The Personal Sensor Network: a User-Centric Monitoring Solution

G. Giorgetti[*], G. Manes
Dip. Elettronica e Telecom., Università degli Studi di Firenze, ITALY

gianni.giorgetti@unifi.it
gianfranco.manes@unifi.it

J.H. Lewis, S.T. Mastroianni,
Embedded Systems Research,
Motorola Labs, Tempe, US

jonathan.lewis@motorola.com
s.mastroianni@motorola.com

S.K.S. Gupta
IMPACT Lab, School Comput. & Info.,
Arizona State University, Tempe, US

sandeep.gupta@asu.edu

ABSTRACT
Server-based architectures used in traditional Wireless Sensor Network (WSN) applications are not suitable when the sensors are installed near the user and local access is desirable. We address this problem by proposing the Personal Sensor Network (PSN), a *computer-less* architecture that enables users to access the sensor data using their cell phones or any other Bluetooth enabled devices. The use of COTS hardware and widely available software resources results in a solution easy to implement and simple to interface with other WSNs.

Categories and Subject Descriptors
C.2.3 [**Computer Communication Networks**]: Network Operations - *network management, network monitoring*.

General Terms
Design, Experimentation.

Keywords
Wireless Sensor Network, IEEE 802.15.4, Bluetooth, Cell phone.

1. INTRODUCTION
In traditional WSN applications, the access to the network is mediated by a central server that collects the information before dispatching the results to the final user. While this model has been extensively developed and adopted in many remote-monitoring deployments, imposing a server-based solution seems unnatural when the sensors are installed near the user (e.g. ambient intelligence, smart-home) or even on his person (body area networks). In all these cases, allowing a direct interaction between the user and the nearby nodes would result in a more intuitive access to the sensor data, reduce the transmission delays, and in some cases, it would even eliminate the need to maintain a server and a remote link.
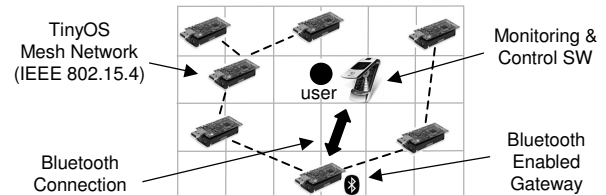
Figure 1. The Personal Sensor Network architecture.

We have explored the idea of a *computer-less* architecture by implementing a *Personal Sensor Network* (PSN) consisting of ready-to-use sensor nodes and a small Bluetooth gateway. The PSN is based on a *user-centric* model where the presence of a server is unnecessary because the user can access the network using the same cell phone (or any handheld device) that he usually carries around in his everyday activities. In addition, since all network devices are compact and lightweight, the PSN can be easily transported and used whenever a monitoring system is needed. For example, the safety of first responders to an incident scene could be increased by a PSN near, or on their body, that provides immediate access to the local sensor information.

In the next section we discuss the characteristics of the system implemented. The application should be considered as a *proof-of-concept* solution aiming at demonstrating the synergy possible by combining established and rapidly maturing technologies (IEEE 802.15.4, Bluetooth and cell phones). While other phone-based approaches have been proposed in conjunction with custom solutions for healthcare applications [1][2], our approach entirely relies on COTS components and widely available software resources. The result is a solution that is easy to implement and simple to interface with existing sensor networks.

2. SYSTEM OVERVIEW
The PSN architecture (see Figure 1) is based on a TinyOS network consisting of IEEE 802.15.4 nodes and a gateway (GW) with support for Bluetooth (BT) connectivity. Using a cell phone, or any other BT enabled device, the user can connect to the GW and obtain real-time information from the sensors, send control commands or communicate specific information (e.g. a preference profile). Since the user can access the network while freely moving among the sensors, the phone not only replaces the PC, but also supports a more natural and intuitive interaction with the WSN. The BT interface on the gateway was designed to achieve a seamless integration between the phone and the sensor nodes. In pursuing this goal we were able to maintain the network structure unchanged to ensure compatibility with existing applications and
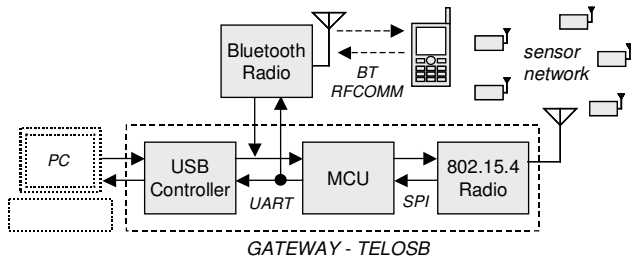
Figure 2. The gateway architecture.

Figure 3. The Bluetooth gateway (left) and a detail of the phone interface (right).

allows the use of a PC when permanent data logging or complex computation is required.

