Bluetooth Core Specification Version 5.2 Feature Overview

Bluetooth® Core Specification version 5.2 includes three primary updates. This document summarizes and explains each change.

Author: Martin Woolley

Version: 1.0.1

Revision Date: 9 December 2020

Bluetooth®

Revision History

Version	Date	Author	Changes
1.0.0	6 January 2020	Martin Woolley	Initial Version
1.0.1	9 December 2020	Martin Woolley	Language Changes

Table of Contents

Table of Contents

Table of Contents

At a Glance

1. Enhanced Attribute Protocol

An improved version of the Attribute protocol (ATT), called the Enhanced Attribute protocol (EATT), has been introduced along with some associated improvements to the Generic Attribute Profile (GATT).

EATT supports concurrent transactions, allows the interleaving of L2CAP packets relating to ATT packets from different applications, and allows the ATT Maximum Transmission Unit (MTU) to be changed during a connection. Collectively, these changes can provide an improved user experience on devices where there are multiple applications using the Bluetooth Low Energy (LE) stack at the same time, through reducing instances where one application's use of the stack temporarily blocks that of another. This can reduce the end- to- end latency of one or more of the applications and improve the user's experience of responsiveness.

In support of EATT, a new L2CAP mode has been defined. The new mode is called the L2CAP Enhanced Credit Based Flow Control Mode. This mode provides flow control and so allows applications to regard the protocol as reliable.

EATT has security advantages over unenhanced ATT as it may only be used over an encrypted connection.

2. LE Power Control

The new LE Power Control makes it possible for devices to dynamically optimize the transmission power used in communication between connected devices. Bluetooth LE receivers may now monitor signal strength and request transmission power- level changes in connected devices, typically to maintain an optimal signal strength from both a signal quality and low- power-use perspective. The Bluetooth controller can monitor and report path loss changes to the Bluetooth host using the concept of *zones*, which some application types will find useful.

Benefits of the LE Power Control feature include:

1. The reduction of overall power consumption by transmitters through dynamic power management conducted between connected devices.

2. Improvements in reliability through the active maintenance of receiver signal strength so that it stays within the optimal range supported by the receiver.

3. Improvements relating to coexistence with other wireless devices that are in the environment and are using the 2.4 GHz frequency range. This benefit applies to all such devices, not just those that are using Bluetooth.

3. LE Isochronous Channels

This feature, which was primarily designed to support LE Audio, the next generation of Bluetooth audio, allows the communication of time-bound data to one or more devices for time-synchronized processing. It can be used over connections or be broadcast to an unlimited number of devices in a connectionless fashion.

New use cases and topologies are made possible by LE Isochronous Channels. An audio source can transmit audio for synchronized playback by small, private groups of devices (*personal audio sharing*) or to large collections of devices of unlimited sizes in public spaces, such as cinemas.

Music sharing is only one application, however. LE Audio, built on top of the new LE Isochronous Channels, will offer a new standard for hearing aids and support assisted hearing systems in diverse locations, such as theaters, conferences, lecture halls, and airports. It is anticipated that multi-language audio systems will become possible, in part, thanks to LE Isochronous Channels and new audio Bluetooth profiles which will use it.

A more comprehensive explanation of each of the new features follows. The Bluetooth Core Specification version 5.2 should be consulted for full details.

1. Enhanced Attribute Protocol

1.1 Background

This section reviews pertinent aspects of version 5.1 of the Bluetooth Core Specification to provide context and background which will support the examination of the new enhanced ATT protocol in section 1.2 below. Where a specification version is not explicitly stated in this section, it should be assumed that version 5.1 is being referred to.

1.1.1 The Bluetooth Low Energy Stack with GATT, GAP, and ATT
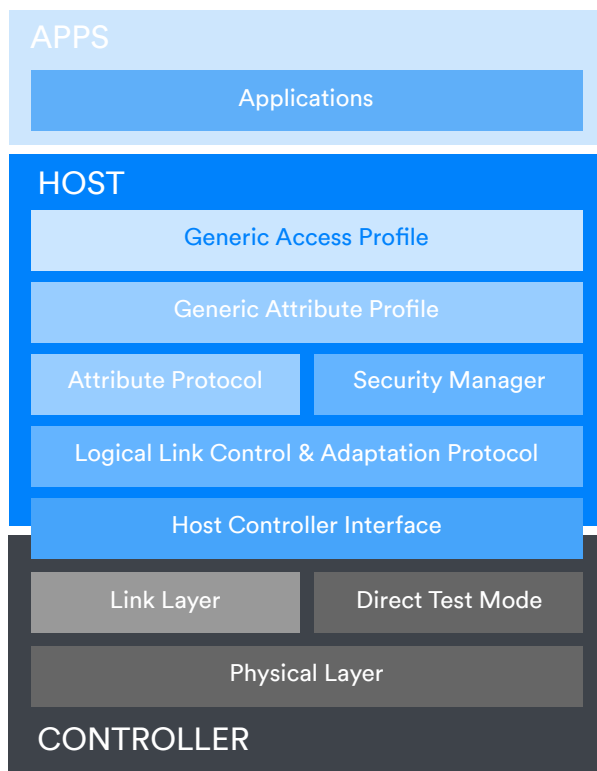
Figure 1 - Bluetooth LE with GATT, GAP, and ATT

A Bluetooth Low Energy (LE) stacks consist of two major architectural components, known as the *host* and *controller*, each of which contain various stack layers. The Host Controller Interface (HCI), defines a series of *commands* which the host can use to communicate with the controller and events which are used by the controller to communicate with the host.

A number of standard stack configurations are defined, and Figure 1 depicts an example of the one that is typically used by Bluetooth LE devices, such as smartphones, heart rate monitors, key finders, and many others.

An awareness of the stack in Figure 1 will help you appreciate the changes which have been made in version 5.2 of the Bluetooth Core Specification.

Bluetooth mesh devices have a different host component, which contains the layers of the mesh protocol stack. This variant of the Bluetooth stack is not relevant to the changes described in this overview.

1.1.2 Attribute Protocol

Bluetooth devices may contain a collection of special data entities, known as *services*, *characteristics*, and *descriptors*, each of which is a type of *attribute*. Attributes of all types are organized within something called an *attribute table*.

Characteristics are instances of state data and reflect some aspect or capability of the device.

Services group associated characteristics together and provide context from which to determine the meaning and rules for using the characteristics that the service contains.

Characteristics can have zero or more descriptors associated with them. A descriptor provides metadata relating to its owning characteristic or a means of configuring or controlling a behavior associated with that characteristic. For example, the Client Characteristic Configuration Descriptor (CCCD) allows a remote device to enable or disable the sending of notification PDUs which contain the value of the characteristic.

The Attribute Protocol (ATT) is used by an ATT client to discover details of the attribute table in a remote, connected device which is known as the ATT server. Client and server can each use the attribute protocol to interact with the other in a variety of ways. For example, a client can use ATT to ask the server to send it the current value of a particular characteristic using one of several types of *read request*. A server can inform the client whenever the value of a characteristic changes by sending it a *notification*, for which a response is not required, or an *indication*, which does require the client to respond.

ATT is one of the primary mechanisms by which applications in connected Bluetooth devices interact with each other, using the PDUs the protocol defines, and procedures defined in higher level specifications, such as the Generic Attribute Profile (GATT).

ATT defines the concept of a *transaction*. Request PDUs from a client require a response PDU to be returned by the server. *Indications* sent by a server must be replied to by the client with a *confirmation* PDU. Each request/response pair or indication/confirmation pair forms a transaction. Most ATT PDU types are transaction- oriented, but ATT also includes a few PDU types which are not associated with transactions, namely *commands* and *notifications*.

1.1.3 Logical Link Control and Adaptation Protocol (L2CAP)

L2CAP is responsible for protocol multiplexing, flow control, and segmentation and reassembly of service data units (SDUs). Each of these concepts is explained below.

L2CAP uses the concept of *channel* to separate sequences of packets passing between layers of the stack. Fixed channels require no set-up, are immediately available and are associated with a particular higher layer protocol such as ATT.
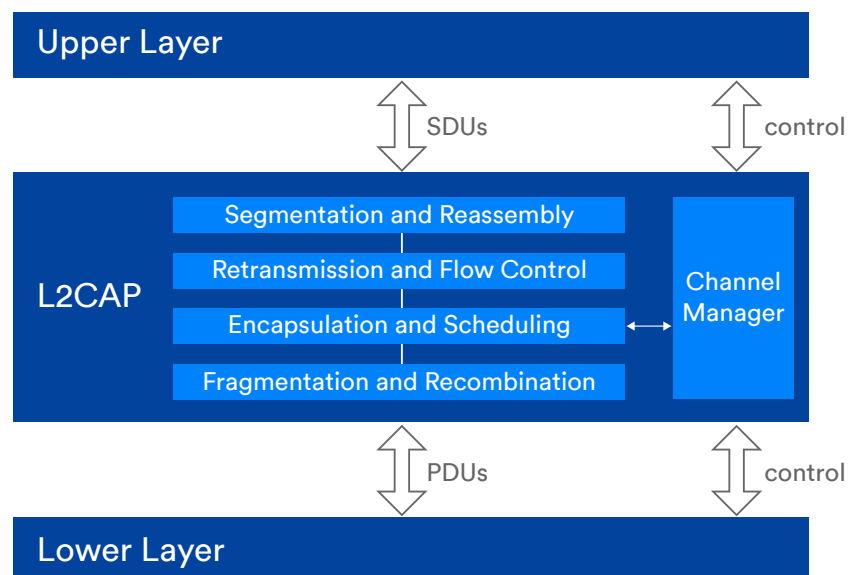

Figure 2 - L2CAP primary functions

Channels may also be dynamically created and associated with a protocol through a specified Protocol Service Multiplexer (PSM) value.

