CS 439: Wireless Networking

MAC Layer – Bluetooth

Fall 2025

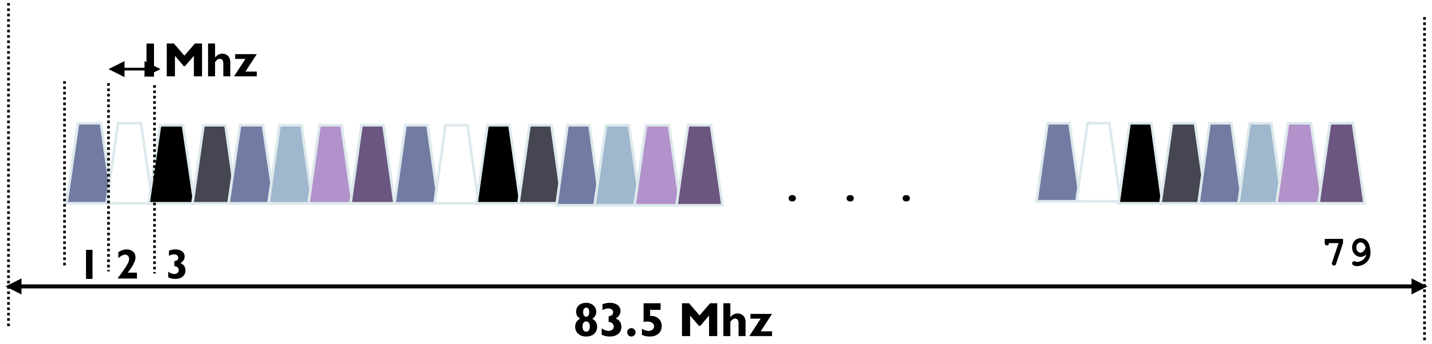
Bluetooth

- Harald Blaatand "Bluetooth" II
  - King of Denmark 940-981 AC
- Runic stones in his capital city of Jelling
  - The stone's inscription ("runes") says:
    - Harald Christianized the Danes
    - Harald controlled the Danes
    - Harald believes that devices shall seamlessly communicate [wirelessly]

Classic Bluetooth

- Cable replacement
  - 2.4 GHz
  - FHSS over 79 channels (of 1MHz each), 1600hops/s
  - 1Mbps
    - Upgraded to 1 or 2 Mbps in 5.0
  - Coexistence of multiple piconets
  - 10 meters (extendible to 100 meters)
    - Max Tx Power 10dB (extendible to 20dB in 5.0)

Bluetooth Radio

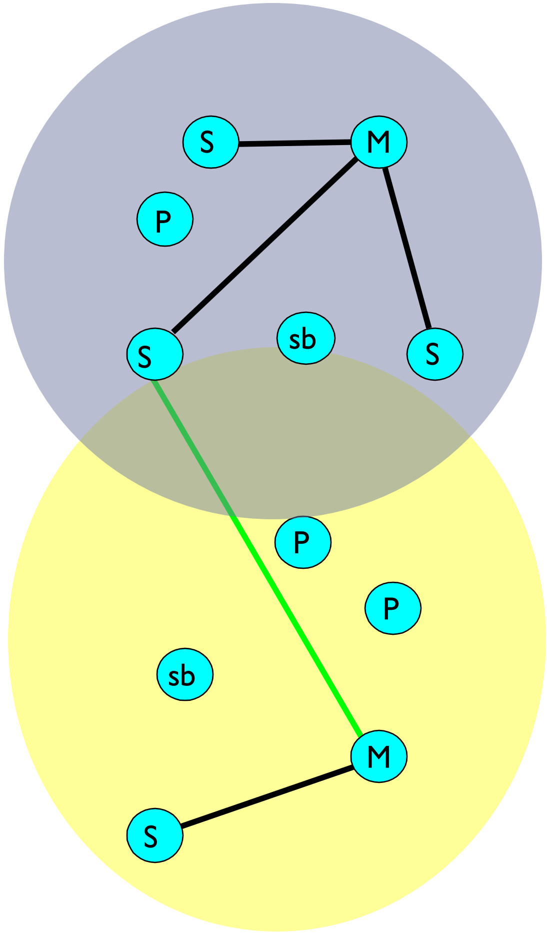


- MA scheme: Frequency hopping spread spectrum.
  - 2.402 GHz + k MHz, k=0, …, 78
  - 1,600 hops per second.
  - 1 Mbps data rate.