3. SENSOR NODES

The PSN is based on TelosB boards, commercially available sensor nodes designed for low power operations and compatible with the IEEE 802.15.4/ZigBee standard. The nodes run a TinyOS based software that monitors a set of parameters relevant to the user: for example, in our application the PSN nodes are configured to report temperature, humidity, battery voltage and radio strength; some units were also interfaced to external CO_2 sensors. The advantage of using TelosB nodes is that we can leverage TinyOS and its support for ad hoc multi-hop protocols (e.g. MultiHopLQI) to create a self-configuring network. Since the nodes are able to automatically select the most reliable route to the GW and the network topology dynamically adapts to changes in the system (e.g. nodes added or removed, radio link failures), deploying a PSN reduces to powering the nodes up and placing them close to the location to monitor.

3.1 NETWORK GATEWAY

One of the TelosB nodes serves as a gateway for the network and is usually connected to a PC through the USB interface. In the PSN architecture we extended the GW functionalities by adding a Bluetooth module that shares the same UART lines used to communicate with the PC (see Figure 2). By adopting this simple solution (only a pull-up resistor and a couple of diodes are used to isolate the TX lines), the gateway can serve a phone, a PC or both devices at the same time without modification to the software structure. Whenever a radio packet is received and the GW forwards it to the USB controller, a copy will also be received by the BT radio. Similarly, messages sent by the PC or the phone[1] will be transparently forwarded to the IEEE 802.15.4 network.

The Bluetooth module (BlueRadios BR-C40A [3]) installed on the GW is a compact unit that offers a simple configuration interface based on AT-commands. The unit supports the Serial Port Profile and is configured to bridge the communication between the internal UART and the virtual port on the BT connection (RFCOMM). Although not all the functionalities are accessible (for example, it is not possible to create and advertise custom services through the Service Discovery Protocol), the available AT commands offer the possibility to tweak some low-level parameters useful to support battery-powered operations. In
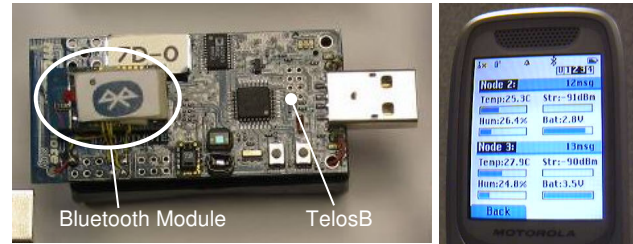
particular, increasing the *page scan* interval and activating the *sniff mode* reduce the current consumption to 2mA when the device is waiting for a connection; and, to 1mA when the device is connected (but no data is exchanged).

3.2 PHONE SOFTWARE

The software running on the user phone is a MIDlet that can be installed on any device supporting JAVA Micro Edition and Bluetooth communication. The software is responsible for handling the connection with the Bluetooth radio on the GW, converting the raw sensor reading in engineering units and providing the user with an intuitive interface to access the sensor data and control commands (see Figure 3). Given the gateway architecture, the only requirement imposed on the software was to handle communication with the GW using the same packet structure adopted within the TinyOS network (in facts, the GW routes the packets between the WSN and the phone without modification). This feature was enabled with limited effort by utilizing part of the JAVA code distributed with TinyOS. Using a stripped-down version of the PC-based classes that implements packet handling over serial communication, we were able to sensibly reduce the prototyping time for the SW. Slightly modified versions of the phone software were created to run on a PDA and a Bluetooth Notebook.

4. CONCLUSIONS

The PSN demonstrates that with limited effort it's possible to seamlessly integrate mobile computing solutions with sensor network technology. In particular, the use of a cell phone is appealing since many people find this technology familiar and intuitive to use, a fact that was confirmed by the opinions of the users of our system. And, even though the aim of this work was to explore connectivity with nearby sensors, because many handheld devices have LAN capabilities as well as the WAN capabilities of the cellular network, each individual local cluster of PSN could also provide data to other centralized data servers.

5. REFERENCES

[1] S. Krco, *Implementation solutions and issues in building a personal sensor network for health care monitoring*, 4th Int. IEEE Conf. on Information Tech. and Applications in Biomedicine, pagg 350-353, 2003.

[2] L. Zhong, M. Sinclair, and R. Bittner, *A Phone-Centered Body Sensor Network Platform: Cost, Energy Efficiency User Interface*, BSN 2006.

[3] Blueradios BR-C40., http://www.blueradios.com

---

[1] Message collisions can occur if the GW is connected to both a PC and a Phone, and both devices attempt to write on the UART lines at the same time. This does not pose a problem if commands to the WSN are issued sporadically.