2.4

A SENSORSPACES™ SYSTEM FOR DATA ACQUISITION AND REAL-TIME MANAGEMENT OF FIELD MEASUREMENTS

Young Yee *
Mkey Technologies, LLC
3903 Azalea Dr
Las Cruces, NM 88005
TEL: (505)-640-3046
EMAIL: yyee@aol.com

Edward Vidal, Jr
Sergio Cabrera
Gerardo Rosiles
Department of Electrical and Computer Engineering
University of Texas at El Paso
El Paso, TX 79968

1. INTRODUCTION

As the number of sensors and the quantity of collected data increases for field measurements, there has been much interest in sensor networks. The scientific community faces tremendous challenges in the collection, dissemination, and archiving of vast amounts of raw and processed data. In large field experiments such as high-resolution urban dispersion studies or global climate investigations, data can originate from various types of sensors, sources, and computing processes. Also, data collections can be sporadic and their publication can materialize in an assortment of formats and through dissimilar channels. To resolve many of these challenges, we propose a sophisticated yet simple and elegant architecture called a SensorSpaces™ system based on the concept of "tuplespaces" (Dr. David Gelernter, Yale University). A SensorSpaces system provides a virtual space to collect and disseminate sensor data objects that are well defined and easily shared. This concept provides loose coupling of sensors, systems, and models in a networked environment. The objective is to demonstrate that tuplespaces implementations can greatly simplify acquisition and distribution of field measurements. A model of a SensorSpaces system to collect and archive field measurements will be presented. Issues regarding sensor interfaces, legacy instrumentation, wireless communications, data base management, scalability of the number of sensors, distributed processing of loosely coupled systems, and performance of the data acquisition system will be discussed.

From a distributed array of commercial sensors, specialized embedded systems for the collection and management of field data were developed. Issues that needed to be addressed are sensor interface, data base management, scalability regarding the number of field sensors, distributed processing of loosely coupled systems, and performance of the data acquisition system (Vidal and Yee, 2003). In the present application, precise time tagging of the data records as it is being collected and navigation coordinates are important elements in the management and handling of these field measurements. The real-time collection and

* *Corresponding author address*: Young Yee, Mkey Technologies, 3903 Azalea Dr, Las Cruces, NM, 88005. EMAIL: yyee@aol.com

pre-processing of the data can be accomplished using micro-controller technology at the sensor source. In this way, near autonomous and self configuring sensors can be arbitrarily placed at optimal locations or they can be mounted vertically on a single field tower and networked together. Assembled data from each local network of sensors can be transmitted either wirelessly or hardwired to a central data collection node. The advantages and limitations of integrating specialized embedded systems inside field sensors will be discussed.

2. REQUIREMENTS

Present day computerized component and instrument technologies are advancing at a very rapid pace such that data taken during a comprehensive field experiment can be overwhelming. Recent terrorist activities in urban environments have prompted field studies in densely populated city business districts. To properly characterize these complex topologies and obtain quality high-resolution data, sensors will need to be mounted in numerous locations and data collection will pose difficult challenges. For example, in the Joint Urban 2003 Atmospheric Dispersion Study (2003), characterization of the dispersion and transport of toxic chemical/biological clouds required large arrays of field measuring instruments.

In many typical field data acquisition systems, specialized data loggers and multi-port serial boards for PC's are used to gather data from sensors (Figure 1). As technology has advanced, these systems have the following disadvantages:

- Limited by the number of physical serial COM ports that a PC can handle
- Software is usually specialized and specific to customized data loggers.
- Data distribution and data processing during data collection is restricted by the data logger software.
- Newer computer systems have abandoned older com ports in favor of the faster USB data ports.
- Conflicts with COM port system interrupts

