Haute école d'ingénierie et d'architecture Fribourg
Hochschule für Technik und Architektur Freiburg

INTERNET DES OBJETS

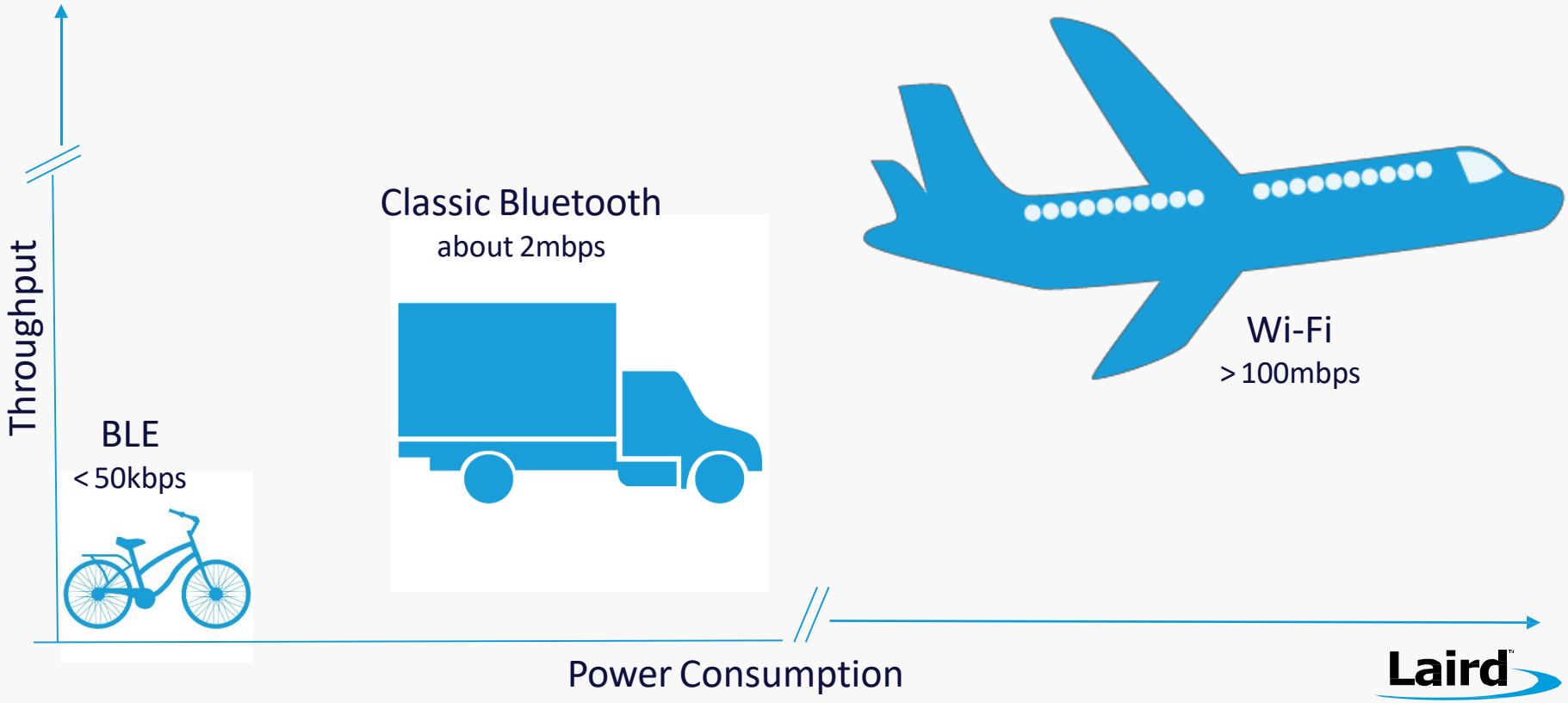
BLUETOOTH LOW ENERGY

SERGE AYER - HEIA-FR – TÉLÉCOMMUNICATIONS
CLASSES ISC-2D // 2023-2024

WHY BLUETOOTH LOW ENERGY ?

- Bluetooth Low Energy has been adopted on all smartphones and tablet platforms
  - Full smartphone connectivity and interoperability
- The entire stack has been standardized for ensuring interoperability at the application level
- Low power
- Low cost
- Prediction:
  - Bluetooth (Low Energy) will be to smartphones what USB is to PCs
  - Connecting a smartphone to any device will happen through Bluetooth (Low Energy)

WHY BLUETOOTH LOW ENERGY ?



WHAT IS BLUETOOTH ?

- Wireless technology
- For short-range communications systems intended to replace the cables
- Operates in the 2.4 GHz ISM (Industrial, Scientific and Medical) band
- Uses FHSS (Frequency Hopping Spread Spectrum) to combat interference
- Range up to 100 m based on the radio class
- No line of sight required

WHAT IS BLUETOOTH ?

- Two Forms
  - BR/EDR – Basic Rate/Enhanced Data Rate
  - Low Energy(BLE)
- Both forms include
  - Device discovery
  - Connection establishment and management
  - Data Transfer

BLUETOOTH TECHNOLOGY EVOLUTION

2004

- V2.0 EDR - Added Enhanced Data Rate boosting throughput from 1 to 3 mbps

2007

- V2.1 EDR - Secure Simple Pairing allows secure device pairing with a button press, numeric entry, numeric compare, and OOB

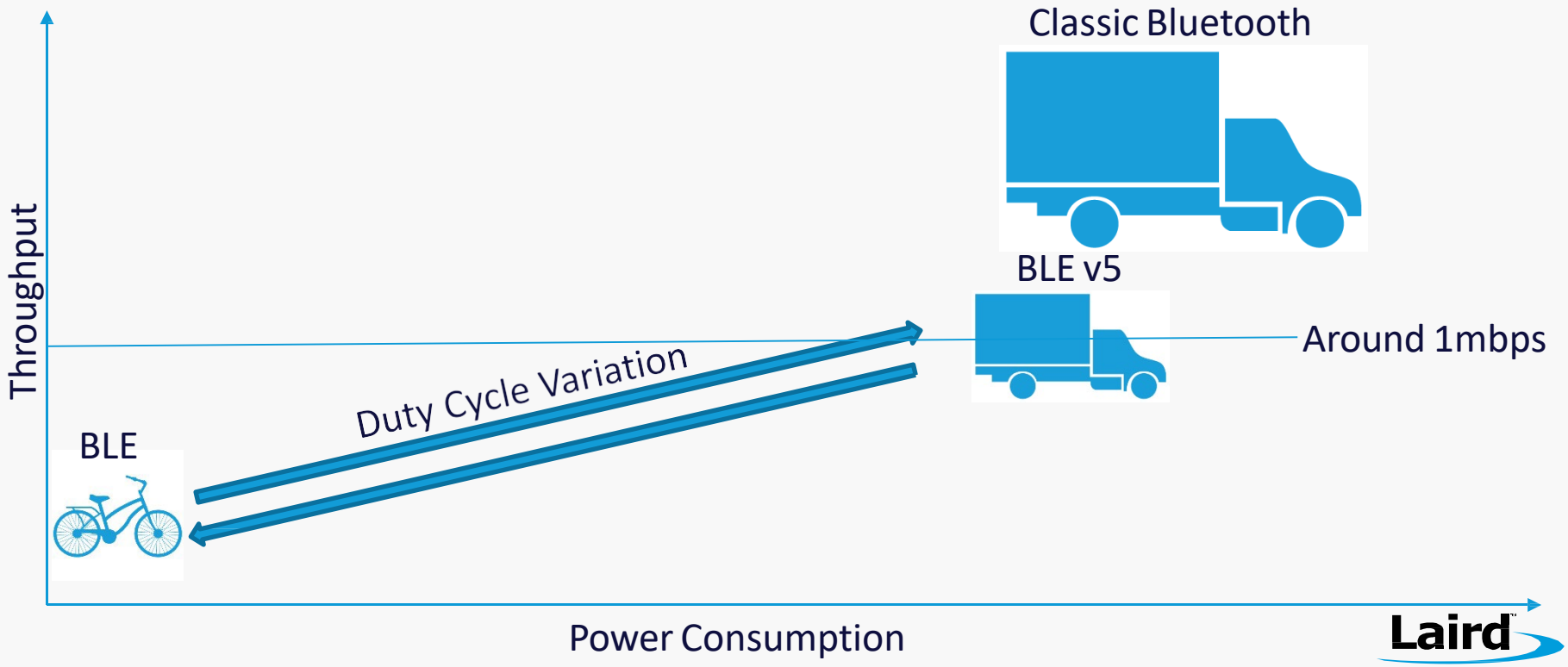
2009

- V3.0 High Speed - Enables applications to use 802.11 MAC/PHY through addition of Generic Alternative MAC/PHY architecture

2010

- V4.0 Low Energy - Enables new applications in different markets including healthcare, sports/fitness, security, home entertainment

