Smartphones

Editor: Roy Want ■ Google ■ roywant@acm.org

Bluetooth LE Finds Its Niche

Roy Want, Bill Schilit, and Dominik Laskowski

For years, researchers have been writing about the Internet of Things and sensor networks, describing them as integral to the future of computing. However, despite the advantages they offer, neither has experienced widespread adoption. Here, we discuss the new Bluetooth Low Energy (LE) protocol, which might be the missing link for these technologies. Bluetooth LE fills a gap, efficiently linking smartphones and low-power sensors previously unsupported by wireless standards such as Classic Bluetooth and Wi-Fi.

THE ROAD TO BLUETOOTH LE

In the late 1990s, cell phone and laptop manufacturers, along with vendors of headsets, mice, and other accessory devices, realized that connecting their products would enable new types of multidevice applications. However, the contemporary practice of employing cables was cumbersome. Clearly, wireless technology would be more convenient, but it required the right combination of operating characteristics.

Standards such as IEEE 802.11b, now known as Wi-Fi, were on the verge of commercialization to support conventional local-area networking over wireless. However, unlike laptops with large batteries or desktops using mains power, feature phones and accessories had relatively meager power supplies. Wi-Fi, in its original form, was an 11 Mbps network, with power usage on the order of 500 mW. If small, power-constrained devices were to employ a wireless network, that network needed to consume less power and cost less—commensurate with the hardware it was augmenting.

The solution was to define a new wireless standard that wasn't a full-blown network solution but instead a wireless cable replacement technology at 1 Mbps. In 1998, Ericsson, Nokia, and Intel formed a special interest group (SIG) and developed the standard known as Bluetooth—named after King Harald Bluetooth, who united the countries of Scandinavia in 970 AD. The name was apt given that Bluetooth aimed to "unite" cell phones, accessories, and laptops. Furthermore, Ericsson and Nokia are based in Scandinavian countries (Sweden and Finland, respectively).

The first Bluetooth v1.0 standard was released in 1999 and became the IEEE standard 802.15.1 in 2002, but it didn't live up to its namesake. Manufacturers of small, very-low-power devices (including mice, heart rate monitors, and home security sensors) often adopted other, sometimes custom, wireless technologies or alternate standards, such as ZigBee.

The most recent revision, Bluetooth v4.0 (www.bluetooth.org), is fully compatible with classic Bluetooth but is extended by the new Bluetooth LE or "Smart" protocol. This extension provides an even lower-power interface than the classic Bluetooth and is optimized to communicate with very-low-power devices, such as wireless sensors. The LE extensions were inspired by the need to support sensors in home and office environments, including motion sensors, light detectors, thermostats, pedometers, and heart monitors.

The design's power target was to run a wireless sensor for at least one year on a single coin cell (approximately 200 mAHr) and to communicate with it using a laptop or cell phone at a data rate suited to these applications. In fact, Bluetooth LE is limited to approximately 200 kbps but is adequate to support common sensor applications that require only a small amount of data to be transmitted periodically, perhaps once per second or minute, depending on the situation.

LE WIRELESS TECHNOLOGY

LE was designed to exploit the existing 2.4GHz Bluetooth radio band to reduce cost and complexity in existing radio solutions. The transmit output power is variable up to 10 mW, with a maximum device current of approximately 15 mA at 3 volts (the sleep current is approximately 1uA). Instead of using 79 frequency-hopped channels, as in Classic Bluetooth, LE uses a larger modulation index with 40 channels on a 2 MHz spacing, and of these, three channels are used for advertising: 37, 38, and 39. Many of the LE interactions use an asynchronous connection-less MAC protocol, in which most transactions occur within three milliseconds. Packet structures are uniform, and only a few packet types are needed.

Published by the IEEE CS ■ 1536-1268/13/$31.00

Advertising Data

Unlike Classic Bluetooth, there's no lengthy discovery process to find nearby devices. Instead, *peripheral devices* can broadcast on the three advertising channels by sending an ADV_IND packet. Such packet transmissions are kept short, with only 31 bytes available for application payload data, in addition to the necessary preamble, packet header, MAC address, and checksum fields (Figure 1).

The structure of the data payload is a series of parameters, defined by the Bluetooth SIG, and includes a manufacturer-specific data field. The simplest use of ADV_IND packets is to broadcast a sensor value once per second in the manufacturer specific data field, and then sleep between transmissions. If 31 bytes isn't enough for an application, there are two other ways to use the protocol providing more flexibility.

