How to Build an Auracast™ Transmitter

This document covers the basic design principles for building an Auracast™ transmitter, explaining the features and the configuration requirements. It also discusses a range of different considerations for physical form factors.

Author: Market Development

Version: 1.0

Revision Date: 14 May 2024

Table of Contents

Table of Contents

1.0 Designing Auracast™ Transmitters

The Auracast™ transmitter is the simplest of the three devices that make up the Auracast™ ecosystem, alongside the Auracast™ receiver and Auracast™ assistant. In this document, we will look at the design considerations that an implementer needs to take when putting together a design for an Auracast™ transmitter. It should be read in conjunction with the Auracast™ transmitter guidelines document [1]. which provides specific information on the requirements for an Auracast™ transmitter.

1.1 Nomenclature

The Bluetooth® LE Audio specifications each introduce a variety of different roles taken by the three devices that comprise the Auracast™ ecosystem. As using all of these can be confusing when providing an overview, this document introduces and uses the generic, descriptive names of Auracast™ transmitter, Auracast™ receiver, and Auracast™ assistant. These are used to describe both the physical devices and their roles. They are based on the definitions of roles in the Basic Audio Profile, but also encompass the roles identified in various higher level specifications which are listed in Table 1.1, which are the Basic Audio Profile (BAP)[2], the Public Broadcast Profile (PBP) [3], the Common Audio Profile (CAP) [4], the Telephony and Media Audio Profile (TMAP) [5], and the Hearing Access Profile (HAP) [6].

Name	Includes the role of	Specification
Auracast™ Transmitter	Broadcast Source Public Broadcast Source Initiator Broadcast Media Sender	BAP 2.2.2.1 PBP 3.1 CAP 2.1.1 TMAP 3.5.2
Auracast™ Receiver	Broadcast Sink Public Broadcast Sink Acceptor Hearing Aid Broadcast Media Receiver	BAP 2.2.2.2 PBP 3.2 CAP 2.1.2 HAP 3.2 TMAP 3.5.2
Auracast™ Assistant	Broadcast Assistant Public Broadcast Assistant Commander	BAP 2.2.2.3 PBP 3.3 CAP 2.1.3

Table 1.1: Underlying specification roles covered by the Auracast™ terminology in this document

1.2 Audience

This document assumes that readers have a basic knowledge of Bluetooth Low Energy (LE) and are familiar with concepts like GATT, characteristics, and client/server architectures. A basic understanding of Bluetooth LE Audio is also assumed. An overview of the Auracast™ ecosystem is available in the companion document, An Overview of Auracast™ Broadcast Audio [7].

2.0 Introduction

Auracast™ transmitters can be implemented in many different forms. They can be designed as:

- Standalone devices which accept either an analog or digital audio input or have a built-in microphone or other internal audio source
- Modules which are incorporated inside devices such as TVs
- Chip-level implementations inside mobile phones, laptops, and tablets

They may be targeted at consumers who will plug them into the audio output of an existing device and expect them to work out of the box, or they can be part of complex, managed audio installations in theatres, conference centers, and airports.

The flexibility of the Bluetooth® LE Audio specifications and the multiplicity of applications means that there will be many different design decisions that need to be made by anyone building an Auracast™ transmitter. This document will discuss the different options and explain how they can be addressed.

2.1 An Overview of the Auracast™ Topology

The three devices that make up the Auracast™ topology work in very different ways.

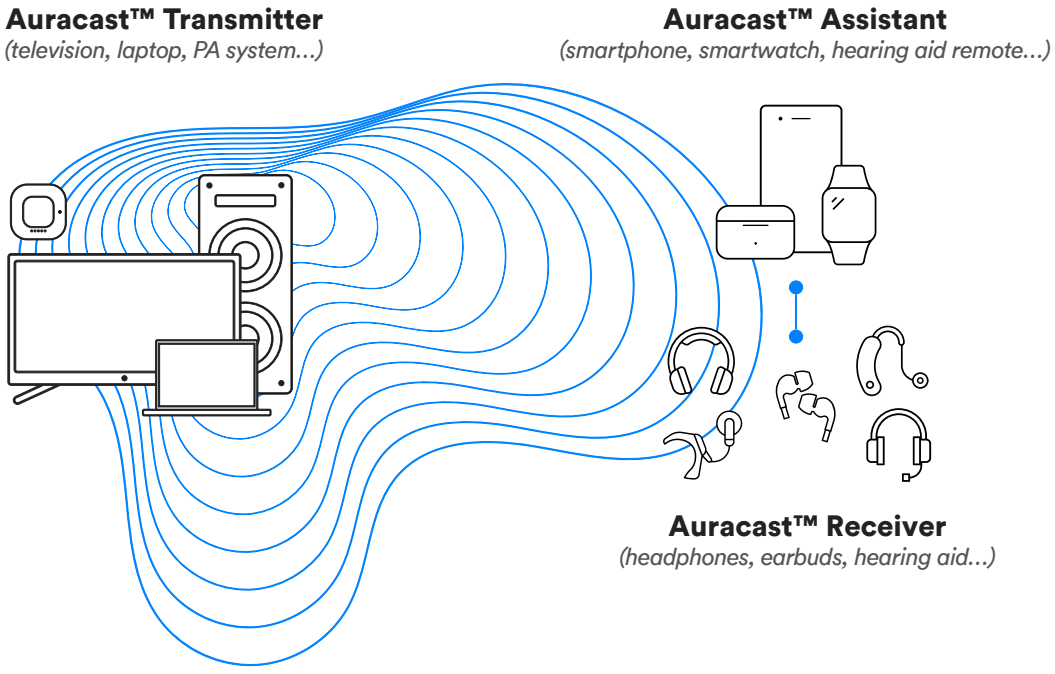

Figure 2.1: The three elements of the Bluetooth® LE Audio broadcast topology

The Auracast™ transmitter operates autonomously with no idea of whether any other device is listening to it. It does not need to be paired to any other device. In many cases, once it is turned on, it will start transmitting. Any number of Auracast™ receivers can listen to it, but the Auracast™ transmitter never knows whether any of them are.

If more than one Auracast™ transmitter is within range of an Auracast™ receiver, then the Auracast™ receiver will normally use one or more Auracast™ assistants to help it find them and select which one to listen to. How to design Auracast™ receivers and assistants is covered in two companion documents [8] and [9].

2.2 The Basic Elements of an Auracast™ Transmitter

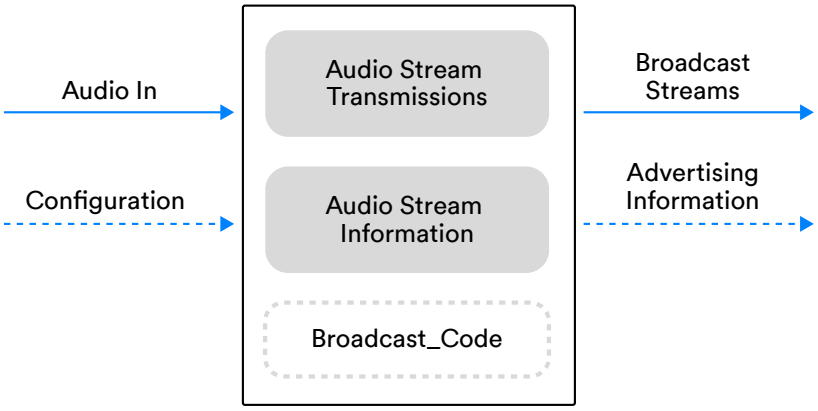

Figure 2.2: The basic elements of an Auracast™ transmitter

In essence, there are three things that an Auracast™ transmitter needs to do:

- Convert the incoming audio into the correct encoded format, using the LC3 codec, and then transmit it
- Expose configuration information about the audio streams using advertisements to allow Auracast™ assistants and Auracast™ receivers to discover them
- Store a Broadcast_Code if the audio streams need to be encrypted

Each Auracast™ transmitter needs to be configured in terms of the number of streams it transmits and how they're encoded, but, for most transmitters, the bulk of that will be set at manufacture. Some of the parameters that identify the Auracast™ transmitter, such as its name, should be configurable by the user. For a TV or smartphone acting as an Auracast™ transmitter, this will normally be accessed via a menu, but, for other Auracast™ transmitters, a setup application should be supplied.

