RC 22109 (W0107-009) 6 June 2001 **Computer Science**

IBM Research Report

An Overview of the Bluetooth Wireless Technology

Chatschik Bisdikian
IBM Research Division
Thomas J. Watson Research Center
P.O. Box 704
Yorktown Heights, NY 10598

Research Division
Almaden - Austin - Beijing - Haifa - India - T. J. Watson - Tokyo - Zurich

An Overview of the Bluetooth Wireless Technology

Chatschik Bisdikian
IBM Corporation
Thomas J. Watson Research Center
30 Saw Mill River Road
Hawthorne, NY 10532
bisdik@us.ibm.com

Abstract:

The Bluetooth wireless technology is designed as a short-range connectivity solution for personal, portable and handheld electronic devices. Since, May 1998 the Bluetooth SIG steers the development of the technology through the development of an open industry specification, including both protocols and application scenarios, and a qualification program designed to assure end-user value for Bluetooth products. This article highlights the Bluetooth wireless technology.[1]

[1] Any opinions expressed in this article represent only the personal opinions of the author and do not reflect a position of the author's employer or any other entity's.

1 INTRODUCTION

For the last few years, the wireless world has been bombarded daily about a new generation of radio frequency (RF) technologies that would have a profound impact, if not revolutionarize, the way we live and contact our businesses. This new generation of technologies spans across the full spectrum of wireless communications coverage. Third generation (3G) wireless technologies are being developed to enable personal, high-speed interactive connectivity to wide area networks (WANs). The IEEE 802.11b wireless local area network (LAN) technology, also referred to as Wi-Fi, finds itself with an increasing rate in corporate and academic office spaces, buildings, and campuses. Furthermore, with slow, but steady rate, the 802.11b technology makes inroads in public areas like airports and coffee bars.

WAN and LAN technologies enable device connectivity to infrastructure based services, either through a wireless carrier provider or through a campus or corporate backbone intranet. The other end of the coverage spectrum is occupied by the short-range personal wireless connectivity technologies allowing personal devices to communicate with each other directly without the need of an established infrastructure. The infrared (IR) technology is a widespread example of such a personal connectivity solution. Moving a step further in functionality, the development of the Bluetooth? wireless technology brings the benefits of the omni-directionality and absence line of sight requirement of RF-based connectivity in the personal connectivity space. The personal connectivity space resembles a communications bubble follows people around that empower them to connect their personal devices with other devices that enter the bubble. Connectivity in this bubble is spontaneous and ephemeral and can involve several devices of diverse computing capabilities; this is unlike wireless LAN solutions that are designed for communication of devices of sufficient computing power as well as battery capabilities.

The Bluetooth[2] wireless technology[3] will serve primarily as a replacement of the interconnect cables between a variety of personal devices, including notebook computers, cellular phones, personal digital assistants (PDAs), digital cameras, etc. The Bluetooth wireless technology aims in serving as the universal low cost, user friendly, air-interface that will replace the plethora of proprietary cables that people need to carry and use to connect their personal devices. While, typically, personal devices communicate based on the RS-232 serial port protocol, proprietary connectors and pin arrangements make it impossible to use the same set of cables to interconnect devices from different manufactures, and, some times, even from the same manufacturer. The primary focus of the Bluetooth wireless technology is to provide a flexible cable connector with reconfigurable pin arrangements permitting several personal devices to interconnect with

[2] Bluetooth is a trademark owned by the Bluetooth SIG, Inc., USA.

[3] According to the Bluetooth brand requirements document, the term “Bluetooth” must always be used as an adjective. Furthermore, when the term “Bluetooth” is used to denote the corresponding technology, the term “wireless” must be inserted between Bluetooth and technology, as in “Bluetooth wireless technology”. The author recognizes that the above rules are not always followed and the term “Bluetooth” has grown to represent both the technology and the whole industry behind it.

each other. Thus, a wireless headset will connect to a cellular phone with the same ease that it will connect to a notebook computer, no matter who manufactures the headset or the cellular phone or the notebook computer. Similarly, one will be able to connect computer peripherals ranging from keyboards and mice to printers and office appliances without the worry of having available and utilizing the right interconnect cables.

Another focus of the technology is to enable a uniform interface to data access points for accessing data services. A user using any number of data capable devices will be able to connect to a LAN access point that provides access to, say, the corporate intranet infrastructure and services. Likewise, the user will be able to connect to her cellular phone and access WAN data services. Applications can then be written that could provide the user with a similar connectivity experience connecting to data service in either manner. Connecting to data services through one's cellular phone gives rise to the concept of a *personal gateway*. People will carry their personal gateways anyway they go. The personal gateway will serve as a facilitator in accessing remote data services, with the added convenience that it can be kept hidden, away from the line of sight of its communicating Bluetooth partner. The Bluetooth wireless technology enables the unobtrusive separation of the functionality of connecting to a data service from viewing and interacting with the information provided by the data service. Thus, a PDA can be used as a more convenient I/O device for entering and receiving data, while using the personal gateway purely for communicating with the wireless data carrier.

Yet another focus item for the Bluetooth wireless technology is to enable ad hoc connectivity among personal devices. This will permit individuals to form collaborative groups, for example, during a conference meeting, to exchange data without the need to rely on an infrastructure to support their communication.

The above focus points have shaped the Bluetooth wireless technology from its outset. Actually, the above focus points have set the development of this technology apart from other wireless efforts. To guarantee that users will benefit from the technology from the outset, the Bluetooth specification relates with a set of simple yet useful usage scenarios. In particular, the specification defines the methods by which these usage scenarios should be implemented in various Bluetooth devices. The definitions of these methods, referred to as *profiles*, provide the guidelines according to which a selected set of applications enabled by the Bluetooth wireless technology can be developed in an interoperable fashion. We will discuss about the Bluetooth specification and its profiles in later chapters.

Incidentally, the name Bluetooth comes from the Danish king *Harald Blåtand* (Bluetooth). King Bluetooth is credited for uniting the Scandinavian people during the 10^{th} century. Similarly, the Bluetooth wireless technology aims in uniting personal computing devices. The name was chosen temporarily to describe the yet unannounced development project. However, the search for a new name never came to a successful fruition and the temporary name became permanent. In retrospect, the selection of this joyful name can be credited about as much as the potentials of the Bluetooth wireless

technology it represents for the recognition and acceptance the technology has received so far.

This paper is organized as follows. In chapter 2……

2 THE HISTORY OF THE BLUETOOTH WIRELESS TECHNOLOGY

During the first half of 1990s, radio engineers had come to the realization that the technology trends are such that low cost, low power, short-range radios will be possible in the near future. These low cost radios will serve as cable replacements between personal devices and primarily cellular phones and other personal data terminals like computer notebooks. A team of engineers from Ericsson, embarked in developing such a radio technology.

However, for these radios to be of value to the cellular phones that they will attach to, other personal communicating companion devices are needed that will be using the same radio technology and the same upper layer communications protocols to communicate with. In other words, for this new technology to be successful, multiple devices from multiple manufacturers need to incorporate it and interoperate with each other. Therefore, a widely acceptable standard needs to be developed for this technology. The need for an industry standard is further underscored by the strong relation of this technology with consumer products and the requirement of the consumer market for low cost and ease to use, interoperable devices.

The development of the Bluetooth industry standard started late in the winter of 1998 when Ericsson, IBM, Intel, Nokia, and Toshiba formed the Bluetooth Special Industry Group (SIG) to develop and promote a global solution for short-range wireless communication operating in the unlicensed 2.4 GHz ISM (industrial, scientific, medical) band. The SIG comprises several working groups including a program management group, which currently comprises a board of directors, that sets-up the directions and oversees the activities of the SIG. There is a marketing team that not only orchestrated the promotion of the technology in the market, but guided and steered its development to satisfy important usage scenarios. There are technical working groups that are responsible for the development and maintenance of the Bluetooth industry specification. There is a regulatory group to interface with authorities responsible with regulating the usage of radio frequencies in various countries. There is a legal group that deals, among other things, with issues related to the management of the IP (intellectual property) associated with the technology, and a number of other groups.