The first is a simple mechanism that supports active scanning. If an ADV_IND packet is transmitted with the appropriate flag set, it indicates that it supports active scanning. This tells a host, or *central device* in LE terminology, that more data is available, and it can transmit a scan request packet (SCAN_REQ) to the peripheral after the ADV_IND packet is received. In response, the peripheral device will send back a SCAN_RSP packet which contains another 31 bytes of application data. The request/response pair is effectively a lightweight remote procedure call returning data to the host. Using this mechanism, up to 62 bytes of data can be transferred to the host.

However, in the general case, applications might need bidirectional and unconstrained data transfer. Using a lightweight client-server mechanism, enabled by the Generic Access Profile (GAP) profile, can offer such data transfers. Furthermore, GAP supports predefined, or user defined, Generic Attribute (GATT) services.

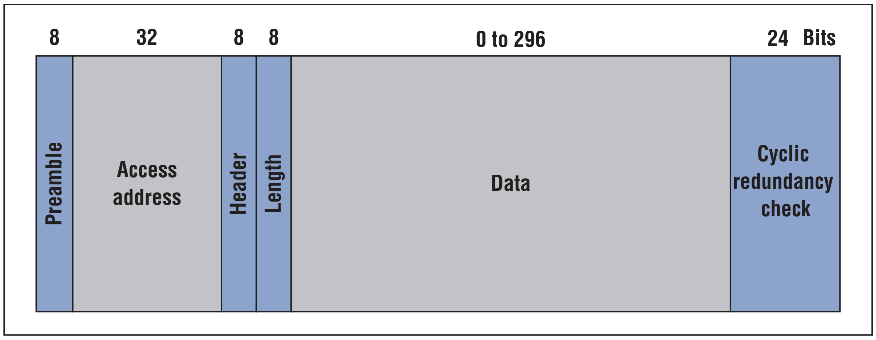

Figure 1. An example Bluetooth LE packet format. Packet transmissions are kept short, with only 31 bytes available for application payload data, in addition to the necessary preamble, packet header, mac address, and checksum fields.

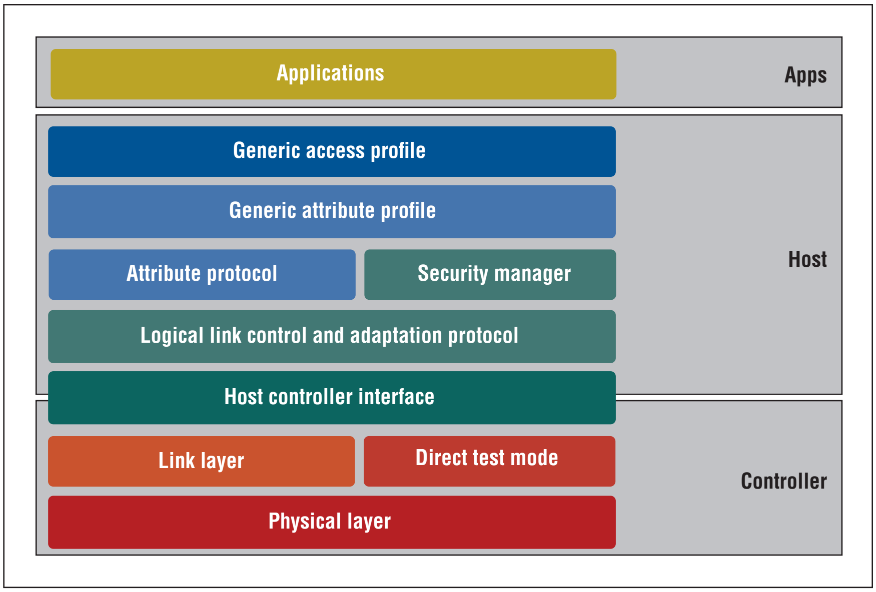

Figure 2. The Bluetooth Low Energy (LE) stack.

GAP and GATT Services

A GAP device can support one or more GATT services, and a host device can connect to these services to read and write attributes using the Attribute protocol (ATT). The Bluetooth SIG defines some of these services represented by a 16-bit address format, but manufacturers can also define their own services using a 128-bit universally unique identifier. A summary of services that a peripheral device supports can be included in the device's ADV_IND packet, providing a hint of its functionality to a host device that's considering whether to establish a connection.

However, because the ADV packets have limited space, in practice, the service list will be a subset of what's actually available. The full list of services can be read by connecting to the GATT layer (see Figure 2) and requesting all of the service identifiers and their corresponding attributes. Attributes are defined as *characteristics* of the service, where each one has a value and a set of descriptors (see Figure 3). A characteristic descriptor provides type information, such as the value's format and how it's represented—in integer or Boolean format, for example. In this way, a host can read the descriptors from a peripheral and write values that control its operation.