… AND NOT BLE 5 IS THE NEW STANDARD



ANOTHER STEP IN MAKING BLE A TECHNOLOGY FOR THE IOT

BLE MESH IS ALSO PART OF THE GAME SINCE 2017

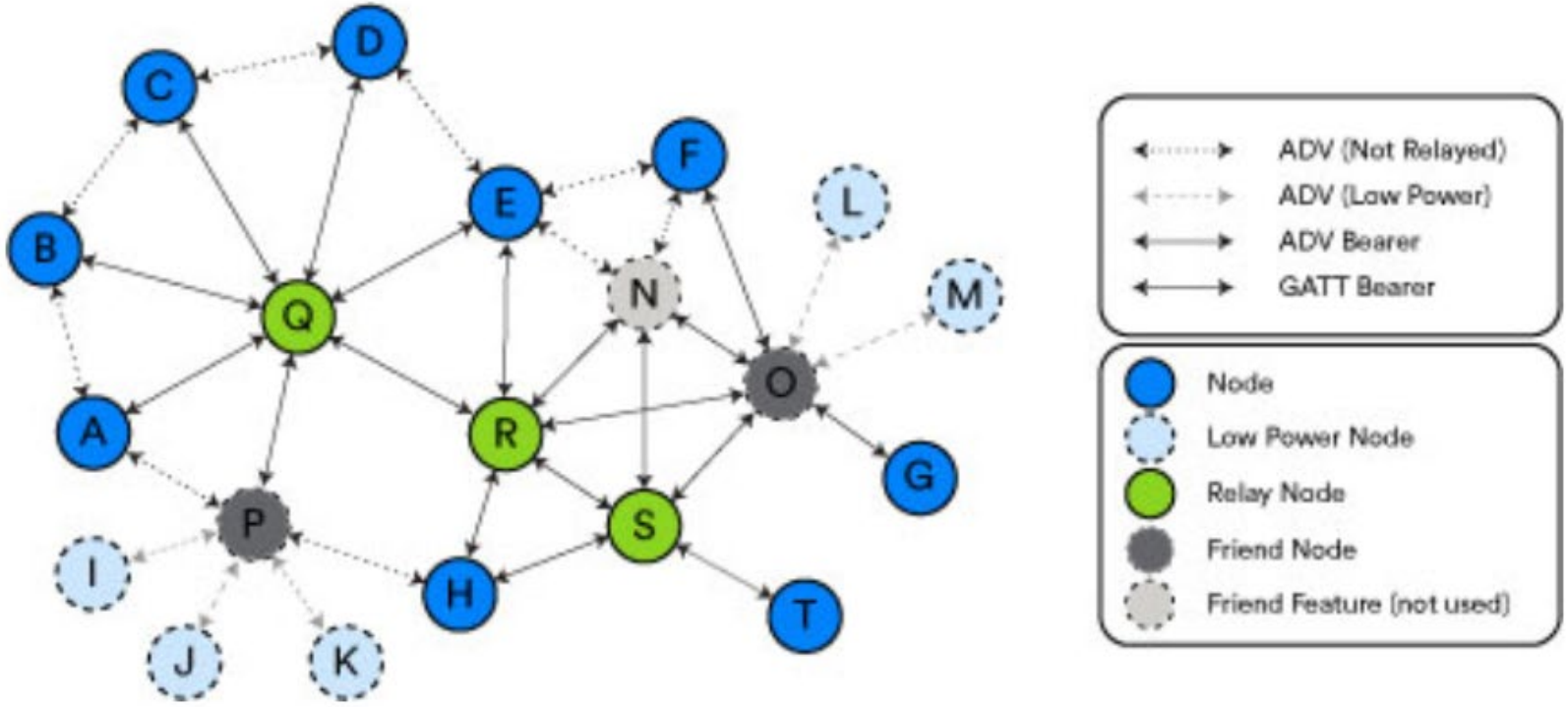


Figure 1 - example topology of a mesh network

BLUETOOTH ECOSYSTEM TODAY

Bluetooth® SMART READY


Droid RAZR

Windows 8


iPhone 4s


Mac Mini


MacBook Air

Bluetooth® SMART

Bluetooth®

4+ Billion *Bluetooth* products today

WHAT MAKES BLE DIFFERENT ?

- Intimately tied to the phenomenal growth in smartphones, tablets and mobile computing.
- Need for connectivity with the outside world on these devices
  - Low barrier to adoption
- Task-specific, creating and innovative products on a relatively modest design budget
  - SoC solutions can be bought under $2, in low volumes

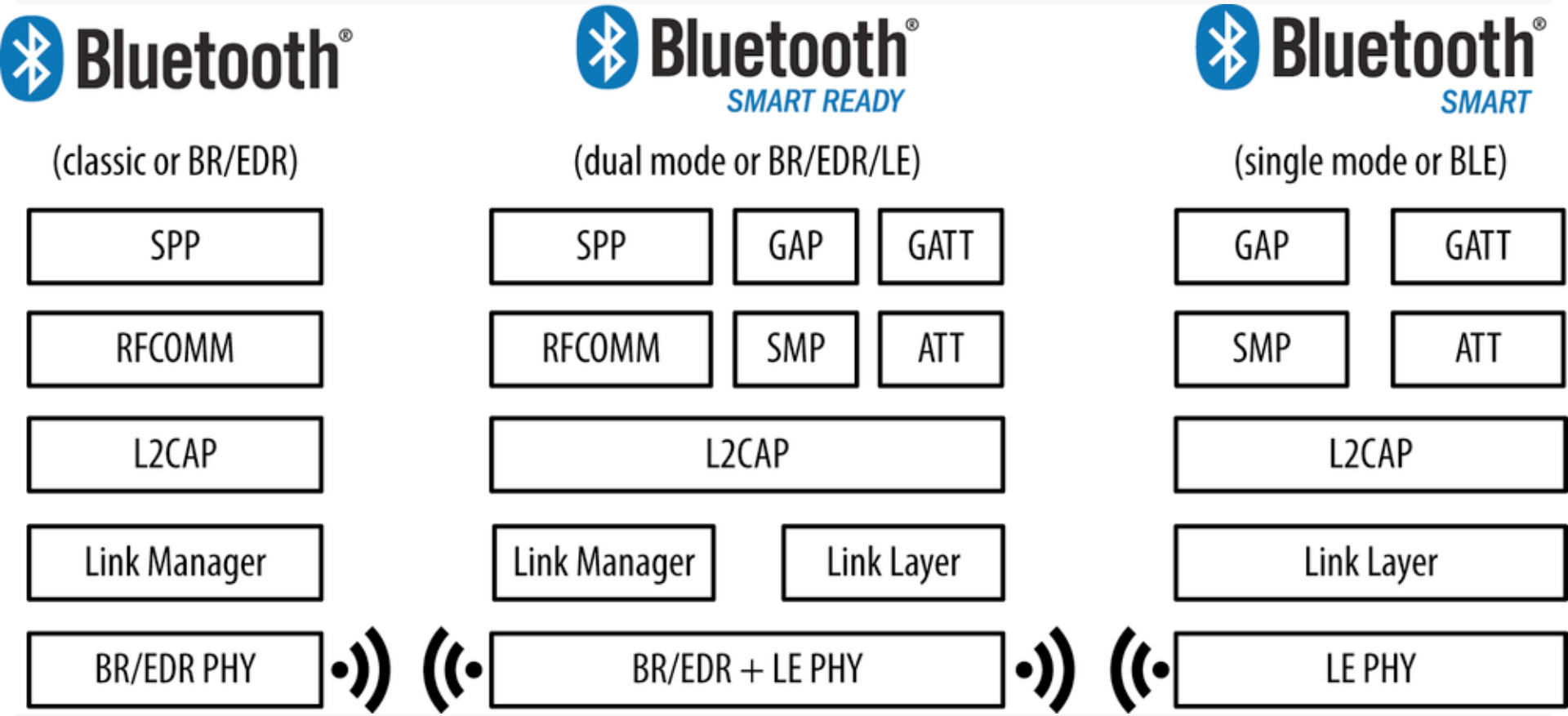
WHAT MAKES BLE DIFFERENT ?

- Is an extensible framework to exchange data among devices, unlike other standards
  - Bluetooth 4.0 released in 2010
  - Bluetooth 4.1 released in December 2013
  - Bluetooth 4.2 released in December 2014
  - Bluetooth 5.0 released in December 2016
- No licensing costs, no fees, lean overall protocol stack
- Last but not least: It is Low Energy !

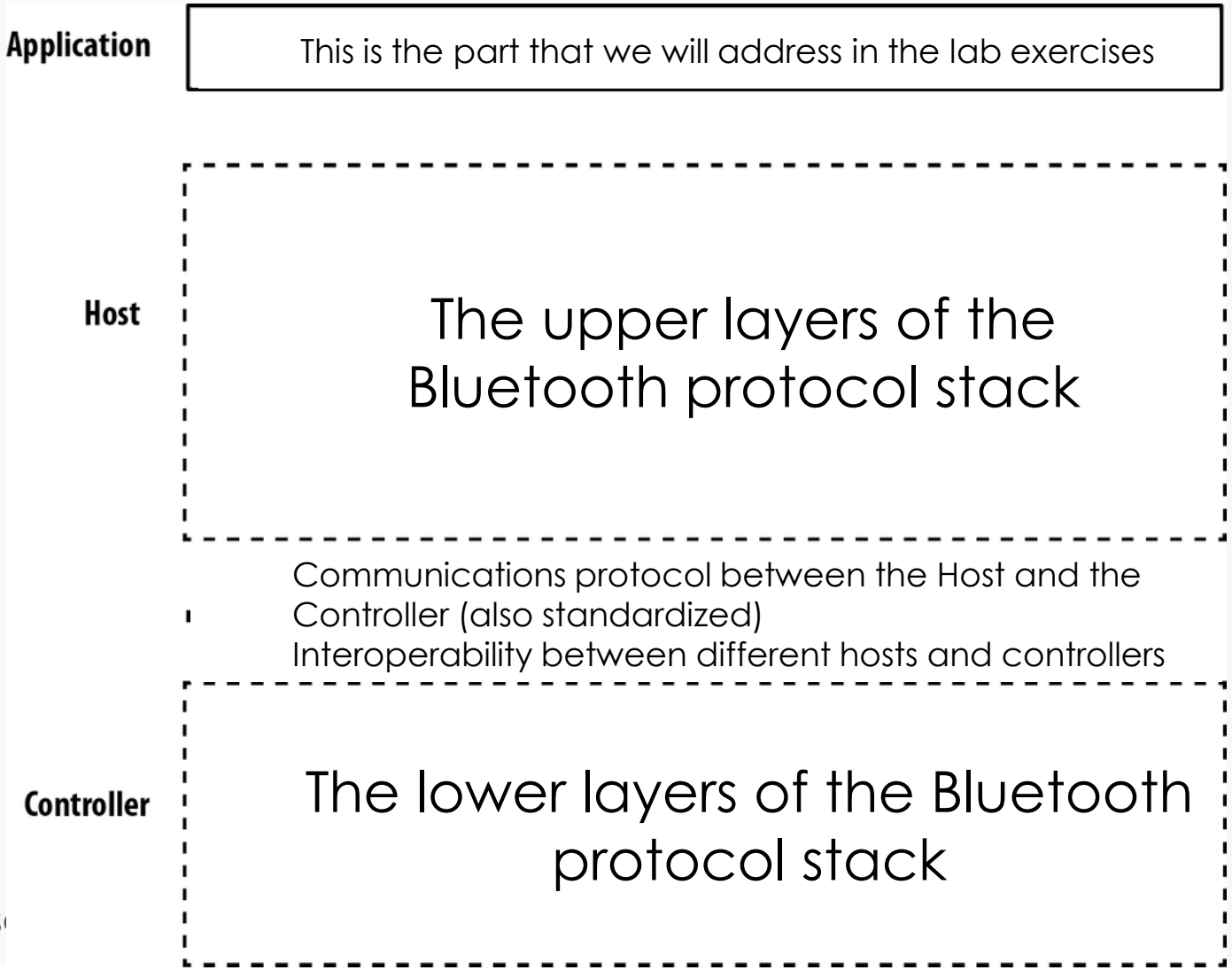
DEVICE CONFIGURATIONS

- Pre-4.0: only BR (Basic Rate) / EDR (Enhanced Data Rate) mode
- 4.x Single-Mode (Bluetooth Smart): only BLE
  - Can communicate with Single-Mode and Dual-Mode devices – not with pre-4.0 devices supporting only BR/EDR
- 4.x Dual-Mode (Bluetooth Smart Ready): both BR/EDR and BLE
  - Can communicate with any Bluetooth device