1.1.3.1 L2CAP and Protocol Multiplexing

Sitting above L2CAP in the stack are layers which use distinct protocols, such as the attribute protocol and SMP, the security manager protocol used by the security manager layer. L2CAP protocol multiplexing makes sure that SDUs get passed up the stack into the appropriate layer for processing.

1.1.3.2 L2CAP and Flow Control

Flow control is concerned with making sure that the rate at which packets are produced by one layer of a stack does not exceed the rate at which a layer in the same stack or on a remote device processes those packets. Without flow control, there is a risk of issues, such as buffer overflow arising.

Credit-based flow control is one of many possible approaches to flow control. In outline, it works as follows:

- The transmitting device knows the capacity of the receiving device in terms of the number of PDUs it can handle without losing data (e.g. through its buffer overflowing). It acquires this capacity information through configuration or via an exchange made between the two devices before data transfer starts.

- The transmitter sets a counter to this receiver capacity limit. Every time a PDU is sent by the transmitter, the counter is decremented. When the counter value reaches zero, the transmitter knows the receiver is at full capacity and so stops sending further PDUs temporarily while the receiver processes its backlog.

- After the receiver reads and processes one or more PDUs from its buffer, it sends back a corresponding number of credits to the transmitter, which uses it to increment its counter. With the counter at a non-zero value, the transmitter may continue to send further PDUs.

L2CAP in version 5.1 of the Bluetooth Core Specification defines several modes of operation, most largely concerned with flow control. ATT over Bluetooth LE connections uses the L2CAP Basic mode, which provides no flow control. This renders ATT at version 5.1 unreliable and applications must accommodate the possibility that transmitted ATT PDUs may be lost by the receiving device. In the case of unacknowledged PDUs, such as notifications, the transmitting device will know whether or not the PDU was received by the stack on the remote device, thanks to link layer acknowledgements, but has no way of knowing whether the PDU was successfully delivered to the receiving application at the top of the stack.

1.1.3.3 L2CAP Segmentation and Reassembly

Layers both above and below L2CAP are subject to a Maximum Transmission Unit (MTU) size which specifies the largest size a PDU of the type created by that layer is

allowed to be. For example, the ATT_MTU parameters defines the maximum size an ATT PDU may be.

L2CAP, itself, and the layers above or below it in the stack may have different MTU sizes and, consequently, it may be necessary to split some PDUs/SDUs into a series of smaller parts which the adjacent layer can handle or, conversely, to reassemble a series of related, smaller parts into full PDUs/SDUs. Those processes, as applied by L2CAP with respect to upper layers, are called *segmentation and reassembly*, while the equivalent processes, as they relate to L2CAP and its relationship with lower layers, are called *fragmentation and recombination*.

1.1.4 Concurrency and Latency

There are a number of issues relating to ATT and L2CAP which have implications for latency when multiple applications are using the stack.

1.1.4.1 ATT Sequential Transaction Model

ATT uses a sequential transaction model. At version 5.1 of the Bluetooth Core Specification, this meant that if an ATT transaction has been started, no further ATT PDUs may be processed by the stack until the current transaction has completed. The transaction is deemed to have completed when the expected response or confirmation PDU has been received from the remote device or when the transaction has timed out after waiting for 30 seconds.

The stack may be providing Bluetooth communication services to more than one application at a time. Consider devices like smartphones which may have a number of applications and system services running concurrently. The rules relating to ATT transactions mean that if an application, *App A*, starts an ATT transaction, another application, *App B*, will have to wait until App A's transaction has completed before being able to start a new transaction of its own. This may result in App B having to wait, and the end- to- end latency associated with the App B device interaction will be increased.

Figure 3 - Two applications using the same stack

1.1.4.2 ATT MTU and L2CAP MTU Sizes

In Bluetooth Core Specification version 5.1, the MTU assigned to the L2CAP layer has a 1-1 correspondence with the MTU value assigned to the ATT layer. This results in L2CAP being unable to interleave the processing of L2CAP SDUs from different applications and different ATT PDUs, which would be advantageous in some circumstances. For example, whilst the version 5.1 sequential transaction model would prevent a second application from starting a new ATT transaction, it does not mean that a non-transactional ATT PDU, such as a value notification, cannot be submitted. But if the L2CAP layer is already processing a large L2CAP PDU, submitting it to the controller for processing, the notification from the second application will still have to wait for L2CAP to become available.

1.1.5 MTU Renegotiation

The ATT MTU may be negotiated immediately after connecting, using the ATT_Exchange_MTU request and response PDUs. Once negotiated, however, it cannot be changed during this connection. If a second application on the same device then starts to use the same connection, and it transpires that the MTU size is not large enough for its purposes, it has no option but to fail, giving the user a poor user experience in the process.

1.2 About Enhanced ATT

This section reviews the new Enhanced Attribute Protocol (EATT) and related changes which have been made in the Bluetooth Core Specification version 5.2.

1.2.1 Capabilities and Benefits

EATT modifies the sequential transaction model, making concurrent ATT transactions possible when conducted over distinct *enhanced ATT bearers*.

A new mode has been added to the L2CAP layer and is used by EATT. The new L2CAP mode provides flow control so that EATT can be regarded as reliable.

MTU values at the ATT layer and L2CAP layer have been made independently configurable, and this may reduce latency experienced by some applications that share the stack with other applications.

1.2.2 Technical Highlights

1.2.2.1 L2CAP Enhanced Credit Based Flow Control Mode

A new L2CAP mode called Enhanced Credit Based Flow Control Mode has been defined. It is always used by L2CAP channels that are used with the enhanced attribute protocol and is similar to the older LE Credit Based Flow Control Mode but can be used with both Bluetooth LE and BR/EDR and allows MTU and Maximum PDU Payload Size (MPS) to be reconfigured dynamically.

An L2CAP channel which is used by EATT and therefore uses the Enhanced Credit Based Flow Control Mode is known as an *Enhanced ATT Bearer*, whereas when an L2CAP channel is used by ATT, it is known as an *Unenhanced ATT Bearer.*

EATT always uses dynamic rather than fixed L2CAP channels but with a new protocol service multiplexer (PSM) value. An L2CAP channel for use by enhanced ATT bearers is created using the new L2CAP PDUs, L2CAP_CREDIT_BASED_CONNECTION_REQ and L2CAP_CREDIT_BASED_CONNECTION_RSP. Note that up to 5 channels may be requested at a time using the new L2CAP_CREDIT_BASED_CONNECTION_REQ PDU.

It is expected that the GATT layer will manage a pool of bearers and allocate them to the servicing of requests, as needed. This is an implementation detail, however.

1.2.2.2 Parallel ATT Transactions

The new enhanced ATT protocol allows concurrent transactions to be handled by the stack.

The sequential transaction rule still exists when EATT is used, but its scope is now defined as being per instance of the Enhanced ATT Bearer. In other words, EATT transactions may be executing in parallel if they are supported by distinct L2CAP channels which use the Enhanced Credit Based Flow Control Mode, i.e. distinct Enhanced ATT Bearers.

When using an Enhanced ATT Bearer, ATT MTU and L2CAP MTU are independently configurable and may be reconfigured during a connection. An increase to the MTU is allowed but reducing its size is not. Allowing MTU to be increased without needing to reestablish the connection means that the risk of a second application, that is using the stack being unable to continue due to the previously negotiated MTU being too small, has been eliminated. Initially, ATT MTU is set to the smaller of the connected devices' L2CAP MTU values, arrived at through an exchange of L2CAP_CREDIT_BASED_CONNECTION_REQ and L2CA_CREDIT_BASED_CONNECTION_RSP PDUs. It may later be increased by exchanging L2CAP_CREDIT_BASED_RECONFIGURE_REQ and L2CAP_CREDIT_BASED_RECONFIGURE_RSP PDUs. Each of these L2CAP PDU types is new to version 5.2 of the Bluetooth Core Specification.