  - Upgraded to 2 Mbps in BT 5.0

Bluetooth Network Topology

- Radio designation
  - Connected radios can be master or slave
  - Radios are symmetric (same radio can be master or slave)
- Piconet
  - Master can connect to 7 simultaneous or 200+ inactive (parked) slaves per piconet
  - Each piconet has maximum capacity (1 Mbps)
  - Unique hopping pattern/ID
- Scatternet
  - High capacity system
  - Minimal impact with up to 10 piconets within range
  - Radios can share piconets!



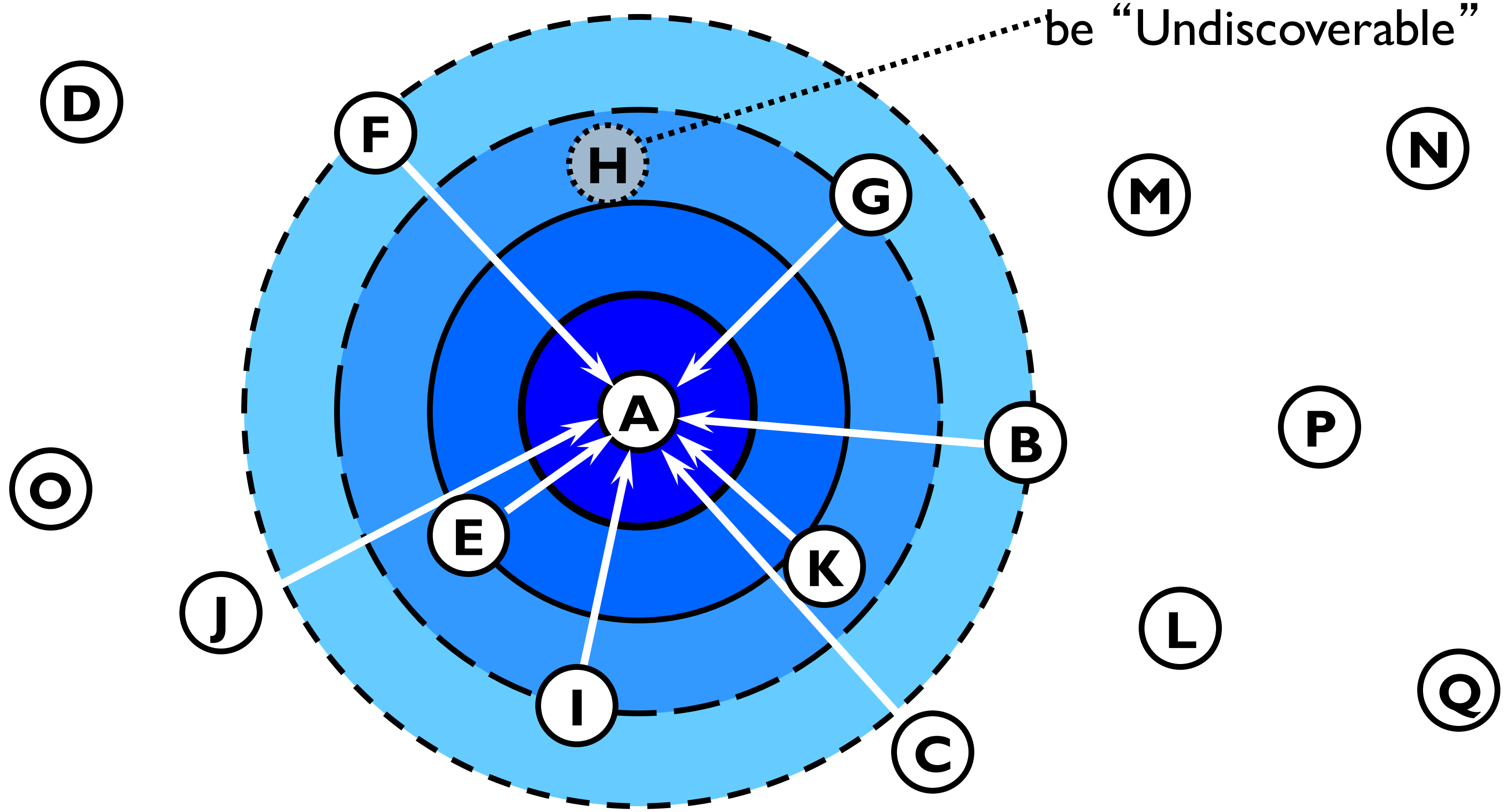
Bluetooth – Contention-free MAC

- Master performs medium access control
  - Schedules traffic through polling.
- Time slots alternate between master and slave transmission
  - Master-slave
    - Master includes slave address.
  - Slave-master
    - Only slave chosen by master in previous master-slave slot allowed to transmit.
  - If master has data to send to a slave, slave polled implicitly; otherwise, explicit poll.