To facilitate the wide acceptance of this new technology, the SIG decided to avail all the technologies explicitly included in the Bluetooth specification royalty-free to adopter members of the technology. In other words, any adopter of the Bluetooth wireless technology can build Bluetooth components and products sharing for free with other

adopter companies any IP (intellectual property) included in the Bluetooth specification.[4] The SIG announced its existence and intentions to the public in May 1998 joined, at the time, by about 70 adopter members; as of this writing there are about 2,500 adopter members. A little over a year later, in the summer of 1999, the over 1,600-page Bluetooth specification version 1.0A became publicly available. The Bluetooth specification was developed by the five promoter (founding) members with substantial contributions from two additional companies, 3Com and Motorola. Adopter members have the opportunity to examine early versions of the Bluetooth specifications and provide feedback as needed. Due to the Bluetooth SIG license agreement, the development of the specification is not made available to the general public until it is finished and approved by the Bluetooth SIG. Adopter members have the privilege to look at the specification prior to its public availability.

The Bluetooth specification ver. 1.0A comprised the following two parts, which we elaborate more later in the paper:

? The core specification defining the radio characteristics and the communication protocols for exchanging data between devices over Bluetooth radio links.

? The profile specification that defines how the Bluetooth protocols are to be used to realize a number of selected applications.

In December 1999, the promoter group increased from five to nine with the addition of: 3Com, Lucent, Microsoft, and Motorola. As of early 2001, Agere, a Lucent spin-off comprising its former microelectronics division, has taken the place of Lucent in the promoters group.

As developers started to build early prototypes and later on products based on the specification, the SIG started collecting comments and suggestions from the developers that were requesting clarifications on the specification. The SIG released additional versions of the specification incorporating, in the form of errata, several of the comments received directly from the developers or indirectly though a series of informal interoperability testing gatherings, referred to as *unplug fests.* During an unplug fest, developers have the opportunity to test their Bluetooth implementations against those of other developers and iron out possible discrepancies among them.

Table 1 highlights some key dates in the history of the Bluetooth wireless technology and its specification.

Mid '90s	Early seeds of the Bluetooth radio technology.
Feb'98	The Bluetooth SIG is formed by the five promoter companies.

[4] Because there are currently several generations of the Bluetooth specification, there are several generations of adopters agreements as well. Thus, certain companies that had signed only the early adopters agreement(s) have free access only to the IP in the early generation(s) of the specification as well.

March'98	First face-to-face technical meeting of the software task force. The "hardware" task force, called the air group, has been meeting prior to the formation of the SIG.
May'98	The SIG announced publicly its existence and its objectives.
Oct'98	The first Bluetooth Developers Conference takes place in Atlanta, GA, USA.
July'99	The Bluetooth spec. ver. 1.0A is release to the general public.
July'99	The Bluetooth SIG submits a proposal to the newly formed IEEE 802.15 working group on wireless personal area networks.
Dec'99	The promoter group increases to nine.
Dec'99	The Bluetooth spec. ver. 1.0B is release to the general public.
Jan'00	New working groups start to be formed to develop additional Bluetooth specifications
Feb'01	The Bluetooth SIG is incorporated in the Bluetooth SIG, Inc.
Feb'01	The Bluetooth spec. ver. 1.1 is release to the general public.

Table 1: *A Bluetooth chronology*

As mentioned earlier, the Bluetooth specification comprises over 1,600 pages of protocol and application specifications. It contains a *core* specification part, which comprises the communications protocols, the testing protocols and procedures, and the compliance requirements. It also contains a *profile* specification part that comprises application and protocol guidelines for building a selected set of interoperable applications. In the following two chapters we present these two parts of the Bluetooth specification, starting with the core specification.

Note that the Bluetooth specification has been written primarily as an implementation manual rather than a formal communications standard document. This aspect of the specification reflects its development process by a group of engineers that actually developed the technology in parallel to the development of the specification. These engineers expressed in the specification their experiences stemmed by their implementations in a prose rather than using strict language formalisms commonly found in a formally developed standard. This approach had its pros and cons. On the upside, the specification is easier to read that a formal standards document. On the downside, using prose, which is naturally imprecise, the specification is open to conflicting interpretation, which are resolved through an errata process.

3 THE CORE SPECIFICATION

The Bluetooth core specification contains both a hardware and a software description. The former pertains to the lowest layers of the protocol stack, like the radio and the baseband, while the latter pertains to higher layers that are typically executed by dedicated microprocessors and/or the processor of a host device.

Figure 1 depicts the Bluetooth protocol stack, which also shows the application and profiles "layer" for completeness. The protocols in the stack have been grouped in two categories: the *transport* and the *middleware* protocols. The transport protocols comprise protocols developed exclusively for the Bluetooth wireless technology. These protocols are involved in all data communications between two Bluetooth devices. The middleware protocols comprise both Bluetooth specific protocols and other adopted protocols. These protocols are used selectively to enable different applications, including both legacy and new applications, to exchange data using the Bluetooth wireless technology. Whenever desired, the middleware protocols shield these applications from the specifics of the Bluetooth transport protocols.

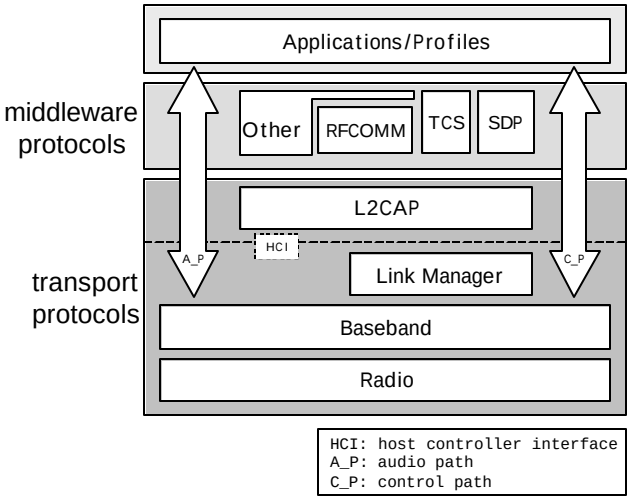

Figure 1: *The Bluetooth protocol stack*

This grouping of the protocols in the Bluetooth protocol stack is not part of the specification. Rather, it is used here as a natural grouping of the protocols for ease of presentation.

3.1 The transport protocols

3.1.1 The radio

The radio layer defines the technical characteristics of the Bluetooth radios. A Bluetooth radio operates on the license-free 2.4 GHz ISM band and is compliant to FCC part 15 regulations for intentional radiators in this band. It employs a fast (1,600 hops/sec), frequency hoping, spread-spectrum (FHSS) technique. The radio hops in a pseudo-random

fashion on 79 one-megahertz channels.[5] The frequencies are located at (2,402+*k*) MHz, *k*=0, 1, …, 78.

The modulation technique is a binary Gaussian frequency shift-keying (GFSK) and the baud rate is 1 Msymbols/sec. Hence, the bit time is 1 ?sec and the raw transmission speed is 1 Mb/sec. The Bluetooth radios come in three power classes depending on their transmit power. Class 1 radios have transmit power of 20 dBm (100 mW); class 2 radios have transmit power of 4 dBm (2.5 mW); and class 3 radios have transmit power of only 0 dBm (1 mW). Due to the power and cost constraints of the various personal devices to use Bluetooth radios, the 0 dBm radios are expected to be the ones mostly used in these devices.

3.1.2 The baseband

The baseband defines the key procedures that enable devices to communicate with each other using the Bluetooth wireless technology. The baseband defines the Bluetooth piconets and how they are created, and the Bluetooth links. It also defines how the transmit resources are to be shared among several devices in a piconent as well as the low-level packet types.

3.1.2.1 The Bluetooth address and clock

Each Bluetooth device has two parameters that are involved in practically all aspects of Bluetooth communications. The first one is a unique IEEE-type 48-bit address assigned to each Bluetooth radio at manufacture time. The *Bluetooth device address* (*BD_ADDR*) is engraved on the Bluetooth hardware and it cannot be modified. The second parameter is a free running 28-bit clock that ticks once every 312.5 ?sec, which corresponds to half the residence time in a frequency when the radio hops at the nominal rate of 1,600 hops/sec.

Bluetooth devices can communicate with each other, by acquiring each other's Bluetooth addresses and clocks, as will further be described in the sequel.

3.1.2.2 The Bluetooth piconet

A piconet is a collection of Bluetooth devices that can communicate with each other. A piconet is formed in an ad hoc manner without any infrastructure assistance and it last for as long as the creator of it needs and is available to communicate with other devices. A piconet contains at least one device identified as the *master* of the piconet and at most 7 other devices identified as *slaves* with which the master is *actively* involved in communications. The terms master and slave are relative to a particular existing piconet. The terms are not assigned to the radio units at manufacture time. A Bluetooth radio may serve either as a master or slave at different times.