Whereas previously, a 1-1 mapping from ATT MTU to the L2CAP MTU existed, with the new Enhanced Credit Based Flow Control Mode, this is no longer the case. If the L2CAP MTU is smaller than the ATT MTU, this forces larger ATT PDUs to be segmented, which in turn allows L2CAP PDUs originating from different applications to be interleaved. This reduces blocking in the stack and prevents the increased latency which would otherwise have been experienced by some applications.

ATT MTU and L2CAP MTU are independent of each other.

Here the L2CAP MTU is smaller than the ATT MTU. Consequently, the large ATT PDU from App A (1) must be segmented and submitted to L2CAP as two SDUs.

L2CAP processes each segment independently, allowing the P2 PDU from APP B (2) to be processed in between the two segments of PDU P1 from App A.

(3) App B does not have to wait for the complete processing of P1 before P2 can be processed.

APP A - - - - - - - ATT clients - - - - - - - APP B

1   Large ATT PDU **P1**

2   Small ATT PDU **P2**

max input PDU size= ATT MDU

ATT

L2CAP MTU is smaller than the ATT MTU

L2CAP   **3**

P1, seg #2/2

P2, seg #1/1

P1, seg #1/2

controller
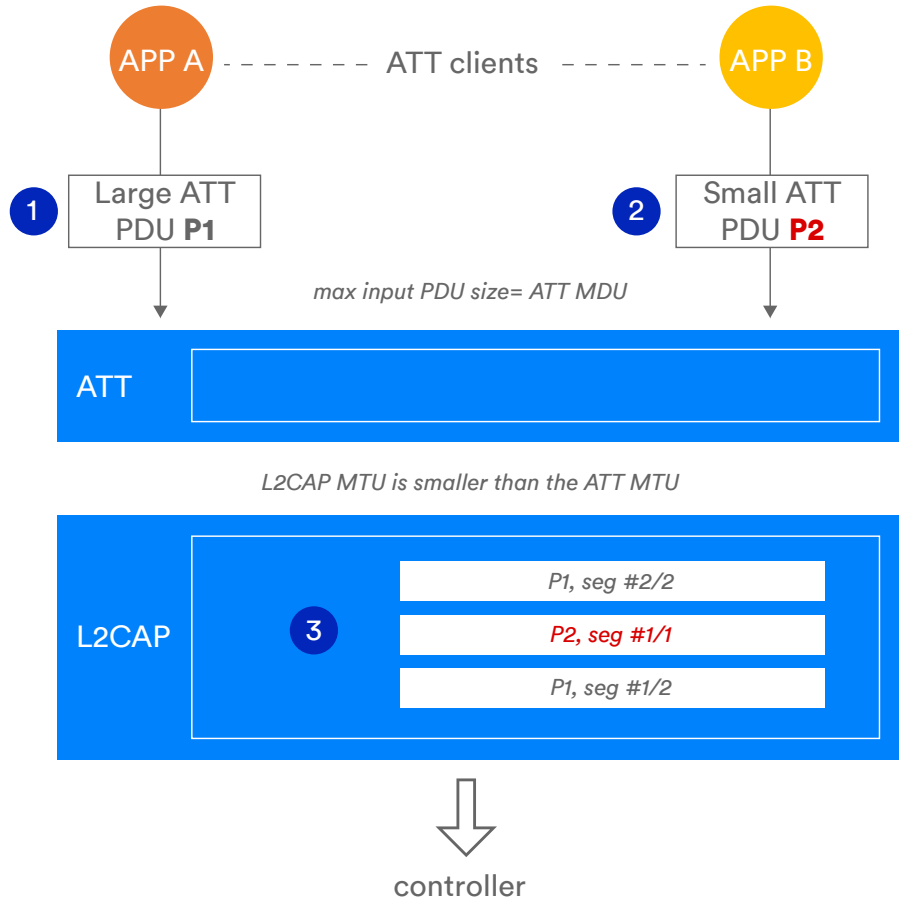
Figure 4 - Interleaved L2CAP SDUs

1.2.2.3 Discovering Support for EATT

The Generic Attribute Profile service has been updated to allow a client to determine whether or not a connected server supports EATT and, conversely, to allow the client to inform the server that it supports EATT.

A new characteristic called *Server Supported Features* has been defined and must be included in the Generic Attribute Profile service if EATT is supported by the server. Bit 0 of the first octet of the value of this characteristic set to 1 means that EATT is supported. A GATT/ATT client can determine whether or not the server supports EATT by reading this characteristic.

The Client Supported Features characteristic was introduced in Bluetooth Core Specification version 5.1. Bits of the characteristic's value indicate support or otherwise for certain features, and version 5.2 of the Bluetooth Core Specification has assigned meaning to two more bits. Bit 1 indicates whether or not the Enhanced ATT Bearer is supported by the client. Bit 2 indicates whether or not a new ATT PDU called Multiple Handle Value Notifications is supported (see 1.2.2.4 below). The client must write an appropriate value to this characteristic to inform the server of the features it supports.

The remote client reads the Server Supported Features characteristic to discover whether the server supports EATT.

The remote client writes to the Client Supported Features characteristic to inform the server of its support or otherwise for EATT.
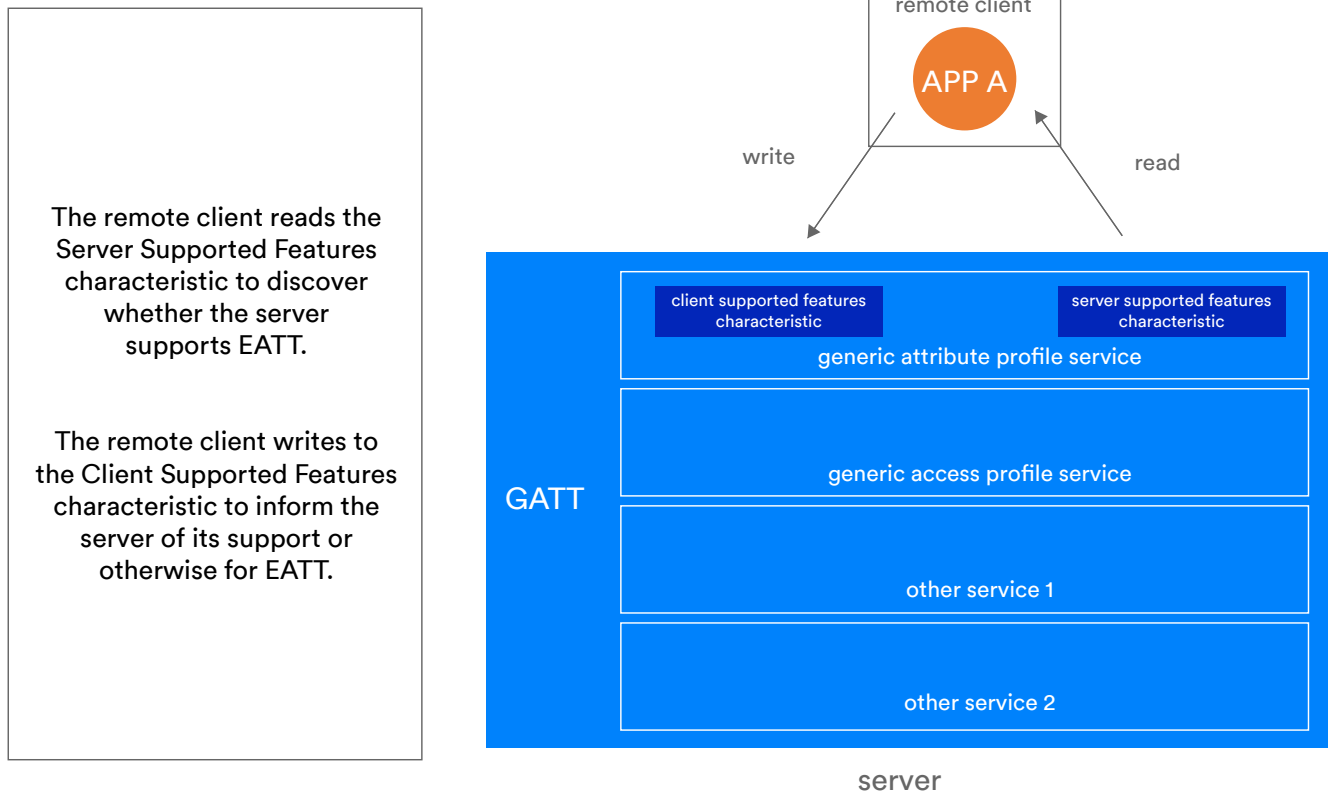
Figure 5 - EATT feature support discovery

1.2.2.4 Attribute Protocol PDUs

The introduction of EATT impacts ATT PDUs in several ways:

1. A number of new PDUs which may only be used over an enhanced ATT bearer have been defined.
2. Some of the ATT PDUs which may be used over an unenhanced ATT bearer may not be used over an enhanced ATT bearer.
3. The specification of the behavior of some PDUs has been refined in terms of ATT bearers.
4. EATT may only be used over an encrypted connection. Unenhanced ATT continues to be allowed over both unencrypted and encrypted connections.