Bluetooth Device Discovery - Inquiry

- Device discovery
  - Sends out an inquire, which is a request for nearby devices (within 10 meters)
  - Devices that allow themselves to be discoverable issue an inquiry response
  - Listeners respond with their address
  - Can take up to 10.24 seconds, after which the inquiring device should know everyone within 10 meters of itself

Bluetooth Device Discovery - Inquiry



Note that a device can be “Undiscoverable”

10 meters

After inquiry procedure, A knows about others within range

Bluetooth Inquiry

- Sender
  - Inquiry sent on 16 different frequencies
  - 16 channel train
    - about 1.28 seconds per channel
    - One full 16 channel train takes 10ms
- Receiver (device in standby mode)
  - Scans long enough for an inquiring device to send the inquiry on 16 frequencies
  - Scan must be frequent enough to guaranteed wake up during a 16 channel train
    - Enters inquiry scan state at least once every 1.28 seconds, and stays in that state for 10ms

Bluetooth Inquiry - Reliability

- Challenge
  - Noisy channels
  - Lost packets
    - Train scan is repeated up to 4 times for each train (10.24 seconds)
    - Designed to successfully communicate at least once with all devices within range

Bluetooth Progression

Bluetooth 1.0

Initial version

Bluetooth Progression

Initial version

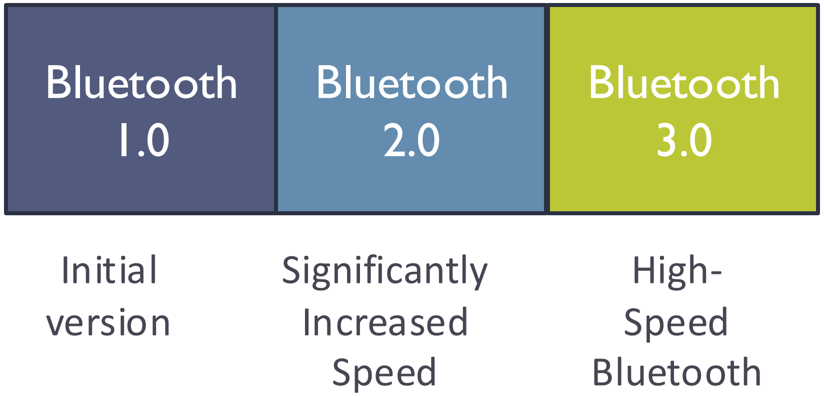
Significantly Increased Speed

Data Rate ↑ (3Mbps↓)

Improved power consumption

Expand to audio devices

Bluetooth Progression



Bluetooth 1.0

Initial version

Bluetooth 2.0

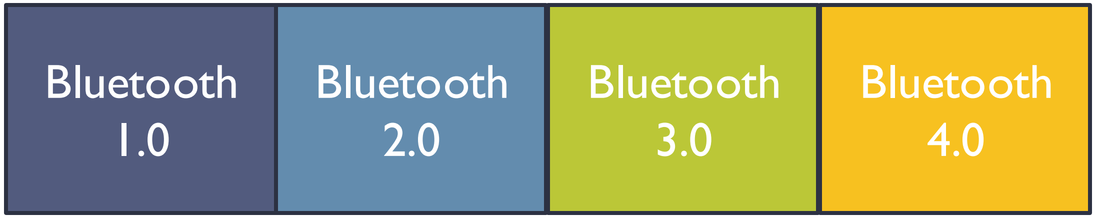
Significantly Increased Speed

Bluetooth 3.0

High-Speed Bluetooth

Bluetooth + Wifi Data Speed (24Mbps)
Efficiency/ Connection Stability/ Power Control …

Bluetooth Progression



Bluetooth 1.0

Bluetooth 2.0

Bluetooth 3.0

Bluetooth 4.0

Initial version

Significantly Increased Speed

High-Speed Bluetooth

Bluetooth Low Energy

Market demands:
Low Power (0.01 - 0.5mW)
Longer Range
Decent speed
Faster discovery