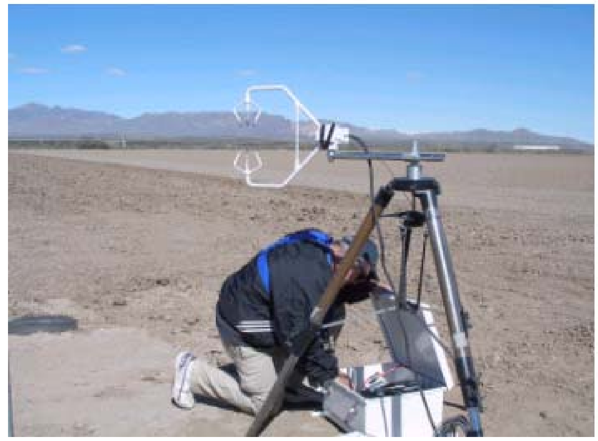

Figure 1. Traditional field measurement system showing sonic anemometer wind measurements using a standalone data acquisition system in an open field. (Photograph by Y.Yee)

In field and urban environments, the data acquisition architecture must be flexible and robust enough for remote site operation. In the Joint Urban 2003 experiments, the Army Research Laboratory deployed an array of wind measuring sensors called sonic anemometers that were networked together and the data transmitted wireless to a remote collection point (Yee, et.al., 2004).

Possible applications for deploying an array of field sensors over a region of interest are outlined.

- Characterize remote conditions near disaster-stricken areas such as locations in or near wildfires, floods, and severe weather.
- Collect data around hazardous waste spills for public safety.
- Measure environmental conditions at remote fields for scientific research studies.
- Provide initial meteorological data to run chemical/biological dispersion models in complex urban terrain.
- Provide critical information to predict advancing hurricanes, tornados, storms, etc.

● Monitor hydrological and meteorological phenomena over mountainous, desert, and/or dense vegetative regions.

3. INITIAL DESIGN AND DEVELOPMENT

The initial design of the SensorSpaces system began with a microprocessor with an Ethernet interface and a typical field sensor as outlined in Figure 2. A micro-controller board communicates directly with the field sensor via the traditional RS-232 communications where Tx is the transmit signal and Rx is the received signal. The data is then processed in real-time and sent out over an ethernet connection using the TCP/IP network protocol. Each controller board/sensor combination is uniquely identified with an IP network address. The initial prototypes were based on the Rabbit 2000 and 3000 micro-controllers and the interface programs were written in C and Java.

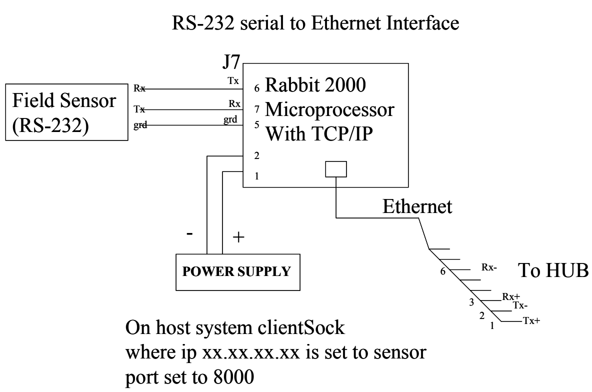

Figure 2. Initial design of the RS-232 serial to Ethernet interface for an individual field sensor.

Currently, a compact, rugged wind and temperature sensor package with GPS-stamped measurements with no moving parts is being developed. The final product would be a miniaturized environmental sensor package for hand deployment in an urban or field environment. The sensors would create an automatic network that would cover the area of interest and data from the sensors including GPS coordinates would be transmitted to a non-intrusive, remote station away from the measurement zone.

4. FIELD SENSOR INTERFACE

Figure 3 is one of the earlier prototype embedded systems that was developed to interface with a sensor's data acquisition board (Yee and Vidal, 2004). The RS-232

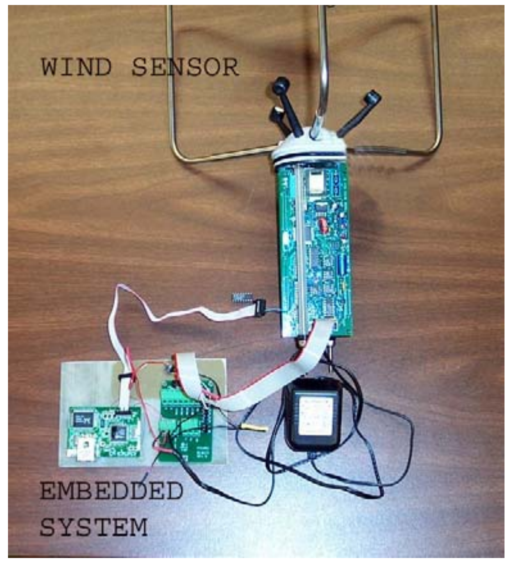

Figure 3. Photograph of a wind sensor connected to an embedded system with power supply.