The following new ATT PDUs, for use solely over an enhanced ATT bearer have been defined.

PDU Names	Related GATT Procedure	Comment
ATT_READ_MULTIPLE_ VARIABLE_REQ and ATT_READ_MULTIPLE_ VARIABLE_RSP	Read Multiple Variable Length Characteristic Values	The request PDU allows an EATT client to ask the server to return the values of two or more attributes. The length of the values of these attributes may be variable or unknown. A set of two or more attribute handles must be included in this request to identify the attributes whose values are requested by the client. The response PDU contains a list of [*attribute length, attribute value*] tuples with the attribute length occupying 2 octets and the value, a variable length as specified by the length part.
ATT_MULTIPLE_HANDLE_ VALUE_NTF	Multiple Variable Length Notifications	Allows the EATT server to send the values of two or more attributes in a single PDU. Support for this PDU type by the server is indicated using the Client Supported Features characteristic

Table 1 - New ATT PDUs for use with enhanced ATT bearers only

The following ATT PDUs, which existed at version 5.1 of the Bluetooth Core Specification, may not be used over an enhanced ATT bearer. Note that PDU names have been changed in version 5.2 of the Bluetooth Core Specification and this is reflected in table 2.

PDU Names	Previously Known As	Comment
ATT_EXCHANGE_MTU_ REQ and ATT_EXCHANGE_ MTU_RSP	Exchange MTU Request and Exchange MTU Response	When using an enhanced ATT bearer, the ATT MTU is initially set to the value with which the L2CAP MTU has been initialized via L2CAP_CREDIT_BASED_ CONNECTION_REQ and L2CAP_ CREDIT_BASED_CONNECTION_RSP PDUs. It may subsequently be increased by exchanging L2CAP_ CREDIT_BASED_RECONFIGURE_ REQ and L2CAP_CREDIT_BASED_ RECONFIGURE_RSP PDUs
ATT_SIGNED_WRITE_ COMMAND	Signed Write	Devices that have paired and exchanged Connection Signature Resolving Keys (CSRKs) may use connection data signing. This is a procedure that appends a digital signature to the Signed Write ATT PDU. However, Bluetooth LE link encryption adds a Message Integrity Check field to all PDUs and so provides for both the authentication and confidentiality of PDUs. Therefore, signatures on Signed Write commands are not checked when sent over an encrypted link because the Message Integrity Code (MIC) is deemed to have provided the required authentication. EATT may only be used over an encrypted link, rendering the Signed Write PDU redundant and so it is not allowed when using an Enhanced ATT Bearer.

Table 2 - ATT PDUs which may not be used over an enhanced ATT bearer

The following ATT PDUs have had their specified behaviors refined to reflect the improved definition of an ATT bearer.

PDU Names	Previously Known As	Comment
ATT_PREPARE_ WRITE_REQ and ATT_PREPARE_WRITE_RSP ATT_EXECUTE_ WRITE_REQ and ATT_EXECUTE_WRITE_RSP	Prepare Write Request and Prepare Write Response Execute Write Request and Execute Write Response	Between them, these PDUs allow a series of characteristic write requests to be grouped into a single atomic operation. *Prepared writes* are queued using the ATT_PREPARE_WRITE_REQ PDUs and eventually are executed as a single atomic operation after the device receives an ATT_EXECUTE_ WRITE_REQ PDU. When used over an Enhanced ATT Bearer, the semantics of *prepared writes* are essentially unchanged except that each related set of *prepared writes* should be executed on the same bearer instance, which can be thought of as having a distinct queue for these PDU types. This allows each client to control the order in which the *prepared write* requests are processed. The specification provides further information about some of the behaviors of prepared writes with multiple enhanced ATT bearers under various circumstances.

2. LE Power Control

2.1 Background

This section examines a number of issues and concepts which have relevance to LE Power Control. Its purpose is to provide context for understanding the new feature, which is described in detail in section 2.2.

2.1.1 Power Control in Bluetooth BR/EDR

Bluetooth BR/EDR includes a power control capability. As at version 5.1 of the Bluetooth Core Specification however, no such capability was defined for Bluetooth LE.

2.1.2 Transmission Power and Electrical Power Consumption

When two connected Bluetooth devices communicate, at a given point in time, one is acting as a transmitter and the other as a receiver. The transmitter will emit its signal at a certain power level, sometimes referred to as *TxPower* and measured in dBm. The electrical power used by the device will increase with higher TxPower levels.

2.1.3 Path Loss

The level of a signal at the receiver will be lower than the original level at the transmitter. This is due to a phenomenon known as *path loss*. Path loss is caused by a number of factors, including the fact that the radio wave-front expands to cover a larger area over which the signal's energy is spread as it emanates outwards from the transmitter. This reduces the energy measured at any point on the wave front. Physical obstacles can reduce signal strength through absorption. Signals can reflect and refract, which also reduces the signal strength. Generally speaking, the further the receiver is from the transmitter, the greater will be the experience of path loss and the lower the received signal strength.

If the original transmit power is known to a receiver, then path loss can be used to estimate the distance that the receiver is from the transmitter.
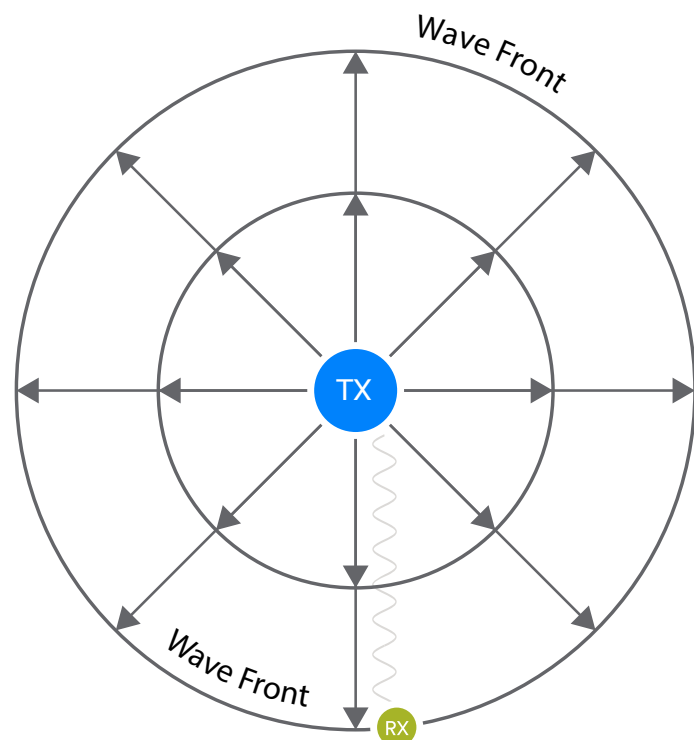

Figure 6 - The expanding wave front of a radio signal

2.1.4 Background Noise, Errors, and Range

Electromagnetic radiation from both man-made and natural sources exists in most environments and is generally referred to as *background noise*. The closer the level of a received signal gets to the lower level of this background noise, the harder it becomes to decode the transmitted signal and extract the encoded data within it without error. According to the Bluetooth Core Specification and its rules about receiver sensitivity, when the basic error rate (*BER*) exceeds 0.1%, communication is regarded as no longer working sufficiently well to be maintained. At this point, the connection between the devices will fail and we say that the receiver and transmitter are now out of range of each other.

2.1.5 Optimal Received Signal Strength

Receivers often have a preferred received signal strength range, within which their performance is optimal. This is known as the *golden range* or sometimes the *golden receive power range*.

Receivers encountering particularly high signal levels, may become saturated and this can cause the link between devices to fail. Conversely, when the receiver signal strength is too low, the error rate will increase and this can impact throughput, with packets that cannot be decoded without error needing to be retransmitted. Ultimately, when the error rate exceeds the permitted limit, the link will fail.

2.1.6 Low-Power Communication and Power Control

A goal of low-power wireless communication is to support the required communication use cases to a suitable level of reliability whilst using as little electrical power as possible.

2.1.7 Coexistence

Bluetooth operates in the 2.4 GHz range. It is not the only technology to use this range, and issues relating to shared use of a given part of the radio spectrum by different radio technologies is known as *coexistence*. Devices which are in- range of each other and using a given frequency must share that part of the radio spectrum in accordance with both applicable national or international regulations and accepted best practice.

2.1.8 Hysteresis

Hysteresis is a phenomenon exhibited by some systems, whereby there is a lag between a change in one property and another which has a causal relationship with it. It is relevant to the subject of radio signal strength and path loss.

2.2 About LE Power Control

This section reviews the new LE Power Control (LEPC) feature and the changes which have been made in the Bluetooth Core Specification version 5.2 in support of it.