[5] The specification permits a reduced channel hop over only 23 channels for countries that have restrictions in their corresponding ISM band.

To identify each slave, the master of a piconet assigns the slaves participating in active communications in the piconet a locally unique *active member address* (*AM_ADDR*). The master regulates and controls who and when transmits. While up to 7 slaves may be actively communicating in a piconet at a time, additional devices may be registered with the master and be invited to become active whenever necessary. These additional devices are called *parked*. Bluetooth devices not associated with any piconet are in *stand-by* mode. Figure DDD shows two piconets with a number of slaves and parked devices associated with them, and a few stand-by devices. Bluetooth piconets can coexist in time and space independently of each other. Furthermore a single device may be a member of several piconets, a case refer to as *scatternet* in the Bluetooth parlance.

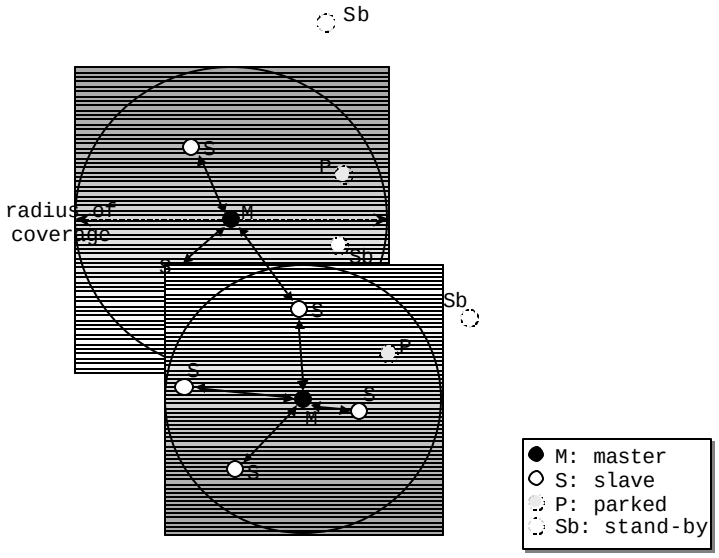

Figure DDD: Bluetooth piconets

The communications channel in a piconet is defined as the sequence of the frequency hops followed by the piconet members in a synchronized manner. The transmit and receive time axis are slotted, with each slot lasting the duration of a nominal frequency hop, 625 ?sec. Each baseband transmission resides fully within the boundaries of a slot. However, multi-slot packets occupying three or five slots, instead, are also allowed. During the transmission of a multi-slot packet, the transmit frequency does not change. When hoping resumes, it resumes with the frequency whose turn would have been if the devices were to use only single-slot transmissions.

To maintain time synchronization for the hops, slaves utilize the Bluetooth clock of the master and the fact that hops occur in multiples of 625 ?sec; slaves actually maintain the offset time between their Bluetooth clock and that of their master. Slots in a piconet are identified as even or odd according to the value of the second least significant bit of the

Bluetooth clock of the master; recall that the Bluetooth clock ticks at a rate twice that of the slot rate. To recreate the frequency hop sequence in a piconet, a slave utilizes the Bluetooth address of the master of the piconet. Furthermore, the Bluetooth clock of the master identifies the particular frequency to be used at a particular slot. Therefore, the communications channel in a piconet is fully identified by the master. As a result, in the case of scatternets, a device can serve as a master for only one piconet for otherwise the two piconets cannot be distinguished from each other.

The master and the slaves alternate transmit opportunities in a *time-division duplex* (TDD) fashion. In particular, the master transmits on even numbered slots, as defined by the master's Bluetooth clock, while the slaves transmit on odd numbered slots; recall that each slot lasts 625 ?sec. A slave can transmit only if the master has just transmitted to this slave. A transmission may last one, three, or five slots; however, the specification requires that only the one-slot transmissions be mandatory. In the case of scatternets, a device cannot receive or transmit data simultaneously in two or more piconets. However, such a device may time-share its participation in each piconet over non-overlapping time intervals.

To engage in communications in a piconet, the slaves in the piconet need to know the *BD_ADDR* and Bluetooth clock of the master. Likewise the master needs to know the identities of the slaves. How this is done, i.e., how a piconet is formed is highlighted next.

3.1.2.3 The piconet creation

The creation of a piconet is a two-step process, depending on whether the master of the piconet knows the identity, *BD_ADDR*, of the device it wants to communicate with. This process comprises an inquiry phase, for locating devices, and a paging phase, for inviting specific devices to join a piconet.

The *inquiry* process is a device discovery process during which the master of a future piconet discovers other devices in its vicinity. The master makes its presence know by transmitting inquiry messages. Devices that perform inquiry scan, i.e., search actively for inquiry messages, respond with inquiry response messages that, among other things, contain the *BD_ADDR* of the device.

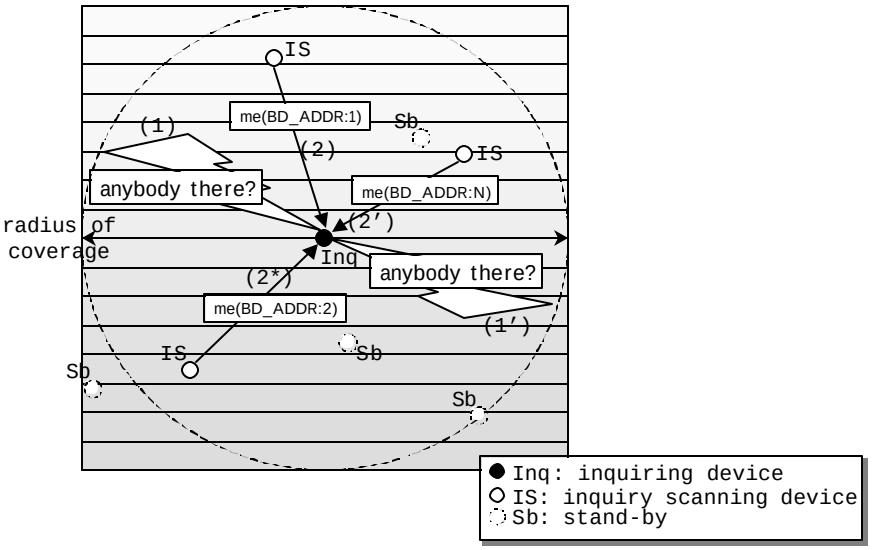

Figure hjk: *The inquiry steps*

Figure hjk outlines the inquiry process. An inquiring device (Inq) transmits inquiry messages: transmissions (1), (1'), etc. These transmissions are made over a well-defined periodic inquiry frequency hoping sequence with period 32 hops. The inquiry frequency hoping sequence is reserved just for the inquiry process. Devices, IS, may actively search for inquiries by scanning for transmissions over the inquiry frequency hoping sequence. Following a receipt of an inquiry message, an inquiry scanning device returns an inquiry response message containing, among other things, the device's *BD_ADDR* and Bluetooth clock value: transmissions (2), (2'), etc. To avoid collisions due to simultaneous responses to an inquiry message by multiple devices, responding devices do not respond immediately, but rather wait a random number of slots and then respond after they receive a second inquiry message.

Since devices operate without any co-ordination, to increase the probability that an inquiring device transmits during the time that an inquiry scanning device listens, the former transmits over two successive frequencies of the inquiry hopping sequence every transmit slot, while the latter listens to a different frequency every 1.28 sec.

Armed with the knowledge of the identity of devices in its vicinity, the master of a piconet may explicitly *page* devices to join its piconet. A master with prior knowledge of the identity of a device, it may skip the inquiry process and go directly into paging the device. If the device does not respond, it may mean that it is not in the transmit range of the paging device.

