GPS TRACKING LOCATION-BASED SERVICE USING WRISTWATCH GEOZIGBEE SENSORS

Alison Brown, Jacob Griesbach, and Bruce Bockius; *NAVSYS Corporation*
Terry Boult, *University of Colorado at Colorado Springs*

BIOGRAPHY

Alison Brown is the Chairman and Chief Visionary Officer of NAVSYS Corporation. She founded NAVSYS in 1986 and served as President and Chief Executive Officer until 2006. She has a PhD in Mechanics, Aerospace, and Nuclear Engineering from UCLA, an MS in Aeronautics and Astronautics from MIT, and an MA in Engineering from Cambridge University. She was a member of the GPS-3 Independent Review Team and the Interagency GPS Executive Board Independent Advisory Team, and is an Editor of GPS World Magazine. She is an ION Fellow and an Honorary Fellow of Sidney Sussex College, Cambridge.

Jacob Griesbach received BS, MS, and PhD degrees in electrical engineering from the University of Colorado at Boulder in 1995, 1997, and 2000, respectively. Starting in 2001, he conducted defense research at the MIT Lincoln Laboratory and joined NAVSYS in 2006, where his research interests include statistical signal and array processing for radar and sonar, GPS signal processing, space-time adaptive processing, filterbanks, adaptive systems, and audio processing.

Bruce Bockius is a Systems Engineer at NAVSYS Corporation where he is responsible for systems engineering, architectural design, software development and testing. He has a B.S. Aerospace Engineering, University of Colorado at Boulder, and is currently working towards his Masters Degree in Computer Science.

Terry Boult is an El Pomar Endowed Professor of Communications and Computations at the University of Colorado at Colorado Springs (UCCS). Before joining UCCS in 2003, he was an endowed professor and founding chairman of Lehigh University's CSE Department. He received his BS in Applied Math (1983), MS in CS (1984), and PhD in Computer CS (1986) from Columbia University and then spent 8 years on the Columbia CS Department Faculty. He has published over 150 Papers and holds 5 patents with 8 pending.

ABSTRACT

The GeoZigBee sensor is a low power GPS wristwatch tracking device that is being developed by NAVSYS Corporation under contract to the US Army. The GeoZigBee utilizes a client/server approach, where the wristwatch client sends raw GPS data to the GeoZigBee Server via a wireless ZigBee® protocol data link. The GeoZigBee wristwatch device includes a low power TIDGET® sensor, flash memory, and a ZigBee wireless data link. The design of the device was selected to minimize the power drain to allow operation for extended periods of time. The TIDGET sensor operates by taking brief snapshots of GPS data when activated. These snapshots are stored in the flash memory and forwarded to the GeoZigBee Server through the ZigBee data link when the device is in range of a ZigBee Gateway.

The GeoZigBee Server integrates a Software Defined Radio navigational processor with an Oracle Application Server. The function of the server is to receive and store the wristwatch data, process the incoming data to compute navigation solutions, and store the results. The Application Server includes a custom web portal that provides location-based services (LBS) to users through a web interface to view, filter, and display the GeoZigBee tracking results.

This paper covers the design of the GeoZigBee system architecture including the wristwatch sensor and the GeoZigBee Server, and presents GPS tracking results. The Web Portal and LBS offered by the Portal for GPS tracking are described with a discussion on the use of open-architecture Open Geospatial Consortium (OGC) standards for sharing and viewing the LBS data through the web as a mapping overlay.

INTRODUCTION

The GeoZigBee wristwatch device was designed specifically to provide an ultra low-power, miniaturized wireless GPS tracking device. It includes a low-power

GPS sensor, a flash memory and a ZigBee® wireless data link. The design of the device was selected to minimize the power drain to allow operation for extended periods of time. The power savings is accomplished by remoting the GPS signal processing from the sensor to the LocatorNet server. This has the advantage of minimizing the power drain within the GeoZigBee device while allowing sophisticated signal processing to be performed at the server to maximize the sensitivity of the GPS signal processing solution. The design of the GeoZigBee device is included in this paper.

The networked system architecture for relaying data between the tracking devices and the LocatorNet Server is illustrated in Figure 1. The ZigBee devices are programmed to search for a ZigBee gateway that provides a connection to the Internet. Once a Gateway is found, the devices upload their GPS sensor data through the Gateway to the LocatorNet Portal for processing. This paper describes the LocatorNet system architecture and the design of the ZigBee Gateways. The ZigBee Gateways have been developed to be an add-on to a standard PC providing a low-cost, secure method of connecting ZigBee to the Portal through a variety of physical network connections including wired Ethernet for fixed connections or WLAN, cellular, or SATCOM services for mobile units.

