro objeto, usando sujetadores de la marca Velcro.

La escala de las etiquetas las pone dentro de una correspondencia aproximada al uno-a-uno con los objetos de escala humana, tales como muebles, paquetes de envíos, computadoras laptop, y regiones específicas dentro de una habitación con escala humana, por ej. una porción de dos metros cuadrados en la pared superior izquierda. Las etiquetas están sueltas de modo que puedan retroceder al fondo, desempeñando sus tareas con una molestia mínima del ambiente y requerimientos de infraestructura mínimos. Este requerimiento de estar sueltas limita la capacidad de una etiqueta: la energía disponible para ella está limitada por su factor de forma

mientras que sus capacidades están limitadas por el costo de energía de su tarea.

En nuestro sistema, las etiquetas están implementadas con una arquitectura modular.

Esto es deseable para propósitos de desarrollo, y deja lugar para mejorar la eficiencia a través de la integración. Los módulos que componen una etiqueta son un sistema acoplado de forma libre conectado por un bus. Cada módulo tiene capacidad computacional local, implementada por un micro-controlador que está interconectado a los recursos locales tales como sensores, actuadores, hardware de comunicación, o memoria. En muchos aspectos, cada módulo puede operar en forma autónoma (*standalone*), despertando a otro módulo solamente cuando los recursos que el módulo controla se necesitan para su tarea.

Los requerimientos del bus son flexibles, para ajustar ciclos de tarea largos. En nuestra arquitectura, un módulo maestro permanece todo el tiempo encendido a frecuencias de reloj sumamente bajas, y actúa como punto central de coordinación. Otros módulos podrán operar en ciclos de tarea fijos o variables. Por ejemplo, un módulo sensor podría estar operando en un modo vigilante de baja potencia, y cuando se detecta una condición particular, despierta al tablero RF y reporta el evento. El tablero RF podrá entonces probar el canal para ver si otras etiquetas están reportando, antes de reportar un evento por sí mismo.

Actualmente estamos desarrollando varios componentes de la plataforma de etiqueta:

- El módulo maestro. Este módulo alberga una cantidad de servicios que necesitarán la mayoría de los sistemas de etiqueta, incluyendo reloj en tiempo real, una memoria EEPROM, y tres UARTs para comunicación con dispositivos externos tales como un sistema más grande basado en el PC-104 o un módulo sensor externo. El módulo maestro puede operar a una muy baja frecuencia de reloj para ahorrar energía, a una variación alta cuando se requiere una coordinación precisa. Tiene también un par de líneas interruptoras que conectan a cada uno de los otros módulos en el sistema. Durante los tiempos de alto gasto de energía, estas líneas interruptoras son usadas para una regulación y sincronización precisa, mientras que durante los tiempos de gasto mínimo de energía, son usadas para despertar a un módulo para que pueda prepararse para recibir un mensaje en el bus de datos seriales.

- El módulo controlador de radio. Este módulo brinda una interfase de red de radio. Una radio de banda base RFM está incorporada dentro de este módulo controlado por un micro-controlador. El micro-controlador implementa una capa MAC así como otros componentes del protocolo. Una SRAM de gran capacidad (32K) está incluida también en el módulo de radio, permitiendo el almacenamiento de estado perteneciente a protocolos de capas más altas. Las más simples de las aplicaciones previstas, tales como una simple guía o sistema de reporte de sensor, podrán ser implementadas directamente en el controlador de radio, eliminando la necesidad de otros componentes externos.
- Módulo de interfase sensor. Este módulo tiene una variedad de sensores a bordo y varios actuadores, interconectados a un micro-controlador. El micro-controlador puede desempeñar análisis simples de los datos del sensor a medida que fluye dentro del sistema, y puede vigilar ciertos eventos característicos. Puede también registrar series de tiempo, posiblemente enviándolas sobre el bus a la SRAM en el módulo de radio o almacenándolas dentro del módulo maestro EEPROM. El módulo sensor resaltará también un dispositivo despertador remoto de bajo poder que puede despertar al procesador cuando se detecta una cierta frecuencia de sonido. Esto permite que la mayoría de los dispositivos estén completamente apagados, esperando ser despertados por uno de los pocos que quedan encendidos, cuando ocurre un evento.