serial data is collected and sent as data packages for transmission over an Ethernet line (Figure 4). In one embedded system application, the data is formed into an object, which is temporally held in a Java Space. Torres (2001) describes a generic model implemented in Java to develop scalable distributed applications. A loosely coupled application scans the Java Space for new objects and the elements of the object are inserted into the fields of a relational database. A position table in the database contains location information (latitude, longitude, elevation, local relative positions (x, y, z)). Time tagging of the data along with a sensor identification number and trial number is performed and linked with a key field (Vidal and Measure, 2001). This key field relates the series of field measurements together for selective retrievals and cross correlations.

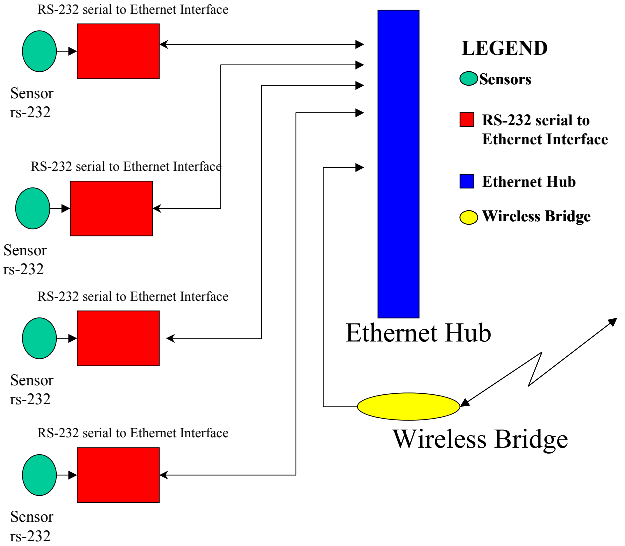

Figure 4. Concept of an array of measuring sensors networked together. The red boxes represent the specialized embedded systems for each sensor.

5. SENSOR NETWORK

The experimental collection of sensor measurements consists of several steps outlined as follows:
1. Select specific locations to place sensors for representative characterization
2. Calculate GPS coordinates including the height above the surface for each sensor.
3. Determine optimum sampling rate.
4. Set the IP network address for each met sensor.
5. Setup wireless LAN bridge connection.
6. Synchronize data collection with the central data acquisition system.
7. Screen for bad data lines and either delete it or process a correction.

6. SENSORSPACES

SensorSpaces™ is based on the concept of the tuple space, as first proposed by Gelernter (1985), which is an implementation of the associative memory paradigm for parallel/distributed computing. The system provides a repository of tuples, i.e., an ordered list of objects of a specified type, that can be accessed concurrently. For example, in the SensorSpaces system, there are a group of sensors that measure physical parameters and collect them as pieces of data. At the same time, there are also a group of processors that can use the data. Sensors that produce data can post their data as tuples in the SensorSpaces, and the users can then retrieve data from the space that match a certain criteria.

Figure 5 shows an overview of the SensorSpaces implementation. SensorSpaces implementation uses a JAVA Programming construct called "JavaSpaces". Tools and interfaces were developed specifically to handle sensor data. The SensorSpaces is a shared-memory space that can be accessed by anyone on the network. This space can be used to collect data, to share data, or to distribute data. Data objects can exist in the space temporarily and then expire i.e. deleted from the space or data objects can persist indefinitely.

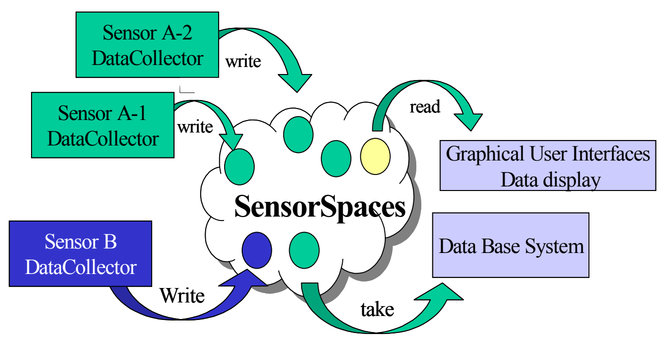

Figure 5. Overall concept of the SensorSpaces for data collection, information sharing, messaging, data display, and data archiving.