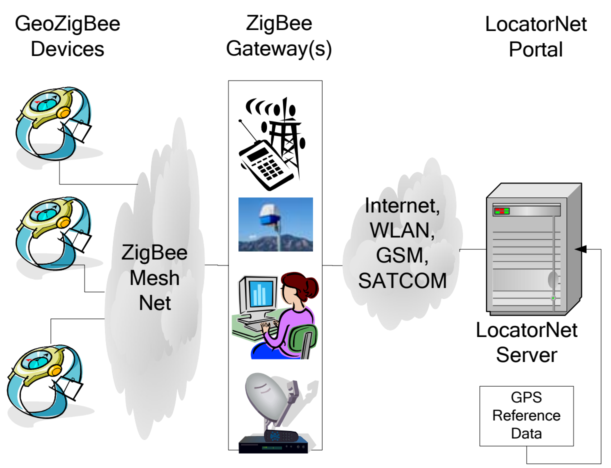

Figure 1 LocatorNet System Architecture

The LocatorNet Server provides the data processing to convert the GPS sensor data into locations. The design of the Software Defined Radio (SDR) used for processing the GeoZigBee GPS data is described in this paper and test results are included demonstrating the system operation. The LocatorNet Server is integrated with a Web Portal that allows users to access the sensor location data for analysis and display. This web-based architecture allows the common infrastructure to be leveraged by a variety of different users. In this paper, we describe an open architecture development approach that allows the LocatorNet server to deliver customized Location-based services using a combination of custom and publicly available data product.

TIDGET GPS SENSOR

The patented TIDGET® ("tracking widget") sensor operates by taking brief snapshots of GPS data when activated[1]. These snapshots are stored in the flash memory and forwarded to the LocatorNet Server through the data link for processing[2].

The TIDGET is built using the RF front-end of a commercial GPS chip (see Figure 2). The device is designed to operate with a variety of different types of data links providing a low-power location solution. Instead of performing the GPS signal processing using an internal baseband processor, the TIDGET device only samples and records the GPS snapshots periodically. While this requires more data to be transmitted across the wireless data link, it significantly reduces the overall power drain of the device making this an ideal solution for low-power tracking applications.

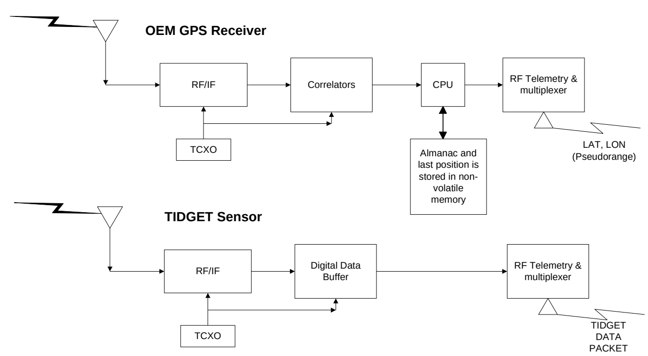

Figure 2 TIDGET Sensor

The TIDGET can be programmed to take different sizes of GPS snapshots depending on the application. The larger the snapshot, the greater the ability of the LocatorNet processing to track low-power GPS satellite signals[3]. For the GeoZigBee device, a 36.4 kbyte snapshot size was selected. This allows GPS signals to be detected to a C/N0 of 26 dB-Hz (-148 dBm) with a SNR detection threshold of 10 dB. When multiple satellites are in view, enhanced signal processing can be used to detect weak signals as low as 20 dB below the strongest GPS satellite signal, down to a threshold of -156 dBm.

This signal tracking performance is comparable with a state of the art GPS chip set such as the SiRFstarIII[4]. This chip set is capable of acquiring GPS signals to a threshold of -147 dBm indoors when using aiding data from GSM, 3G, or CDMA and when one satellite is being received at -147 dBm. The SiRFstar III is able to compute an aided open sky fix using only 100 mJ of power. However, under weak signal conditions the power/fix increases to over 1350 mJ. The TIDGET power consumption by comparison remains constant at 15 mJ for taking a 52 msec (36.4 kbyte) snapshot whether strong or weak signal processing is needed to be performed at the Server.

The total power consumption of the TIDGET device must also include the data link required to transmit the TIDGET data back to the LocatorNet Portal. In Table 1, a list of some of the different wireless data links that could be used for transmitting this data are included. For the low-power GeoZigBee wristwatch application, we have selected the ZigBee data link as providing the best compromise in terms of power consumption, range of operation and data bandwidth.