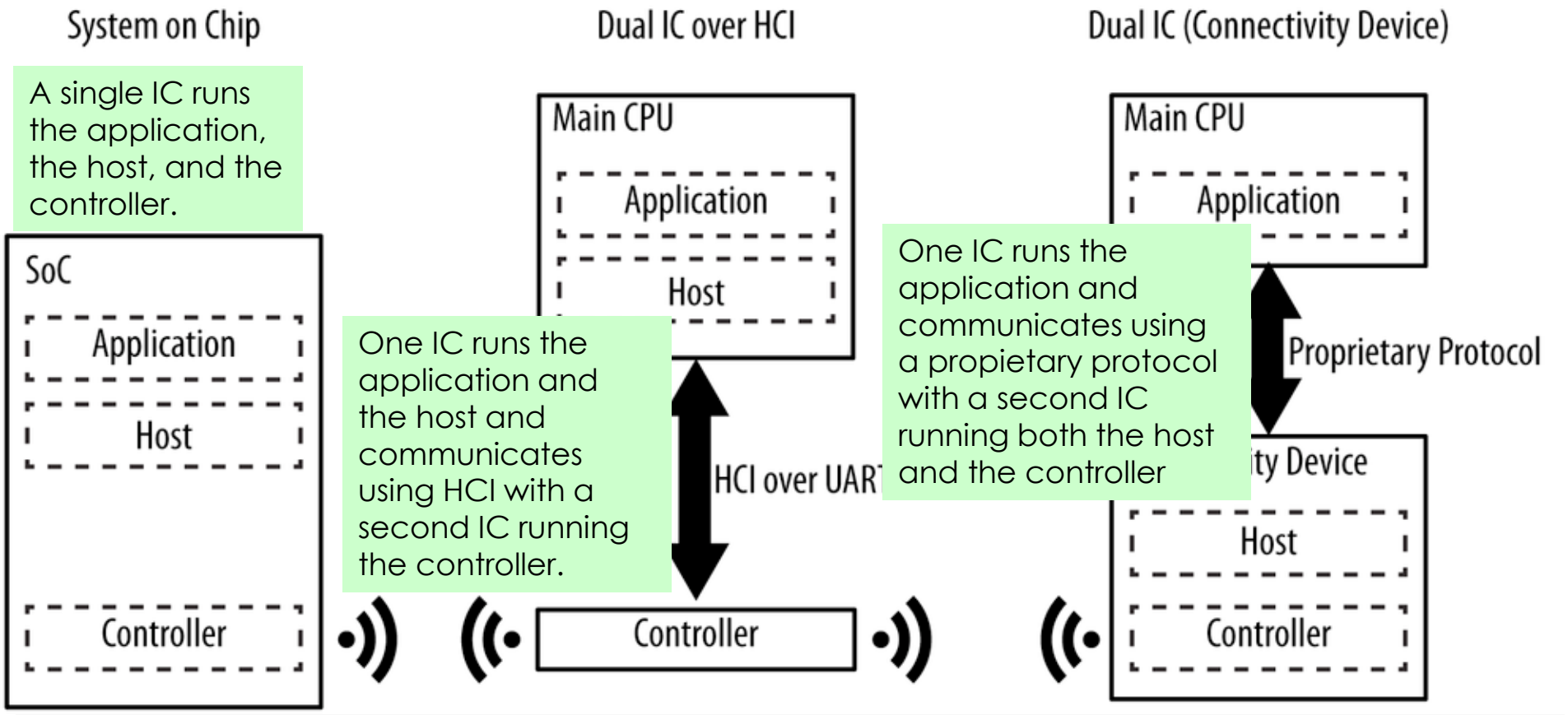
DEVICE CONFIGURATIONS



MAIN BLOCKS IN THE PROTOCOL STACK



CHIP CONFIGURATIONS



BLE LIMITATIONS

- Data throughput
  - Modulation rate at about 1 Mbps, as a theoretical upper limit for the throughput that can be provided
  - There are other limiting factors
    - Bidirectional traffic
    - Protocol overhead
    - CPU, radio limitations
    - Software
  - In practice, plan about 5-10 kBytes/sec

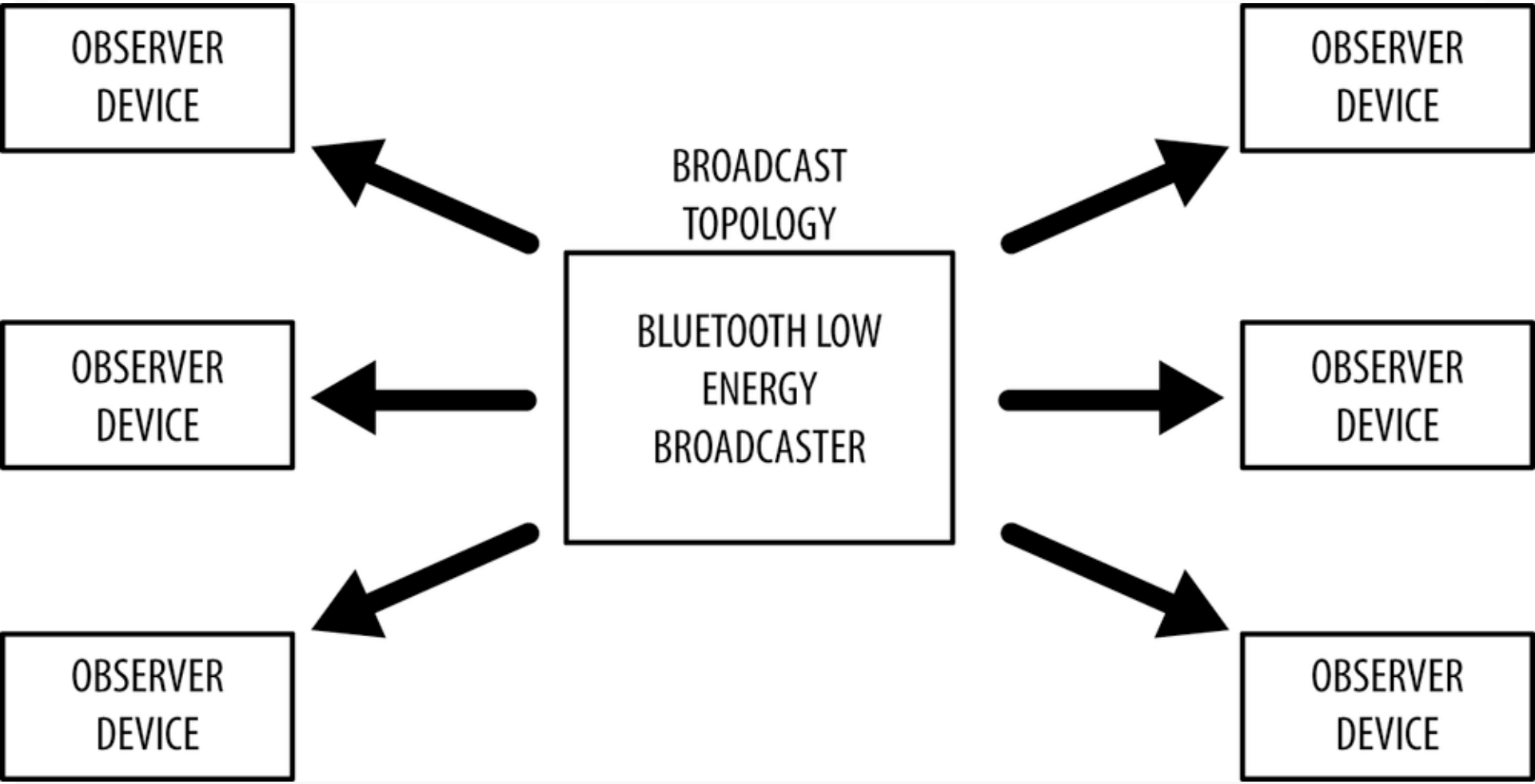
BLE LIMITATIONS

- Never forget: BLE was designed to achieve the lowest power protocol possible
  - Achieved by using short burst of packets (during a connection event) at a certain frequency (connection interval), switching the radio off in between
- Operating range
  - Depends on many factors such as the operating environment, the antenna design, the device orientation, etc..
  - Focused on short-range communication
  - Transmit power can be configured, again for optimized battery lifetime
  - Typically 2-5 meters, possible up to 30-100 meters line-of-sight

NETWORK TOPOLOGY

- Two ways of communicating
  - Broadcasting (the so-called Beacons or iBeacons)
  - Within connections
- Broadcasting
  - Send data out to scanning devices
  - Roles in the topology
    - Broadcaster: sends non-connectable advertising packets
    - Observer: scan those packets
  - It is the only way for a device to transmit data to more than one device at a time
  - No security or privacy