BLE Highlights

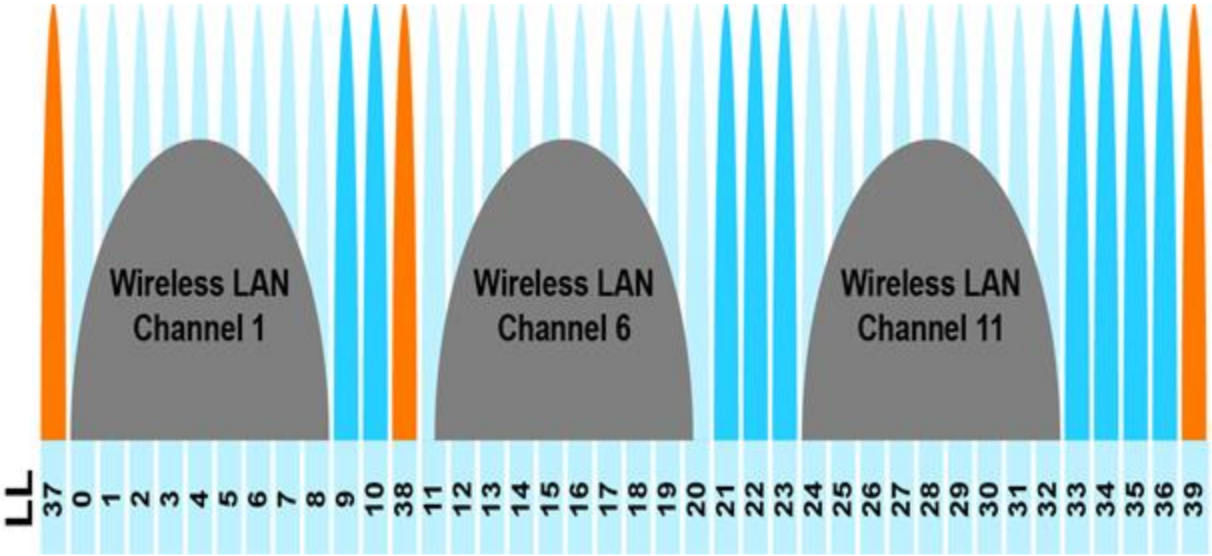
- Shared wireless channel
  - BLE operates in the 2.4 GHz ISM band with Wi-Fi and other technologies (phones, microwave ovens …)
- BLE = Bluetooth Low Energy
  - Improved discovery
  - Key component: Beacons
    - Tags send out advertising beacons (typ. dist 30ft)
    - Phones scan for beaco

Wi Fi ™

ZigBee®

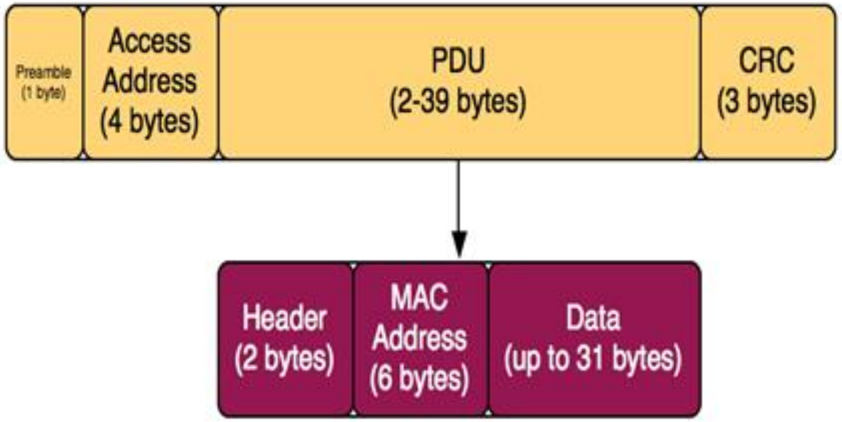
BLE Highlights: Channel Use and Coexistence with Wi-Fi

- Separate advertising and connected channels
  - Key: Three disjoint advertising channels (37, 38, 39)
  - Positioned between Wi-Fi channels (1, 6, 11)