Table 1 Alternative TIDGET Data Links

	ZigBee 802.15.4	Bluetooth® 802.15.1	Wi-Fi® 802.11b	GPRS / GSM 1XRTT / CDMA
Application Focus	Monitoring & Control	Cable Replacement	Web, Video, Email	WAN, Voice/ Data
System Resource	4KB-32KB	250KB+	1MB+	16MB+
Battery Life(days)	100-1000+	1-7	.1-5	1-7
Nodes Per Network	255/65K+	7	30	1,000
Bandwidth (kbps)	20-250	720	11,000+	64-128
Range (meters)	1-100+	1-10+	1-100	1,000+
Key Attributes	Reliable, Low Power, Cost Effective	Cost, Convenience	Speed, Flexibility	Reach, Quality

ZIGBEE DATA LINK

ZigBee[5] is a low-power wireless mesh networking protocol that has been designed for maximum power life. The GeoZigBee wristwatch device includes a ZigBee data link that can be used to connect to a ZigBee Gateway when within range. A networking protocol has been developed that allows the ZigBee wireless data link to transport the data to the Gateway where it is forwarded to the GeoZigBee Server for processing.

Basic ZigBee's initial market profile is for lighting, HVAC, and security systems, all very low-bandwidth applications. ZigBee has built in mesh networking allowing single or multi-hop connections between units to a Gateway. Using the basic ZigBee "reliable mode" of transmission, we were only getting 32Kbps throughput for a single hop connection and only 2Kbps to 3Kbps throughput, with 6-7% packet loss, on a 2 hop (3 node) network. This performance was not sufficient for prompt, reliable wireless uploads of the GeoZigBee data to a Gateway. To address this issue, we developed an application-layer selective retransmission protocol, built on top of the ZigBee best-effort mode. This protocol is more intelligent and provides true network reliability resulting in increased data throughput. With a single hop network, we sustained over 90Kbs, and in many cases, with good signal strength, we could obtain nearly 200Kbps, all with zero packet loss and very few retransmissions. For a two hop network we maintained over 45Kbps with 3% to 5% retransmissions needed to ensure zero overall packet loss[6].

An important factor to consider is that this throughput performance improvement was achieved without any changes to the actual ZigBee Protocol but by developing an application layer enhancement on top of the existing protocol. This way the proprietary part of any ZigBee Stack need not be altered. Since the enhancement will run on the application layer, it could be included in the ZigBee profile as a utility.

The normal range of operation of the ZigBee data link, using onboard or PCB antennas, is within 100 feet of a Gateway. A Gateway can be a conventional PC equipped with a ZigBee USB device and installed with the LocatorNet network upload software. While straight ZigBee dongles exist, the system design uses a GeoZigBee unit configured in a "gateway mode" connected by USB to the PC. The LocatorNet network upload software converts the GeoZigBee data into a sequence of Database update commands and sends those over a TCP/IP connection to the Portal.

Figure 3 Extended Range ZigBee Connection

The range between the GeoZigBee device and the ZigBee gateway can be extended by adding an improved antenna to the ZigBee Gateway. Static directional antennas on both the node and the coordinator can improve the operational distance by a factor of 4-10, but are unacceptable for the watch unit. Larger amplified directional antennas, such as that shown in Figure 3, have allowed us to extend the high-bandwidth single-hop transmissions up to 765m, while only using the onboard PCB antenna for the GeoZigBee node The Phased Array antenna shown in Figure 3 also provides a Linux-based network router making it ideal for a Gateway unit.

GEOZIGBEE DEVICE

As depicted in Figure 4, the GeoZigBee wristwatch unit comprises a ZigBee chip with an embedded microcontroller, a "glue-logic" CPLD programmable logic device, GPS cache SRAM, bulk-storage Flash memory, a GPS front-end RF chip, antenna and TCXO, a USB interface chip and power management circuitry.

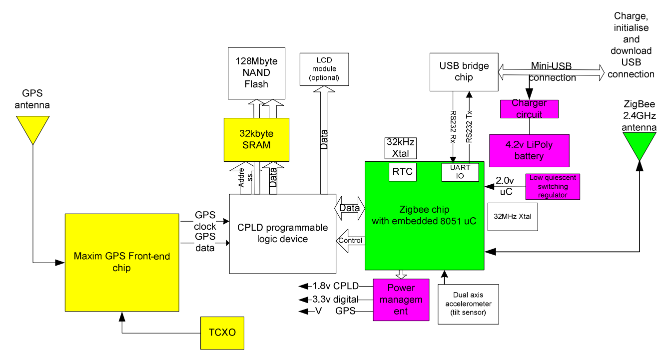

Figure 4 GeoZigBee Device Components

The host microcontroller acts as system controller, spending most of its time in a very low-power power-down mode (16uA, 166 Joules/month), with programmed wake-up by timed events from its onboard Real-Time Clock. On periodic wake up, it checks for available connections to ZigBee Gateways and takes GPS snapshots. To minimize wasting memory, if the device is not oriented to the sky, an onboard dual axis accelerometer can estimate orientation.

If a GPS snapshot is requested, the GPS front-end circuitry will be powered on and the GPS data streamed via the CPLD to an SRAM cache memory. The GPS section is then immediately powered down. The host uC will then copy the memory from SRAM to the Flash file system.