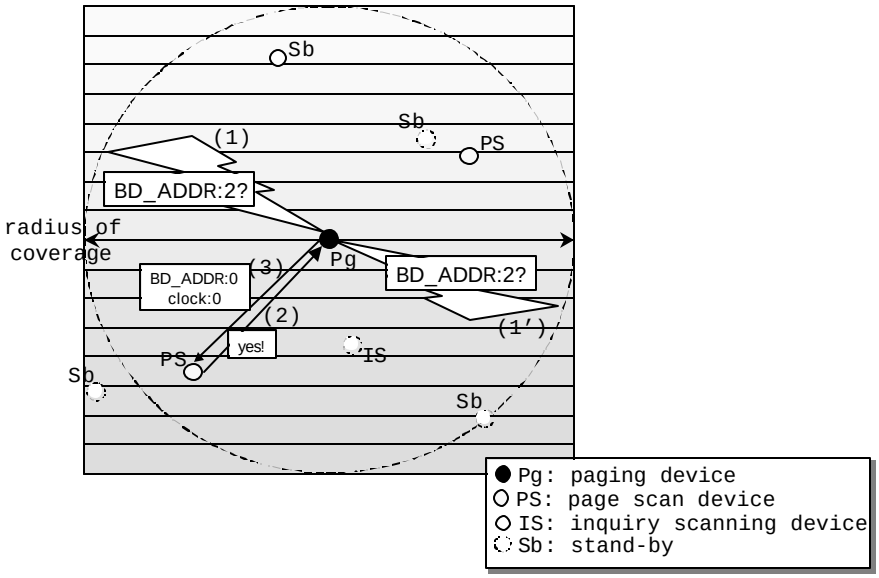

Figure vbn: *The page steps.*

Figure vbn outlines the page process. A paging device (Pg) transmits paging messages to the paged device: transmissions (1), (1'), etc. These transmissions are made over a well-defined periodic page frequency hoping sequence with period 32 hops. The page frequency hoping sequence is defined by the *BD_ADDR* of the paged device. The paged device, PS, may actively search for pages by scanning for transmissions over the paging frequency hoping sequence. Following a receipt of a page message, a page scanning device acknowledges the paging device with a page response message: transmission 2. Subsequently, the paging device will provide the paged device with its *BD_ADDR* and Bluetooth clock value: transmission (3).

Similarly to the inquiry process, to increase the probability that a paging device transmits during the time the paged device listens for pages, the former transmits over two successive frequencies of the page hopping sequence every transmit slot, while the latter listens to a different frequency every 1.28 sec. The paging device may have an estimate of the value of the Bluetooth clock of the paged device either from a previous inquiry or active communication. Hence, the paging device may intelligently select an initial transmit frequency for the pages to maximize the likelihood that the paged device is listening to that frequency.

With the information sent by the paging device to the paged device, the paged device can now join as a slave the piconet whose master is the paging device. After joining the

piconet, the master and the slave may negotiate reversal of roles in which case, the (original) master becomes a slave in the piconet whose master will be the (original) slave.

Next we present how masters and slaves exchange data.

3.1.2.4 The Bluetooth links and baseband packets

There are two types of links supported in the Bluetooth piconet. Between a master and a slave there is a single *asynchronous connectionless* (ACL) link supported. Optionally, a piconet may support *synchronous connection-oriented* (SCO) links. Up to three SCO links may be supported in a piconet.

The ACL link is a best effort link appropriate for asynchronous data transmissions. It maintains integrity by using retransmissions and sequence members, as well as forward error correction (FEC) if necessary. The SCO link supports periodic audio transmissions at 64kbps, in each direction. SCO traffic is not retransmitted, but it can use FEC mechanisms to recover from transmission errors when they occur.

Figure lkj shows the various baseband packet types. They all contain an *access code* (AC) field which is used to distinguish transmissions in different piconets. With the exception of the ID packet, all other packets have the header portion, and with the exception of the poll and null packets, all other packets also have a payload section.

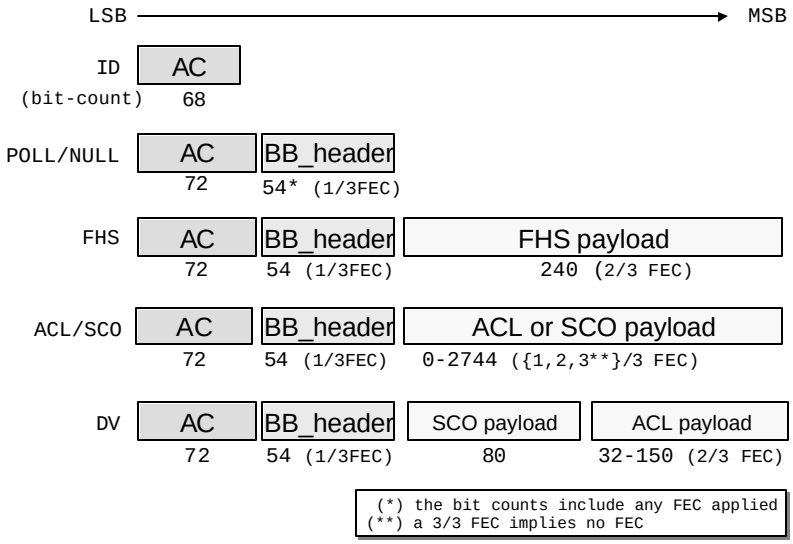

Figure lkj: *The baseband packet types*

The *poll* packet is used by the master to explicitly poll a slave when no payload information needs to be sent to the slave. The *null* packet is used to acknowledge a transmission when no payload information needs to be sent.

The *frequency hope sequence* (FHS) packet is used during the creation of a piconet and it is used to pass address (*BD_ADDR* and *AM_ADDR*) and clock information between future masters and slaves. The payload of an FHS packet is encoded with a shortened Hamming code with rate 2/3. The number of bits shown in the figure is after the application of the FEC.

The ACL or SCO packets carry asynchronous and synchronous data in their payload respectively. The payload of ACL packets may be encoded with an FEC with rate 2/3, or not encoded at all. The payload of SCO packets may be encoded with an FEC with rate 2/3 or 1/3, or not encoded at all. When the FEC with rate 1/3 is used, each bit is simply repeated three times. The *data voice* (DV) packet is a packet that contains both ACL and SCO data and is transmitted at the periodic instances of regular SCO packet, whenever there is a need to send ACL data to the recipient device of the SCO transmission.

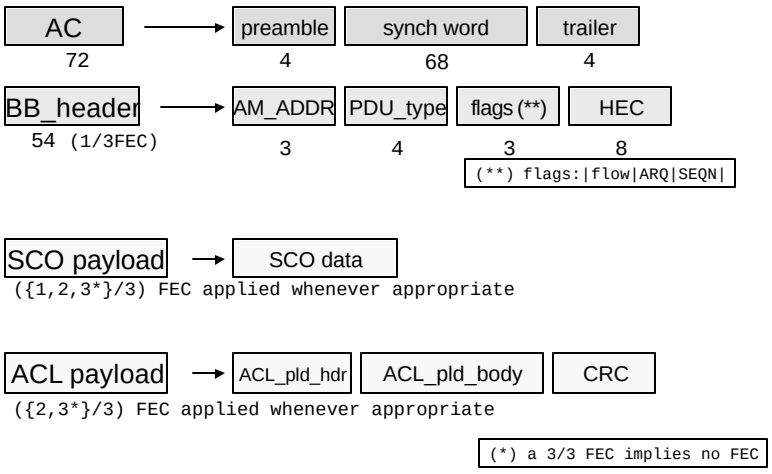

Figure BB1: *The baseband packet fields*

Figure BB1 depicts the fields in the header and the payload of ACL packet. The *AM_ADDR* field identifies the destination slave of a master transmission or the source slave of a slave transmission. The *PDU_type* field identifies the type of baseband packet as shown in Figure lkj. The flags are used for controlling the transmission and

retransmission of ACL packets. In particular, ACL packets use a stop-and-go ARQ scheme and a 1-bit sequence number. Furthermore, the ACL link is flow controlled. The header is protected by an 8-bit header error check (HEC) code. The ACL payload has its own header and body portion, see also figure qwe, and it is protected with a 16-bit cyclic redundancy check (CRC).

When *AM_ADDR* = b‘000’, then the packet is a broadcast packet from the master to all the slaves. Broadcast packets are not acknowledged and are not retransmitted.