6.1 Features

The salient features of the SensorSpaces are highlighted as follows:
● Any number of data collecting processes can connect to a (networked shared-memory) space to
● Read, write, or take objects of interest from the space.

● Processes interact with the space by simply executing read, write, or take operations.
● Data objects reside in the space until a programmed time-out expires or a process takes it out of the space.
● Mechanisms exist to make contents of the space persistent in case of system or network failures.
● The implementation is platform independent
● Work can be distributed among different computers

6.2 Performance Issues

While the simplicity and elegance of implementing the tuplespaces concept is very appealing, optimum distributed computing performance can only be realized through careful planning, setup and analysis. Torres and Vidal (2003) have reported that data object size becomes an issue when there are large numbers of weather intelligence entries to be exchanged through the space. Because of the diverse and varied types of measurements that field sensor can collect, it is highly recommended that the user conduct a series of data acquisition experiments to determine system bottlenecks and to reduce the overhead of data transmission and storage.

Another issue could be the reliability of the tuplespaces environment. As with any networked application, tuplespaces process performance can be subject to network reliability and bandwidth availability.

7. SENSORSPACES DEMONSTRATION

For the SensorSpaces demonstration, a variety of sensors were purchased with varying capabilities. Some of the sensors were wireless, some sensors could measure temperature, some sensors could measure vibrations, some sensors could measure tilt (Figure 6).

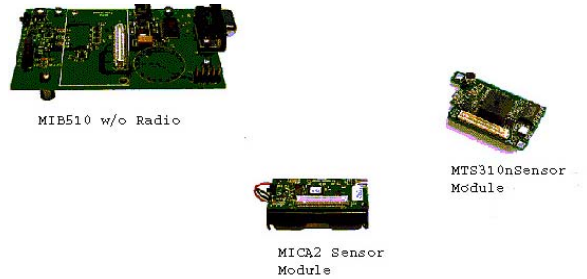

Figure 6. Commercially available devices called MOTES by Crossbow were used in one of the SensorSpaces simulation.

Figure 7 is a depiction of a simple embedded system configuration using a Rabbit 2000 micro-controller board to convert the RS-232 data from Crossbow's MIB510 Wireless Sensor Network base station to Ethernet transmission. In this configuration a low cost Rabbit 2000 can replace the laptop to communicate with the Wireless Sensor Network. Here the Rabbit 2000 is functioning as a gateway for several wireless sensors Motes (Processor Radio Boards) with sensor modules. The Tx from the MIB510 is connected to Rx of port C of the Rabbit 2000. The baud rate on the Rabbit is set to 57600 which is the default for the MIB510 with 8 data bits 1 stop bit and no flow control. The IP of the rabbit is currently set to 192.168.0.60. The program is listening at

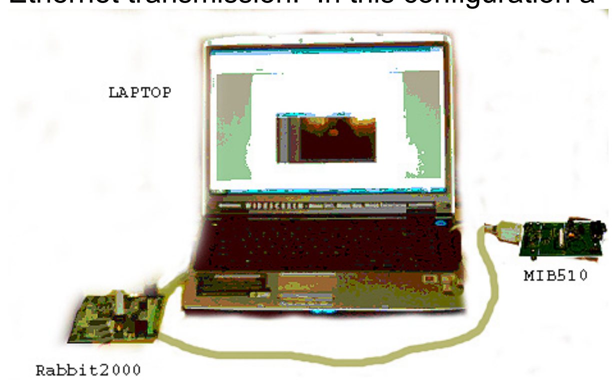

Figure 7. A commercially available low resolution RS-232 sensor (MIB510 device that measures temperature and vibrations) was connected to an Ethernet interface via the Rabbit2000 microprocessor.

port 8001 while the sonic sensor uses port 8000. In Figure 8, a Mote Interface Board MIB510 is connected to a laptop PC to provide radio communication and data collection from the other mica2 sensor devices in the network.