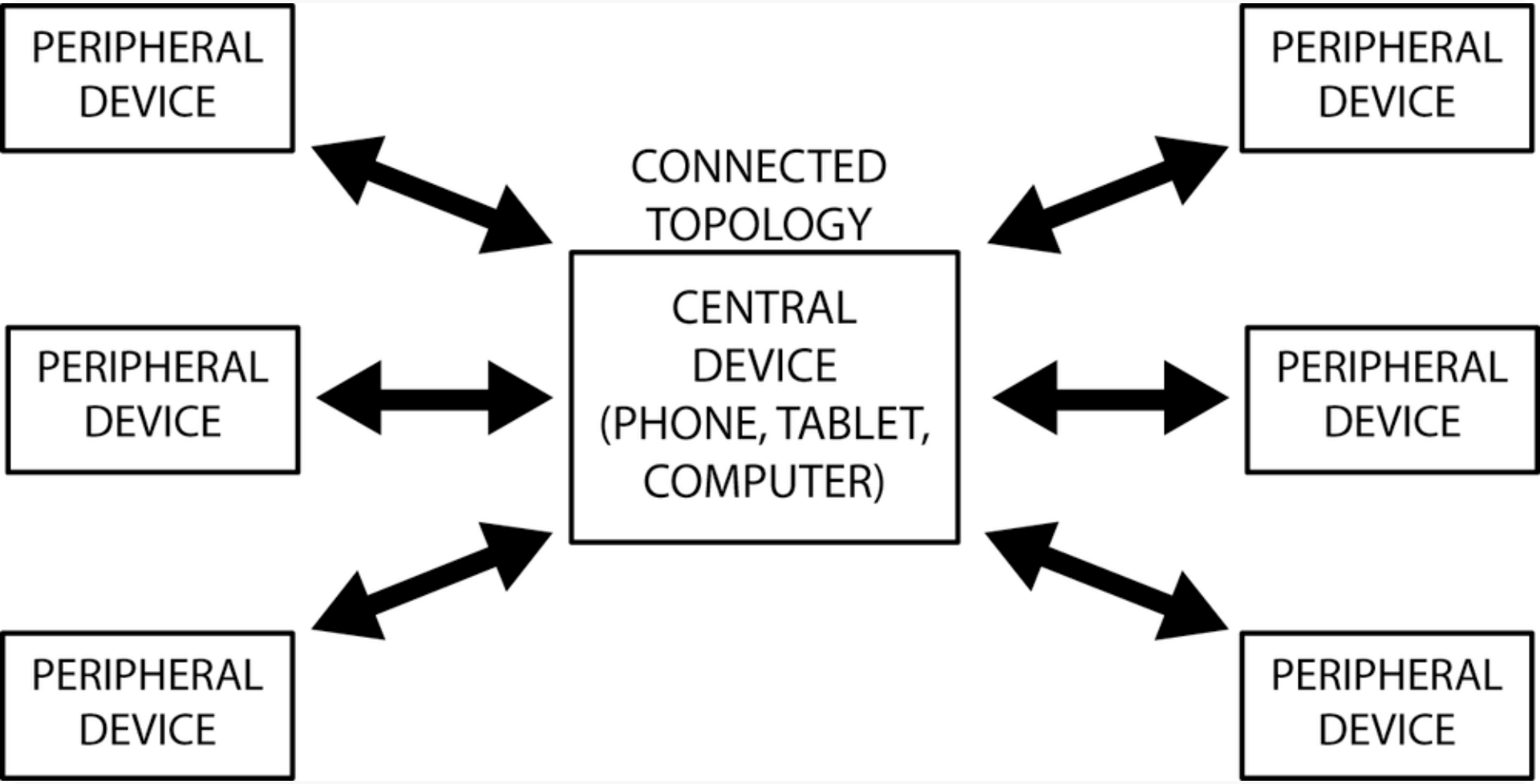
NETWORK TOPOLOGY: BROADCASTING



NETWORK TOPOLOGY: CONNECTIONS

- Required for transmitting data in both directions
- Required if more payload is requested than available in advertising packets
- Exclusive, permanent, periodical data exchange with packets between two devices (it is thus private - outside sniffing)

NETWORK TOPOLOGY: CONNECTIONS



NETWORK TOPOLOGY: CONNECTIONS

- Roles in the topology
  - Central or Master:
    - Listen for connectable advertising packets
    - Initiates the connection
    - Manages timing and initiates data exchanges
  - Peripheral or Slave:
    - Sends advertising packets
    - Accepts incoming connection requests (and stops advertising when connected)
    - Follows central's timing and exchanges data with it

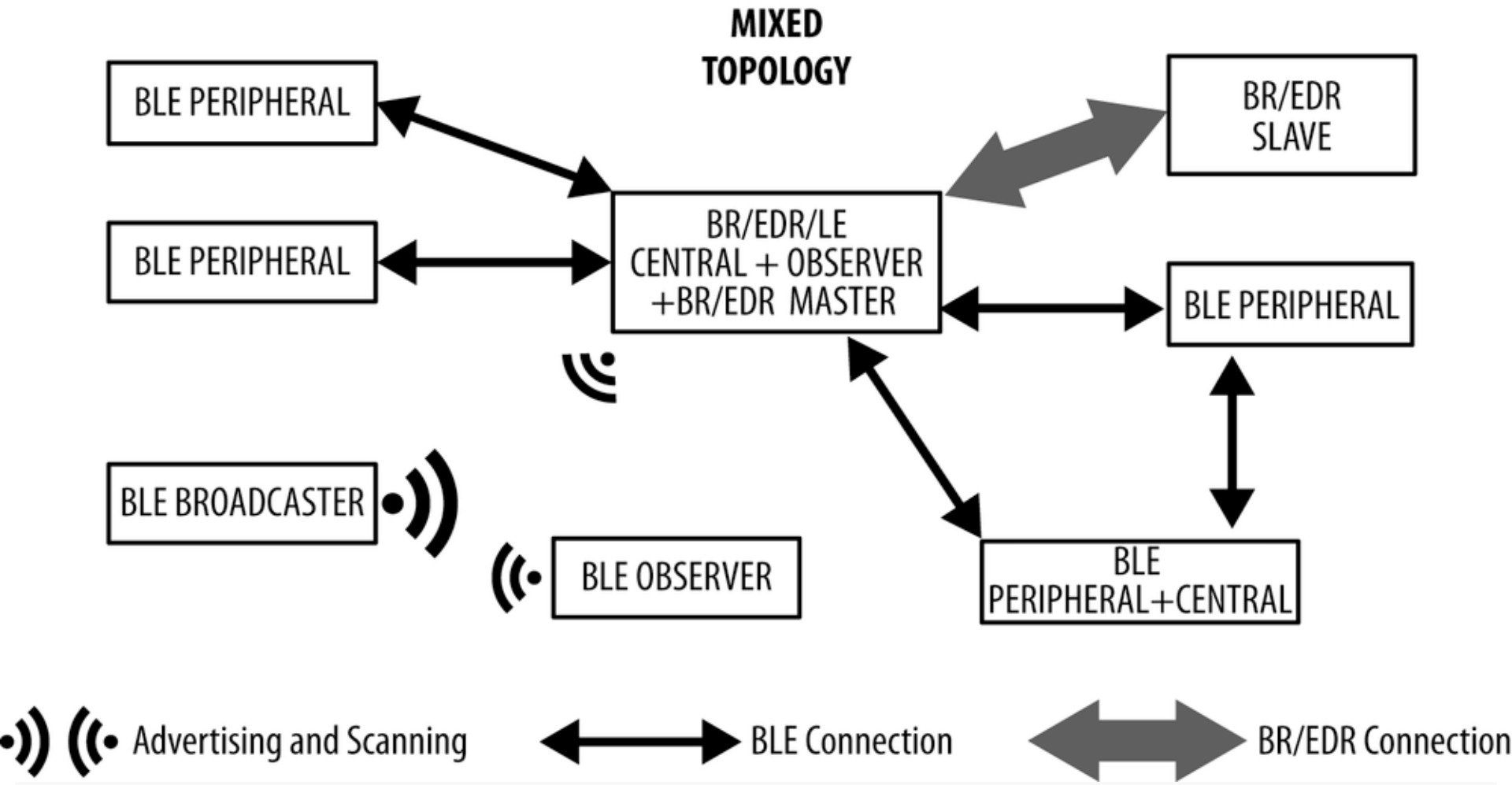
NETWORK TOPOLOGY: CONNECTIONS

- A device can act as a central and peripheral at the same time (since 4.1)
- A central can be connected to multiple peripherals
- A peripheral can be connected to multiple centrals (since 4.1)

NETWORK TOPOLOGY: CONNECTIONS

- Rich, layered data model: data is organized around units called services and characteristics (specified in the Generic Attribute Profile or GATT)
  - Multiple services organized in a meaningful structure
  - Services can contain multiple characteristics
  - Characteristics can define their own access rights and descriptive metadata
- Uses less power than in broadcast mode
  - Can extend the delay between connection events
  - Can push large chunks of data upon data changes only
  - Timing for connection events is known from both peers