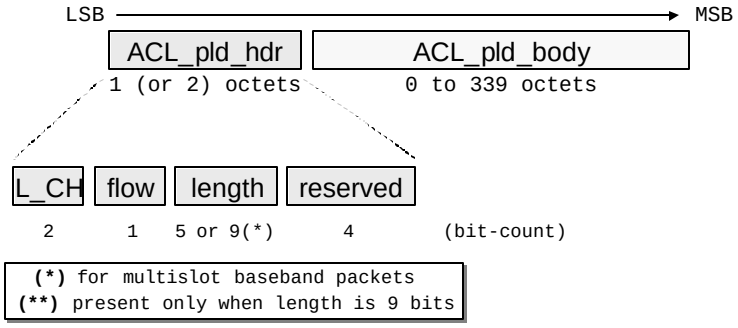

Figure qwe: *The ACL packet payload format*

The *L_CH* field in figure qwe is used to identify the logical channel for this baseband transmission. When *L_CH* = b‘11’, then the body of the ACL packet payload is passed to the link manager and is used for the configuration of the Bluetooth link. When *L_CH* = b‘01’ or b‘01’ then the body is passed to L2CAP for further processing.

Table 2 summarizes the effective payload rates achieved on a Bluetooth link. The notation DM stands for medium rate data, with a 2/3 FEC; the notation DH stands for high rate data with no FEC. In the symmetric case, two communicating devices, say, Dev_A and Dev_B, use the same exactly packet types to communicate. In the asymmetric case, the maximum effective throughput of Dev_A is shown in the forward direction (from Dev_A to Dev_B), when in the opposing direction (from Dev_B to Dev_A), Dev_B uses one-slot packets.

SCO	64 Kbps in each direction						
DV packets carry 64 Kbps SCO traffic and and 57.6Kbps of ACL data; ACL data use 2/3-rate FEC							
ACL	Slot size *x*	Symmetric (Kbps)		Asymmetric (Kbps)			
		DM*x*	DH*x*	DM*x*		DH*x*	
				forward	reverse	forward	reverse
	1	108.8	172.8	108.8	108.8	172.8	172.8
	3	258.1	390.4	387.2	54.4	585.6	86.4
	5	286.7	433.9	477.8	36.3	732.2	57.6

Table 2: *Bluetooth link capacities*

3.1.3 The Link Manager Protocol (LMP)

The link manager protocol is a transactional protocol between two link management entities in communicating Bluetooth devices whose responsibility is to set-up the properties of the Bluetooth link. For LMP packets, the *L_CH* field in Figure qwe is set to the binary value b‘11’.

Through LMP transactions, a device may authenticate another one through a challenge response mechanism. For authenticated devices, the link may further be encrypted. Two link managers may learn each other’s features, for example whether the devices support SCO links, what size of packet transmission do they support, or whether they support any of the low power consumption modes. SCO connections are established using LMP transactions, polling intervals and agreed upon packet sizes are also set-up through LMP transactions.

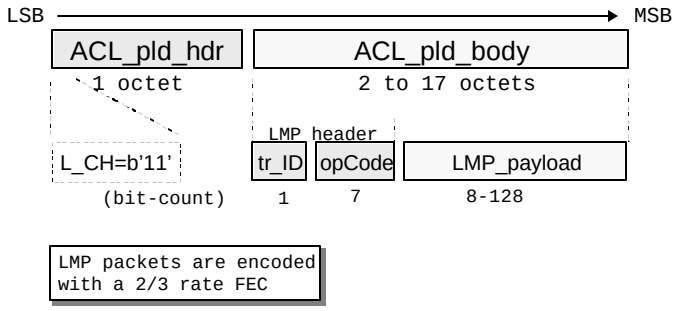

Figure qwt: *The LMP packet format*

Figure qwt shows the format of an LMP packet. The LMP packets are carried in the payload of DM1 ACL packets with logical channel *L_CH* = b‘11’; optionally, when supported, DV packets may also be used. The header of an LMP packet is just one octet. The *tr_ID* (for transaction ID) simply signify the initiator of an LMP transaction, which could be either the master (*tr_ID* = b‘0’) or the slave (*tr_ID* = b‘1’).

3.1.3.1 Security procedures

The algorithms for authentication and encryption are part of the baseband portion of the Bluetooth specification. However, the act of authentication, as well as the negotiation for encrypting the link between two devices are part of the LMP specification. Thus, the discussion about the security procedure is included here.

Bluetooth devices may be authenticated and links may be encrypted. Due to the ad hoc nature of Bluetooth communications and the fact that Bluetooth devices do not depend on infrastructure services for communications, certificate and public key infrastructure (PKI) approaches for authentication do not apply directly to Bluetooth piconets. Instead, the authentication of Bluetooth devices is based on a challenge/response mechanism based on a commonly shared secret.

Authentication of devices may happen at any time during the lifetime of a connection between two Bluetooth devices. The authentication starts with the transmission of an LMP

challenge packet. The challenge packet contains a random number generated by the *challenger*, which is the device that attempts to authenticate the other device. The receiver device of the challenge, called the *claimant*, operates on the challenge using a 128-bit authentication key. The claimant returns the result of the operation to the challenger, who can then compare the result with the expected outcome of the operation and, thus, verify the identity of the claimant.

To perform an authentication, each device is associated with *unit* key. In addition, each pair of devices may have a separate key, call the *combination* key, used for authenticating the specific two devices. Whether the unit key or the combination key is used for authentication of devices the same procedure is followed. When two devices are unaware of any link keys for performing authentication, e.g., when they connect for the first time, a personal identification number (PIN) is used to initialize the authentication process. The same PIN must be provided in both devices. Devices may have pre-configured PINs, e.g., a device without a user interface, then for authentication of such devices this PIN must be used.

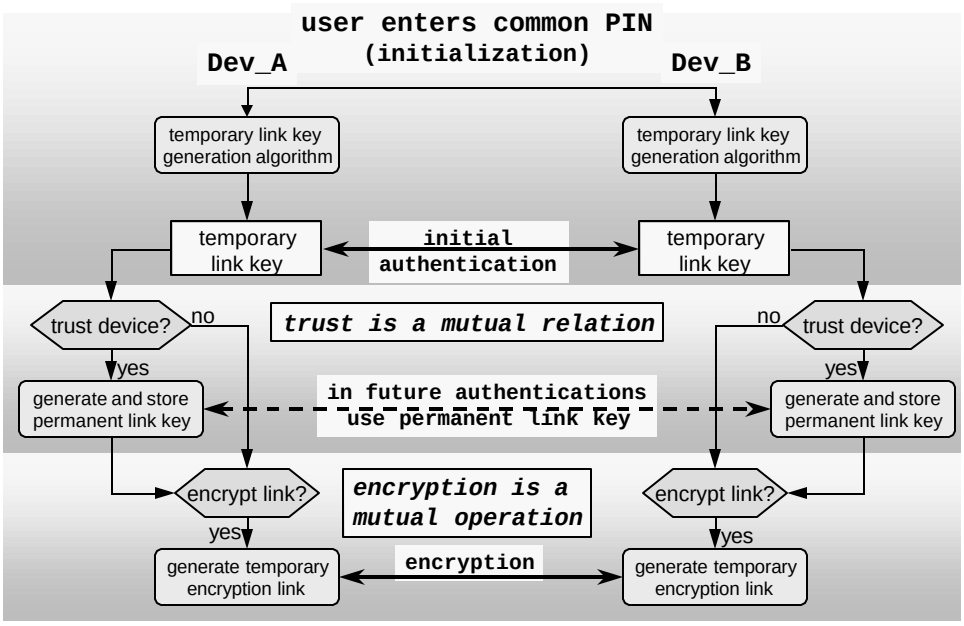

Figure 123: *The security steps*

The authentication process is highlighted in figure 123 using two devices, Dev_A and Dev_B, which we assume to be foreign to each other. We further assume that the devices have authentication enabled.

Following inquiry, if needed, and paging, Dev_A and Dev_B become members of the same piconet; for this discussion, it is immaterial which device is the master or the slave. Device, say, Dev_A will initiate authentication of Dev_B. In the absence of a link key for this authentication, the devices will request the entry of a PIN at the user level. The same PIN needs to be entered in both devices. The PIN will result in temporary link key that will be used for an initial authentication of Dev_B.

If the devices are to be trusted for future communications, both devices will generate and *store* a permanent link key for communicating with each other. This key will be used for future authentication of the devices. Following authentication in one direction, authentication may occur in the opposite direction as well, where Dev_B authenticates Dev_A.

Following device authentication, the devices may further encrypt the link between them to protect against eavesdropping. Using the link key, the devices will generate a sequence of encryption keys to encrypt their transmissions. The encryption key changes with each packet transmission.

Encryption is a mutual operation, and encryption encrypts the whole link, both the asynchronous and synchronous transmissions. For broadcast transmission, a *master* key is created by the master that is then passed to slaves using a regular link key. Since, encrypted broadcasts can be decrypted only by the slaves with the master key, the use of the encrypted broadcast link can serve as a means to implement multicasting by distributing the master key only to the members of the multicast group.