At its most basic, an Auracast™ transmitter only needs to use two of the Bluetooth® LE Audio specifications:

- The Basic Audio Profile (BAP) [2]
- The Public Broadcast Profile (PBP) [3]

These are shown in Figure 2.3. An Auracast™ transmitter requires a chip which supports the Bluetooth LE Audio features of Bluetooth Core Specification version 5.2 or above [10]. It does not need to support any of the classic BR/EDR Bluetooth features.

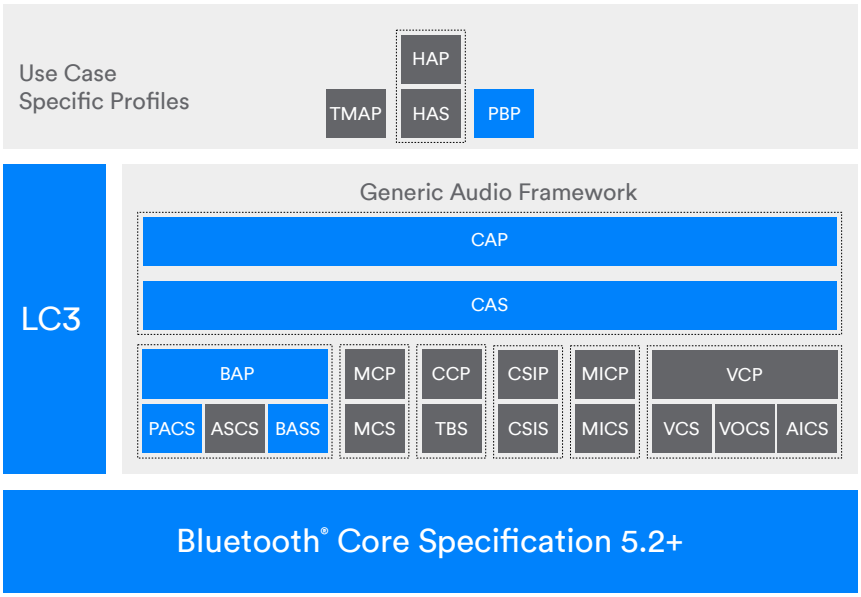

Figure 2.3: The Bluetooth® LE Audio specifications required for an Auracast™ transmitter

2.3 The Design Steps for an Auracast™ Transmitter

There are some basic design choices which need to be made to design an Auracast ™ transmitter, which everyone needs to follow. They are:

1. Decide the market segment. Is it consumer or professional? This will dictate the features and the configuration and commissioning requirements.
2. Decide the number and type of audio inputs which need to be supported and the conversion requirements for each of them.
3. Decide the number of output broadcast streams. This normally determines the number of LC3 codec instances.
4. Decide on the codec configurations required. All Auracast™ transmitters must support at least one broadcast stream that uses one of the Standard Quality codec configurations which are defined in PBP. Support for other codec configurations is optional.
5. Decide if the broadcast audio streams need to be encrypted.

Once you've worked those out, you're ready to start designing your Auracast™ transmitter. The information in the following sections will guide you through the process.

2.4 The Basics of Broadcast Audio Streams

It's important to understand how an Auracast™ transmitter takes in its audio input and converts it into one or more broadcast streams. This is defined in BAP. The specification is very flexible, allowing an extremely wide range of configurations for an Auracast™ transmitter, which is shown in Figure 2.4.

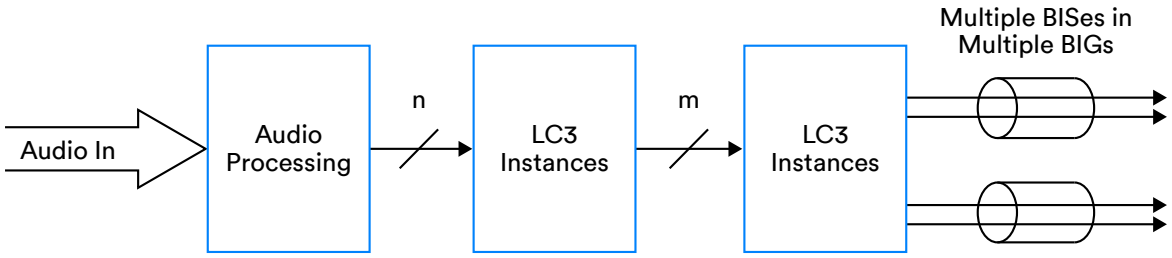

Figure 2.4: Generic architecture of an Auracast™ transmitter

The diagram shows an audio input on the left. It may be a mono stream, a stereo stream, an arbitrary number of different language streams, or multichannel sound. It may be analog, digital, or optical from an external source, or generated internally as in a phone, TV, media player, or microphone. These inputs are processed into a number of different streams (n) for LC3 encoding and subsequent Bluetooth® transmission.

Some implementations may support multiple input audio formats with an option for the user to select which they want to use. Audio inputs will generally need to be converted into PCM streams for the LC3 codec, although that will depend on the specific requirements of the platform being used for the Auracast™ transmitter. The input conversion is outside the scope of the Bluetooth specifications and developers will need to implement this based on their product configuration.

In most implementations, the number of output streams (m) will be the same as the number of input audio streams (n) with a direct mapping from input to output. However, an Auracast™ transmitter may decide to encode the same audio input stream at different codec settings, resulting in more transmitted audio streams than input streams. Auracast™ transmitters which want to support both Standard and High Quality streams at the same time need to do this (Standard and High Quality stream configurations are defined in PBP [3].). The Auracast™ transmitter may also downmix a stereo input to a single mono output. These are all implementation decisions.

Each input stream requires its own LC3 instance to encode it before it is passed on for transmission as an independent1 Broadcast Isochronous Stream (BIS). Depending on the Auracast™ transmitter configuration, these may all be contained within a single Broadcast Isochronous Group (BIG) or separated into multiple Broadcast Isochronous Groups.

In Figure 2.4 and subsequent figures illustrating broadcast isochronous configurations, single arrows represent a BIS and the larger pipe around them represents a BIG as shown below.

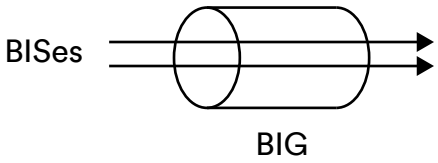

Figure 2.5 Representation of BISes and a BIG

1 *The Bluetooth® LE Audio specifications allow encoded audio streams to be multiplexed into a single audio stream after LC3 encoding. This is not recommended for Auracast™ applications, as it increases the resources required in earbuds and hearing aids and is not a mandatory requirement for Auracast™ receivers.*

3.0 Transmitting Auracast™ Streams

In this section, we will look at the most common stream arrangements for Auracast™ transmitters. The Bluetooth® LE Audio architecture is very flexible, but the vast majority of implementations will be designed to support a fixed stream configuration for the life of the product.

3.1 The Mono Auracast™ Transmitter

The simplest implementation takes a single audio input and transmits it as a single BIS in a single BIG. It is illustrated in Figure 3.1 and will be the configuration used in many Public Auracast™ transmitters. The input could be any audio source – either an output from a music player, TV or mixing desk, or a microphone.

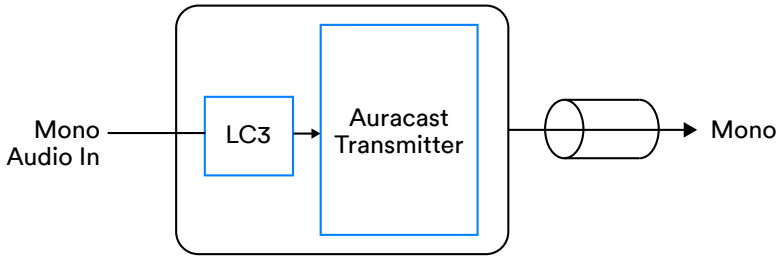

Figure 3.1: A mono Auracast™ transmitter[2]

The Auracast™ Transmitter Recommendations [1] require that Auracast™ transmitters that are designed for Public Transmitter applications[3] must transmit the stream with one of the Standard Quality codec configurations defined in the Public Broadcast Profile. For live applications, where the user may hear ambient sound as well as the Bluetooth stream, this will normally be the 24_2_1 low latency configuration[4], which minimizes any echo effects where a listener can hear both the ambient and transmitted streams.

The Auracast™ transmitter shown in Figure 3.1 can be built into a microphone, providing a convenient hand-held solution which can broadcast to earbuds, hearing aids, and PA speakers in a venue.