2.2.1 Capabilities and Benefits

2.2.1.1 Power Management

The LE Power Control feature provides Bluetooth LE devices with the ability to exercise power management by optimizing transmit power levels dynamically. A receiving device, monitoring the level of the signal (the *RSSI*) from a connected device may request a change in the transmit power level used by its peer in either direction. It may, for example, ask the remote device to increase its transmit power level when the RSSI is getting lower or to reduce it when the RSSI is getting high and approaching the point at which saturation might be experienced. Transmitting devices may change their transmit power level autonomously and inform the other device that this has happened, along with various parameter values that include the new transmit power level.

The ability to exercise dynamic management of transmit power levels allows the optimization of the use of electrical power, especially when the range between devices often varies during normal use. No more power will be used than is needed for effective communication.

By keeping the RSSI within the range of levels that produce best performance from the receiver, the quality of the signal can be kept high and error rates low. This can yield a better experience for users in terms of their experience of throughput and responsiveness.

The new ability for Bluetooth LE devices to tune transmission power levels to no more than that which is required, improves coexistence in general and benefits all users of the 2.4 GHz frequency range, not just Bluetooth devices.

Connected devices which support LE Power Control must use it for power management purposes.

2.2.1.2 Path Loss Monitoring

Path loss is defined in the Bluetooth Core Specification, in the context of LE Power Control, as *the difference between the remote transmit power and the average local RSSI measurement for the connection*. The LE Power Control feature allows the host to define two or three path loss zones, known as Low, Middle, and (optionally) High. The controller can then inform the host of changes in the zone which the path loss falls within, using new Host Controller Interface (HCI) events.

2.2.2 Technical Highlights

In this section, key technical changes relating to LE Power Control from the Bluetooth Core Specification version 5.2 are described.

2.2.2.1 Link Layer

Three new Link Layer control PDUs have been introduced for LEPC. The new PDUs are *LL_POWER_CONTROL_REQ, LL_POWER_CONTROL_RSP* and LL_POWER_CHANGE_IND.

LL_POWER_CONTROL_REQ

This PDU allows a device to ask a connected peer to report various power control parameter values. It may also use the same PDU type to request that the connected device change its transmission power level for a specified PHY.

The LL_POWER_CHANGE_REQ PDU includes the three fields shown in table 3.

Field	Description
PHY	An 8-bit collection of binary flags which identify the PHY(s) that the power control request relates to. 4 of the available 8 bits have meanings defined: Bit 0 : LE 1M Bit 1 : LE 2M Bit 2 : LE Coded with S=2 Bit 3 : LE Coded with S=8 Consult the Bluetooth Core Specification for further details about each PHY type.
Delta	The amount by which the sending device is requesting that the recipient device adjust its TxPower. This may be a positive or negative amount measured in dB or one of two special values: 0×00 or 0×7F. A value of zero allows the sender to query the recipient's power control parameters without it making a change. 0×7F indicates that the device receiving this PDU should increase its TxPower to the maximum level that it supports.
TxPower	This field must be set to the transmit power level of the device sending this PDU and is expressed in dBm. If this value is not available, it must be set to 0×7F.

Table 3 - LL_POWER_CONTROL_REQ

LL_POWER_CONTROL_RSP

This PDU is sent as the response to a LL_POWER_CONTROL_REQ PDU. It contains the fields described in table 4.

Field	Description
Min	A single bit flag which is set if the sender device is currently using the minimum TX power level it supports.
Max	A single bit flag which is set if the sender device is currently using the maximum TX power level it supports.
Delta	The actual change in the transmit power which was made in response to the request to which this response PDU relates. Measured in dB.
TxPower	The transmit power level of the specified PHY, in dBm or one of two special values. 127 means the value is unavailable. 126 means the device is not managing power levels for the PHY specified in the request PDU.
APR	Acceptable Power Reduction. Indicates the largest decrease in transmit power level that the device sending this PDU could now accommodate.

Table 4 - LL_POWER_CONTROL_RSP

LL_POWER_CHANGE_IND

This PDU is used to notify a remote device whenever the sender's transmit power level has been changed autonomously for any one or more PHYs, when the maximum power level on the PHY currently used for transmit purposes has been changed or when management of power levels for this PHY has started or has been stopped. The fields this PDU contains are the same as those in a LL_POWER_CONTROL_RSP PDU except that it does not include the APR field.

2.2.2.2 Host Controller Interface Commands and Events

HCI defines several new commands and events that are related to LE Power Control. Both power management and path loss monitoring use cases are provided for.

New HCI Commands

Command	Description
LE Enhanced Read Transmit Power Level	Allows the host to determine the values of the currently used and maximum transmit power levels of a specified connection in its local controller for each PHY.
LE Read Remote Transmit Power Level	Allows the host to determine the transmit power used by a remote device on a specified connection. The local controller may satisfy this request using data it already possesses from a previously received LL_POWER_CHANGE_IND or LL_POWER_CONTROL_RSP PDU. Alternatively, it may send the remote device a LL_POWER_CONTROL_REQ and use the data contained within the response. If the controller has no previously obtained data, it has no choice but to send a request to the remote device. The maximum age that previously obtained power level data may be before it is deemed necessary to request new data from the remote device is an implementation decision.
LE Set Path Loss Reporting Parameters	Allows the host to specify the path loss reporting parameters which the controller must use for a specified connection. Parameters include high and low path loss thresholds which the controller must observe being crossed for a specified minimum amount of time. Each threshold is expressed as an absolute path loss amount measured in dB and an associated hysteresis amount.
LE Set Path Loss Reporting Enable	Allows the host to enable or disable path loss reporting from the controller.
LE Set Transmit Power Reporting Enable	Allows the host to enable or disable transmit power reporting from either or both of the local and remote controller. A connection handle parameter identifies the connection to which the command applies.

New HCI Events

Event	Description
LE Transmit Power Reporting	This event is generated whenever the local transmit power changes and local reporting has been enabled with the *LE Set Transmit Power Reporting Enable* command. It is also generated whenever the remote transmit power has changed, if remote reporting has been enabled. It includes the following parameters:
	Reason: 00: local transmit power changed, 01: remote transmit power changed, 02: An *LE Read Remote Transmit Power Level* command completed.
	PHY: The PHY to which the change relates.
	Transmit Power Level: The transmit power in dBm or 0×7E (remote device is not managing power levels on this PHY) or 0×7F (not available).
	Transmit Power Level Flag: 2 bits which between them can indicate the related transmit power being at its minimum (bit 0) and/or maximum (bit 1) level.
	Delta: The change in transmit power in dB.
LE Path Loss Threshold	Sent when a path loss threshold, defined by a LE Set Path Loss Reporting Parameters command, is crossed.

2.2.2.3 Discovering Support for LE Power Control

Devices can only use a particular LE power control procedure if both devices support the relevant parts of this feature. Therefore, a way of discovering feature support information about devices is required.

The Link Layer defines the Feature Exchange procedure and this allows peer devices to exchange a bit mask called the FeatureSet. The bit values in the FeatureSet field indicate which of the various link layer features a device supports. Three of the previously unused bits in the FeatureSet field have been assigned for the purpose of indicating support for aspects of the LE Power Control feature, as shown in table 5.

Bit	Feature
33	LE Power Control Request
34	LE Power Change Indication
35	LE Path Loss Monitoring

Table 5 - LE Power Control and Link Layer FeatureSet bits

3. LE Isochronous Channels

3.1 Background

This section reviews a number of basic concepts as well as pertinent aspects of version 5.1 of the Bluetooth Core Specification to provide context and background which will support the examination of the new Bluetooth LE Isochronous Channels feature in section 3.2 below. Where a Bluetooth Core Specification version is not explicitly stated in this section, it should be assumed that version 5.1 is being referred to.

3.1.1 Physical Layer Variants

Bluetooth LE defines several physical layer variants, known as PHYs. The supported PHYs offer different symbol rate and range capabilities.

Phy	Characteristics
LE 1M	1 mega-symbol per second symbol rate. The default Bluetooth LE PHY which all LE implementations must support.
LE 2M	2 mega-symbol per second symbol rate offering a higher communication speed than LE 1M. Support for LE 2M is optional.
LE Coded (S=2)	Uses forward error correction to improve reliability at lower signal-to-noise ratios and hence increases range while reducing the data rate. An approximate doubling of range can be achieved with the parameter S set to 2, with data rate being halved. Support for LE Coded is optional.
LE Coded (S=8)	Uses forward error correction to improve reliability at lower signal-to-noise ratios and hence increases range while reducing the data rate. An approximate quadrupling of range can be achieved with the parameter S set to 8, with data rate being reduced to one eighth. Support for LE Coded is optional.

3.1.2 Slots and Channels

Modern, digital radio communications technologies employ one or more schemes which allow the radio medium to be shared by multiple transmitting devices. These schemes are typically referred to as multiple access methods. The issue being addressed by such schemes is that given a collection of radio devices, all within range of one another, only one may transmit on a particular frequency at a time. If more than one transmission occurs on a particular frequency at the same time, packets are said to *collide* and data becomes

corrupted and must be retransmitted or be lost.