Operations on a value might be linked to additional components, such as physical actuators though

SMARTPHONES

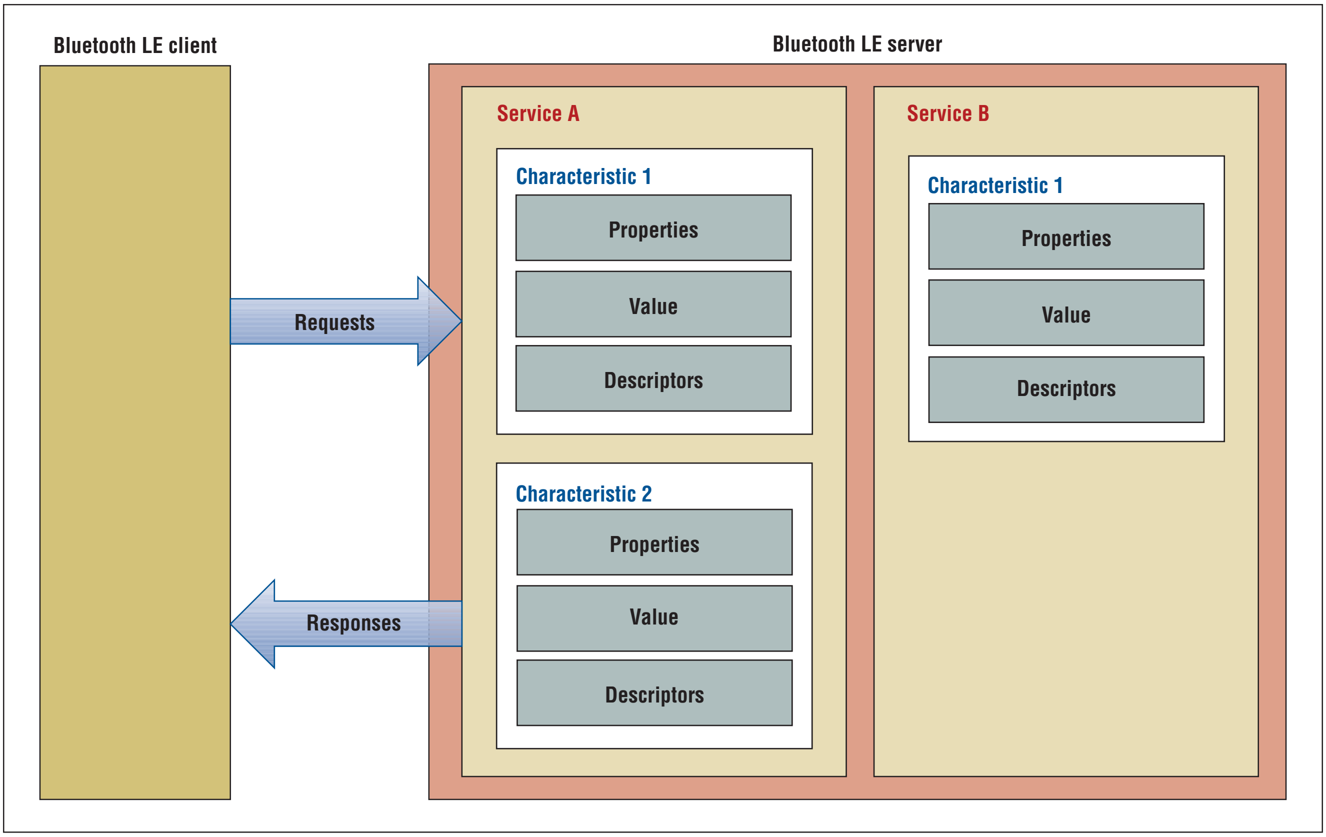

Figure 3. The Bluetooth LE Attribute protocol, implemented by Generic Attribute (GATT) services containing characteristics, each with an attribute and corresponding value.

bit-mapped input/output, to produce a physical result when written. Similarly, a value can be read from a characteristic that reports the current state of a real-time sensor. The simplicity of providing a general-purpose interface enables reading and writing of attribute values, with low overhead and implementation cost, which lowers the barrier for innovation in this area.