The Auracast™ Transmitter Recommendations also define a Personal Transmitter, which can default to a higher 48kHz sampling frequency. However, to ensure that all Auracast™ receivers can work with it, the device must allow the transmitted streams to be switched between either 16/24kHz or 48kHz sampling rates or, alternatively, support both codec configurations by simultaneously transmitting two streams – one at each sampling rate.

---

2 *The Auracast transmitter block in this diagram incorporates the LC3 codec. It has been split out to help illustrate the different configurations*

3 *Public and Personal transmitters are defined in the Auracast™ Transmitter Recommendations [1].*

4 *The 24_2_1 is codec configuration defined in Table 6.3 of BAP [2]. It signifies a sampling rate of 24kHz, a frame size of 10ms and a low latency. Public transmitters may also use 16kHz sampling rates.*

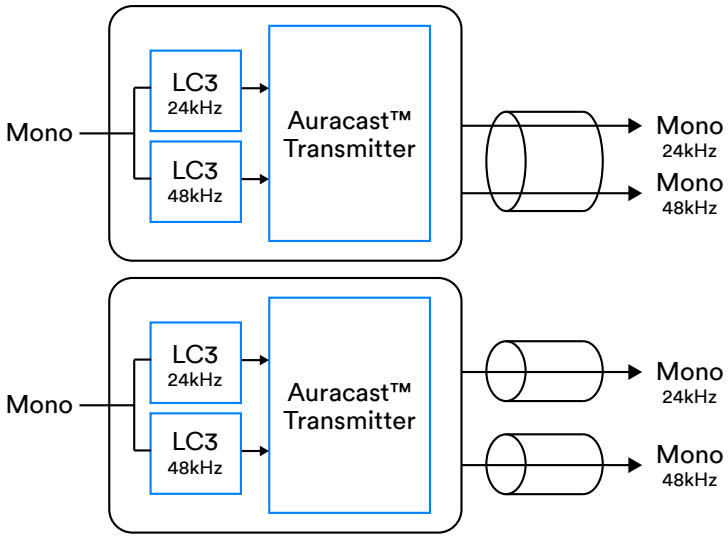

Figure 3.2: BIG options for transmitting streams encoded at different sampling rates

Figure 3.2 shows the two different ways that an Auracast™ transmitter can be arranged to support two simultaneous output streams at different codec settings for a mono application. In most implementations, the first option of placing them into the same BIG would be chosen, as it is the simpler implementation.

Using separate BIGs, which is shown in the lower drawing in Figure 3.3, has an advantage, as it allows each group to be started and stopped independently. It also results in a slightly more optimal use of airtime, as each BIG can be sized to fit the length of the encoded audio packets.

3.2 The Stereo Auracast™ Transmitter

A stereo Auracast™ transmitter adds only a minor amount of additional complexity to the transmitter. Figure 3.3 shows the format for a stereo Public Auracast™ transmitter. This will probably be the most common implementation for Auracast™ transmitters, supporting inputs from a variety of audio devices. The audio input may be analog audio via a 3.5mm or RCA connector or digital audio via USB, HDMI ARC, or optical. The left and right streams will be fed separately to two 24kHz LC3 instances, encoded, and then transmitted as two separate BISes in a single BIG.

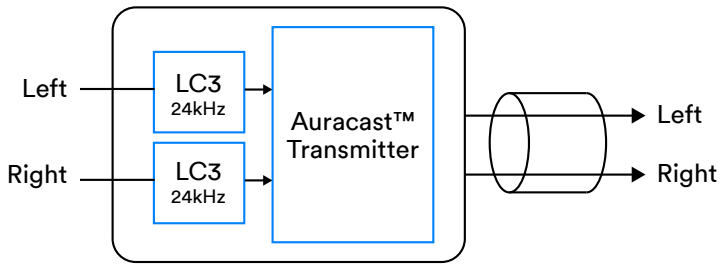

Figure 3.3: The format of a stereo Auracast™ transmitter

To ensure broad interoperability with all Auracast™ enabled receivers, transmitters wanting to be classified as Auracast™ transmitters must support at least one Standard Quality broadcast stream at either 16 or 24kHz. If the transmitter also supports 48kHz, the transmitter must contain a means for the user to switch the codec support to either 16/24kHz or 48kHz. Alternatively, the transmitter can be configured so that both streams are always transmitted. Figure 3.4 shows an example of how to separate the 24kHz and 48kHz stereo streams into two separate BIGs in a device that classifies itself as a Personal Auracast™ transmitter. This arrangement requires four instances of the LC3 codec. As the 24kHz and 48kHz stereo streams are in separate BIGs, they can be turned on or off independently, depending on the user's requirement.

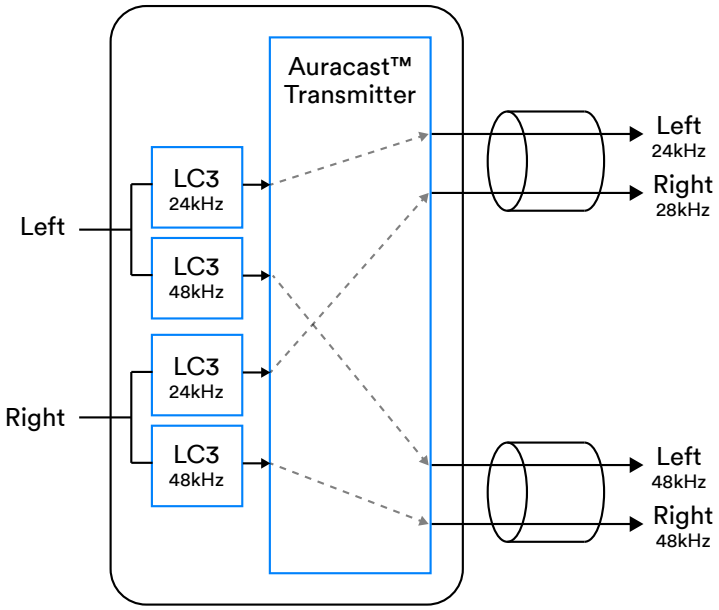

Figure 3.4: Simultaneous transmission of both 24kHz and 48kHz stereo streams

3.3 Configuring the Auracast™ Stream Structure

Most Bluetooth® LE Audio radios will be able to support all of the configurations shown above. A designer needs to decide which audio stream arrangement they want their Auracast™ transmitter to support and configure it by setting the BIG parameters and mapping the inputs to the output streams. If more than one BIG is used in the implementation, then each BIG needs to be configured separately. The configuration will normally be fixed for the life of the product.

An Auracast™ transmitter has a very simple state machine, as shown in Figure 3.5[5]. For most devices, when they are switched on, they will immediately be configured by their firmware and then automatically move to the streaming state. Depending on the silicon platform being used, the controller configuration may be done through HCI commands or a proprietary interface supplied by the platform manufacturer.

5 *Defined in Section 6.2.1 of the Basic Audio Profile.*

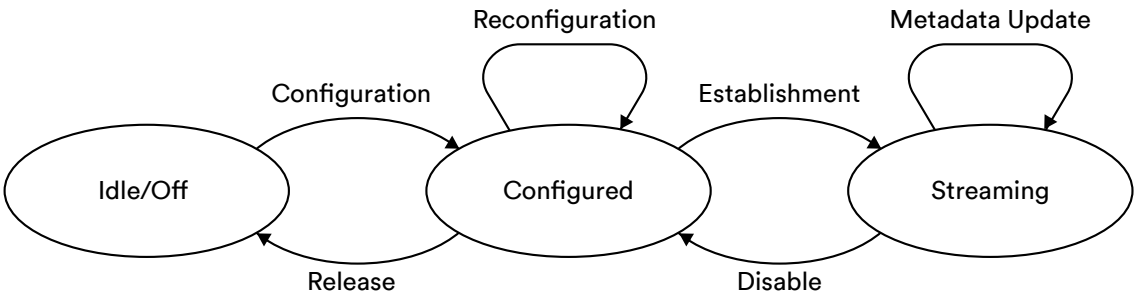

Figure 3.5: The Auracast Transmitter state machine

Regardless of the method, there are five sets of information which need to be provided for the internal configuration process, which control the format of the broadcast transmissions and accompanying advertisements. These can be broken down into:

- User-viewable data, which is used by an Auracast™ assistant to display and select streams
- Codec configuration data, which is used for the transmitter's LC3 instances
- Rendering data, which tells an Auracast™ receiver which streams to render and when to do it
- Encryption data, which informs the Link Layer whether it needs to encrypt the steams
- Stream structure data, which describes the content of each BIS

Each of these are described below. For most devices, the majority of these will be fixed for the life of the product. In some cases, they may be adjustable by a user action, either by a configuration application or a physical switch.