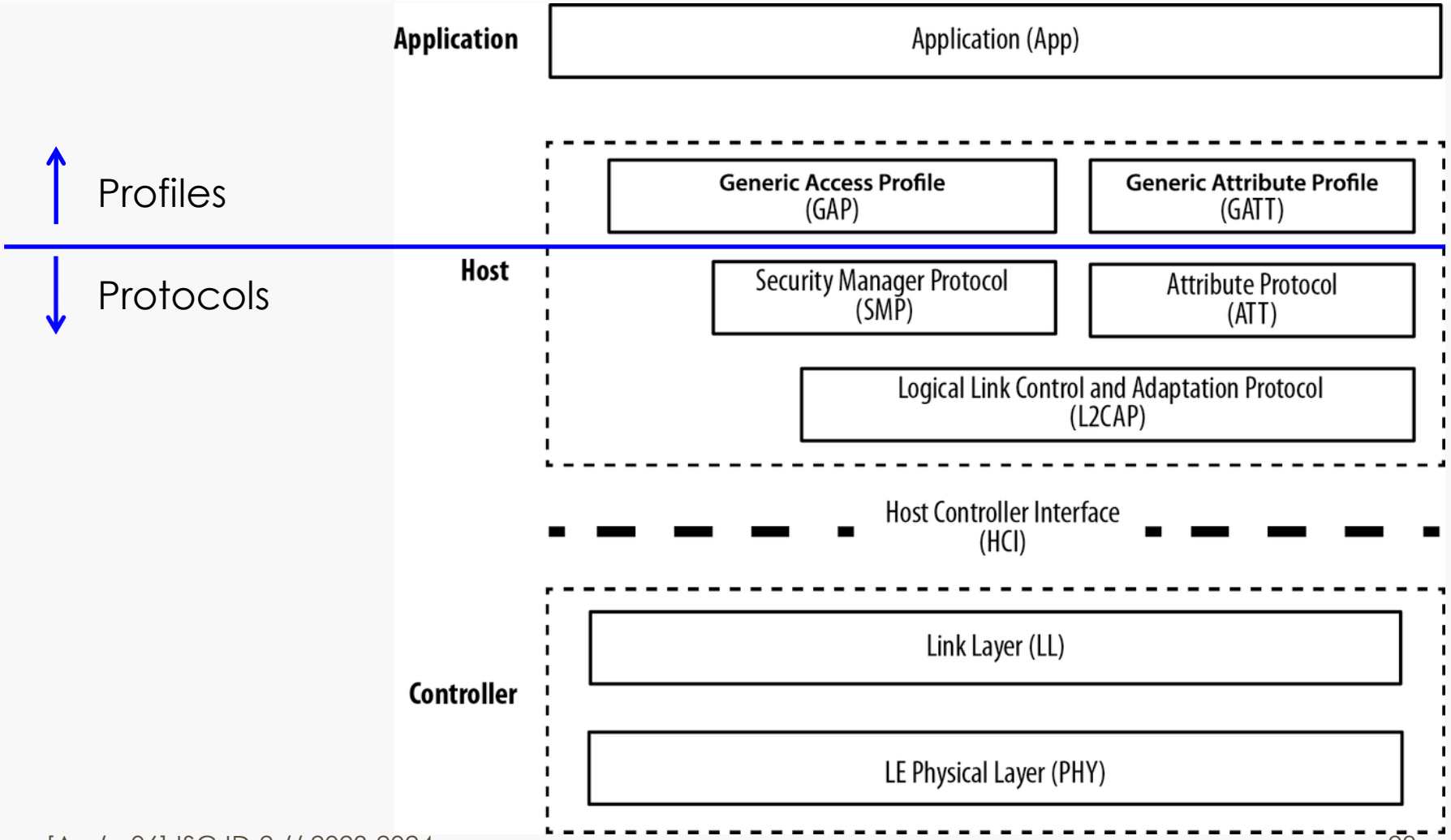
NETWORK TOPOLOGY: MIXED TOPOLOGY



PROTOCOLS VS. PROFILES

- Clear separation between protocols and profiles
- Protocols
  - Layers that implement the different packet formats, routing, multiplexing, encoding, and decoding
- Profiles
  - Functionality covering
    - Basic modes of operation for all devices (Generic Access Profile (GAP) or Generic Attribute Profile (GATT))
    - Specific standardized use cases (e.g. Proximity Profile, Environmental Sensing Profile)
    - Vendor specific profiles
  - Defines how protocols must be used to achieve a particular goal

PROTOCOLS VS. PROFILES



GENERIC PROFILES: GAP

- Generic Access Profile (GAP)
  - Roles, procedures and modes to
    - Broadcast data
    - Discover devices
    - Establish connections
    - Manage connections
    - Negotiate security levels
  - Is the topmost control layer
  - Mandatory for all BLE devices

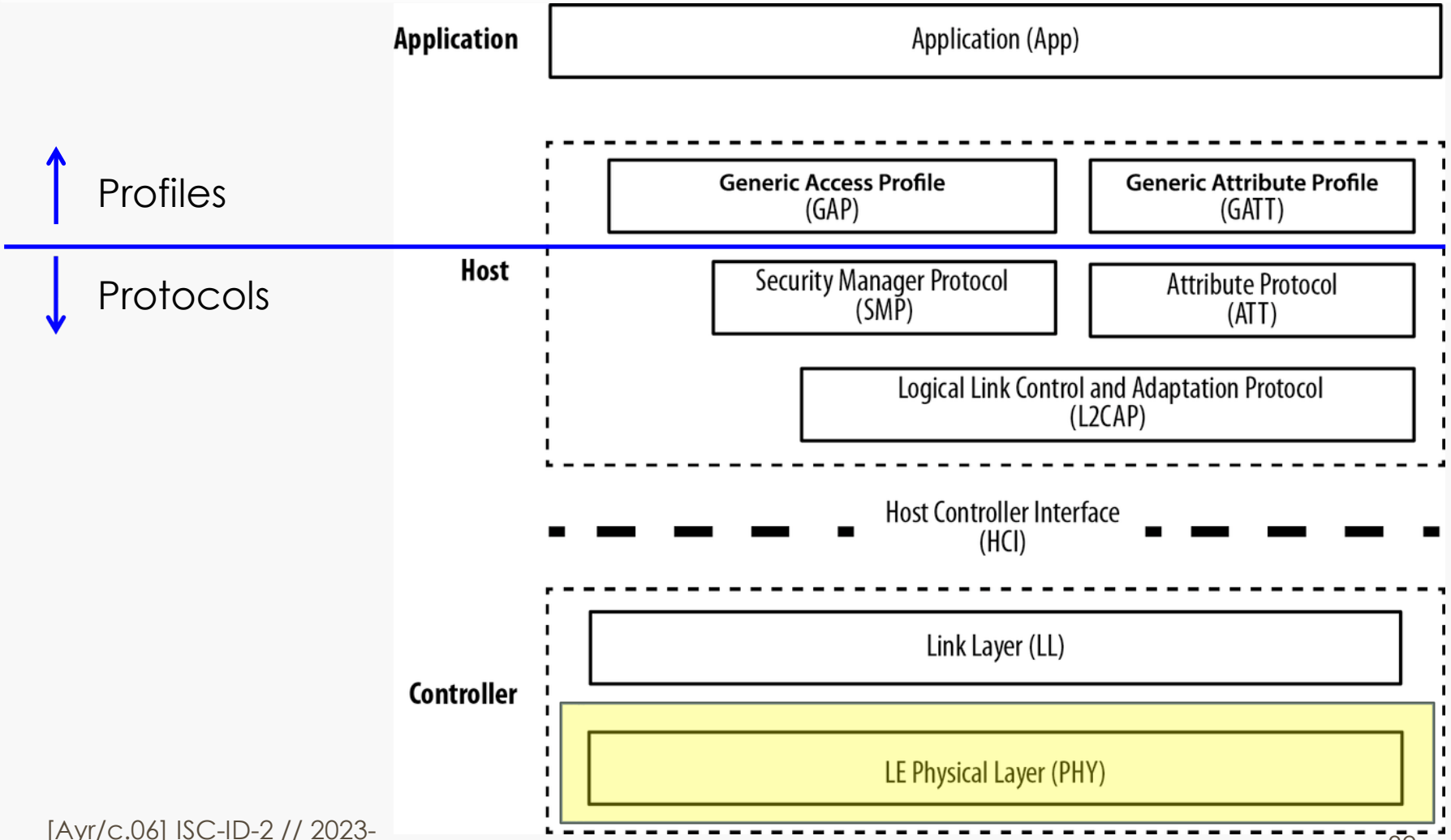
GENERIC PROFILES: GATT

- Generic Attribute Profile (GATT)
  - Data model and procedures to discover, read, write and push data elements between devices
  - Is the topmost data layer
  - (Almost) mandatory – other ways of exchanging data have been introduced with 4.1

USE CASE SPECIFIC PROFILES

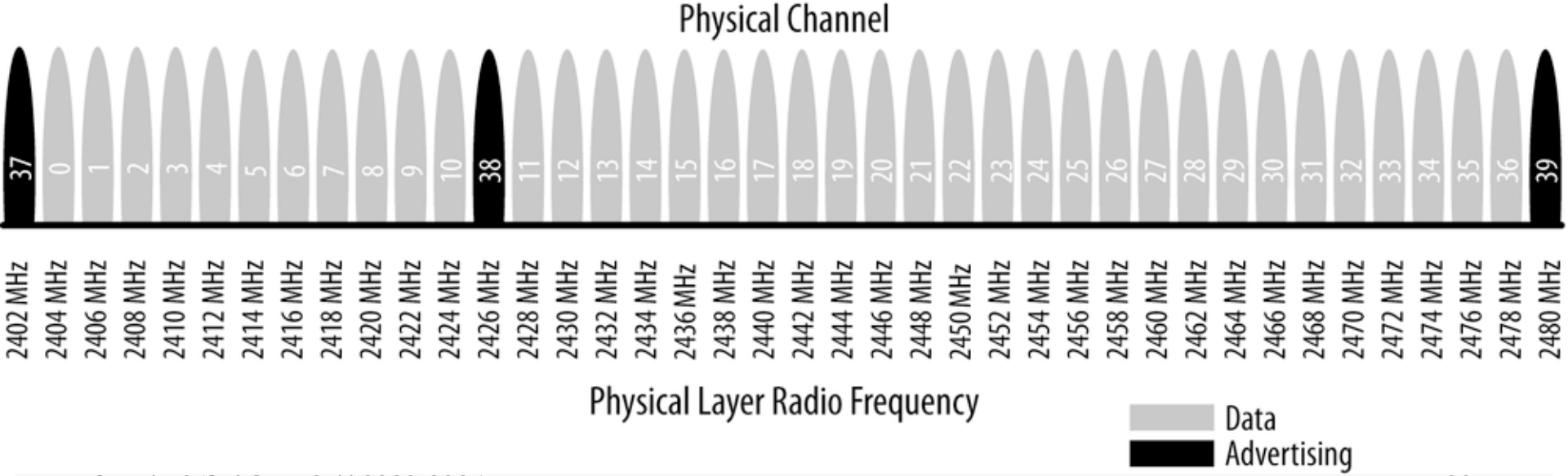
- Proximity profile
  - Detects the presence or absence of nearby devices
  - Use case: beep if a piece of luggage is beyond a certain distance
- Health Thermometer Profile
  - Transfers body temperature readings over BLE
- Cycling Speed and Cadence Profile
  - Allows sensors on a bicycle to transfer speed and cadence data to a smartphone or tablet
- Vendor specific profiles
  - Can be kept private or be published

BLUETOOTH STACK: PHYSICAL LAYER



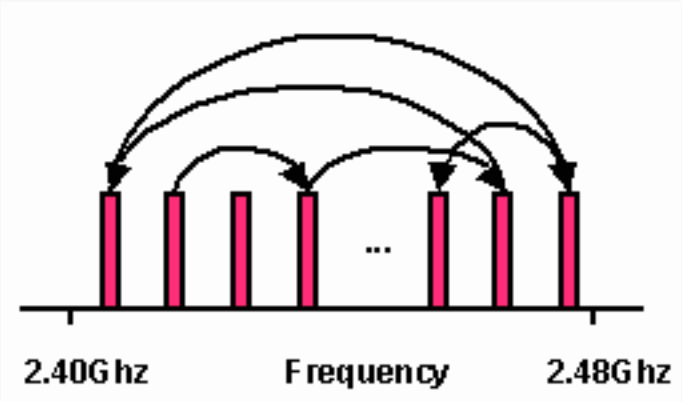
PHYSICAL LAYER

- 2.4 GHz ISM (Industrial, Scientific, and Medical) band
- 40 channels from 2.4000 GHz to 2.4835 GHz
- 37 channels for connections, 3 channels for advertising