The encryption key can be up to 128 bits long. However, the size of the encryption keys is ultimately regulated by government authorities. The authentication and encryption keys are generated based on the SAFER+ algorithm; for more information on the algorithm see http://csrc.nist.gov/encryption/aes/round1/conf2/papers/massey.pdf.

3.1.3.2 The low power modes

Likewise to the security algorithms, the actual low power mode of operation are part of the baseband. However, these modes can be configured and activated via LMP transactions and they are highlighted here.

In the *sniff* mode, a slave agrees with its master to listen for master transmissions periodically, where the period is configured through LMP transactions.

In the *hold* mode, a device agrees with its communicating partner in a piconet to remain silent (in the particular piconet) for a give amount time. A device that has gone into hold mode, does not relinquish its temporary address, *AM_ADDR.*

Finally, in the *park* mode a slave device agrees with its master to park until further notice. As a device enters the park mode, the device relinquishes its active member address, *AM_ADDR*. While parked, a device will periodically listen to beacon transmissions from the master. A device may be invited back to active communications using a broadcast transmission during a beacon instant. When the slave the wants to be unparked, it would send a message to the master in the slots following the beacon instant.

The above modes of operation are designed for reducing the power consumption of a device. However, they are optional features. While in any of these modes, a device may be involved in other tasks, like entering inquiry scans, participating in active communications in another piconet, etc. Hence, the low power modes of operation, while designed for this purpose, enable additional modes of operation for a device.

3.1.4 The Host Controller Interface (HCI)

As the name states, this is not a protocol per se. It is rather an interface for host devices to access the lower layers of the Bluetooth stack through a standardized interface. Through the HCI, a host device passes and receives data destined to or coming from another Bluetooth device. Through the HCI also, a host may instruct its baseband to create a link to a specific Bluetooth device, execute inquiries, request authentication, pass a link key to the baseband, request activating a low power mode, etc. The HCI will not be discussed further here; for more information, see the Bluetooth specification.

3.1.5 The Logical Link Control & Adaptation Protocol (L2CAP)

The L2CAP layer shields the specifics of the Bluetooth lower layers and provides a packet interface to higher layers. At the L2CAP layer, the concepts of master and slave devices do not exist anymore. The L2CAP supports the multiplexing of several logical channels over the device's ACL links; note that a slave has only one ACL link while a master has one for each slave that it actively communicates with.

L2CAP packets can be much larger than the baseband packets and they may need to be segmented prior to transmission over the air, and reassembled following their receipt. For L2CAP packets, *L_CH* field in Figure BB2 is set to the binary value b'10' for the transmission of the first segment of an L2CAP packet, and b'01' for subsequent segments.

L2CAP traffic flows over logical channels terminating at the L2CAP layer of communicating devices. Channels may either be connectionless, or connection oriented. A channel end-point is identified by a two-octet *channel identifier* (CID); within each device, CIDs of various channels are unique. The connectionless CID has the reserved value 0x'0002'. The connection-oriented channels go through a set-up process using L2CAP signaling. The CID of the signaling channel has the reserved value 0x'0001'.

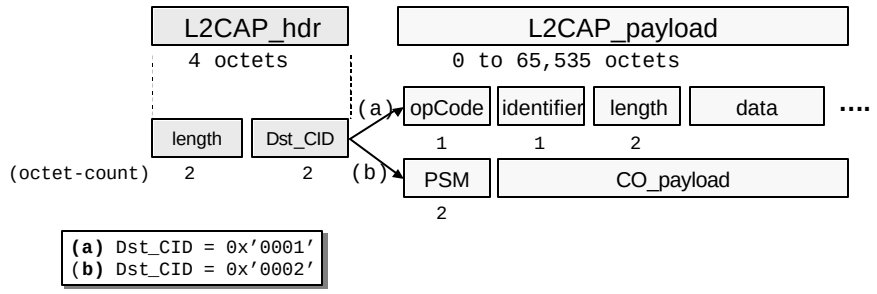

Figure rty: *The L2CAP packet format*

Figure rty summarizes the various types of L2CAP packets. They comprise a header, which is four octets long, and a payload portion, which could be up to 65,535 octets long. Devices with limited capabilities may be able to handle only much smaller packets. Learning the features supported from the other L2CAP entities is part of the L2CAP connection set-up process.

As figure rty shows, the payload of the L2CAP signalling packets (CID = 0x'0001') contains signalling information that is formatted with the following fields: (a) a one-octet *opCode* field to identify the particular signalling data; (b) a one-octet *identifier* field used to match responses to requests; (c) a two-byte *length* field containing the length of the data field; and (d) the signalling data. An example of a signalling packet is given in figure sdf.

Figure rty also shows the format of a connectionless L2CAP packet. Its payload carries asynchronous data and the PSM field used for protocol multiplexing as discussed shortly.

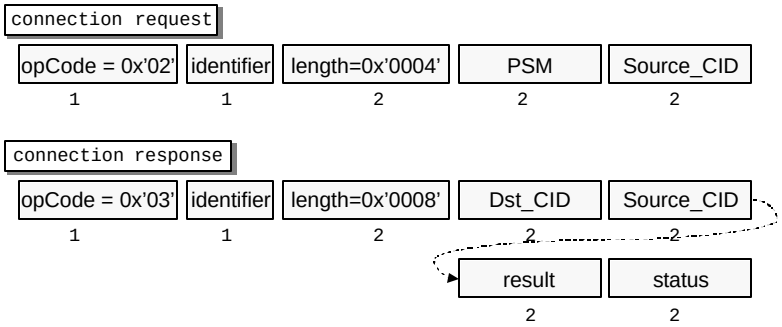

Figure sdf: *The L2CAP connection request and response packets*

Figure sdf shows the payload of the connection request and response packets. In requesting a channel creation, the requesting L2CAP entity notifies the other side of the local CID for the requested channel. Furthermore, it notifies the protocol engine that will process the incoming L2CAP payloads. The *protocol and services multiplexor* (PSM) field is used for supporting protocol multiplexing. The field is typically two octets long, but it can be extended if necessary. The PSM values below 0x'1000' are reserved. As figure rty shows, each connectionless L2CAP packet needs to carry its own PSM information. Upon receipt of a connection request, the receiving L2CAP entity will respond on whether it accepts, or rejects, or needs to process further the connection request. In the latter case, the status field provides information about the need for additional processing, e.g., authentication pending.

A channel could further be configured for the *maximum transmit unit* allowed in each direction of the transmission. Also, a quality of service negotiation is possible, however, currently only best effort traffic is fully supported over L2CAP channels.

3.2 The middleware protocols

While the transport protocols are involved in every communication of application data over Bluetooth links, not every middleware protocol participates in Bluetooth communications at all times.

3.2.1 The service discovery protocol (SDP)

I order to support the rich application space envisaged for Bluetooth devices, a service discovery protocol (SDP) was added to the basic Bluetooth protocols. Using this protocol a Bluetooth device can inquiry of the services that another device across a Bluetooth link may have and learn about how to get access to it. The SDP only provides information about services, it does not provide access to them. A Bluetooth device may access the service via different means using the information learned through service discovery.

SDP assumes that a Bluetooth device maintains a logical registry of services it supports that are accessible through its Bluetooth interface. SDP is a transactional protocol comprising of a sequence of requests and responses. The SDP defines how to formulate requests for information to this registry and how the registry formulates responses to these requests. The SDP does not define how the registry is implemented, but only how to submit valid questions to the registry and how to interpret the answers received from the registry.

SDP is optimized for usage of devices with possibly limited capabilities over wireless links. Bandwidth is preserved by utilizing binary encoding of information over the air. Universally unique identifiers (UUIDs) are used to describe services and attributes of these services in a manner that may not require a central registration authority for registering services. Typically the UUIDs are 128-bit long, however, for known services 16-bit and 32-bit UUIDs may also be used.

SDP packets are carried over connection-oriented L2CAP channels between communicating devices. The PSM value for SDP is reserved and has the value 0x'0001'. There are three types of service request/response transactions. Requests are send by the service discovery client to a service discovery server and replied the other way around.