All of this information is exposed to Auracast™ receivers and Auracast™ assistants which are scanning for Auracast™ transmissions.

3.3.1 User Viewable Data

There are four data fields which are used by Auracast™ assistants to filter broadcast transmissions when they are scanning and to display information to a user allowing them to make an informed selection of an audio stream.

Field	Value / Description
Broadcast_Name (Mandatory)	This is used to provide a user-readable name for Broadcast Assistants and is a text field which is used in a similar manner to the SSID in Wi-Fi routers. It should be set to a unique value at manufacture, but users should be able to change it. It is a separate instance for each BIG, but the same name is usually used for all BIGs implemented in an Auracast™ transmitter.
Program_Info_LTV (Optional - Application dependent)	A metadata structure used where more granular stream data is known, similar to program information in an electronic program guide. This is normally only used in embedded applications, such as TVs, where it can be populated automatically. It is a user-readable string, which is specific to a subgroup in each BASE[6].
Local Name (Recommended)	The Local Name (sometimes called the Friendly Name) is a user-readable string, which is used to identify the device. It is a standard Bluetooth® field, not an Auracast™ field. his is not mandatory for an Auracast™ transmitter but is needed if the device needs to pair for configuration.
Appearance Type (Optional)	The Appearance Type is defined in Assigned Numbers and provides information on the type of device. It is a code, not a user-readable string, but it can indicate if the Auracast™ transmitter is a phone, a TV, a microphone, etc. It is typically used by an application to provide an icon to represent the device or can be used to filter specific types of device.

Table 3.1: User viewable data configuration fields

3.3.2 Codec Configuration Data

Each LC3 instance that is being used to encode a broadcast audio stream needs to be configured, using the Codec Configuration settings from Table 6.4 of BAP [2]. An Auracast™ compliant device must use a subset of these, as specified in the Public Broadcast Profile [3].

All public Auracast™ transmitters must support at least one of the codec configurations shown in Table 3.2 for transmitting Auracast™ Streams. The codec configurations for Auracast™ transmitters match the mandatory decoding capabilities of all Auracast™ receivers. This ensures universal interoperability.

Codec Configuration (See BAP Table 6.4)	Typical Application
16_2_1	Low-latency voice (Standard Quality)
16_2_2	High-reliability voice (Standard Quality)
24_2_1 (recommended)	Low-latency media (Standard Quality)
24_2_2	High-reliability media (Standard Quality)

Table 3.2: All Auracast™ transmitters must support at least one of these codec configurations

Personal Auracast™ transmitters may additionally support one or more of the codec configurations shown in Table 3.3. Note that not all Auracast™ receivers are capable of decoding and rendering streams encoded at 48kHz, hence the requirement that the user must be able to configure the Auracast™ transmitter to add, or to revert to a Standard Quality broadcast audio stream.

6 *The Broadcast Audio Stream Endpoint structure (BASE) is defined in Section 3.7.2.2 of BAP.*

Codec Configuration (See BAP Table 6.4)	Typical Application
48_1_1	Low-latency media (High Quality)
48_2_1	Low-latency media (High Quality)
48_3_1	Low-latency media (High Quality)
48_4_1	Low-latency media (High Quality)
48_5_1	Low-latency media (High Quality)
48_6_1	Low-latency media (High Quality)
48_1_2	High-reliability media (High Quality)
48_2_2 (recommended)	High-reliability media (High Quality)
48_3_2	High-reliability media (High Quality)
48_4_2	High-reliability media (High Quality)
48_5_2	High-reliability media (High Quality)
48_6_2	High-reliability media (High Quality)

Table 3.3: Personal Auracast™ transmitters must support at least one of these codec configurations (BAP Table 6.4)

The codec configurations shown above are used to send preferences to the controller but leave the final choice of Link Layer parameters to the controller. Designers should ensure that they do not use values which are not supported by the intended Auracast™ receivers. This is particularly important for Auracast™ transmitters designed for public locations.

3.3.3 Rendering Data

The parameters in Table 3.4 are used to provide information to an Auracast™ receiver about where and when to render streams. These do not affect the transmitter timings but are used to guide timings on Auracast™ receivers

Parameter	Description / Value
Presentation Delay (Mandatory)	The Presentation Delay defines a time when a receiver renders an audio stream after all audio data has been transmitted. This value affects the overall latency of the broadcast audio. All Auracast™ receivers must support a value of 40ms. All HAP and TMAP-compliant Auracast™ receivers must support any value from 20ms to 40ms so an Auracast™ transmitter designed to operate only with these devices can set a value down to 20ms. Auracast™ transmitters should not set a Presentation Delay value lower than 20ms but may set the Broadcast_Audio_Immediate_Rendering_Flag LTV, which instructs Auracast™ receivers to use the lowest value of Presentation Delay which they support. The value of the Presentation Delay will normally be set by the manufacturer. Auracast™ transmitters designed for professional installation may allow this value to be adjusted by the installer to cope with latency in larger venues.

Parameter	Description / Value
Audio_Channel_Allocation LTV (Conditional)	An LTV structure defines the Audio Location for each transmitted stream, which is where the audio is expected to be rendered. It is used to map each BIS to its intended rendering location. If the stream has no intended location, it is assumed to be mono and the Audio_Channel_Allocation LTV must be omitted. For all other locations, it is mandatory.
Streaming_Audio_Contexts LTV (Mandatory)	A metadata structure which is used to identify the use case of the audio stream. It aids a Broadcast Scanner in filtering the available Broadcast Transmissions it finds. "Undefined", "Live" or "Media", depending on the intended usage of the device. The value may be configurable by the user through a configuration app or physical switch.
Broadcast_Audio_Immediate_ Rendering_Flag LTV (Optional)	This metadata flag instructs an Auracast™ receiver to use the lowest value of Presentation Delay which it supports (which may be below 20ms). This is useful where the lowest overall latency is required. It is up to earbud and hearing aid manufacturers to ensure that pairs of left and right devices always use the same value when choosing their own, lower value. The effect of rendering early is unlikely to be observable for speakers. Early rendering is most beneficial for head-worn devices where ambient audio is present. If the Broadcast_Audio_Immediate_Rendering_Flag LTV is present, different pairs of Auracast™ receivers may render the transmitted audio stream at slightly different times.

Table 3.4: Auracast™ transmitter parameters related to rendering and use cases

3.3.4 Encryption Data

The final element of configuration for an Auracast™ transmitter is the choice of whether or not the audio streams are encrypted. If encryption is enabled, the controller needs to use the value of the Broadcast_Code to set an encryption key.

For public Auracast™ transmitter applications, encryption is often not required, so devices designed for these applications can be set to operate only in the unencrypted mode.

Field	Value / Description
Encryption (Mandatory)	ON / OFF This is an input which is used by the LE Create BIG HCI Command to inform the controller whether encryption is required or not. This may be preset in the implementation or be changed by the user. All BISes in a BIG are encrypted. It is not possible to encrypt individual BISes. To encrypt its streams, the Controller needs to be given the Broadcast_Code by the host application. The encryption state can be changed between encrypted and unencrypted, but, if it is changed, the current BIG must be stopped and restarted.

An Auracast™ transmitter needs access to the Broadcast_Code if it wants to encrypt its audio streams. Normally a unique code will be programmed into the Auracast™ transmitter at the

manufacturer,with a means provided for the user to update this with their own string. An Auracast™ transmitter uses this value to encrypt its streams. Auracast transmitters do not transmit the value of the Broadcast_Code using a Bluetooth signal. Auracast™ transmitters need to use an out-of-band method to deliver this value to an Auracast™ assistant. This may be by providing a printed passcode in a similar manner to public Wi-Fi access points, by scanning a QR code which could either be static or displayed on a TV or smartphone, or by additional, alternative out-of-band method.

3.3.5 Stream Structure - The Broadcast Audio Stream Endpoint

The structure of streams in each BIG is represented by the Broadcast Audio Stream Endpoint structure, which is referred to as the BASE. This is defined in Section 3.7.2.2 of BAP, which includes the relatively complex example of an Auracast™ transmitter with two separate stereo streams in a single BIG, shown in Figure 3.6.