The microcontroller extensively controls power management of the device, only powering up the sections of the circuit that are needed at the appropriate time. Power is supplied by means of an onboard 140mAhr Lithium Polymer cell. This cell is charge via the wrist watch USB connection. The USB connection also gives a means to initialize the device and an alternative means of downloading stored data.

Each captured GPS snapshot requires 15mJ to acquire under all circumstances. To transmit the data to the gateway, the circuit will consume between 70mJ and 231mJ per snapshot depending on range and signal environment. Total battery energy available to the device is 2000 Joule.

LOCATORNET SERVER GPS PROCESSING

The LocatorNet Portal is based on an Oracle Application Server. The ZigBee Gateway software is designed to "publish" data into the LocatorNet Portal which initiates a data processing sequence using the LocatorNet Server signal processing software. The LocatorNet Server GPS processing is implemented using an SDR architecture where the GPS signal generation and code correlation functions are performed in software. The GPS Navigation data is loaded into the LocatorNet Portal from reference station sites across the Internet allowing world-wide tracking of GPS data. The LocatorNet Portal also can access digital terrain data allowing altitude-aided solutions to be calculated in the event that only three GPS satellites are tracked. Using a special purpose high speed signal processor, the time-to-compute-fix (TTCF) for the TIDGET data is less than 200 msecs. With a Pentium 4 processor the TTCF is approximately 1 minute. Figure 5 and Figure 6 illustrate the navigation accuracy associated with the snapshot signal processing. A measured CEP of 5.14 m was demonstrated with this data.

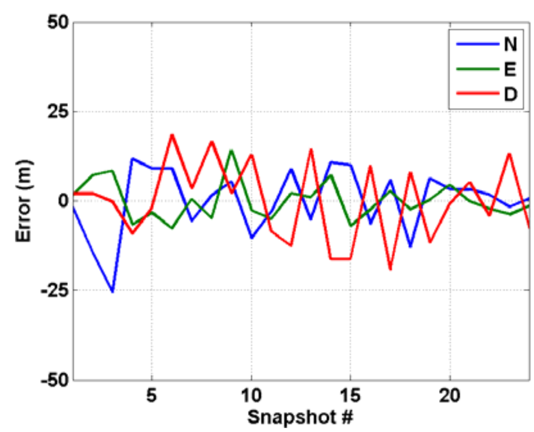

Figure 5 Snapshot NED Comparison to Reference

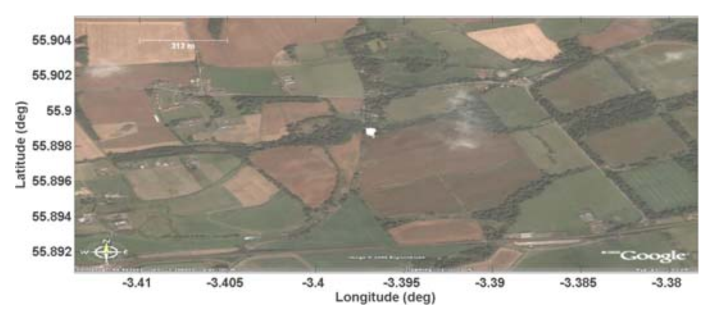

Figure 6 Snapshot Position Solution

LOCATORNET LOCATION BASED SERVICE ARCHITECTURE

The LocatorNet Portal provides a secure repository for the GeoZigBee tracking data allowing only authorized subscribers access to individual GeoZigBee device results. A Web-based interface is provided to display the data results overlaid on a map.

The Portal repository schema is illustrated in Figure 7. Gateways, GeoZigBee Devices, Wearer IDs, and User Subscribers are each individually registered in the GATEWAY_ID, TIDGET_DEVICE, WEARER_ID, and SUBSCRIBER_INFO tables respectively. Also, the configuration parameters for all devices are stored into the DEVICE_TYPE table that is referenced by each TIDGET_DEVICE entry. When new data arrives from a gateway, it is placed into the TIDGET_DATA table with a TIDGET_HEADER entry providing a rough position estimate from the gateway. As the data is processing, pseudorange entries are stored in the RNG table for both C/A and P(Y) measurements. Afterwards, the RNG entries for a given snapshot are used to compute a navigation solution entry in the TIDGET_SOLUTION table that includes an accurate position and velocity measurement. These position and velocity measurements are used to provide the data for the LocatorNet geographical displays.