(INSERTAR FIGURA)

**Figura 3:** Prototipo del módulo controlador de radio para usar con nuestra plataforma "etiqueta".

- Módulo CPU o DSP. Algunas aplicaciones podrán requerir más poder computacional y más memoria cuando ocurre un evento. Para manejar estos casos, puede desarrollarse un módulo computacional. Este módulo podría ser invocado cuando sea necesario realizar un cálculo complejo y luego volver a dormir.

La implementación de la etiqueta favorece el COTS, la simplicidad de diseño y la flexibilidad, sobre la optimización del diseño del hardware. Nuestras etiquetas requieren una arquitectura modular. Cada

módulo tiene un procesador barato y conecta con otros módulos a través de un bus de sistema. Los módulos podrán ser potenciados a través de interruptores controlados por software en cada tablero. Ninguno de los módulos requiere un diseño complejo. Sin embargo, la modularidad permitiría que otros tipos de sensores sean agregados como resultado de otros proyectos de investigación ahora o en el futuro.

Calculamos que para obtener una plataforma experimental útil serán necesarios los siguientes módulos:

- Módulo de suministro de energía/interfase huésped. Este módulo brinda un reloj de tiempo real, energía regulada para el bus y una o más interfaces seriales para el output de *debugging/logging* (detección y remoción/registro), o para interconectar con un procesador huésped tal como un nodo SENSIT o PC.
- Módulo de controlador de radio. Este módulo brinda una interfase de red de radio para otros módulos en el sistema. Una radio de banda de base RFM es incorporada dentro de este módulo con un micro-controlador para manejarlo. Las aplicaciones más simples previstas podrán ser implementadas directamente en el controlador de radio, eliminando la necesidad de otros componentes externos.
- Módulo de muestreo y almacenamiento de sensor. La forma más simple de este módulo toma una muestra a partir de un convertidor A/D y almacena la serie de tiempo resultante en una memoria serial. Luego del muestreo, los datos pueden ser cargados sobre el bus dentro de otro módulo *para procesarlos*. Una versión más poderosa con una CPU más veloz será desarrollada como parte del desarrollo de un sensor de rango acústico basado en ultrasonido.

(INSERTAR FIGURA)

**Figura 4:** "Mota COTS" de Pister, desarrollada en la U.C. Berkeley [Pister99]

- Módulo despertador remoto. Este módulo brinda una manera para que un nodo sea "despertado" de un estado "dormido" de muy baja potencia. Para que tal módulo sea útil, el módulo despertador debe consumir mucha menos energía que el nodo que está

despertando. Un ejemplo trivial de módulo despertador es un botón "on/off". Planeamos brindar soporte a otras tecnologías existentes, tal como un módulo que pueda despertar nodos en respuesta a pulsos RF de alta energía.

#### 6.1.4 Motas

Las motas son los componentes más pequeños de nuestra arquitectura de sensor en hileras. Allí donde las etiquetas están en una escala comparable a la escala humana, las motas son mucho más pequeñas y numerosas. Se prevé que las motas sean lo suficientemente pequeñas como para flotar en el aire [Pister99] o para ser adjuntadas a una superficie en grandes cantidades [Abelson99]. Se pretende que el costo de las motas sea lo suficientemente pequeño como para que estén completamente disponibles. Se piensa que la tecnología para producir motas proviene del área de MEMS.

Las motas se comunican con los objetos de gran escala a través de RF de baja potencia y a través de la luz visible. A través de la integración extensiva la RF de baja potencia puede ser incluida en el mismo dado de silicio de la propia mota, y la antena podrá servir como una "cola" que le permita flotar en una corriente de aire. Otra técnica de comunicación posible utiliza reflectores gobernables de esquina cónica para reflejar la luz laser que se refleja sobre un grupo de motas. Una tercera técnica de comunicación utiliza un espejo gobernable para apuntar un rayo laser. Todas estas opciones se apoyan en la tecnología MEMS para brindar esta funcionalidad en tan pequeño paquete.