Bluetooth uses a combination of two multiple access methods to share the radio medium. They are called Frequency Division Multiple Access (FDMA) and Time Division Multiple Access (TDMA).

FDMA involves using a series of different radio frequencies to support communication. Packets transmitted at the same time on different frequencies, cannot collide and so multiple devices may communicate simultaneously. Bluetooth LE has 40 distinct radio channels available to it, each of which is 2MHz wide.

TDMA involves defining time slots during which transmission may take place. Receiving devices are required to respond after a specified time interval. In the Bluetooth Core Specification, a slot in which transmission may occur is called an event.

Using FDMA and TDMA in combination makes Bluetooth able to support large numbers of devices in a given environment.

Bluetooth uses a channel selection algorithm to choose a radio channel for each event. The quality associated with each channel is monitored, and the system adapts to the conditions encountered, ceasing to use channels which are found to be experiencing problems, perhaps due to unusually high use by other (not necessarily Bluetooth) devices. This has the benefit of making Bluetooth communication robust in busy radio environments.

3.1.3 The Bluetooth Data Transport Architecture

The architecture of Bluetooth defines a number of concepts, which collectively constitute the Bluetooth data transport architecture. Key amongst these concepts are the Physical Channel, Physical Link, Logical Link, and Logical Transport. Certain combinations are defined for use in support of different application types, each with particular requirements regarding issues such as topology, timing, reliability, and channel use.

A Physical Channel defines one of several different ways of communicating using Bluetooth. For example, communication can take place between two connected devices using the LE Piconet Physical Channel, which involves adaptive frequency hopping across 37 channels. Alternatively, the LE Advertising Physical Channel can be used for broadcast, connectionless communication from one device to an unlimited number of other devices. The LE Periodic Physical Channel can be used to broadcast data too, but on a regular basis with a deterministic time schedule. Observer (receiver) devices are able to determine that time schedule and use it to synchronize their scanning schedules.

A Physical Link is based on a single, specific physical channel and specifies certain characteristics of that link, such as the use or otherwise of power control.

Logical links and transports have various parameters which are designed to provide a suitable means of supporting a particular set of data communication requirements over a physical link, using a particular physical channel type.

For example, reliable, bi-directional, point- to- point communication in Bluetooth LE uses the LE asynchronous connection logical (ACL) transport, with either an LE-C link for control data or an LE-U link for user data, over a physical link based on the LE Piconet Physical Channel.

Physical Channel LE Piconet	Physical Link	Logical Transport LE asynchronous connection (ACL)	Logical Link LE-C or LE-U

On the other hand, unreliable, uni-directional broadcast communication in Bluetooth LE uses the LE Advertising Broadcast (ADVB) logical transport, with either an ADVB-C link for control data or an ADVB-U link for user data, over a physical link based on the LE Advertising Physical Channel.

Physical Channel LE Advertising	Physical Link	Logical Transport LE Advertising Broadcast (ADVB)	Logical Link ADVB-C or ADVB-U

3.1.4 Extended Advertising

Advertising is a Bluetooth LE procedure which provides a connectionless communication mechanism. It is used either to allow the advertising device to be discovered so that it can be connected to by another device or as a means of communicating broadcast data to one or more other, unconnected devices. Devices receiving broadcast advertising data are called *observers*.

Advertising uses two sets of channels, known as the primary and secondary channels. There are three primary channels and another thirty-seven secondary channels. Large advertising data payloads may be offloaded to one or more packets sent on the secondary channels. This allows larger amounts of data to be broadcast and brings all 40 Bluetooth LE channels into play for advertising purposes.

The scheduling of standard advertising events is not deterministic and includes a degree of intentional, random timing variation that is designed to avoid persistent packet collisions. A special advertising mode called Periodic Advertising exists, however, and this allows advertising events to be scheduled according to fixed, deterministic intervals. A procedure, known as the periodic advertising synchronization establishment procedure, allows observer devices to determine the periodic advertising schedule of the advertising device so that they can synchronize their scanning schedule with it.

3.1.5 Bluetooth Audio

Bluetooth technology is used in a diverse range of product and application types. The earliest uses of Bluetooth gave rise to products like the wireless mouse. In more recent years, Bluetooth has gained the ability to form networks of tens of thousands of devices or to provide high-accuracy, location-based services.

Audio, however, continues to be the largest application category for Bluetooth devices, with about 1 billion devices forecast to have shipped in 2019. It's also

one of the longest established uses of Bluetooth, with devices such as hands-free, mobile headsets amongst the very-first-ever Bluetooth products.

The Advanced Audio Distribution Profile (A2DP) defines how Bluetooth can be used for high-quality audio applications, such as the streaming of music in a point-to-point topology from a smartphone to a set of Bluetooth headphones. A2DP is a very widely adopted profile that has been successful in delivering a quality listening experience to music lovers, free from the inconvenience of cables. A2DP uses the older Bluetooth BR/EDR technology as opposed to the newer Bluetooth LE.

New audio use cases and product ideas have been emerging for some time, and, in some cases, A2DP is not able to meet these new requirements. In particular, the topologies it can support, which define the permitted relationships between audio sources and *sinks* (devices like speakers which receive and *render* audio) is limited to one or more point-to-point arrangements. There is no way to make sure that multiple sinks render their audio stream at exactly the same time, so that playback is synchronized across a series of associated devices. Consequently, the greater majority of devices which use A2DP will only work when connected to one other device at a time.

3.2 About LE Isochronous Channels

This section reviews the new Bluetooth LE Isochronous Channels feature and related changes which have been made in the Bluetooth Core Specification version 5.2.

3.2.1 Capabilities and Benefits

3.2.1.1 Time-Bound Data and Synchronized Processing

The Bluetooth LE Isochronous Channels feature is a new way of using Bluetooth LE to transfer time-bounded data between devices. It provides a mechanism that makes sure that multiple sink devices, receiving data from the same source, will render it at the same time. Data has a time-limited validity period, at the end of which it is said to expire. Expired data which has not yet been transmitted, will be discarded. This means that receiver devices only ever receive data which is valid with respect to rules regarding its age and acceptable latency.

3.2.1.2 New Audio Use Cases

LE Isochronous communication was primarily designed for use in audio products and systems. It provides the means by which audio, delivered from a source to multiple sinks, can be rendered at the same time, for properly synchronized playback. Audio, which for some reason is delayed after being generated at the source, expires and is discarded so that it does not affect the listening experience at the sink(s).

> New use cases, new topologies, and, ultimately, new product types will be possible with the new Bluetooth LE isochronous channels feature. The use of Bluetooth LE for audio is called LE Audio.

Here are a few examples of possible LE Audio scenarios.

Use Case	Description
Personal Audio Sharing	Groups of friends can simultaneously enjoy the music playing on one smartphone via their Bluetooth headphones. This is an example of a private group of audio sink devices sharing a single audio source.
Public Assisted Hearing	The dialogue of a theater production can be broadcast such that all Bluetooth LE hearing aid users in the audience can hear it.
Public Television	At the gym, all attendees with Bluetooth LE headphones or ear buds can listen to the television audio track.
Multi-Language Flight Announcements	Passengers on an aircraft could be able to connect their headphones to the flight information system, specify their preferred language, and hear flight information in that language.

3.2.2 Technical Highlights

3.2.2.1 Isochronous Communication and the Bluetooth Data Transport Architecture

The Bluetooth data transport architecture has been enhanced to support isochronous channels, which may be connection-oriented or connectionless.

In both cases, isochronous communication uses the new LE Isochronous Physical Channel. This physical channel uses frequency hopping and specifies the timing of the first packet, which then acts as an anchor point for the timing of subsequent packets. LE-S logical links are used for streaming data such as audio, which has no inherent structure whereas LE-F is intended for use with framed data packets.

Any one of the four Bluetooth LE PHYs (see 3.1.1) may be used with the new LE Isochronous Physical Channel.