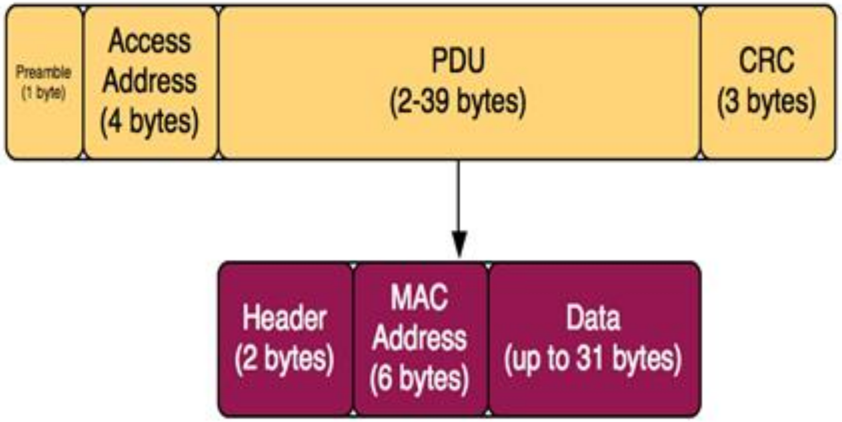
BLE Highlights: Advertising

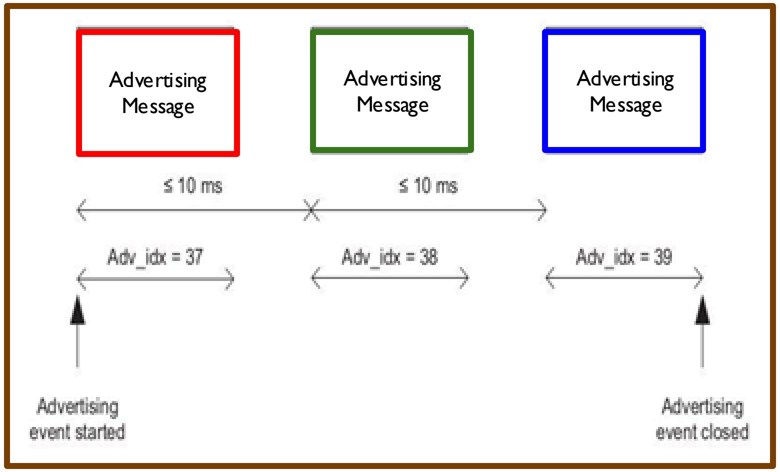
- Advertising Tags
- Advertising Messages
  - Header + MAC Address + up to 31 Bytes of data
    - ~200 - 400 usec per packet
  - Two types: Non-scannable, Scannable



BLE Highlights: Advertising

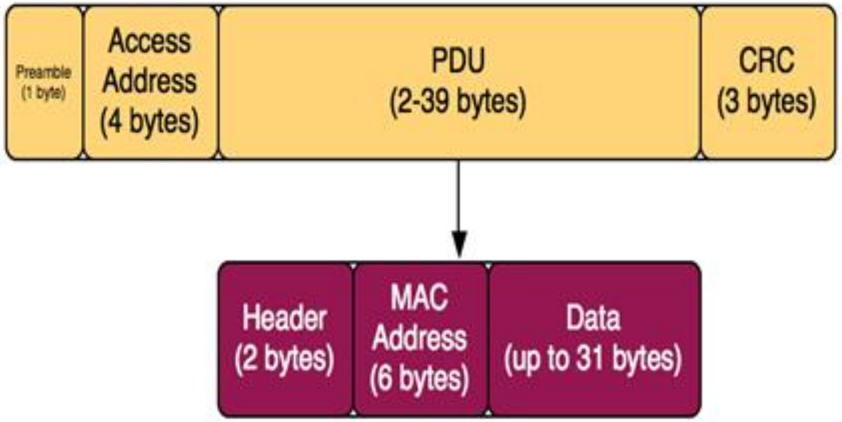
- Advertising Tags
- Advertising Messages
  - Header + MAC Address + up to 31 Bytes of data
    - ~200 - 400 usec per packet
  - Two types: Non-scannable, Scannable
- Advertising Event
  - One advertising message sent out on each advertising channel (37, 38, 39)