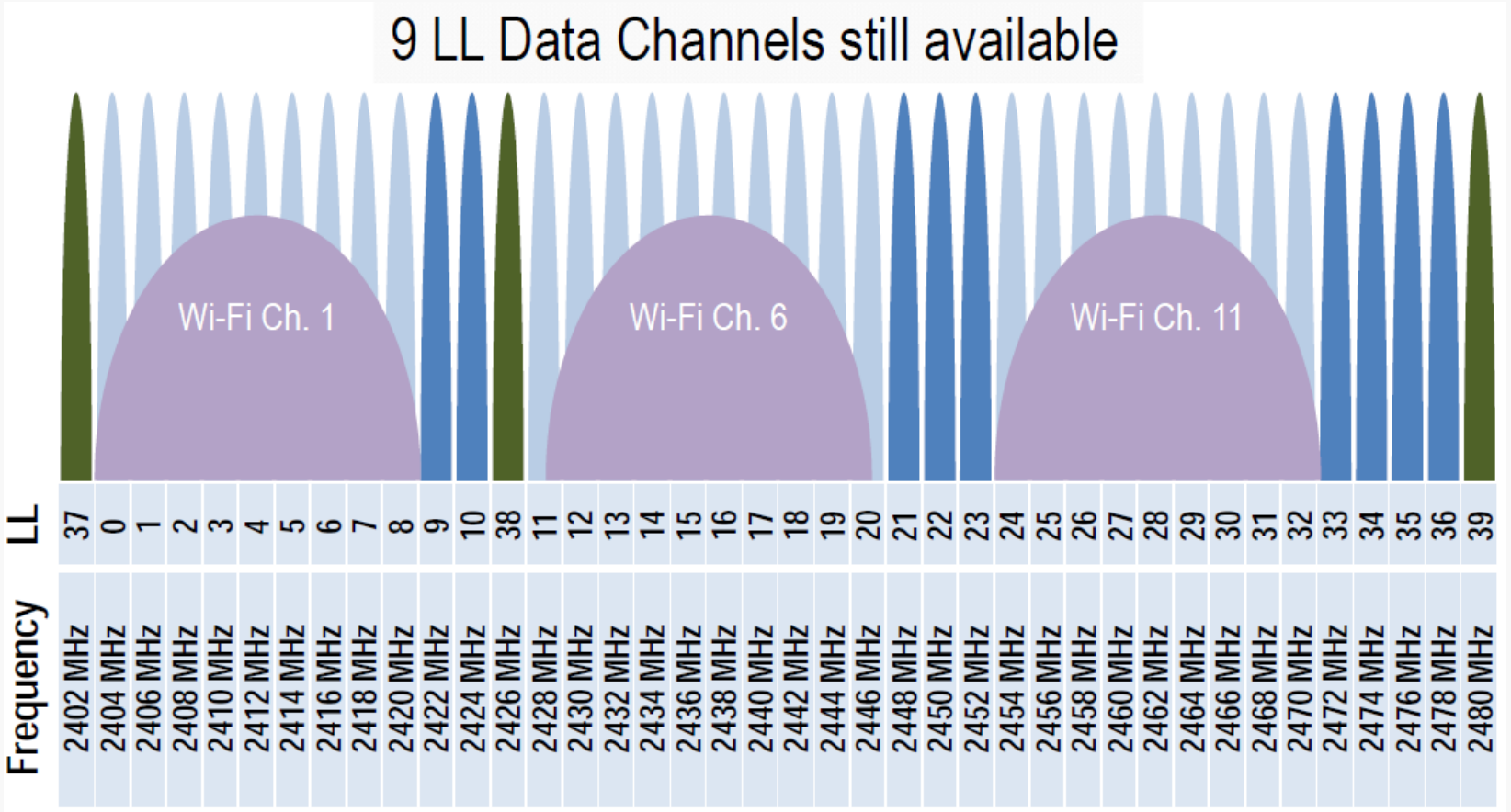
PHYSICAL LAYER: FREQUENCY HOPPING

- Frequency hopping spread spectrum
  - Rapid switching among many frequency channels
  - channel = (curr_channel + hop) mod 37
  - Hop value communicated when the connection is established
  - Minimizes the effect of radio interference potentially present in the band (Wifi and classic Bluetooth)
  - Modulation rate is fixed at 1Mbit/s (upper physical throughput limit)

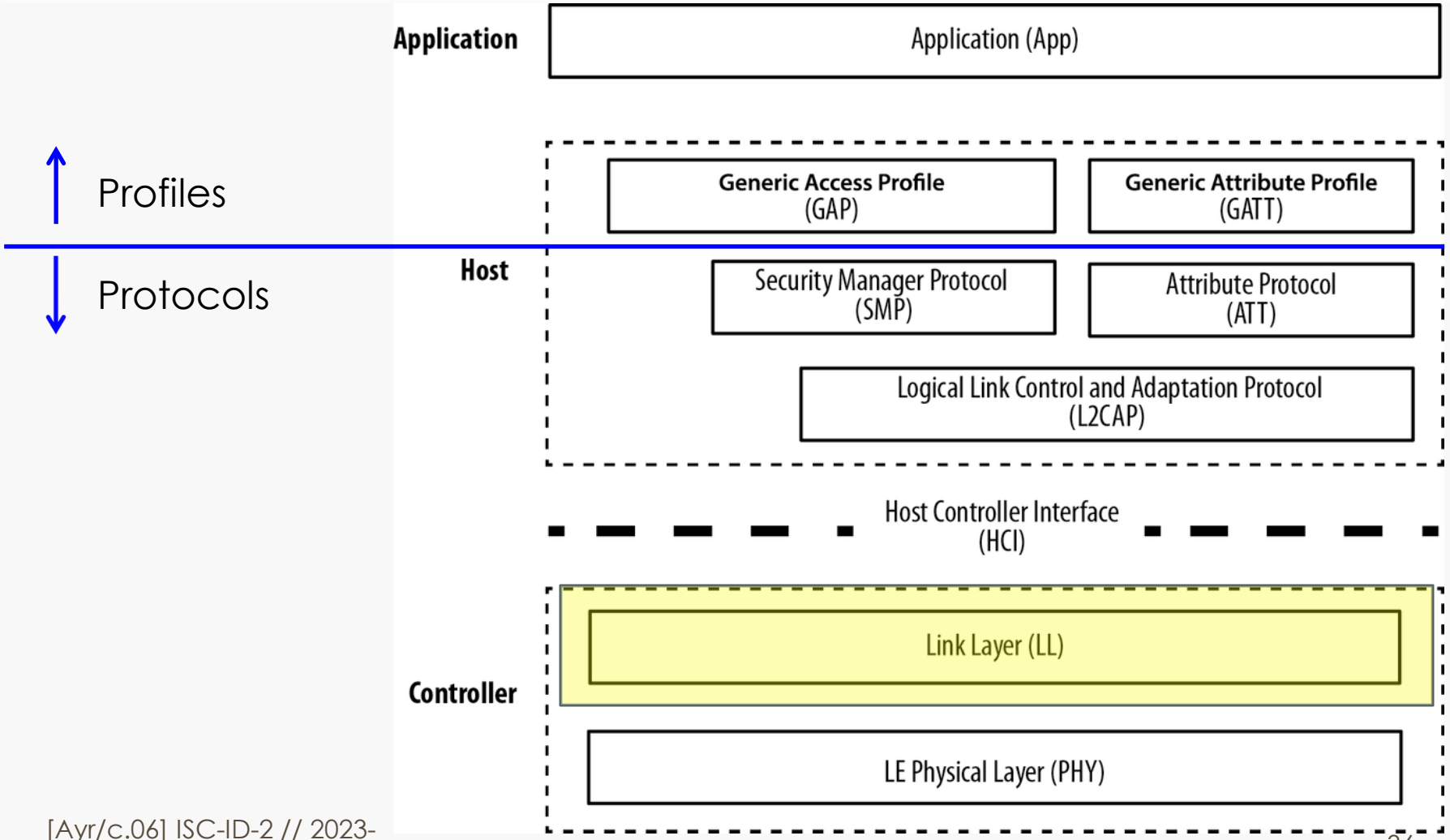


PHYSICAL LAYER

- Advertising channels avoid 802.11



BLUETOOTH STACK: LINK LAYER



LINK LAYER

- Interface to the physical layer
- Interfaces to the higher layers through the Host Controller Interface (HCI)
- Implemented as a combination of software and hardware
  - Has hard real-time constraints

LINK LAYER

- Defines the following roles:
  - Advertiser: a device sending advertising packets
  - Scanner: a device scanning for advertising packets
  - Master: a device that initiates a connection and manages it
  - Slave: a device that accepts a connection and follows the master's timing
- Defines the device addresses (acting like MAC addresses)
  - Public, fixed factory-programmed (must be registered with the Registration Authority)
  - Random, pre-programmed or dynamically generated

LINK LAYER: PACKETS

- One packet format

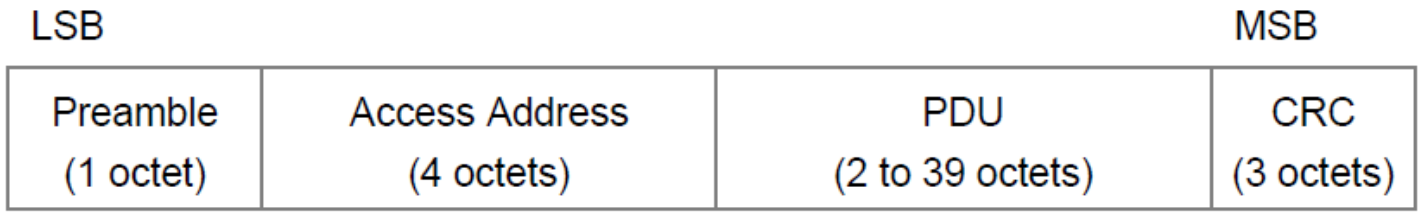


Figure 2.1: Link Layer packet format

- Two packet types: advertising and data
- Preamble:
  - Used for frequency synchronization and timing at the receiver
  - Fixed for advertising packets
- Access address:
  - Fixed for advertising packets
- PDU (Protocol Data Unit)
  - Advertising Channel PDU
  - Data Channel PDU