Peripheral LE Solutions

Although most mobile device Bluetooth solutions support LE, they also provide the full classic Bluetooth functionality. When designing a tiny, wireless, battery-operated sensor, only the LE wireless functions are needed, providing an opportunity to reduce power consumption, silicon size, and device cost. Today, several commercial semiconductor solutions provide just the Bluetooth LE capability in a form that supports a wide variety of embedded applications.

Some popular LE peripheral solutions include the Texas Instruments CC2450,[2] based on an 8051 internal microcontroller, and the Nordic nRF51822,[3] based on an embedded ARM core. Both companies provide a wealth of supporting materials, including developer kits, prototype sensors and tags, protocol analyzers, and example code for common LE applications.

There are also numerous third-party vendors that are integrating these chips into their own module, sold as a complete Bluetooth LE solution—such as the BlueGiga module and development kit (see Figure 4). These devices encapsulate the RF design, which includes the radio, ground-plane, antenna, shielding, and interface components. This approach tends to be more expensive than engaging in a completely custom solution, but has some benefits when integrating it into a new product, because the module has already passed FCC-compliance testing requirements.

SMARTPHONE PLATFORM SUPPORT

Most smartphones and laptops designed in the last two years include a Bluetooth v4.0 chipset that integrates the Bluetooth LE hardware. As with many innovations, it takes time for a platform operating system to incorporate the necessary drivers and APIs. The iOS operating system was the first mobile OS to do this, and, as a result, many products based on Bluetooth LE only operate in conjunction with an iPhone app.[4]

At the Google-IO developers forum in San Francisco in May 2013, Android support for Bluetooth LE was announced in the Android 4.3 release of JellyBean.[5] The Microsoft Windows phone v8.1 OS release is also expected to have Bluetooth LE support in the near future. Now that the major phone operating systems support LE, and with others following soon, there's likely to be an explosion of new applications and products in this area.

Application Categories

Many types of applications can use this potentially ubiquitous low-power technology, including those for

- environmental control (monitoring heating, ventilation, air conditioning, humidity, and light);
- security (sensing motion or window and door movement);
- health monitoring (tracking blood pressure, heart rate, pulse, blood oxygen, and posture);
- sports and activity monitoring (identifying when a person is running, bicycling, walking, or sitting, or counting steps while hiking);
- proximity detection (locating devices and sending alerts when devices are out of range);
- user interface (UI) extensions (showing phone alerts on a smart watch); and
- power conservation (monitoring power for smart devices).

Other applications include using Bluetooth LE to transfer credentials between devices that enable higher-bandwidth connections. Similar to near-field communication, the Bluetooth LE protocols can bootstrap the connection between other wireless technologies that rely on shared credentials to ensure a channel is secure. Because LE can be constrained to operate over a short range, reducing the radio's transmit power can limit the distribution of those credentials to devices in the immediate vicinity.

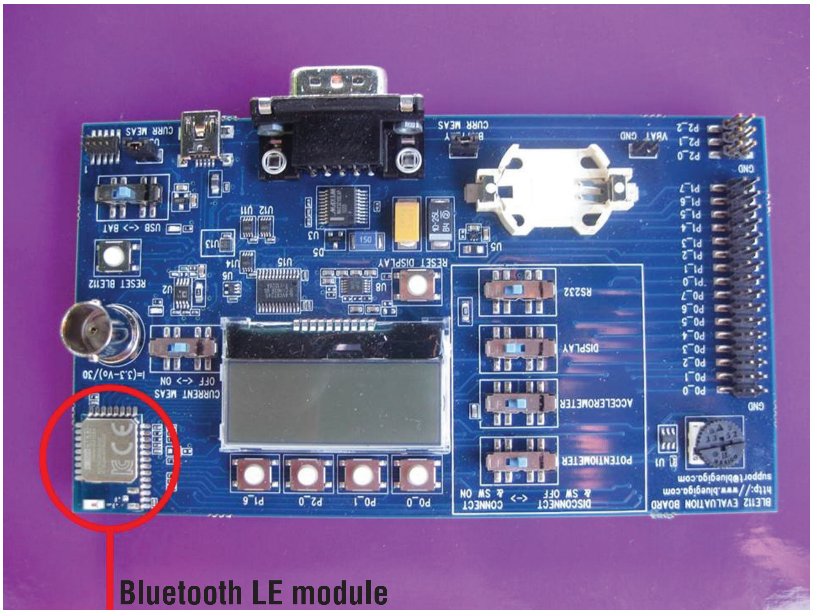

Figure 4. A BlueGiga Bluetooth LE development kit. The Bluetooth LE module is incorporated into the development board and could become the core component of future products.