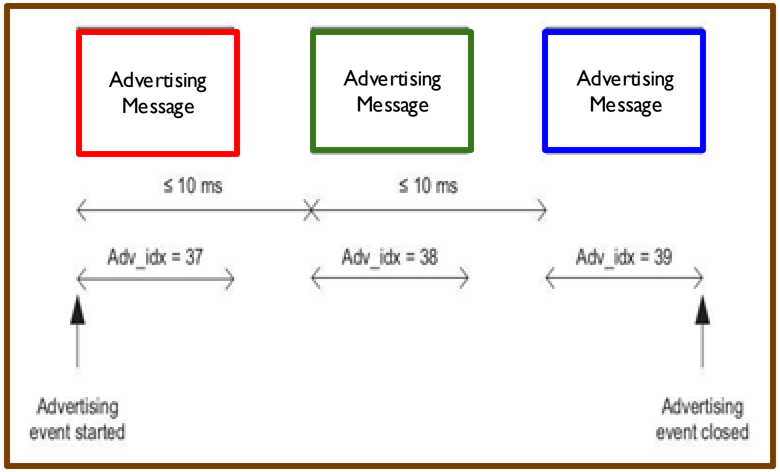




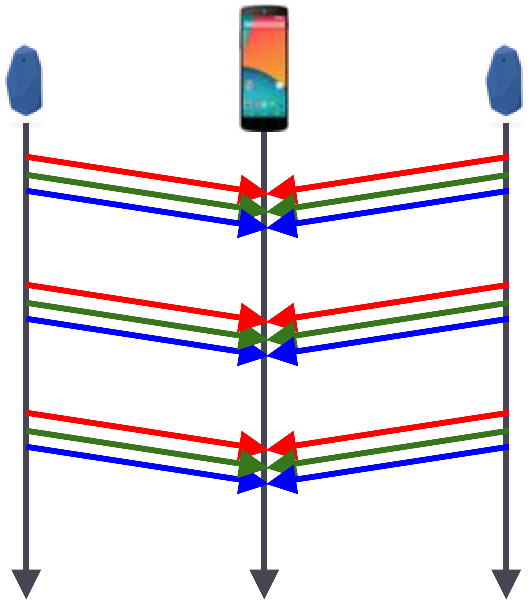
BLE Highlights: Advertising

- Advertising Tags
- Advertising Messages
  - Header + MAC Address + up to 31 Bytes of data
    - ~200 - 400 usec per packet
  - Two types: Non-scannable, Scannable
- Advertising Event
  - One advertising message sent out on each advertising channel (37, 38, 39)
- Advertising Interval
  - One advertising event per advertising interval
  - e.g., every 1 sec or 100 msec



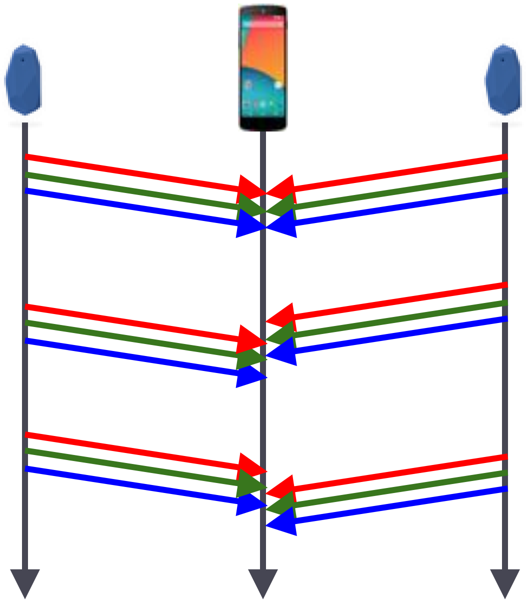


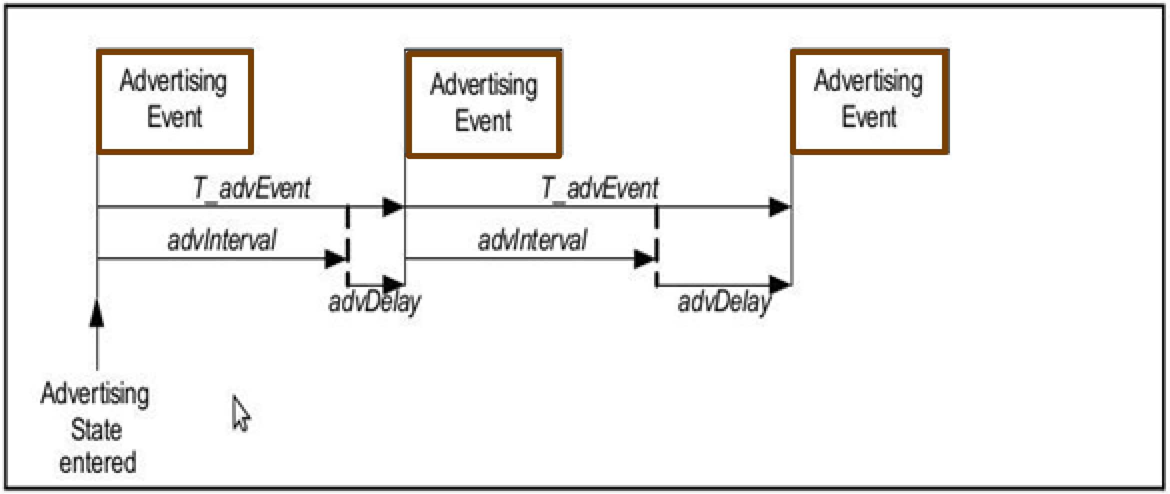
BLE Highlights: Advertising and Collisions