A pesar de que las motas del tamaño de las motas de polvo todavía están distantes, la investigación actual está avanzando en el uso de dispositivos muy pequeños y simples que integran un micro-controlador con sensores y una radio sobre un pequeño tablero.

Estamos usando la mota desarrollada por Kris Pister en la UC Berkeley [Pister99], mostrada en la figura 4.

## 6.2 Software

### 6.2.1 Controladores del dispositivo Radiometrix

Este paquete [RDD] contiene un controlador de dispositivo Linux para el modelo de radio RPC (Controlador del Paquete de Radio) fabricado por Radiometrix. El RPC es una radio "plug-on", de rango corto, autocontenido, y de relativamente baja potencia. Ha sido una parte crítica de la infraestructura de nuestro banco de prueba para la implementación y validación de la difusión dirigida y otros algoritmos.

### 6.2.2 Emlog

El Emlog [EML] es un módulo de núcleo Linux que hace fácil el acceso a la más reciente (y sólo a la más reciente) emisión (output) de un proceso. Funciona tal como "tail-f" en un archivo log, excepto que el almacenamiento requerido no crece jamás. Esto es muy importante para nuestras instalaciones de registro, y corrección de errores (*logging* y *debugging*) en sistemas embebidos donde no hay suficiente memoria o espacio en disco para mantener archivos de registro completos, pero donde algunas veces se necesitan los más recientes mensajes de *debugging* [por ej., luego de que se observa un error].

### 6.2.3 Parapin

Parapin [PAR] facilita la escritura de código en C bajo el Linux que controla los pins individuales en un puerto paralelo de PC. Este tipo de control es muy útil para los proyectos de electrónica que usen el puerto paralelo del PC como una interfase digital genérica I/O. Parapin hace grandes esfuerzos para aislar al programador de la -en cierta forma- compleja interfase de programación de puerto paralelo suministrada por el hardware del PC, facilitando el uso del puerto paralelo para I/O digital. Por la misma razón, esta abstracción hace también que el Parapin sea menos útil en aplicaciones que necesitan realmente usar al puerto paralelo como un puerto paralelo (por ej. para hablarle a la impresora).

## 7. CONCLUSIONES

Hemos descrito algunos de los bloques de construcción de sistema que estamos desarrollando para las redes sensoras distribuidas, hemos presentado detalles de nuestros lechos de prueba experimentales para el desarrollo de redes sensoras, y descrito la aplicación de monitoreo del hábitat para la cual están siendo desarrolladas. Muchos de nuestros bloques de construcción de sistema están siendo desarrollados en paralelo con la meta de una eventual integración dentro de un sistema desplegable.

## 8. AGRADECIMIENTOS

Este trabajo fue apoyado por DARPA bajo el subsidio N° DABT63-99-1-0011 como parte del proyecto SCADDS, subsidio de NSF ANI-9979457 como parte del proyecto SCOWR, y fue también hecho posible en parte debido al apoyo de Cisco Systems.

## Referencias

[Abelson99] H.Abelson, D.Allen, D.Coore, C.Hanson, G.Homsy, T.Knight, R.Nagpal, E.Rauch, G.Sussman y R.Weiss, "Amorphous Computing", MIT AI Memo 1665, Agosto 1999.