Connection-oriented isochronous channels use the LE-CIS (LE Connected Isochronous Stream) logical transport and support bi-directional communication. A single LE-CIS stream provides point-to-point isochronous communication between two connected devices. A flushing period is specified for the LE-CIS logical transport. Any packet which has not been transmitted within the flushing period will be discarded.
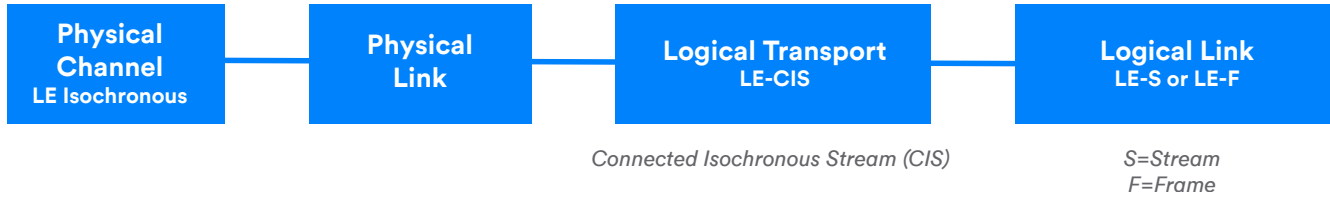
Physical Channel LE Isochronous	Physical Link	Logical Transport LE-CIS	Logical Link LE-S or LE-F

Connected Isochronous Stream (CIS)

S=Stream
F=Frame

Figure 7 - Connection-oriented isochronous channel data transport

Connectionless isochronous communication uses Broadcast Isochronous Streams (BIS) and only supports uni-directional communication. LE-S or LE-F logical links over the LE Isochronous physical channel are used for user data with the new LEB-C broadcast control logical link used for control requirements, such as the communication of channel map updates. A single BIS can stream identical copies of data to multiple receiver devices.
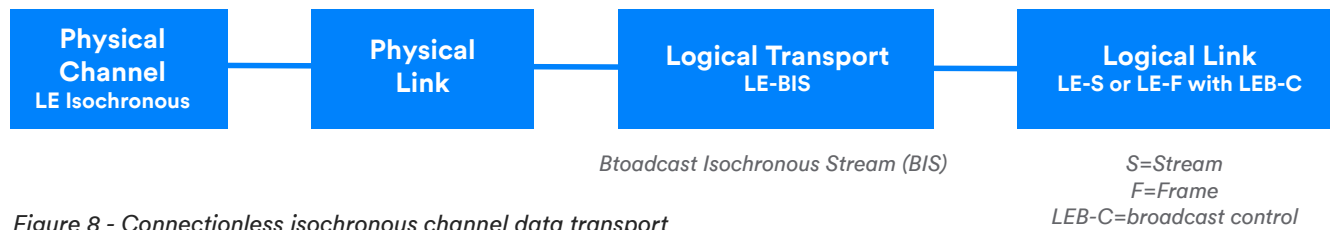
Physical Channel LE Isochronous	Physical Link	Logical Transport LE-BIS	Logical Link LE-S or LE-F with LEB-C

Btoadcast Isochronous Stream (BIS)

S=Stream
F=Frame
LEB-C=broadcast control

Figure 8 - Connectionless isochronous channel data transport

3.2.2.2 Groups, Streams, Events, and Sub-Events

CIS streams are members of groups called Connected Isochronous Groups (CIG), each of which may contain multiple CIS instances. Within a group, and for each CIS, there exists a schedule of transmission and receiver time slots known as events and subevents.

CIS instances in the same CIG have common timing reference data, which is used in the synchronization of isochronous data processing (typically, audio rendering) by receivers across all of the streams in the group.

Each event occurs at a regular interval called the ISO Interval, which may be anywhere in the range 5ms to 4s in multiples of 1.25ms. Each event is divided into one or more subevents. In a connected isochronous stream, during a subevent, the Central (C) transmits once and the Peripheral (P) responds as shown in Figure 10. Note that the channel
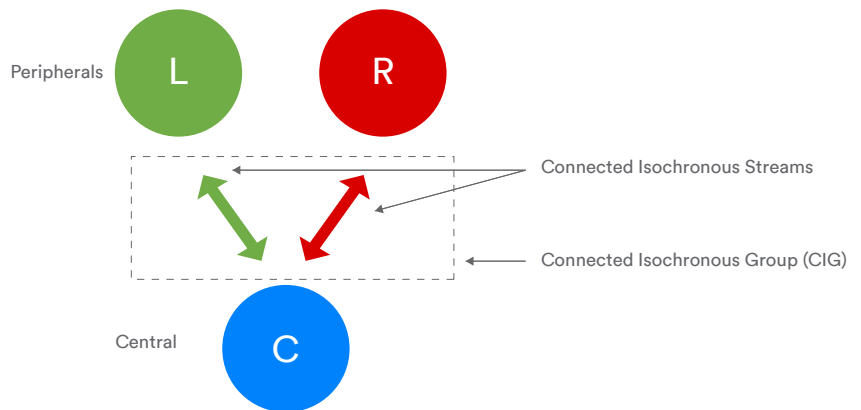

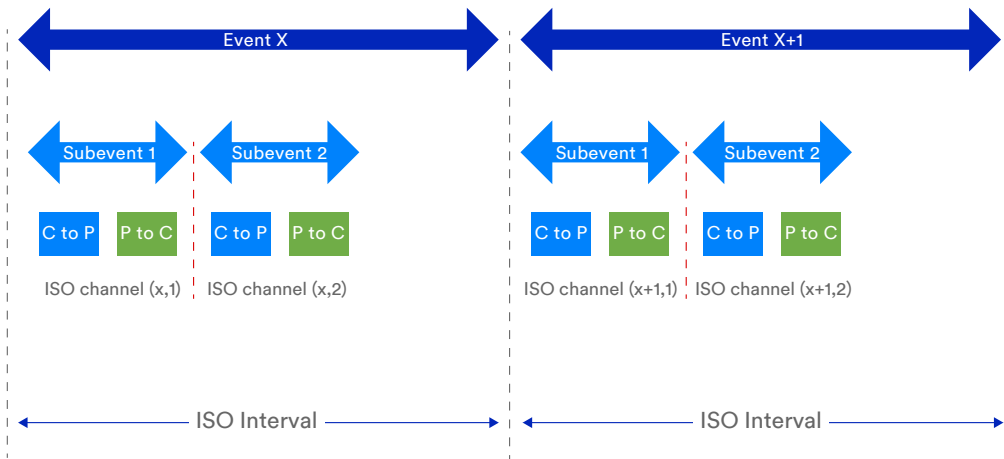*Figure 9 - Connection-oriented isochronous streams and group servicing left and right stereo ear buds*
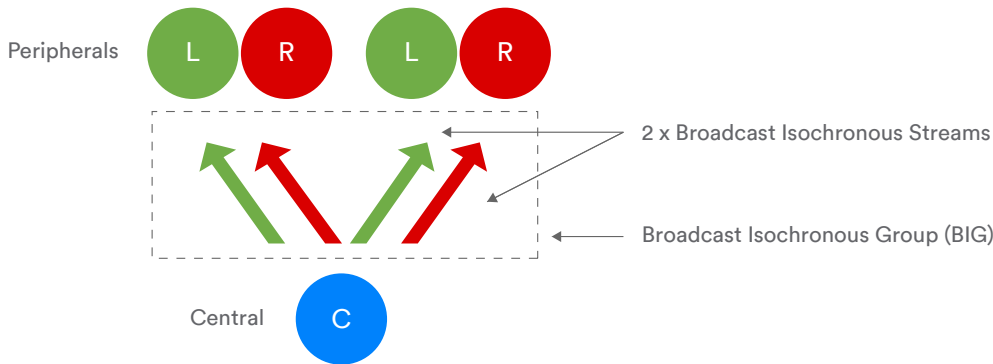
Figure 10 - CIS events and subevents

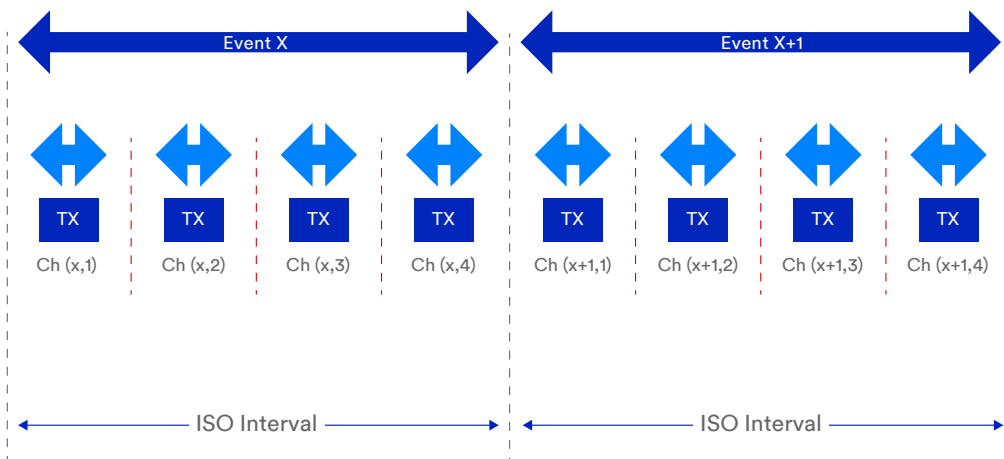*Figure 11 - Broadcast isochronous stream and group*

Figure 12 - BIS events and subevents

is changed at each subevent.

Multiple CIGs may be created by the Central device.

BIS streams are members of groups called Broadcast Isochronous Groups (BIGs), each of which may contain multiple BIS instances. Multiple BIGs may be created by the Central device.

Within a BIG, and for each BIS, there exists a schedule of transmission time slots known as events and subevents.