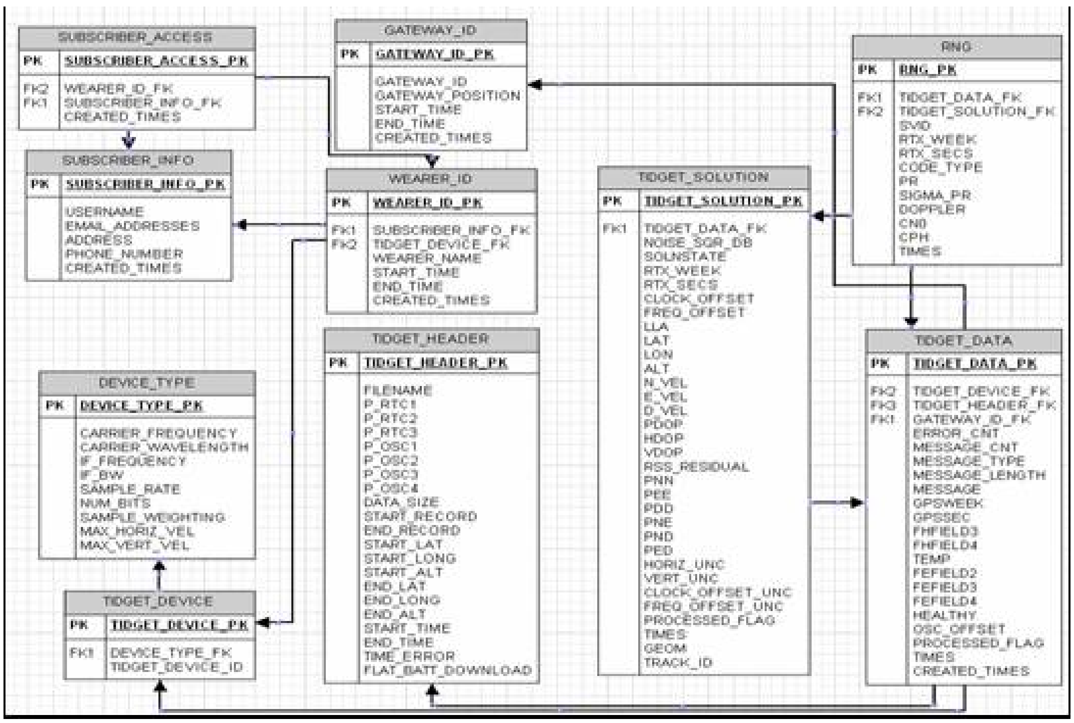

Figure 7: Portal Database Schema

As shown in Figure 8, in order to offer Location based Services, it is necessary to combine the TIDGET SOLUTION data with mapping and feature content. In developing the LocatorNet architecture, we considered three alternative implementation options for providing generic LBS functionality.

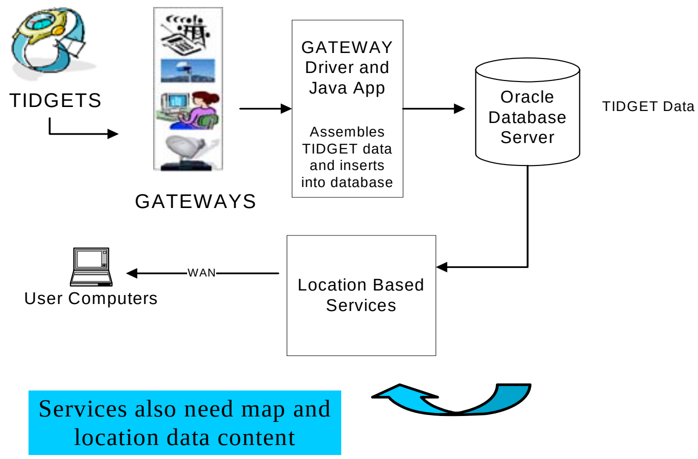

Figure 8 LocatorNet LBS Architecture

The first LBS implementation method considered was to populate the LocatorNet server with static GIS mapping and feature data from companies like NAVTEQ and TeleAtlas. Available web-based development tools for building LBS applications using this approach include ArcWeb, MapPoint and NavBuilder. The advantage of this development approach is that you ended up with a fully contained, completely customized solution. The disadvantages are: a relatively long time-to-market for building custom applications, expensive development costs and an unending requirement (and cost) to keep the external GIS data up-to-date. This method results in products that aren't standards-based and can't easily accept information from outside data sources or share your content with other LBS applications.

The second LBS architecture considered was to make use of existing web-based products such as Google Maps or Yahoo Maps. These products allow you to overlay your own data on top of their maps. The advantages of this approach are quick and easy development and excellent map content and coverage. The disadvantages include licensing costs and/or restrictions, limited customizability, inability to control the end product, and a lack of interoperability. As examples, Yahoo map's licensing terms prohibit displaying GPS-derived data less than 6 hours old on their maps[7], and Google requires standard map pages to be publicly accessible on the Internet[8]. Either may choose to add advertising to their maps in the future. These solutions also don't allow you to share your data with other applications using open standards.

The third LBS architecture considered was to serve up the GIS mapping and feature data using an off-the-shelf standards-based web mapping server. An example of this is eSpatial's iSMART5 server[9]. iSMART serves up a map display web page on demand, generated by fusing together different layers of data retrieved using web services from other standards-based servers. Web Map Services (WMS) are an Open Geospatial Consortium (OGC) standard that allows GIS-based applications to share GIS data through web services. As an example, the TIDGET feature data included in the Oracle database and mapping and feature data retrieved from other servers can be fused together into a single web display that can be viewed by LBS users through a standard web browser. The iSMART server can also "publish" the local GeoZigBee data as a making it available to other data consumers.