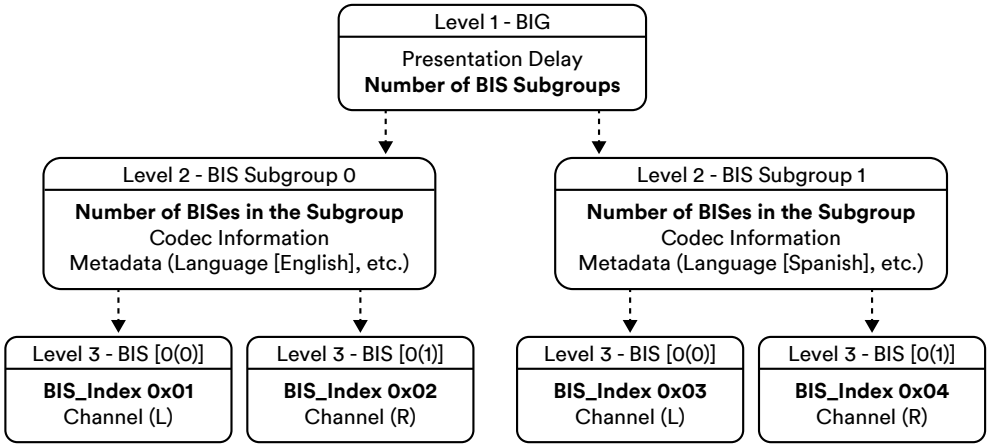

Figure 3.6: Structure of a BASE with two subgroups, each containing two BISes

The BASE is divided into three levels. Level 1 contains information which is relevant for all of the BISes within all of the subgroups. Currently, it is limited to containing the value of Presentation Delay and the number of subgroups within the BIG.

Subgroups contain audio streams which are intended to be rendered together, such as a left and right music stream. Audio streams which would normally be rendered separately, such as a downmixed mono stream or a stereo stream encoded with a different sampling frequency should be placed in their own subgroups. Different language variants should always be placed in their own subgroups.

The Level 2 information is specific to each subgroup. It starts with the number of BISes within that subgroup, followed by the codec configuration for those BISes. Level 2 also contains metadata which is relevant to that subgroup, such as the Program_Info, Language, and Streaming_Audio_Contexts Type LTVs. If the information is common for other subgroups, it still needs to be repeated in their Level 2 data where they apply to all BISes within that subgroup.

Level 3 is concerned with information which is specific for each BIS in that subgroup. Each BIS needs to have a Level 3 entry. It contains the unique BIS_index, which is used to identify the position of each BIS within the BIG, along with its Audio_Channel_Allocation LTV which maps that BIS to its intended

rendering point. For most Auracast™ transmitters, that value is likely to be left or right. For example, a Level 3 entry with the BIS_index value of 0×02 and an Audio_Channel_Allocation of 0×00000002 would mean that the BIS was the second BIS in the BIG and that it is intended to be received by Auracast™ receivers which want to render a right stream.

Mono streams are indicated by the absence of a Audio_Channel_Allocation LTV. They should be placed in a separate subgroup from BISes which have an Audio_Channel_Allocation LTV. Only one mono stream is allowed in a subgroup.

3.3.5.1 A BASE for a Stereo Auracast™ Transmitter

Most Auracast™ transmitters will be simpler than the example above, supporting either a single mono stream or a single stereo stream. Figure 3.7 illustrates the more complex of these two options – the stereo transmitter with its BASE structure. It is followed by a detailed example of the BASE structure parameters for the case where the stereo input streams are sampled at 24kHz and are being used for a media use case.

Examples of other BASE structures are available in the Appendix of the Auracast™ transmitter Guidelines [1].

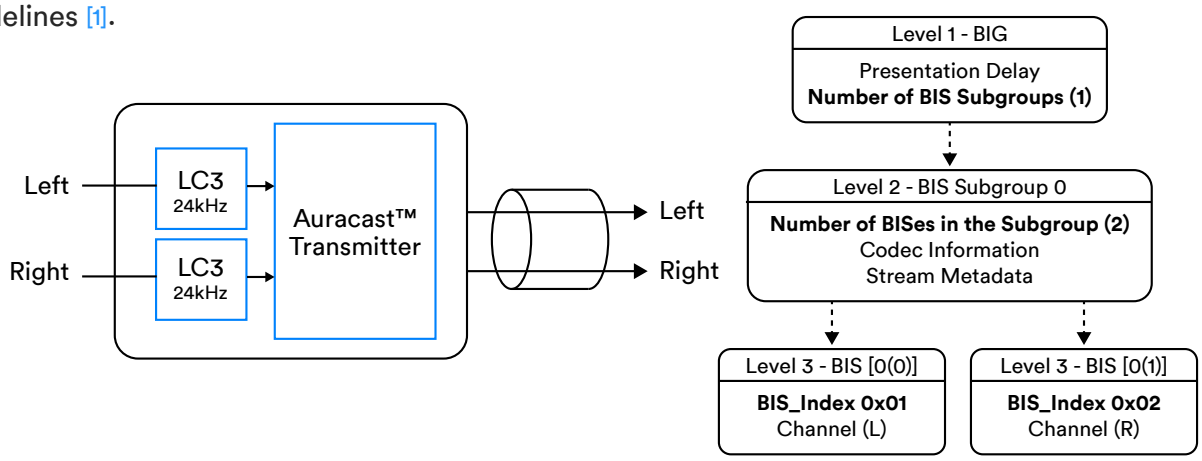

Figure 3.7: The BASE structure for a stereo public Auracast™ transmitter

Level	Parameter	Size (Octets)	Value and Description
	Length	1	Length of Type and Value fields for AD data type: 0×2F = 47 octets
	Type: «Service Data - 16-bit UUID»	1	0x16 (Defined in Bluetooth® Assigned Numbers)
	Value		2-octet Service UUID followed by additional service data
	Basic Audio Announcement Service UUID	2	0×1851 (Defined in Bluetooth Assigned Numbers)
BASE Structure			
1	Presentation_Delay	3	40 ms
1	Num_Subgroups	1	0x01: 1 Subgroup

Level	Parameter	Size (Octets)	Value and Description
2	Num_BIS[0]	1	0x02: 2 BISes in Subgroup[0]
2	Codec_ID[0]	5	Octet 0: 0x06 = LC3 Coding_Format Octet 1–2: 0x0000 Octet 3–4: 0x0000
2	Codec_Specific_Configuration_Length[0]	1	Length of the Codec_Specific_Configuration for Subgroup[0]: 10 octets - 0×0A
2	Codec_Specific_Configuration[0]	10	3 LTV structures for Subgroup[0] defining: LTV 1: Sampling_Frequency: 24000 Hz LTV 2: Frame_Duration: 10 ms LTV 3: Octets_Per_Codec_Frame: 60 octets
2	Metadata_Length[0]	1	Length of Subgroup[0] Metadata: 6 octets
2	Metadata[0]	6	1 LTV structure for Subgroup[0], defining: LTV 1: Streaming_Audio_Contexts: Media
3	BIS_index[0[0]]	1	0x01
3	Codec_Specific_Configuration_ Length[0[0]]	1	Length of the Codec_Specific_Configuration for BIS_index 0x01: 0x06 octets
3	Codec_Specific_Configuration[0[0]]	6	1 LTV structure for BIS_Index 0x01, defining: LTV 1 = Audio_Channel_Allocation: FL
3	BIS_index[0[1]]	1	0x02
3	Codec_Specific_Configuration_ Length[0[1]]	1	Length of the Codec_Specific_Configuration for BIS_index 0x02: 0x06 octets
3	Codec_Specific_Configuration[0[1]]	6	1 LTV structure for BIS_Index 0×02, defining: LTV 1 = Audio_Channel_Allocation: FR

Figure 3.8: The Basic Audio Announcement containing the BASE structure for a stereo Auracast™ transmitter

The contents of the BASE will be populated automatically, based on the implementation configuration, and are normally static. Some of the Level 2 metadata may be configurable by a user in more complex implementations. Appendix B describes the currently defined metadata LTV structures and where they can be used.

During the life of a broadcast session, i.e., the lifetime of a BIG, the only elements of a BASE which can be changed are the metadata elements of Level 2. A change in any other parameter, including the Presentation Delay or codec configuration requires the BIG to be torn down, reconfigured, and restarted.

4.0 Exposing the Audio Stream Structure

Once the design decisions of Section 2 have been made, and the mapping of inputs to outputs and codec configuration is fixed, the Auracast™ transmitter now needs to make them publicly available to allow scanning devices to detect its presence.