- If tags get synchronized, all advertising messages will collide

BLE Highlights: Advertising and Collisions

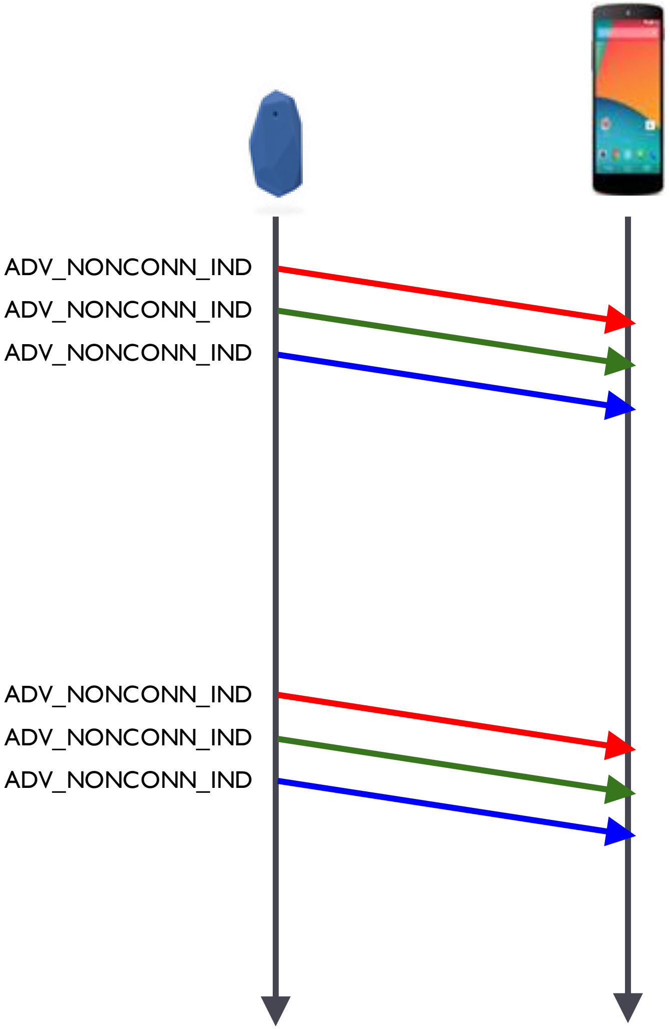


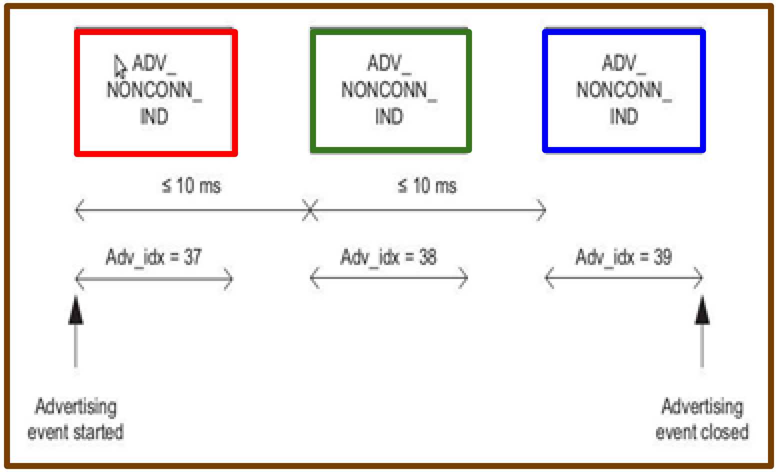


- Collision avoidance
  - Jitter advertising times
  - advDelay is added on to the end of each advertising event
  - advDelay = rand [0,10ms]

BLE Highlights: Tags Types - Non-Scannable

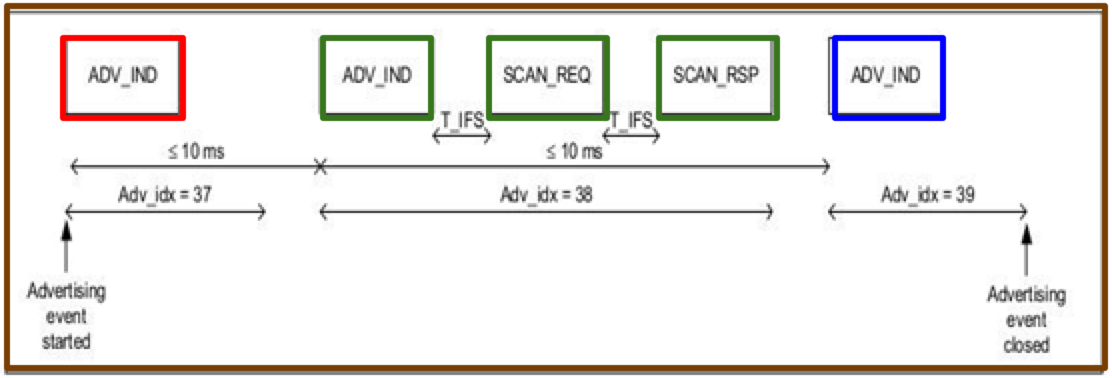
- Non-Scannable Tags
- Ex. gBeacon v3, iBeacon (?)
- Tags send ADV_NONCONN_IND messages
- Typically sent back-to-back
- Scanners listen, but do not respond