The open architecture OGC compliant web service based design approach was selected for the LocatorNet Server. The advantages of this approach are: rapid deployment by using an off-the-shelf product, access to remote data sources, and the ability to customize the user experience through custom-written web pages that contain the generated map images. Since most map data is obtained through remote data sources there is no local cost or effort required to stay current. Local map data can still be seamlessly integrated for unique features not available elsewhere, or as a backup when the remote data is unavailable.

Figure 9 through Figure 13 (GeoZigBee and the City Streets overlay) can be seamlessly integrated with the remote WMS data (imagery, topographic map, and weather RADAR) and presented to the user in an easy-to-navigate fashion.

OPEN GEOSPATIAL CONSORTIUM

The LocatorNet Portal was designed to be compatible with the web services that have been standardized by the Open Geospatial Consortium (OGC)[10]. OGC have developed a family of web services that can be used for sharing and distributing mapping and feature data between web servers.

The OpenGIS® Web Map Service (WMS) Implementation Specification provides three operations (GetCapabilities, GetMap, and GetFeatureInfo) in support of the creation and display of registered and superimposed map-like views of information that come simultaneously from multiple remote and heterogeneous sources. When client and server software implements WMS, any client can access maps from any server. Any client can combine maps (overlay them like clear acetate sheets) from one or more servers. Any client can query information from a map provided by any server.

The OpenGIS® Web Feature Service (WFS) Implementation Specification allows a client to retrieve and update geospatial data encoded in Geography Markup Language (GML) from multiple Web Feature Services.

The specification defines interfaces for data access and manipulation operations on geographic features, using HTTP as the distributed computing platform. Via these interfaces, a Web user or service can combine, use and manage geodata -- the feature information behind a map image -- from different sources. The following WFS operations are available to manage and query geographic features and elements: create a new feature instance; delete a feature instance; update a feature instance; lock a feature instance; and get or query features based on spatial and non-spatial constraints.

The OpenGIS® Web Coverage Service (WCS) Implementation Specification extends the Web Map Server (WMS) interface to allow access to geospatial "coverages" (raster data sets) that represent values or properties of geographic locations, rather than WMS generated maps (pictures).

The Web Coverage Service supports electronic interchange of geospatial data as "coverages" - that is, digital geospatial information representing space-varying phenomena. A WCS provides access to potentially detailed and rich sets of geospatial information, in forms that are useful for client-side rendering, multi-valued coverages, and input into scientific models and other clients. The WCS may be compared to the OGC Web Map Service (WMS) and the Web Feature Service (WFS); like them it allows clients to choose portions of a server's information holdings based on spatial constraints and other criteria. Unlike WMS which filters and portrays spatial data to return static maps (rendered as geo-registered pictures by the server), the Web Coverage Service provides available data together with their detailed descriptions; allows complex queries against these data; and returns data with its original semantics (instead of pictures) which can be interpreted, extrapolated, etc. -- and not just portrayed. Unlike WFS which returns discrete geospatial features, the Web Coverage Service returns representations of space-varying phenomena that relate a spatial-temporal domain to a (possibly multidimensional) range of properties.

The OpenGIS® Location Service (OpenLS) Implementation Specification: Core Services describes OpenGIS Location Services (OpenLS). The primary objective of OpenLS is to define access to the Core Services and Abstract Data Types (ADT) that comprise the GeoMobility Server, an open location services platform. OpenLS Core Services are specified in five parts. The Directory Service is a network-accessible service that provides access to an online directory (e.g., Yellow Pages) to find the location of a specific or nearest place, product or service. The Gateway Service is a network-accessible service that fetches the position of a known mobile terminal from the network; this interface is modeled after the Mobile Location Protocol (MLP), Standard Location Immediate Service. The Location Utility Service provides a Geocoder/Reverse Geocoder; the Geocoder transforms a description of a location, such as a place name, street address or postal code, into a normalized description of the location with a Point geometry usually placed using Cartesian coordinates, often latitude and longitude. The Presentation (Map Portrayal) Service portrays a map made up of a base map derived from any geospatial data and a set of Abstract Data Types as overlays. The Route Service determines travel routes and navigation information between two or more positions.

EXAMPLES OF WMS DATA FEEDS

A variety of different WMS data feeds are already available on the web from public sources and more are being added. An example of some of the data that has been linked into the LocatorNet Portal are shown in Figure 9-Figure 13. Figure 9 shows current TIDGET data for a user's area of interest that has been extracted from the local LocatorNet database. In Figure 10 the user adds in vector street data to provide location context. This street data is stored on the server database, and thus is available even when the server is not able to access the Internet – useful for some mobile and military applications. In Figure 11 the user has requested a USGS supplied topographic map overlay in lieu of the vector streets. This data was requested on-the-fly from the USGS by the iSMART server through the Internet[11]. Figure 12 shows a high-resolution aerial image obtained from another USGS Web Map Service as the background[12]. Lastly Figure 13 shows how remote data services can also provide time-sensitive information – in this case Doppler RADAR data[13].