As there is no connection between an Auracast™ transmitter and Auracast™ receivers, the transmitter uses extended advertisements to let other devices know of its presence. Once the Auracast™ transmitter is turned on, it will configure itself and then proceed to transmit its audio streams.

Figure 4.1 illustrates the structure of primary, extended, and periodic advertisements which are used by Auracast™ transmitters to expose this information. A Bluetooth® LE Audio platform will use the information from its broadcast configuration to populate these advertisements, which Auracast™ receivers and assistants will then use to determine whether they want to receive any of the streams.

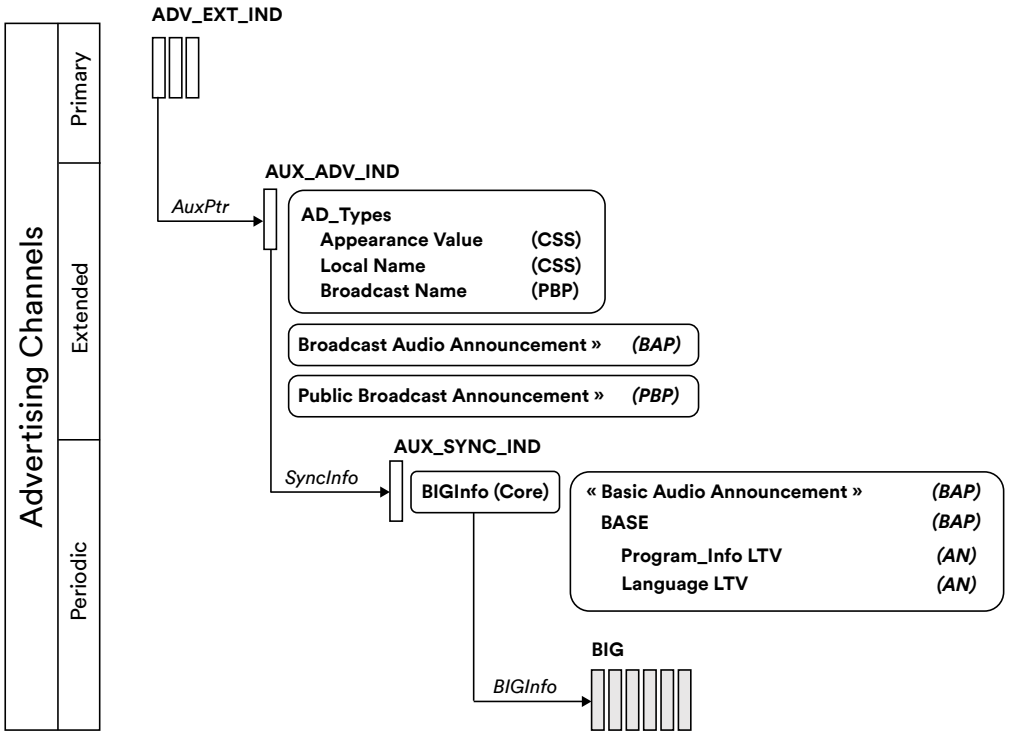

Figure 4.1: The arrangement of advertisements and announcements used by an Auracast™ transmitter[7]

Primary advertisements indicate the presence of extended advertisements by including the AuxPtr, which tells scanners where to find them. All Bluetooth LE Audio broadcasters use extended advertisements to provide information about their broadcasts, so scanners only need to look at advertisements sets which include the AuxPtr. They need to discover the presence of the Broadcast Audio Announcement in the extended advertisement, which confirms that this is a

7 *CSS = Core Specification Supplement, PBP = Public Broadcast Profile, BAP = Basic Audio Profile, AN – Assigned Numbers.*

Broadcast transmitter. It contains the Broadcast_ID – a randomly generated identifier which remains static for the lifetime of the BIG. The Broadcast_ID is important for identifying BIGs. The value must be different for different BIGs in an Auracast™ transmitter. It is recommended that the Broadcast_ID value should be reused for BIGs which serve the same use case for the life of the Auracast™ transmitter.

Having found the Broadcast Audio Announcement, scanners will look for the Public Broadcast Announcement, which is a mandatory feature of all Auracast™ compliant transmitters[8]. The Public Broadcast Announcement contains a bitfield which currently contains three features, shown in Table 4.1. This provides high-level information about the streams within the BIG.

Bit	Feature	Description
0	Encryption	Indicates whether the BIG is encrypted
1	SQ stream available	Confirms that an SQ (16 or 24kHz) audio stream is available
2	HQ stream available	Confirms that an HQ (48kHz) audio stream is available

Table 4.1: The Public Broadcast Announcement features field

The Public Broadcast Announcement allows a scanning device to perform a basic check to determine whether it can support a stream. If it cannot support it, generally because it cannot support 48kHz streams or encrypted streams, it does not need to expend time and energy looking at the announcements in the periodic advertisements but can move to find the next broadcast transmitter.

The value of the bitfield in the Public Broadcast Announcement is valid for the lifetime of the BIG. A change in any of the bitfield values will require the termination, reconfiguration, and restart of the BIG.

The extended advertisement carrying the Public Broadcast Announcement also includes the Broadcast Name, which is the primary human-readable field that an Auracast™ assistant will use to display the presence of an Auracast™ transmitter to the user. All Auracast™ transmitters must include the Broadcast Name AD Type. It is recommended that the Broadcast_Name metadata LTV, which carries the same value, should be present in the metadata section of the Public Broadcast Announcement.

The extended advertisements also include the syncinfo pointer, which directs a scanner to the periodic advertising train. This carries the detailed configuration information describing the configuration of the BIG. This is the Basic Audio Announcement, which contains the BASE structure described above.

When all of the above elements have been set up and transmission of the advertising set has started, the Broadcast state machine will be in the configured state. At this point, no audio streams are being transmitted.

8 *Auracast™ transmissions are a subset of Bluetooth® LE Audio broadcasts, using a limited range of codec configurations to ensure interoperability with other Auracast™ compliant devices.*

4.1 Moving to the Streaming State

In most Auracast™ transmitters, the broadcast Audio Stream state machine will transition automatically to the Streaming state once confirmation has been received from the controller that the BIG has been configurated and established. At this point, audio data should be transmitted on all BISes and the controller should start transmitting the BIGInfo structure in its periodic announcements. Unlike Connected Isochronous Groups, all BISes in the BIG enter the streaming state and remain active for the duration of the BIG, although they may contain null packets.

The BIGInfo provides the information needed by an Auracast™ receiver to synchronize with BISes and defines the timing structure of the BIG. The data in the BIGInfo should be created automatically by the controller in the Bluetooth® LE Audio chip, using the information in the BASE structure, along with codec-specific parameters which are set by the manufacturer.

There is no way to turn individual BISes within a BIG on or off. Once a BIG enters the streaming state, data is transmitted in every BIS.

4.1.1 Controller Options for Increased Robustness

The Codec Configurations for Broadcast in Table 6.3 of BAP provide recommendations that the Host sends to the Controller when it is configuring a CIG. These are summarized for the most common options that an Auracast™ transmitter is likely to use in Table 4.2.

Set Name	Application	RTN	Max Transport Latency (ms)	Auracast™ Transmitter Support	Auracast™ Receiver Support
16_2_1	Low Latency	2	10	Mandatory	Mandatory
16_2_2	High Reliability	4	60	Mandatory	Mandatory
24_2_1	Low Latency	2	10	Optional	Mandatory
24_2_2	High Reliability	4	60	Optional	Mandatory
48_2_1	Low Latency	4	20	Optional	Optional
48_2_2	High Reliability	4	65	Optional	Optional

Table 4.2: Recommended Codec Configuration parameters

The Bluetooth® Core Specification includes a range of Link Layer parameters which can increase the robustness of a connection by spreading retransmissions across multiple isochronous intervals. These are the Burst Number (BN), Immediate Repetition Count (IRC), and Pre_Transmission_Offset (PTO). By increasing the diversity of transmission, their use should improve robustness against interference. There is a trade-off in using these parameters. They should enhance robustness, but that comes with an increase in latency. Examples of their use are provided in Bluetooth Core Specification Vol 6, Part B, Section 4.4.6.6. Most platforms allow Auracast™ transmitter designers to adjust these based on their expected product application.

Designers should be aware that there are no mandatory requirements for Auracast™ receivers to support these features. The mandatory requirement for the 16_2_2 and 24_2_2 codec configurations implies that all Auracast™ receivers should accept an RTN value of up to 4. However, all Auracast™ receivers should be designed to cope with other LL parameters, although that may result in them ignoring some retransmissions. Table 6.5 in BAP provides recommendations for Link Layer parameters for the 48_2_2 codec configuration, showing how they may be used for different transmitter applications. Complex transmissions schemes may be ignored by some Auracast™ receivers, so manufacturers should test against as many products as possible.