LINK LAYER: ADVERTISING

- Two purposes
  - For advertising data outside of a connection
  - For device (slave) discovery for establishing a connection
- Sent over the air without knowing whether a scanner is present
- Sent at fixed rates, between 20 ms and 10.24 secs
- Sent over the 3 advertising channels

LINK LAYER: ADVERTISING

- Advertising channel PDU

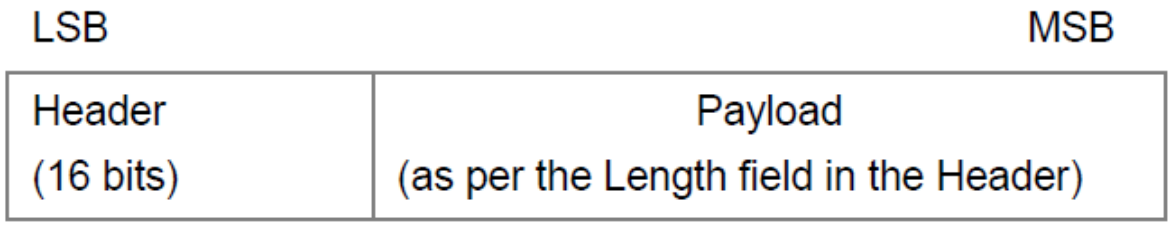


Figure 2.2: Advertising channel PDU

- Advertising channel PDU header

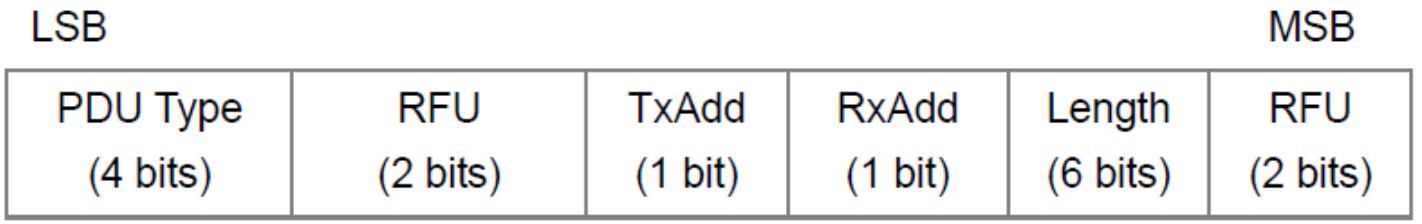


Figure 2.3: Advertising channel PDU Header

LINK LAYER: ADVERTISING

- Advertising packet
  - Sent by the Link Layer in the Advertising State
  - Received by a Link Layer in the Scanning or Initiating State
- Advertising packets have three different properties:
  - Connectability:
    - tells to the scanner whether the advertiser may accept a connection (connectable/non-connectable)
  - Scannability:
    - tells to the scanner whether the advertiser may accept a scan request (scannable/non-scannable)
  - Directability:
    - tells whether the advertising packet is directed at a specific scanner or not (directed/undirected)
    - It if is directed, then the packet contains only the advertiser's and scanner's addresses and it is thus connectable.

LINK LAYER: ADVERTISING

- ADV_IND: Connectable Undirected Advertising
  - Connectable, Scannable, Not Directed
  - Payload

Payload	
AdvA (6 octets)	AdvData (0-31 octets)

Figure 2.4: ADV_IND PDU Payload

  - AdvA: Advertiser's public or random device address (indicated by TxAdd)
  - AdvData: Advertising data from the advertiser's host

LINK LAYER: ADVERTISING

- ADV_DIRECT_IND: Connectable Directed Advertising
  - Connectable, Not Scannable, Directed
  - Payload

Payload	
AdvA (6 octets)	InitA (6 octets)

Figure 2.5: ADV_DIRECT_IND PDU Payload

  - AdvA: Advertiser's public or random device address, as indicated by TxAdd
  - InitA: Public or random address of the device to which this PDU is addressed, as indicated by RxAdd
  - Designed for allowing fast reconnection, with very specific timing requirements

LINK LAYER: ADVERTISING

- ADV_NONCONN_IND: Non-Connectable Undirected Advertising
  - Not Connectable, Not Scannable, Not Directed
  - Payload

Payload	
AdvA (6 octets)	AdvData (0-31 octets)

Figure 2.6: ADV_NONCONN_IND PDU Payload

  - AdvA: Advertiser's public or random device address, as indicated by TxAdd
  - AdvData: Advertising data from the advertiser's host

LINK LAYER: ADVERTISING

- ADV_SCAN_IND: Scannable Undirected Advertising
  - Not Connectable, Scannable, Not Directed
  - Payload

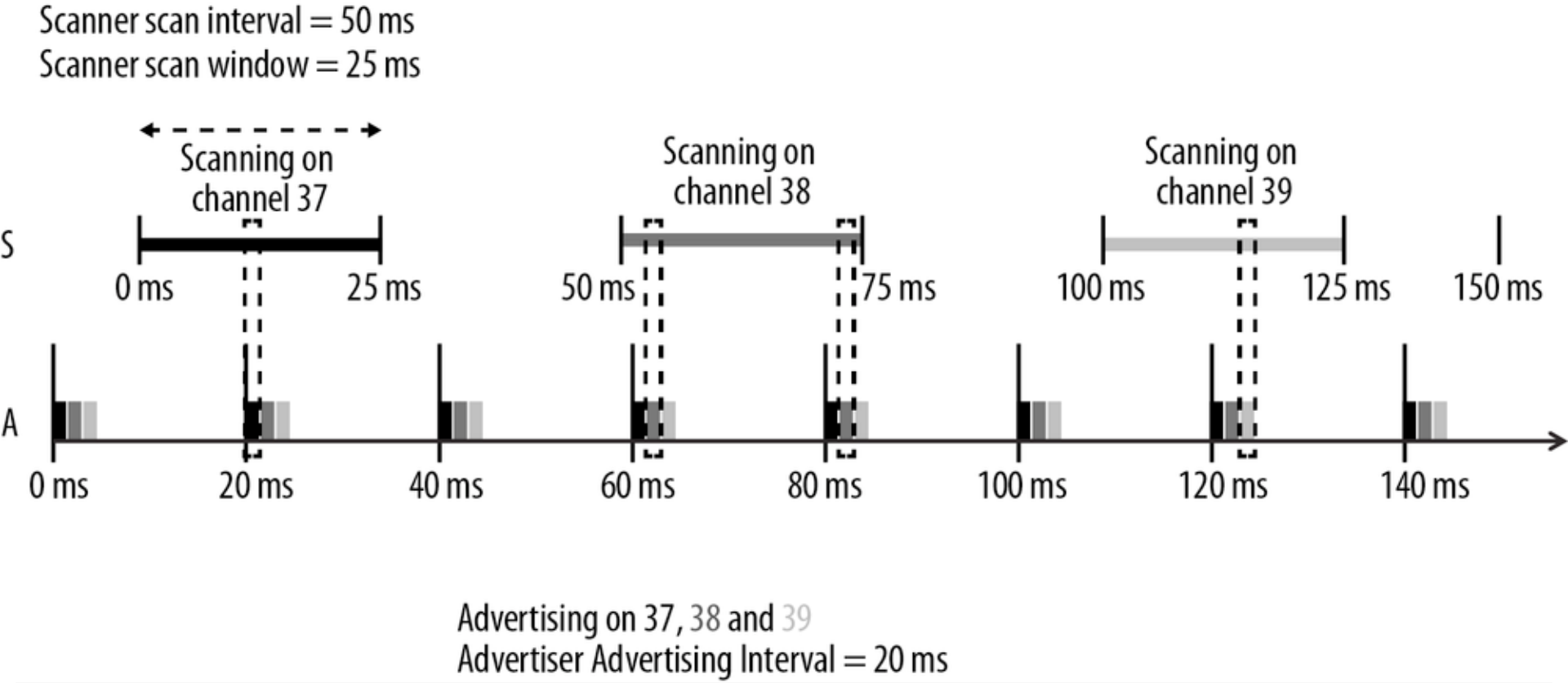
Payload	
AdvA (6 octets)	AdvData (0-31 octets)

Figure 2.7: ADV_SCAN_IND PDU Payload

  - AdvA: Advertiser's public or random device address, as indicated by TxAdd
  - AdvData: Advertising data from the advertiser's host

LINK LAYER: SCANNING

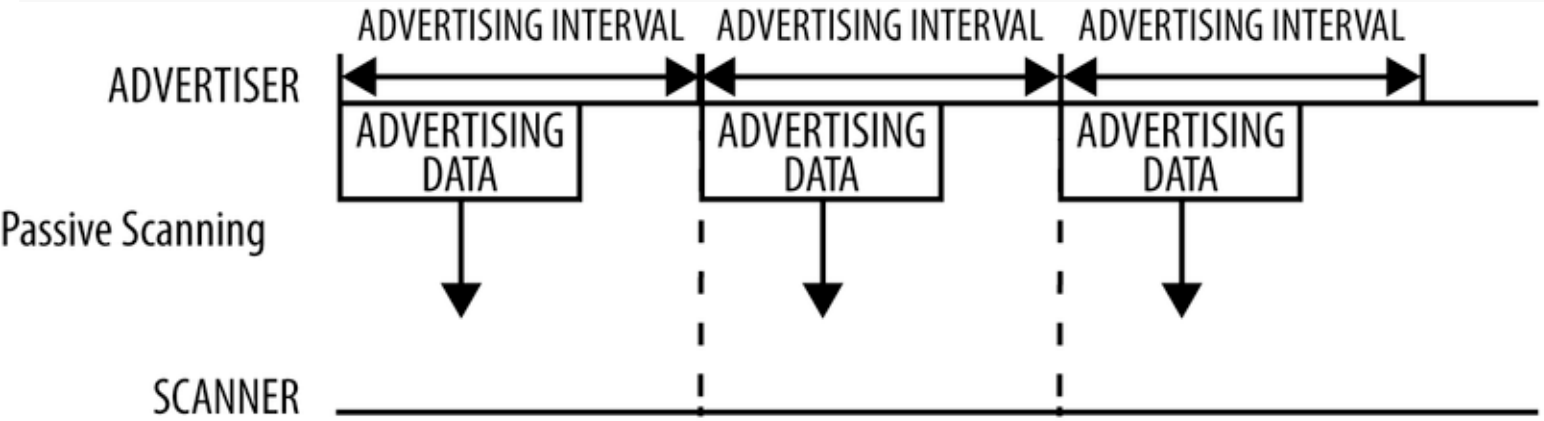
- Scan interval and scan windows influences the discovery speed, but also the power consumption