Example Products

One of the most prolific types of Bluetooth LE devices marketed to date has been "finder" technologies for device tagging. These small, lightweight tags can be attached to things, such as keys, computers, bags, and books, in a nonintrusive way, and located with a mobile device (or smartphone) application when lost. Products in this space include Proximo (www.kensington.com), Tile (www.thetileapp.com), and StickNFind (www.sticknfind.com), each with a small form factor solution well suited to this application (see Figure 5).

A smartphone application will typically locate a tag by showing a graphic depicting what tags are in the vicinity and indicating their range, which is estimated based on a combination of the advertised transmission signal strength and the signal strength measured by the receiver. If a device isn't located, the user can walk around trying to pick up the signal or can listen for a sound alert made by the app when detected. The application can also provide information about when and where the device was last detected. This is particularly useful when you've lost your keys but still have your smartphone handy.

The Internet of Things

As long ago as 1996, researchers at Xerox PARC were experimenting with systems for bridging the physical world with the virtual world of the Internet using RFID tags. This potential evolution of the World Wide Web would let computers be aware of not only the virtual digital objects in their file servers but also objects, people, and things in the physical world. In other words, a future Internet search could make a reference to an object in the real world, possibly letting you control and probe its state.

Around 1999, the "Internet of Things" was coined as a popular term to describe this concept, and many white papers were written about its potential. There's even a related conference (www.china-iot.net/iThings2013.htm), which started in 2008. However, despite all the work and interest in this area, we haven't seen the technology take off and become adopted in mainstream work practices. The reasons for this might be subtle, but it's probably an indication that the key ingredient—the catalyst for adoption—hasn't been invented.

Although RFID has been in existence for over 40 years, common standards, low-cost tags, and a widely deployed reader system are still missing. The

SMARTPHONES

Figure 5. A proximity solution: (a) a smartphone app with (b) a locator fob and (c) the Kensington Proximo LE tag.

smartphone industry has made some steps toward solving this problem with the recent adoption of NFC readers (a subset of the RFID standards) in a few phone models. However, given that the majority of smartphones already have Bluetooth LE radio technology built in, the emergence of the Bluetooth LE standard might present the first practical opportunity to building out the IoT.

Sensor Networks

Similar to the IoT, sensor networks have held the promise of microscale monitoring of our environment, encompassing pollution control, structural monitoring (seismic and corrosion), and better weather prediction. Such networks might also monitor energy usage, detect energy waste, and improve security. The key advantage of such networks is that data is sampled at a high spatial resolution, far higher than would be possible with conventional wired solutions.

One reason why the technology has had minimal adoption is because of the costs associated with maintaining numerous sensors. Even supporting their power requirements has been an issue—changing batteries on this scale is labor intensive, and scavenging environmental energy isn't always practical.

The low energy requirements of Bluetooth LE might help reduce the financial burden for sensor networks, either because battery replacement is needed far less frequently, or because a sensor battery lasts long enough for the entire module to be expendable.

Finally, as with the IoT example, the millions of smartphones in the world act as a platform that can link data captured by Bluetooth LE sensors to cell towers with wired Internet connections. Thus, the cost of readers, or Internet gateway devices, is dramatically reduced by using the serendipity of nearby smartphones.

As with many early ideas—including early visions of e-books, tablet computers, and RFID—the underlying technology and supporting ecosystem must mature to the point where the value proposition outweighs the investment, not only in terms of investment costs but also user time, overhead, usability, and maintenance. Bluetooth LE might be the missing link, enabling pervasive adoption of the IoT and sensor networks. However, it will be a few years of development before we really know if Bluetooth LE is the critical ingredient to make these visions a reality. P

REFERENCES

1. "CC2450 Data Sheet," Texas Instruments, 2013; www.ti.com/lit/ds/symlink/cc2540.pdf.
2. "nRF51822 Data Sheet," Nordic, 2013; www.nordicsemi.com/eng/Products/Bluetooth-R-low-energy/nRF51822.
3. "Core Bluetooth Framework Reference," iOS Developer Library, 2013; http://developer.apple.com/library/ios/documentation/CoreBluetooth/Reference/CoreBluetooth_Framework/_index.html.
4. "Bluetooth Low Energy," Android Developers, 2013; http://developer.android.com/guide/topics/connectivity/bluetooth-le.html.

Roy Want is a research scientist at Google. Contact him at roywant@acm.org.

Bill Schilit is a research scientist at Google. Contact him at schilit@computer.org.

Dominik Laskowski is a computer engineering student in his final year of a Bachelor of Engineering at McGill University. Contact him at dominik@laskowski.com.