4.2 Advertising Timing

The repetition rate of the advertising set (comprising primary advertisements, extended advertisements, and periodic advertisements) will influence how quickly a scanning device is able to detect an Auracast™ transmitter. The Auracast™ transmitter guidelines provide a recommendation of 30ms for the extended advertising interval. This should allow most scanners to acquire the extended advertisements within 10 seconds. Increasing the interval will slow down the acquisition process and may also increase the power consumption of the scanner.

5.0 Encryption and Broadcast_Code

Auracast™ transmitters designed for public use do not generally require encryption. However, there are cases where it is desirable. Bluetooth® signals will penetrate walls, so people in neighboring rooms may be able to receive audio streams. Where the content is confidential or personal, encryption should be employed. The Bluetooth LE Audio specifications and the Bluetooth Core Specification define a Broadcast_Code which is used by Auracast™ transmitters to encrypt their audio streams. The same Broadcast_Code is used by Auracast™ receivers to decrypt the stream.

Broadcast_Codes are sent from an Auracast™ assistant to the Auracast™ receivers. They are not included in the advertising set of Broadcast transmitters, so they need to be acquired by Auracast™ assistants by an out-of-band method. How that is accomplished is not defined in the Bluetooth LE Audio specifications. It is anticipated that the most common method will be similar to that used in Wi-Fi, where it will be shared with users via a printed notice or a QR Code, as in the example below.

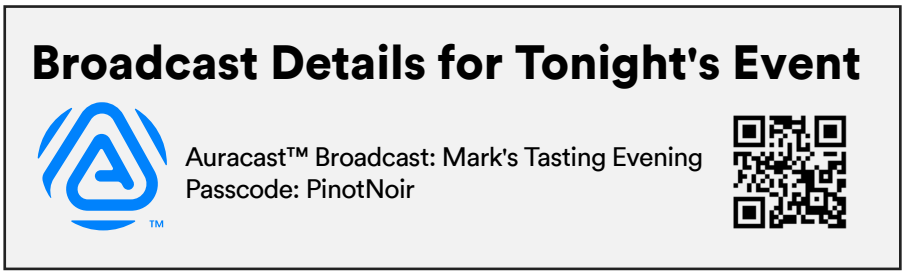

Figure 5.1: An example of a public display of a Broadcast_Code

Manufacturers of Auracast™ transmitters should set the Broadcast_Code to a unique default value at manufacture, but users will appreciate the ability to provide their own text string through a configuration utility. The format of the Broadcast_Code is a string of at least four, and at most 16, octets when represented in UTF-8 (See Bluetooth Core Specification Volume 3, Part C, Section 3.2.6).

A QR Code can be included for an even simpler user experience. Multiple QR codes can be produced to direct an Auracast™ assistant to select specific streams from an Auracast™ transmitter. Details of how to use QR codes are covered in the Broadcast Audio URI specification [11].

6.0 User Interface Considerations

Auracast™ transmitters will come in many different forms. Many will be fit-for-purpose transmitters for public infrastructure, while others will be embedded in devices, typically phones, TVs, and PCs. These are likely to have user applications or extensions to settings menus to configure them. Auracast™ transmitters for public infrastructure are are likely to be designed for professional installation and remote management, so they will come with comprehensive configuration options.

In the short-to-medium term, the largest volume of Auracast™ transmitters is likely to consist of products which work with existing consumer products, adding Auracast™ transmission capability to legacy phones, TVs, and sound systems. There will also be totally new products, such as Auracast™ microphones, which combine a microphone with the Auracast™ transmitter which will broadcast both to speakers in an auditorium as well as being able to be accessed by people wearing hearing aids and earbuds.

6.1 User Configurable Options

There are a few parameters which should be accessible on all Auracast™ transmitters so that a user can change them. These are:

- Overwriting the default Broadcast_Name
- Overwriting the default Broadcast_Code, if encryption is supported
- Overwriting the Local Name[9]

When also supporting high quality broadcast audio streams, Auracast™ transmitters[10] need a method for a user to:

- Select 16kHz, 24kHz streaming, or 48kHz streaming if concurrent streaming is not implemented

Generally, the simplest method to provide configuration access to the transmitter is with a Bluetooth® app on a phone which is paired to the Auracast™ transmitter. To prevent malicious tampering, the application should be password protected, with the default password supplied with the Auracast™ transmitter. Implementers should look at the interfaces of broadband routers and consider following the same design principles.

9 *The Local Name (also called the Friendly Name) is used for initial pairing. Although Auracast™ Transmitters do not pair with other Auracast™ devices, they often need to pair with a smartphone to access an application for their initial configuration. As with other Bluetooth products, users may want to customise this name to help them identify the device.*

10 *Defined in the Auracast™ transmitter recommendations document [1].*

6.2 Status Feedback

Product designers should consider what level of user feedback is relevant to allow the user to ascertain the correct operation of an Auracast™ transmitter. This might include indications of the following states:

- Active transmission of audio streams
  Encryption ON or OFF
- Standard Quality and/or High Quality transmission present

For testing the installation, it may be useful to include the ability to transmit an internally stored audio sample to allow users to test the operation of the Auracast™ transmitter.

6.3 Volume Control

The Bluetooth® LE Audio specifications assume that all broadcast transmissions are transparent from the source with no adjustment of the volume of the signal. Therefore, there is no requirement for volume control on an Auracast™ transmitter. Volume control is left to each individual Auracast™ receiver. This avoids the situation where the volume is turned down on an Auracast™ transmitter, resulting in a user having to increase the local gain to maximum. This may result in painful levels of sound when users move to a different audio stream.

Some products which include the input of ambient audio, such as, such as an Auracast™ microphone, may be an exception as they may need the user to set the microphone gain.

The Auracast™ transmitter recommendations provide more information on optimum settings for the audio input, which take advantage of the dynamic range of the LC3 codec.

6.4 Transmit Power

Some designs will support a range of transmit power, allowing users to customize the range. Where this is done, an application or physical switch should be provided. In general, devices should be set to the minimum power which gives reliable coverage in the intended coverage area (e.g. conference room).

Auracast™ broadcasts have a significantly greater range than Bluetooth Classic Audio signals. Having the ability to control the transmit power may be a useful feature where coverage needs to be limited.

7.0 References

[1] Auracast™ Transmitter Guidelines document

[2] Basic Audio Profile v1.0.1

[3] Public Broadcast Profile v1.0

[4] Common Audio Profile v1.0

[5] Telephony and Media Audio Profile v1.0

[6] Hearing Access Profile v1.0

[7] An Auracast Overview

[8] How to Design an Auracast™ Earbuds

[9] How to Design an Auracast™ Transmitter

[10] Bluetooth® Core Specification version 5.2 or later

[11] Broadcast Audio URI specification

8.0 Appendix A: Audio Quality and Airtime

When designing Auracast™ transmitters, designers need to be aware of the trade-offs between audio quality, the number of streams, and airtime.

8.1 Codec Configuration and Packet Size

Unlike Bluetooth® Classic Audio and LE Audio unicast applications, there are no acknowledgments from receiving devices with a broadcast transmission. This means that Auracast™ transmitters always retransmit unconditionally. Where higher sampling rates are used, this leads to an increase in the number of bytes transmitted in each packet, which has an impact on airtime usage.

Table 8.1 shows the airtime for a single Standard Quality BIS for low-latency and high-reliability settings. The airtime figures include the total BIS airtime, including packet headers and interframe spacing. They do not include the advertising set overhead, which accounts for around 0.6 percent of additional airtime for each BIG, depending on the amount of metadata included. All numbers assume the use of a 2MB PHY.

Codec Config	Category	Retransmissions	Total Airtime
16_2_1	Low latency	2	10.6%
16_2_1	High reliability	4	17.7%
24_2_1	Low latency	2	13.0%
24_2_2	High reliability	4	21.7%

Table 8.1: Airtime for Standard Quality codec settings

Table 8.2 shows the equivalent airtime for a single High Quality BIS with low-latency and high-reliability settings. For HQ, both low-latency and high-reliability settings use four retransmissions. These tables only show 10ms codec configurations, as they provide better subjective audio quality than 7.5ms configurations. The 7.5ms settings are typically only used for systems that are connecting to legacy Bluetooth devices at the same time as broadcasting. Auracast™ receivers do not need to support 7.5ms encoded streams, so they should not be used for Public Auracast™ applications.