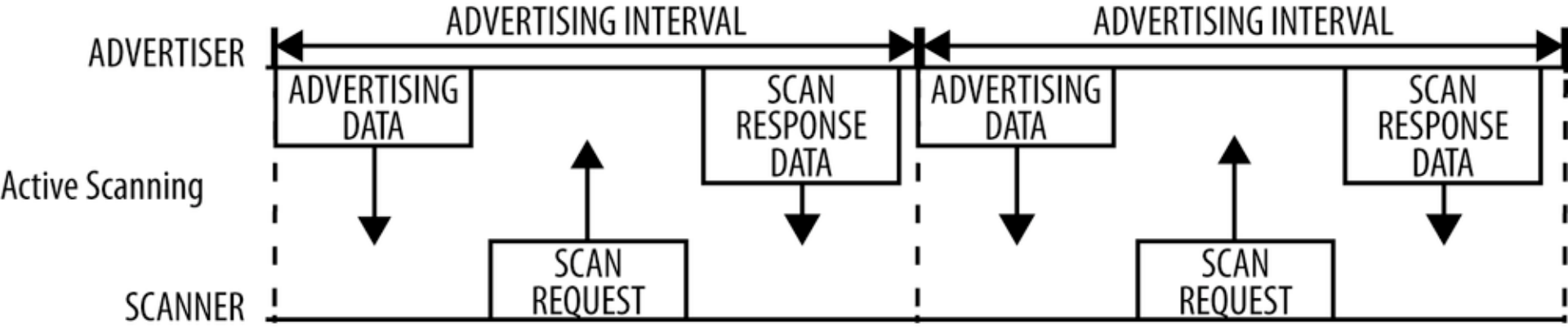


LINK LAYER: SCANNING

- Passive scanning
  - Scanner simply listens
  - Advertiser does not know whether any scanner received any packet
- Active scanning
  - Scanner issues a ScanRequest packet after receiving an advertising packet
  - Advertiser responds with a Scan Response packet
  - This mechanism doubles the effective payload that the advertiser is able to send to the scanner, without connection
  - This does not allow data transfer from the scanner to the advertiser

LINK LAYER: SCANNING





LINK LAYER: SCANNING

- SCAN_REQ:
  - Sent by the Link Layer in the Scanning State
  - Received by a LinkLayer in the Advertising State
  - Payload

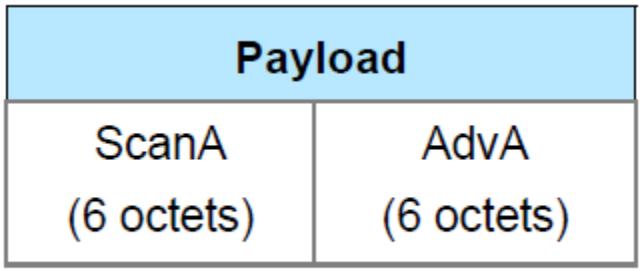


Figure 2.8: SCAN_REQ PDU Payload

  - ScanA: Scanner's public or random device address, as indicated by TxAdd
  - AdvA: Advertiser's public or random device address, as indicated by RxAdd

LINK LAYER: SCANNING

- SCAN_RSP:
  - Sent by the Link Layer in the Advertising State
  - Received by a LinkLayer in the Scanning State
  - Payload

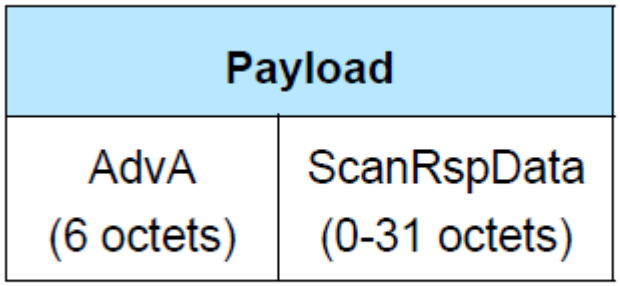


Figure 2.9: SCAN_RSP PDU payload

  - AdvA: Advertiser's public or random device address, as indicated by TxAdd
  - ScanRspData: any data from the advertiser's Host

LINK LAYER: ADVERTISING CHANNEL

- Advertising channel PDU Type

PDU Type $b_3b_2b_1b_0$	Packet Name
0000	ADV_IND
0001	ADV_DIRECT_IND
0010	ADV_NONCONN_IND
0011	SCAN_REQ
0100	SCAN_RSP
0101	CONNECT_REQ
0110	ADV_SCAN_IND
0111-1111	Reserved

Table 2.1: Advertising channel PDU Header's PDU Type field encoding

LINK LAYER: CONNECTIONS

- A master scans for advertisers that are accepting connections
  - The master can filter such devices by device addresses or based on the advertising data (e.g. filtering for a specific service).
  - The master can use white lists for filtering devices.
- The master then sends a connection request packet to the slave and, provided that the slave responds, establishes a connection with the slave.
- The connection packet includes information about how the connection must be operated

LINK LAYER: CONNECTIONS

- CONNECT_REQ:
  - Sent by the Link Layer in the Initiating State
  - Received by the LinkLayer in the Advertising State
  - Payload

Payload		
InitA (6 octets)	AdvA (6 octets)	LLData (22 octets)

Figure 2.10: CONNECT_REQ PDU payload

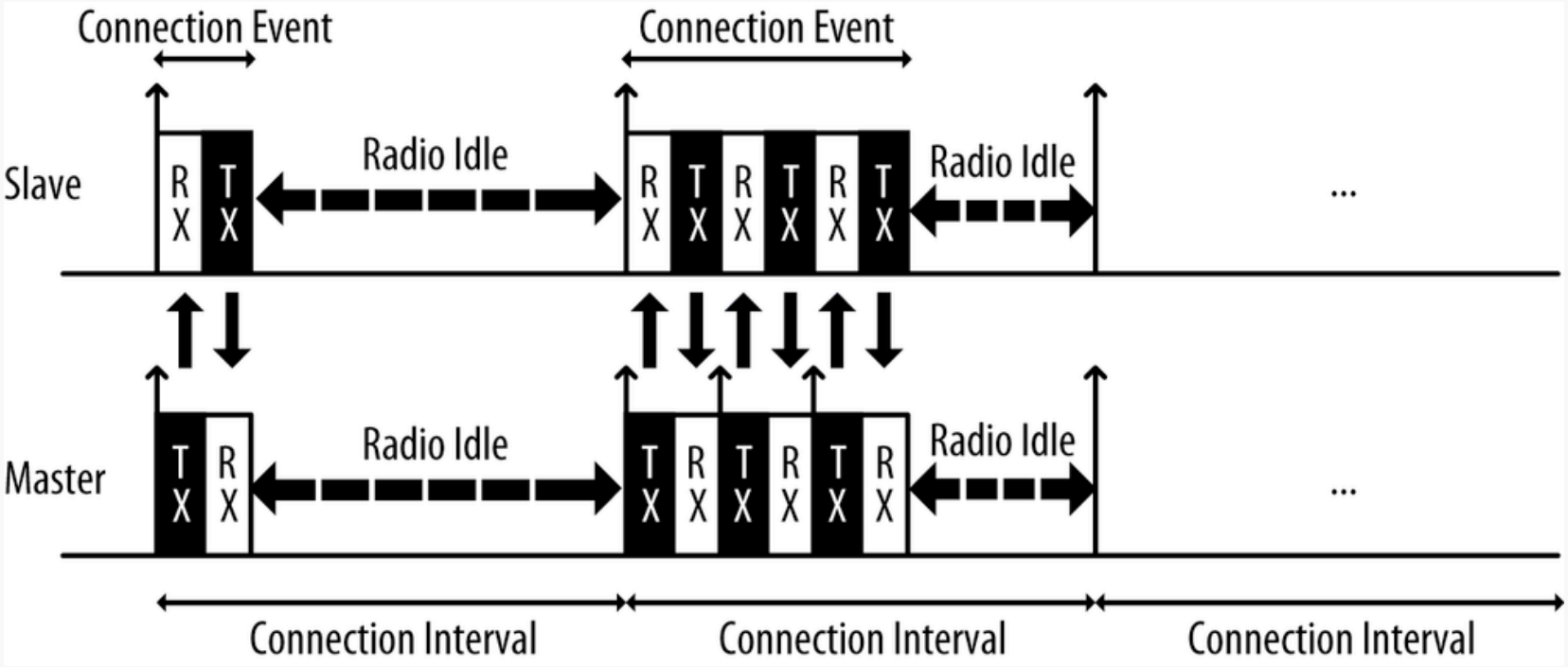
  - InitA: Initiator's public or random device address, as indicated by TxAdd
  - AdvA: Advertiser's public or random device address, as indicated by RxAdd
  - LLData

LLData									
AA (4 octets)	CRCInit (3 octets)	WinSize (1 octet)	WinOffset (2 octets)	Interval (2 octets)	Latency (2 octets)	Timeout (2 octets)	ChM (5 octets)	Hop (5 bits)	SCA (3 bits)

Figure 2.11: LLData field structure in CONNECT_REQ PDU's payload

LINK LAYER: CONNECTIONS

- A connection = sequence of data exchanges between the slave and the master at predefined times



LINK LAYER: CONNECTIONS

- Connection interval
  - Time between the beginning of two consecutive connection events
  - Ranges from 7.5 ms (high throughput) to 4 s (lowest possible throughput)
- Slave latency
  - Number of connection events that a slave can choose to skip without risking a disconnection
- Connection supervision interval
  - Maximum time between two received valid data packets before a connection is considered lost
- Hop increment
  - Defines the hopping sequence that both the master and the slave will follow during the lifetime of the connection