- [Colwell98] Colwell, Rita, 1998. Testimonio de la Dra. Rita Colwell, Directora de la National Science Foundation, ante el Sub-comité de Investigación Básica, Comité de Ciencia Doméstica, Audiencia sobre Sensores remotos como herramienta de investigación y manejo. 28 de setiembre de 1998.  
<http://www.nsf.gov/od/lpa/congress/rc80928.htm>
- [Estrin.et.al.99] Deborah Estrin, Ramesh Govindan, John Heidemann y Satish Kumar "Next Century Challenges: Scalable Coordination in Sensor Networks", ACM MobiCom 99, Agosto de 1999, Seattle, WA.
- [Gell-Mann95] Gell-Mann, Murray, 1995. What is Complexity?, Complexity, Vol.1, no.1 John Wiley & Sons, Inc.
- [Hamilton00] Hamilton, Michael P. 2000. Hummerncams, Robots, and the Virtual Reserve. Directors Notebook, sitio web del James San Jacinto Mountains Reserve. 6 de febrero de 2000.  
<http://www.jamesreserve.edu/news.html>
- [Hamilton-Flaxman92] Hamilton, M.P. y M. Flaxman. 1992. Scientific data visualization and biological diversity: new tools for spatializing multimedia observations of species and ecosystems. Landscape and urban planning. 21:285-297.
- [Intanago-et.al.00] Intanagonwiwat, C., Govindan, R., Estrin, D. "Directed Diffusion: A Scalable and Robust Communication Paradigm for Sensor Networks", ACM MobiCom 2000, Agosto 00, Boston, MA.
- [Keitt-etal97] Keitt, T.H., D.L. Urban, y B.T. Milne 1997. Detecting critical scales in fragmented landscapes. Conservation Ecology [online] 1(1): 4. Disponible en Internet.  
<http://www.consecol.org/vol1/iss1/art4>
- [Lohr-etal95] Lohr, S.A., P.G. Connors, J.A. Stanford and J. S. Clegg. 1995. A New Horizon for Biological Field Stations and Marine Laboratories, Miscellaneous Publication 3, Rocky Mountain Biological Laboratory, Crested Butte, CO, 36 pp.
- [Mataric95] Maja J. Mataric, "Issues and Approaches in the Design of Collective Autonomous Agents", Robotics and Autonomous Systems, 16(2-4), Dic. 1995, pp. 321-331
- [Michener-et.al98] Michener, W.K., J.H. Porter y S.G. Staford, 1998. Data and Information Management in the Ecological Sciences: A Resource Guide. LTER Network Office, University of New Mexico, Albuquerque, NM. 138 pp. ("DIMES report").  
<http://www.lternet.edu/ecoinformatics/guide/frame.htm>
- [Mills94] David L. Mills. Internet Time Synchronization: The Network Time Protocol. En Zhonghua Yang y T. Anthony Marsland, editores, *Global States and Time in Distributed Systems*. IEEE Computer Society Press, 1994.
- [Pister99] J.M. Kahn, R.H. Katz y K.S.J. Pister, "Mobile Networking for Smart Dust", ACM/IEEE Intl.Conf. on Mobile Computing and Networking (MobiCom 99), Seattle, WA, Agosto 17-19, 1999.
- [Pottie.Kaiser00] G.Pottie y W.Kaise, "Wireless Sensor Networks", Communications of the ACM, Vol. 43, No 5, Mayo de 2000, pp. 51-58
- [Sukhatme99] Gaurav S. Sukhatme, James F. Montgomery, y Maja J. Mataric, "Design and Implementation of a Mechanically Heterogeneous Robot Group", Proceedings of Mobile Robots XIV - SPIE99, Boston, MA.
- [Sukhatme00] Gaurav S. Sukhatme y Maja J. Mataric, "Embedding Robots into the Internet", Comunicaciones del ACM, Mayo de 2000, 43(5), pp.67-73.
- [Walker-Steffen97] Walker, B., y W. Steffen. 1997. An overview of the implications of global change for natural and managed terrestrial ecosystems. Conservation Ecology [online] 1(2) <http://www.consecol.org/vol1/iss2/art2>

## Referencias de páginas web

- [RDD] Controladores de dispositivo Radiometrix  
<http://www.circlemud.org/jelson/software/radiometrix>
- [EML] Paquete de Software Emlog,  
<http://www.circlemud.org/jelson/software/emlog>
- [PAR] Biblioteca de control de pins Parapin PC  
<http://www.circlemud.org/jelson/software/parapin>
- [RPC] Controlador del Paquete de Radio,  
<http://www.radiometrix.com>
- [104] Consorcio PC104 , <http://www.pc104.org>
- [PAL] 3COM Dispositivos Palm Pilot, <http://www.palm.com>
- [TIN] TINI: Ibutton, Interfase de Internet Tiny,  
<http://www.ibutton.com/TINI/index.html>
- [UCS] Linux embebido / Proyecto micro-controlador,  
<http://www.uclinux.com>
- [CCL] CCLinux: La distribución Linux mínima,  
<http://www.cosmicchaos.com/CCLinux/index.shtml>