Codec Config	Category	Retransmissions	Total Airtime
48_2_1/2	All	4	29.7%
48_2_3/4	All	4	33.7%
48_2_5/6	All	4	40.7%

Table 8.2: Airtime for High Quality codec settings

It is clear from these numbers that an Auracast™ transmitter can only support a limited number of BISes. For the example in Figure 3.6, with both English and Spanish stereo streams, this can only be achieved using the 24kHz sampling settings.

There is another subtlety in scheduling. When a BIG is scheduled, it allocates the same amount of airtime for each BIS. This means that it always allocates the amount of space necessary for the data packets corresponding to the largest BIS.

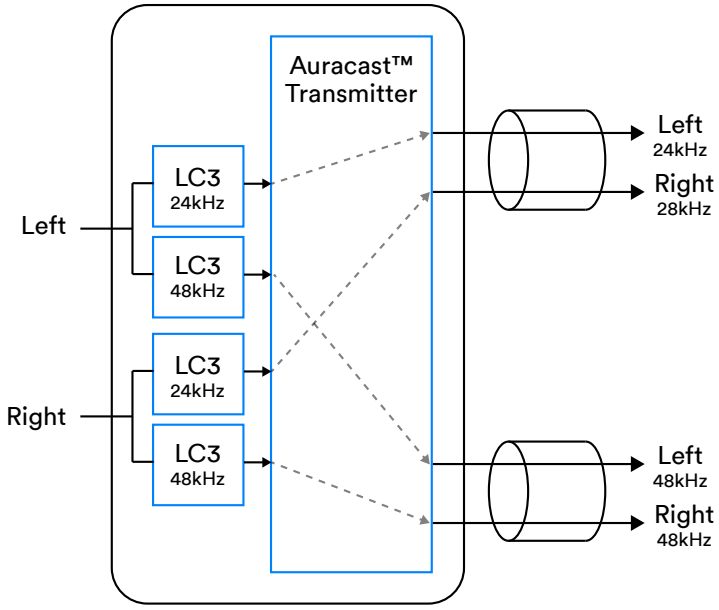

Figure 8.1: Arrangement to simultaneously transmit both 24kHz and 48kHz stereo streams

If we look at the example of Figure 3.4 (repeated above as Figure 8.1), where both 24kHz and 48kHz streams are being transmitted, you can see why they have been separated into two different BIGs. In this case, if the 24kHz is encoded using the 24_2_1 configuration, that BIG will consume 26.0 percent of airtime. The 48kHz BIG configured with the 48_2_1 setting will consume 59.4 percent, resulting in a total of 85.4 percent of airtime. The advertising sets for the two BISes add around 1.2 percent of additional airtime, taking the total to just under 87 percent. This is high, but it is possible, which means that these four BISes can be scheduled in the two separate BIGs.

In contrast, if all four BISes were placed in the same BIG, it would be scheduled as four BIS events sized to fit the 48_2_1 packet, which would require 118.8 percent of airtime, which is not possible.

An added advantage of using two separate BIGs is that they can be started and stopped independently. In the example above, the 24kHz stereo stream can be transmitted at all times, with the higher quality 48kHz stream turned on or off as required. Or vice versa.

The good news is that the LC3 is a very effective codec. Independent listening tests have shown that most users are satisfied with the quality of 24kHz encoding, even when listening in perfect listening conditions. To optimize the use of airtime, there are good reasons to default to 24kHz configuration for public locations or to consider transmitting in mono.

8.2 Mono vs Stereo

Although stereo is now the default standard for almost all recordings, many public audio locations use mono. If we think about public announcements, public TVs, PA systems, ceiling speakers, and many other voice-based applications, mono is sufficient. There are also times when it may make sense to generate a separate mono stream. One of the aims of Auracast™ boradcast audio is to become the next-generation public audio infrastructure, so devices should be designed first and foremost to deliver reliable, clear audio.

The philosophy of public Auracast™ installations is very different from consumer audio, which pushes the limit of audio reproduction, often to limits beyond human hearing. Auracast™ products are designed to provide a long-term audio infrastructure which will work with every Auracast™ receiver both now and in the future. Designers should carefully consider the tradeoffs and look to design products that will have the widest level of interoperability and audibility when they make their design decisions. It is worth remembering that the telecoil implementations in modern hearing aids are still compatible with hearing looks installed over fifty years ago. Public infrastructure needs to be designed for long-term interoperability.

9.0 Appendix B: Choosing and Using Metadata

The Bluetooth® LE Audio specifications include a range of LTV metadata structures which can be used in the Public Broadcast Announcement (PBA) and Level 2 of the BASE structure in the Basic Audio Announcement (BAA). The LTV structures are defined in Section 6.12.6 of the Assigned Numbers document. They are mostly designed to help scanning devices and Auracast™ receivers make choices about whether or not to receive a broadcast transmission. Auracast™ transmitters should use them to help that choice.

Table 9.1 lists all of the current Metadata LTV structures and indicates whether they should be used in the PBA or BAA. Not all are used for broadcast. It is expected that more metadata structures will be introduced as broadcast use cases are developed. Developers should be aware that Auracast™ assistants and receivers treat these as optional information and may not use them. Receivers may not recognize all metadata LTV structures and will ignore those they do not know. The Metadata LTV structures are defined in Section 6.12.6 of Assigned Numbers.

An Auracast™ transmitter may change the metadata present in a BASE structure during the life of a BIG. If this occurs, an Auracast™ receiver may not be aware of the change, unless it regularly scans and parses the BASE structure.

Metadata Name	PBA	BAA	Description
Preferred_Audio_Contexts (6.12.6.1)	n/a	n/a	Not used in broadcast.
Streaming_Audio_Contexts (6.12.6.2)	n/a	Y	Used to provide information about the use case. Generally used to distinguish between Voice (low latency) and Media. An Auracast™ receiver may use this to influence its rendering behavior.
Program_Info (6.12.6.3)	N	Y	Provides further detail about the content being broadcast. The value is different for different subgroups in the same BIG.
Language (6.12.6.4)	N	Y	Indicates the language of the content being broadcast in a subgroup. This will usually be omitted if all streams are in the same language. If used, a value should be present in each subgroup.
CCID_List (6.12.6.5)	n/a	n/a	Not normally used in broadcast applications unless the Auracast™ transmitter contains media control functionality.
Parental Rating (6.12.6.6)	Y	Y	Describes the suggested minimum age for listeners of the broadcast content. This may be used in the Public Broadcast Announcement if it applies to all streams. It may also be repeated in the Basic Audio Announcement.
Program_Info_URI (6.12.6.7)	N	Y	A URI link which a scanner can use to obtain more information about the content of an audio stream. If used, it should normally be included in addition to the Program_Info. It will only be of use to Auracast™ assistants which have active internet access.
Extended_Metadata (6.12.6.8)	N	Y	Not currently used.

Metadata Name	PBA	BAA	Description
Vendor_Specific (6.12.6.9)	N	Y	Vendor-specific information.
Audio_Active_State (6.12.6.10)	Y	N	Indicates that a BIG is not currently transmitting audio data. This is used where audio data is only transmitted intermittently, such as with transport announcements. The BIG should be in the streaming state when this metadata is present so that Broadcast Scanners can synchronize to the streams ready for audio to be available.
Broadcast_Audio_ Immediate_ Rendering_Flag (6.12.6.11)	N	Y	Informs an Auracast™ receiver that it can ignore the Presentation Delay in Level 1 of the BASE and render the audio content as soon as possible. See Table 3.4 for more information on its use.
Assisted Listening Stream (ALS) (6.12.6.12)	Y	Y	Indicates that at least one stream has been processed to increase the intelligibility of audio content for users with hearing loss. When used in the BASE, it should only appear in the Level 2 metadata of subgroups which are Assisted Listening streams. Note that most hearing aids perform personalized audio processing and may prefer not to select an ALS stream. The ALS stream may be of greatest benefit to users wearing Personal Sound Amplifier Products (PSAPs) or normal earbuds.
Broadcast_Name (6.12.6.13)	Y	N	Replicates the content of the Broadcast_Name AD Type. This allows a scanning Auracast™ receiver to share it with non-scanning devices via notifications. It is recommended that this is included in the metadata section of the PBA.

Table 9.1: Available metadata structures and their usage