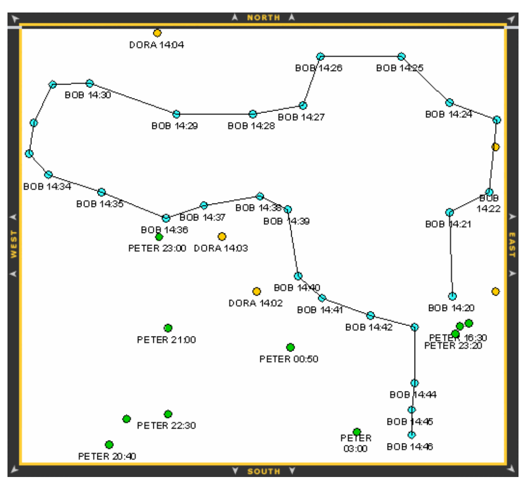

Figure 9 GeoZigBee Data

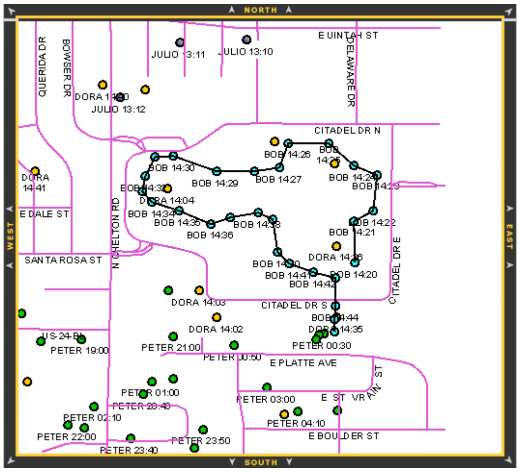

Figure 10 City Streets overlay from local database

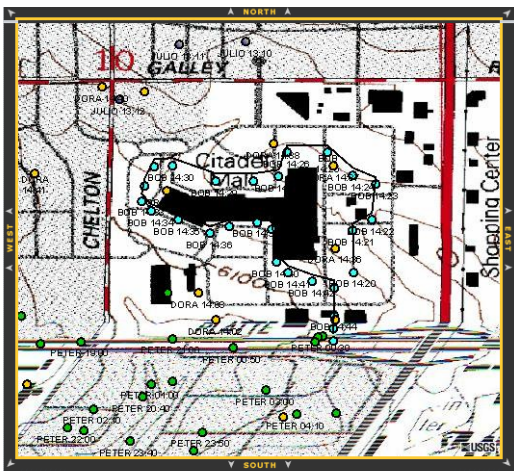

Figure 11 Combining a topographic map pulled over the internet from a USGS Web Map Service

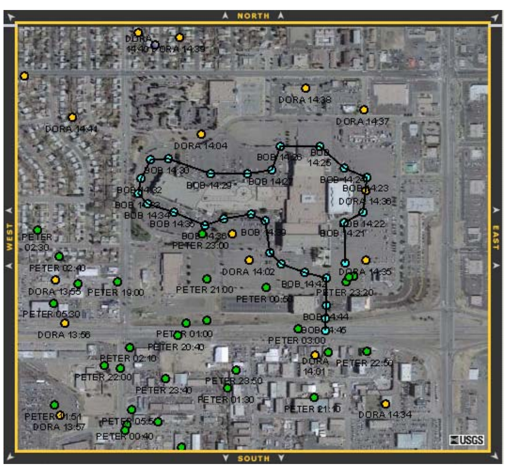

Figure 12 High-resolution Aerial Imagery pulled over the internet from another USGS Web Map Service

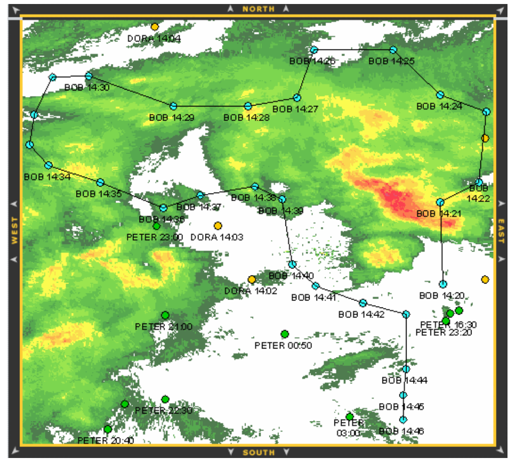

Figure 13 Live Doppler RADAR pulled over the internet from a NOAA Web Map Service

EXAMPLE LOCATORNET APPLICATION

Figure 14 illustrates a basic LocatorNet personnel tracking LBS application. This application was developed based on an Oracle Application Server and iSMART5 mapping server, together with some custom web pages. The Oracle Application Server and database maintains data integrity and security, allowing users access to only the data they need. Oracle Spatial database extensions were used to quickly provide spatial querying capabilities; users can ask questions such as "Who is near the intersection of Powers Blvd and 30th Street," or "Where was User X at 2pm yesterday?" By making use of live WMS feeds such as Doppler RADAR users can even make queries such as "Was User X being rained on a 2 pm yesterday?"