Firstly, a client can inquiry a server using a collection of service names, represented as a list of UUIDs. The server returns a list of handles representing the service records in its service registry. Secondly, using a known service record handle of a service, the client may then retrieve from the server a list of service attributes related to the particular service, i.e., learn more about the specific service. Thirdly, combining the last two types of transactions in one step, a client may inquiry a server for a specific class of services and upon return also provide information regarding specific attributes for these services. The SDP also permits the browsing of services where a client may create a listing of services available in a server. However, there exists no explicit browsing transaction. It is created using the three previously mentioned transaction types.

3.2.2 The RFCOMM protocol

The RFCOMM protocol is an important layer that is used to expose a serial interface to the packet based Bluetooth transport layers. In particular, the RFCOMM layer emulates the signals on the nine wires of an RS-232 interconnect cable. The RFCOMM is based on the ETSI 07.10 standard permits the emulation and multiplexing of several serial ports

over a single transport. The multiple ports are identified using the *data link connection identifier* (DLCI).

RFCOMM enables legacy applications that have been written to operate over serial cables to run on top of a Bluetooth link without modification. Several of the applications developed for Bluetooth use the RFCOMM as part of their implementation stack.

3.2.3 The telephony control signalling (TCS) protocol

Telephony control can be performed using the AT command set. Since, the AT commands have been designed to be passed over serial lines, Bluetooth devices use the RFCOMM to send and receive control signalling based on the AT command set. For example, using these commands, a dialer application in a notebook computer may instruct a cellular phone to dial-up an ISP location.

The AT command set is well-established and it can be used for supporting legacy applications, like the dialer application. In addition to this control protocol, refer to as TCS-AT, the Bluetooth technical groups developed an additional packet-based telephony control signalling protocol, called TCS-BIN (BIN stands for the binary encoding of information). The TCS-BIN protocol is based on the ITU-T Q.931 standard and it runs directly on top of L2CAP. The protocol supports normal telephony control functions like placing and terminating a call, sensing ringing tones, accepting incoming calls, etc. Unlike TCS-AT, TCS-BIN supports point to multi-point communications as well allowing, for example, a cordless base station to pass the ringing signal of an incoming call to several cordless headsets associated with the base station.

3.2.4 Other protocols

To support various applications, a number of industry standards have been adopted. Such protocols include the point-to-point protocol (PPP), an IETF standard, for enabling communications, including IP communications, over serial lines; the object exchange (OBEX) protocol, an IrDA standard, for transporting objects between devices; the infrared mobile communications (IrMC) protocol, an IrDA also standard, for describing and encoding information in business cards, calendar entries, and messages. All these protocols are run on top of RFCOMM.

3.3 The Bluetooth profiles

As mentioned earlier, the Bluetooth specification comprises not only communications protocols but applications as well. This sets the Bluetooth wireless technology apart from many other communications technologies that focus primarily on the physical, data link and possibly networking aspects of communications. Since, the Bluetooth wireless technology is to be used primarily by consumers, the technology must require minimal technical expertise from its users. For this to be possible, a set of simple but useful applications had to be developed to allow Bluetooth devices to perform useful tasks with other Bluetooth devices right out of the box. This would provide value-add to the users of

the technology and aid in establishing this technology as the de facto means for short-range communications of personal devices.

The specifications for building interoperable applications are called *profiles*. Actually, there are two types of profiles, protocol profiles and application profiles. The profiles defined in version 1.1 of the specification are summarized in figure yui. Profiles can be based on other profiles as well, thus figure yui also shows the relation between the profiles.

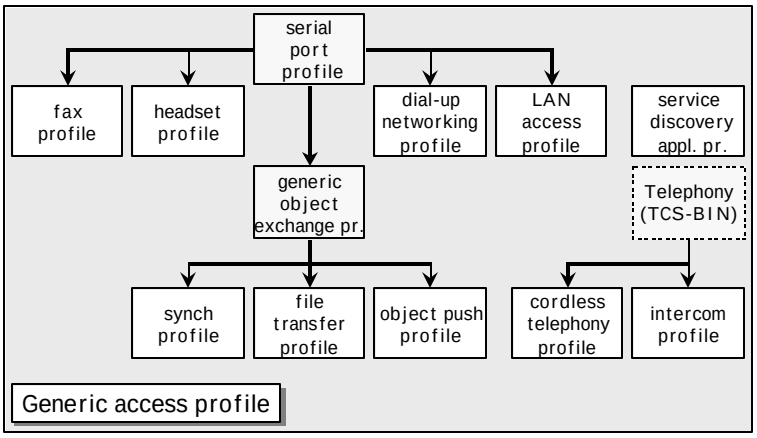

Figure yui: The Bluetooth profiles

All profiles depend on the *Generic Access Profile* (GAP) which defines the basic rules and conditions for connecting devices with each other and establishing Bluetooth links and L2CAP channels. It also defines security levels according to which devices may allow themselves to be discovered, or allowed to be connected, be authenticated or authenticate other devices, etc. It defines the conditions necessary to establish trust relations between devices, see figure 123.

There are two protocol profiles, which depend on each other. The *serial port profile* defines how RFCOMM runs on top of the Bluetooth transport protocols, while the *generic object exchange profile* defines how object can be exchanged using the OBEX protocol running on top RFCOMM as defined in the serial port profile. Depending on the previous profile, there are profiles describing how to synchronize personal information management (PIM) data, how to push (and pull) objects, e.g., business cards, and how to transfer files. Based on the serial port profile there are also additional profiles related to

the use of cellular phones as modems for dial-up networking, connecting to a wireless headset, sending faxes, or accessing LAN services through a LAN access point.

There are two profiles based on the TCS-BIN protocol which describe two aspects of the so called 3-in-1 usage scenario, where a cellular phone can be used as a headset in a cordless telephony system or as an intercom device to communicate directly with other cellular phones.

Finally, there is the service discovery application profile that shows how a service discovery application uses the service discovery protocol and, furthermore, how the latter protocol uses the Bluetooth transports for carrying the service discovery packets between a service discovery client and a server.

4 THE BLUETOOTH QUALIFICATION PROGRAM

The Bluetooth qualification program (BQP) is an integral part of the Bluetooth wireless technology. It is a program designed to test compliance to the specification and product interoperability. The latter implies that not only compliance to the protocol specification is tested but also test whether applications claimed to follow a Bluetooth profile behave as expected. The BQP does not deal with the regulatory type approvals that are administered by national and international authorities. Manufacturers will need type approvals for their products before they can sell them in the market. For example, in the USA, a Bluetooth device needs to be certified as compliant to the FCC part 15 regulations.

Only products qualified through the BQP can be called Bluetooth products, receive the Bluetooth free license, and bear the Bluetooth trademarks. The BQP is the mechanism by which the Bluetooth brand is protected assuring users of the technology a certain level of confidence in the product and also in the functionality of products carrying the Bluetooth trademarks. Only products from adopter members can be qualified. Qualified products are published in the official Bluetooth website.

Testing in the BQP includes the following:

? radio qualification testing;

? protocol conformance testing –this includes the Bluetooth transport protocols and the service discovery protocol);

? profile conformance testing –at a minimum, this includes conformance to the generic access profile;

? profile interoperability testing – for additional profiles.

Not all products need to go through all the tests mentioned above. Tested products may include anything from basic Bluetooth radio components, to complete products, e.g., a wireless headset, or a cellular phone with an integrated Bluetooth subsystem. The breadth of testing depends on the particular product considered.

The program includes implementation conformance statements (ICSs), where manufacturers declare the features that their products support; possibly, product testing by the manufacturer; and testing in a Bluetooth Qualification Testing Facility (BQTF). Manufacturers submit their ICSs and test results to a Bluetooth Qualification Body (BQB), who is a person authorized to review the test reports and manufacturer declarations and approve products. When the BQB approves a product, the new Bluetooth product is added in the listing of qualified Bluetooth products, which is available through the Bluetooth web-site.

5 ADDITIONAL ACTIVITIES

5.1 The SIG

The work in the Bluetooth SIG did not end with the release of the Bluetooth 1.0A specification. The SIG has been heavily involved in promoting the technology and organizing Bluetooth Developers conferences, which are technical forums, where product developers, and potential customers may learn of the latest developments about the technology and have the opportunity to meet with each other. The SIG also encourages "unplug" fests, where manufacturers can test their products against each other and exchange technical information in order to achieve high degree of product interoperability. The unplug fests are not part of the BQP, but they provide a diverse forum for product interoperability testing. The regulatory group of the SIG collaborates with other similar bodies in the industry and is actively engaging authorities in various countries in an effort to harmonize the spectrum allocation in the 2.4 GHz in these countries.