BIS instances in the same BIG have common timing reference data, which is used in the synchronization of broadcast isochronous data processing across the broadcast streams in the group.

When using a Broadcast Isochronous Stream (BIS), a device broadcasts either an isochronous data packet or broadcast control information at each subevent. Devices receiving the broadcast cannot respond to it.

3.2.2.3 Retransmissions and Reliability

Reliability may be enhanced using retransmissions of identical packets within successions of subevents on either BIS or CIS streams. In the case of BIS, retransmissions are unconditional, whereas with CIS, they occur when the Peripheral has not acknowledged a transmission.

In the case of BIS, because there is no requirement to reserve slots for Peripheral responses (as is the case with CIS), twice as many subevents may be scheduled for transmissions during a given amount of air time, so there is a greater opportunity for reliability-enhancing retransmissions.

Retransmissions, due to their occupying distinct subevents, are transmitted on different channels, and selected channels must be at least 6 MHz from the last transmission. This helps mitigate potential packet loss due to interference on a particular channel.

3.2.2.4 Synchronization and Group Events in Connected Isochronous Communication

Synchronized processing of isochronous data, transmitted over Connected Isochronous Streams, is achieved through a concept called the CIG Event and a number of synchronization timing parameters.

Consider the case where audio from a smartphone (the *source*) is to be played in the left and right buds (the two sinks) of a pair of Bluetooth LE in-ear headphones. The left and right buds each establish a CIS stream with the source device. The streams are both members of the same CIG group. A fragment of audio produced by the source is encoded into a packet and a copy is transmitted to each sink device over its stream, one at a time during a series of successive
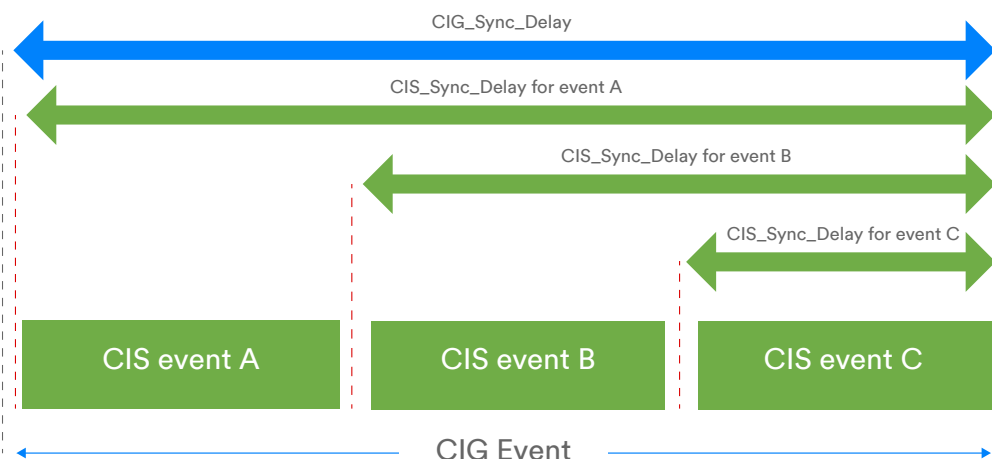

Figure 13 - CIG Event and sync delays

CIS events. Playback must not occur until all devices in the CIG have received the packet and this will take an elapsed time known as the CIG_Sync_Delay. Each CIS event has a delay parameter called the CIS_Sync_Delay which is equal to the CIG_Sync_Delay at the first CIS event but which is then progressively reduced at each subsequent CIS event for the group. This means that packets transmitted earlier have to wait longer before being rendered by

their sink devices, whereas packets transmitted later in the group's event processing wait a shorter time. Higher-layer specifications, such as profiles, may stipulate the use of a further *presentation delay* to use in the calculation of the time at which data should be rendered. The net of this stepped delay system is that each sink device will render the received audio at the same time.

The transmission of a packet to all connected sink devices in a group takes place in a *CIG Event*.

Figure 13 illustrates the way CIS events and sync delay parameters work together to deliver a synchronized audio playback capability to multiple, connected audio devices.

There are two ways in which CIS subevents may be scheduled within this scheme. In the first case, all subevents belonging to a CIS event are scheduled in an uninterrupted sequence. CIS events do not overlap, and all subevents in a CIS event are dealt with, one at a time, before moving on to the next CIS event. This is known as sequential subevent scheduling.

In the second case, CIS events overlap, and the first subevents of each CIS event are handled one after the other and then the second subevent of each CIS event and so on. This is known as interleaved subevent scheduling.

Note that there may be a maximum of 31 CIS instances per CIG.

3.2.2.5 Synchronization and Group Events in Connectionless Isochronous Communication

A device which is broadcasting isochronous data is known as an Isochronous Broadcaster. Other devices wishing to receive data for a particular BIG from an Isochronous Broadcaster must execute a procedure called the Broadcast Isochronous Stream Establishment Procedure. There are two stages to this procedure. First, the receiver must synchronize to a periodic advertising train that is associated with the required stream. AUX_SYNC_IND periodic advertising PDUs contain a field called BIG Info and the data it contains is then used to synchronize with the required BIS. Once synchronization with the BIS has been accomplished, the receiver is known as a Synchronized Receiver.

In the process of establishing the stream, the receiver may also calculate a group session key with which to secure communication (see 3.2.2.8 Stack Impact Summary).

Multiple devices have identical packets delivered to them simultaneously when using connectionless isochronous communication, but synchronized processing at receivers is still an issue. Consider the case where audio is shared with a group of users, each of which is using left and right stereo ear buds. Left and right stereo content must be delivered sequentially to each of the left and right ear buds using different broadcast streams, but they must still be rendered simultaneously.

To address this requirement, sets of Broadcast Isochronous Streams across which synchronized rendering is required must be members of the same Broadcast Isochronous Group. A delay parameter, BIG_Sync_Delay, defines the duration of a full BIG event. The start of the event is termed a BIG Anchor Point and the end of the event a BIG Synchronization Point. These concepts and some other timing parameters allow synchronized rendering of

the data from multiple streams within a BIG to take place.

BIS events involve the transmission and receipt of new BIS Data and BIS Control PDUs.

There may be a maximum of 31 BIS instances per BIG.

3.2.2.6 The Isochronous Adaptation Layer (ISOAL)

ISOAL is a new layer in the Bluetooth stack which sits in the controller above the link layer. It provides flexibility in the way that lower layers of the stack and upper layers work together, allowing the size of isochronous data packets as created and consumed by upper stack layers to be different to the size used by the CIS or BIS logical transport in the link layer. This is accomplished through a segmentation and reassembly service being applied to SDUs to/from the upper layer and fragmentation and recombination being applied to SDUs to/from the link layer.

ISOAL also allows the upper layer to use timing intervals that differ from those used by the link layer so that the rate of SDUs exchanged with the upper layers is not the same as the rate with which they are exchanged with the link layer.

The Host Controller Interface (HCI) will usually be used as the interface from the upper layer to the ISOAL.

3.2.2.7 Security and LE Isochronous Channels

Connection-oriented isochronous communication uses Bluetooth LE security features in the same way as any other connection-oriented application. Devices requiring security must be paired and the security of the communication link stems from this.

Connectionless isochronous communication using the Broadcast Isochronous Stream (BIS) necessitated the introduction of a new LE security mode 3.

LE security mode 3 defines three security levels:
1. No security (no authentication and no encryption)
2. Use of an unauthenticated Broadcast Code
3. Use of an authenticated Broadcast Code

Broadcast Code is a value which is associated with a secure Broadcast Isochronous Group. Users, wishing their device(s) to participate in the BIG must provide their device with the Broadcast Code using a suitable input mechanism.

From the Broadcast Code, a Group Long Term Key (GLTK) is calculated. A Group Session Key (GSK) is then calculated using the GLTK and a Group Session Key Diversifier, which is broadcast in extended advertising packets by the Isochronous Broadcaster in the BIGInfo field. The GSK is used to encrypt and decrypt data broadcast and received in a BIG.

Note that when Broadcast Code appears in a user interface, it will be named *Bluetooth Privacy Code.*

3.2.2.8 Stack Impact Summary

To support LE Isochronous Channels, various parts of the Bluetooth stack have changed.

- A new layer called the Isochronous Adaptation Layer has been added to the controller (see 3.2.2.5)

- The data transport architecture has been modified to support connection-oriented and connectionless isochronous communication.

- There's a new LE security mode 3 which is based around the use of a shared Broadcast Code to allow encryption to be used in Broadcast Isochronous Groups.

- Generic Access profile defines new procedures for setting up isochronous communication in connections or via broadcasting.

- The HCI layer has a number of new commands and events which allow isochronous communication to be configured and used.

- The Link Layer has a number of new PDUs including the Connected Isochronous PDU and the Broadcast Isochronous PDU. LL_CIS_REQ and LL_CIS_RSP are used to create a Connected Isochronous Stream over a connection.

Bluetooth Core Specification version 5.2 should be consulted for full details. ■