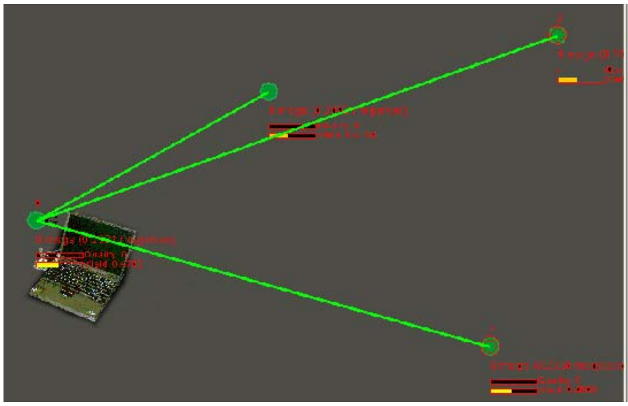

Figure 8. Topology of an array of measuring sensors networked together.

8. SUMMARY AND CONCLUSION

The retrieval of sensor measurements in field or urban type settings requires specialized methodologies to handle the data ingestion of large quantities of data. To accomplish this task, embedded systems were developed to collect traditional serial output data from field sensors, to quality control the data in real-time and to transmit the processed data into an ethernet network. Using microchip technology to collect and process data at the sensor level, the data from individual sensors can be networked together. Assembled data from each local network of sensors can then be transmitted wirelessly over a network bridge to a central node. Data from each sensor can be uniquely tagged with sensor identification, GPS location, and time stamps. Data at the central node can be quality checked for corrupted data entries before being saved in a relational data base system. Implementation of a SensorSpaces environment will allow multiple accesses to the data for distributed applications such as real-time displays, data compression, sorting, and statistical analysis. This architecture is not limited to the collection of one type of measurement. It can be used to collect other types of sensor instrumentation that output serial information such as pressure readings, rain gauge measurements, radiation measurements, etc. Scalability features and loosely coupled distributed data processing are the key highlights of this architecture.

9. REFERENCES

JOINT URBAN 2003 ATMOSPHERIC DISPERSION STUDY, Oklahoma City, July 2003, EXPERIMENTAL PLAN, US DOD Defense Threat Reduction Agency (DTRA), US DOE National Nuclear Safety Administration - Chemical and Biological National Security Program (CBNP).

Gelernter, D., 1985: Generative communication in Linda, *ACM Transactions on Programming Languages and Systems*, Vol 7 No 1.

Torres, Mario, 2001: Developing Scalable Distributed Applications, A Generic Model Implemented in Java, *Dr Dobb's Journal*, Sept, 2001.

Torres, M. and E. Vidal, 2003: Dissemination of weather intelligence information and met sensor data acquisition using Jini/JavaSpaces technologies, Proceedings of the SPIE - The International Society for Optical Engineering, Volume 5101, pp. 257-268.

Vidal, Edward Jr. and Measures Edward M. "Integrating Meteorological Instruments Using Java and Computer Networking", Fifth Symposium On Integrated Observing Systems, Jan 14-18, 2001 90-94.

Vidal, Jr, E. and Y. Yee, 2003: Data Collection Of High Resolution 3D Sonic Anemometer Measurements, 19th International Conference on Interactive Information Processing Systems (IIPS) for Meteorology, Oceanography, and Hydrology, Long Beach, CA.

Yee, Y.P., M. Bustillos, S. Chang, R. Cionco, E. Creegan, D.S. Elliott, D. Garvey, G. Huynh, D. Ligon, E. Measure, D. Quintis, M. Torres, G. Vaucher, E. Vidal, Jr, C. Williamson, J.Yarbrough, "Wind and Turbulence Observations in Joint Urban 2003", The 84th Annual Conference of the American Meteorological Society, Seattle, WA, Jan 2004.

Yee, Y and E. Vidal, Jr, 2004: Embedded Systems for Meteorological Sensor Applications, *International Conference on Embedded Systems and Applications 2004,* Las Vegas, NV.