The technical groups within the SIG are maintaining the specification by administering an errata process and developing a new set of protocols and profiles. New technical work focuses on

? enhanced radio capabilities with backward compatible faster radios;

? study any co-existence issues between technologies sharing the 2.4 GHz ISM band, like the Bluetooth wireless technology, IEEE 802.11 wireless LANs, HomeRF, cordless telephony, etc.;

? support for IP networking without the use of the RFCOMM and PPP;

? support for higher level service discovery methods, like UpnP;

? develop communication solutions for usage scenarios involving:

  ? in vehicle communications, printing, transfer of digital images, richer audio, voice, and video experience, distribution of location information, using the unrestricted digital information (UDI) extensions for 3G handsets in Japan.

5.2 IEEE 802.15

Another forum where personal area networks are being standardized is the IEEE 802 LAN/MAN standards committee. In March 1999, the IEEE 802.15 working group was created to develop a family of communications standards for wireless personal area networks™ (WPANs™).[6] In the first meeting of the new working group in July 1999, the Bluetooth SIG submitted the just created Bluetooth specification as a candidate for an IEEE 802.15 standard. The Bluetooth proposal was chosen to serve as the baseline of the first 802.15 draft standard.

As mentioned earlier, the Bluetooth specification describes a complete solution including communication protocols and applications. The IEEE 802 standard are dealing only with the lower two layers of the OSI stack: the physical layer (PHY) and the medium access control sub-layer (MAC) of the data link control layer. As such, there is no immediate mapping between the Bluetooth specification and an IEEE 802 standard. The 802.15.1 task group decided to base its MAC and PHY on the Bluetooth transport protocols. This was done because all of the latter protocols are involved in data communications between Bluetooth devices. Any smaller subset of protocols would have created a daunting task to dissect the Bluetooth specification in an unnatural, hardly verifiable manner. Any larger set of protocols would have included either protocols not always present in Bluetooth communications, like RFCOMM, or protocols not related with the normal transport of data between Bluetooth devices, like SDP.

[6] Wireless Personal Area Network and WPAN (and their plural versions) are trademarks of IEEE.

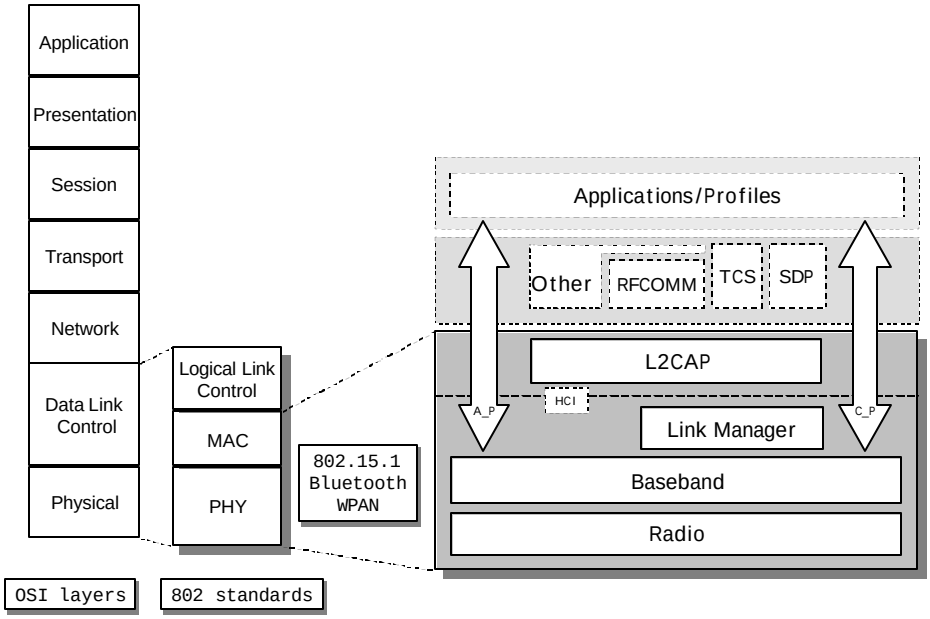

Figure 789: Relation between the OSI layer, the IEEE 802 layers and the Bluetooth stack

Figure 789 depicts the relation between the OSI model layers, the focus of the IEEE 802 standards and how the 802.15.1 effort relates to the Bluetooth stack. As of this writing, the development of the draft standard is at final stages, short of going for sponsor balloting.

Since the formation of 802.15.1 task group, additional task groups have been formed to study additional aspects of WPANs. The 802.15.2 task group studies coexistence issues between 802 wireless technologies. The 802.15.3 task group is developing standards for high rate radios (>20Mbps). Finally, the 802.15.4 task group is developing standards for low rate radios (<200Kbps).

6 SUMMARY

The Bluetooth wireless technology is a specification for short-range, low-cost, and small form-factor that enables user-friendly connectivity among portable and handheld personal devices, and provides connectivity of these devices to the internet. The technology supports both asynchronous data flows and synchronous audio streams over links with raw link speed of 1Mbps. It operates in the 2.4 GHz ISM band utilizing low transmit power radios, typically 0dBm, using a frequency hoping spread spectrum technique. The Bluetooth specification is an on going process steered by the promoters of the Bluetooth SIG and developed by contributing SIG members.

The SIG recognized that the technology would be successful if it is widely available and useful tasks can be done with it from the early days of it use. For this reason, the Bluetooth specification comprises a protocol stack provided by a hardware and software

description and an application framework, called profiles, for building interoperable applications. Furthermore, the technology is provided license-free to the adopters members of the technology. The Bluetooth qualification program, which is applicable only on potential Bluetooth products by adopter members, has been designed to build, promote, and maintain a level of confidence to the users of the technology that they are using a product that was built in manner compliant to the Bluetooth specification. Furthermore, it can interact with other devices and behave as expected when executing any application claimed to be conformant to the Bluetooth profiles.

The Bluetooth movement started in May 1998 and it has been followed very closely by the technical community, the business community, all sorts of market analysts and gurus and the media. As of this writing, almost two years after the release of the Bluetooth specification, it appears that the deployment of the technology is moving slower than expected. That is actually quite understandable and it is common with any new technology; for example, the development of the 802.11 technology starter in the early 1990s, and it took a good portion of a decade before it started spreading.

The slower than anticipated pace of deployment of the Bluetooth wireless technology is understandably notable. However, the expectations for the potentials for the technology does seem to have diminished, only shifted in time. The image of a single technology that enables a personal area network that moves as humans move and brings worry-free, ad hoc connectivity to personal devices at home and in the workplace, in the car and in the mall, in the airport and in the ballpark, etc., is too strong a paradigm to ignore it. Today, as was in the May of 1998, the only technology that still has the possibility to succeed in this space is the Bluetooth wireless technology.

IN MEMORY OF…

Dr. Richard LaMaire, a dear friend and esteemed colleague, that passed away as the finishing touches of this article were being made.

ADDITIONAL RESOURCES

The Bluetooth specification is available free of charge on the official Bluetooth web-site at *http://www.bluetooth.com*. The site contains plenty of information regarding the technology and how the SIG operates. The site has also a listing of the Bluetooth qualified products. A very good technical article written by one of the innovators of the Bluetooth wireless technology, Dr. Jaap C. Haartsen, titled *The Bluetooth Radio System*, can be found in the special issue on "Connectivity and Applications Enablers for Ubiquitous Computing and Communications" of *IEEE Personal Communications*, February 2000. The same issue contains a paper by Tom Siep, et al, on the process of developing the 802.15 standard, titled *Paving the Way for Personal Area Network Standards: An Overview of the IEEE P802.15 Working Group for Wireless Personal Area Networks*; the activities of the IEEE 802.15 working group can be found at h*ttp://grouper.ieee.org/groups/802/15/*. *Bluetooth Revealed* (Prentice-Hall PTR, 2001),

authored by Brent A. Miller and myself, highlights the 1,600 pages of the specification spiced with trivia from the development of the specification as experienced by authors. A good article highlighting the technology with emphasis the qualification program is *Bluetooth's slow dawn*, by Ron Schneiderman, in *IEEE Spectrum*, November 2000. *Bluetooth: Connect Without Cables*, Jennifer Bray and Charles F. Sturman provides a lot of useful information about the Bluetooth specification for implementers and product developers.