The flexible LocatorNet LBS architecture allows additional mapping and feature content to be easily added for customized applications. Maps, images, features from WMS, WFS, WCS sources can easily be added into Location-based services. The Open architecture also allows content to be shared through OpenLS standards.

Figure 14 LocatorNet Application

CONCLUSION

The GeoZigBee tracking sensors have the following advantages over previous GPS tracking solutions.

- Ultra low-power design enabling operation for 30 days using a wristwatch size device and battery
- Wireless networked connectivity using low-cost COTS ZigBee devices
- Improved ZigBee data transfer reliability using an enhanced ZigBee transmission protocol

The open architecture, standard-based design approach for the LocatorNet server has the following advantages for offering Location-based services

- Web-based access and display of the GeoZigBee tracking results using only a thin client
- Powerful search and query engines using Oracle's geospatial application server
- Rich feature content through integration with OpenGIS web services
- Flexible web based user interface design for rapid implementation and deployment of web based services
- Secure data management for maintaining privacy and managing access to GeoZigBee tracking data

A variety of commercial and military applications exist for the GeoZigBee tracking unit and the LocatorNet Location-based services. The first GeoZigBee system is being developed for the U.S. Army TATRC for use in clinical trials. Its initial use is envisioned as a hospital type bracelet that will be used to collect the location and time of treatment applied to trauma patients in support of clinical trials. The data presented in this paper is from a lab GeoZigBee unit. Delivery of the first wristwatch prototype units will occur in early 2007.

The inexpensive, small GeoZigBee devices can be used for monitoring locations of children, pets, parolees or Alzheimer patients as examples. By the Subscriber defining a "Perimeter" at the LocatorNet Portal, automated web services can be used to alert the subscriber when the designated device passes beyond that perimeter.

Applications also exist for First Responders. In many emergency response or disaster recovery scenarios, there is a need to maintain situational awareness on the location of emergency personnel. By placing an extended range ZigBee Gateway (see Figure 3) on an aircraft, the locations of emergency responders on the ground can be routed to the LocatorNet Server and made available to the First Responder command centers through the Internet. This type of system would have immediate benefit for providing real-time situational awareness of personnel in hazardous areas. By overlaying the GeoZigBee personnel locations on a real-time mosaic provided by a system such as NAVSYS' GI-Eye and GRIM[14] products, First Responders would have real-time knowledge of personnel located in danger areas such as fire hot spots or flood zones as in the Hurricane Katrina scenario.

NAVSYS is currently working with industry partners who are interested in producing and distributing the GeoZigBee devices for a variety of commercial applications.

ACKNOWLEDGMENTS

The authors would like to acknowledge the support of Dr. Gary Gilbert and TATRC who have provided funding to support the development of this technology.

REFERENCES

[1] A. Brown, "The TIDGET – A Low Cost GPS Sensor for Tracking Applications," ION 5th International Technical Meeting, Albuquerque, September 1992

[2] GPS tracking system, United States Patent 5,379,224

[3] A. Brown, M. May, B. Tanju, "Benefits of Software GPS Receivers for Enhanced Signal Processing," GPS Solutions 4(1), Summer, 2000 pp 56-66

[4] http://www.sirf.com/Downloads/Collateral/GSC3(f)_6.20.05.pdf

[5] ZigBee Specification v1.0, ZigBee Document 053473r00, Version 1.00, ZigBee Alliance dated December 14, 2004.

[6] Sujeeth Narayan, "Extension of Zigbee stack to achieve higher reliability and throughput for transfer of large data objects," MS Thesis U. Colorado at Colorado Springs, 2006

[7] http://developer.yahoo.com/maps/mapsTerms.html

[8] http://www.google.com/apis/maps/terms.html

[9] http://www.espatial.com/products_ismart5overview.html

[10] Open Geospatial Consortium, http://www.opengeospatial.org/

[11] Via http://terraserver-usa.com/

[12] Via http://terraserver-usa.com

[13] http://maps.customweather.com/image?VERSION=1.1.1&REQUEST=GetCapabilities&Service=WMS

[14] A. Brown, H. Holland and Y. Lu, "Real-Time Web-based Image Distribution using an airborne GPS/inertial Image Server," ION GNSS 2006, Fort Worth, September 2006