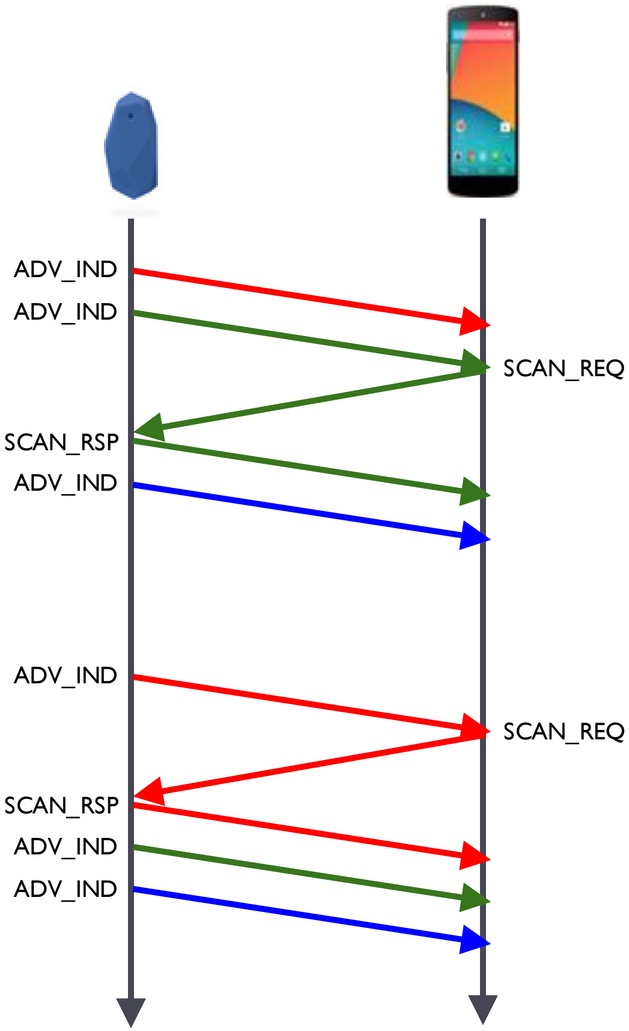




BLE Highlights: Tags Types - Scannable

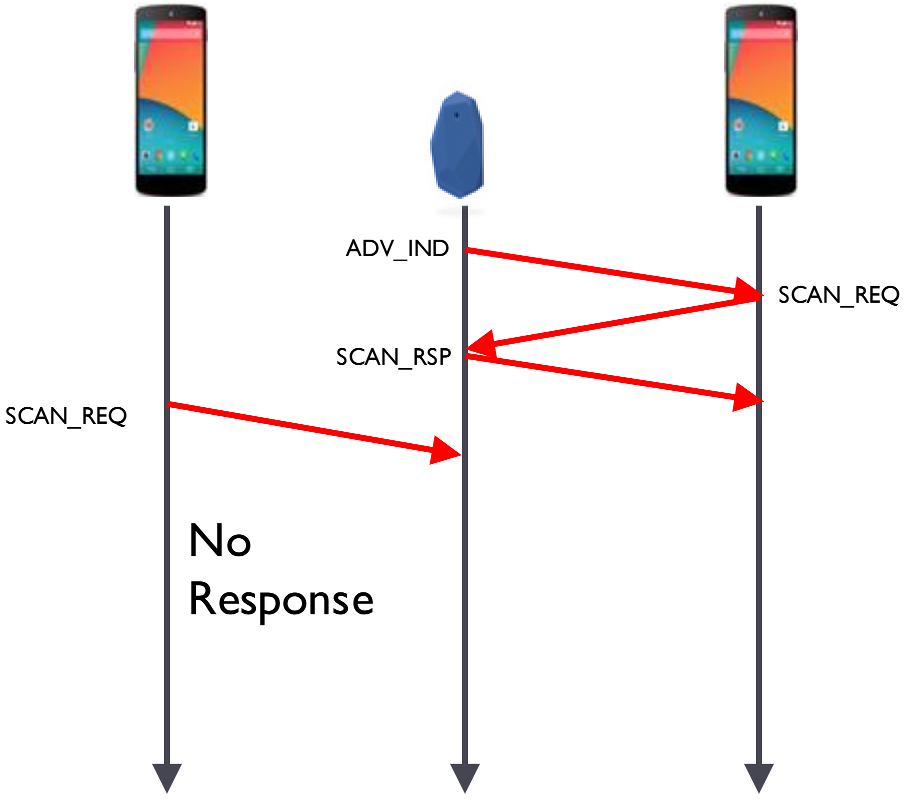
- Scannable Tags
- Ex. gBeacon V1, Estimote
- Tags send ADV_IND messages
- Scanners respond with SCAN_REQ message
- Tags respond with SCAN_RSP message
  - Up to 31 Bytes of extra data
- Tags wait ~150 usec for a request after beacon





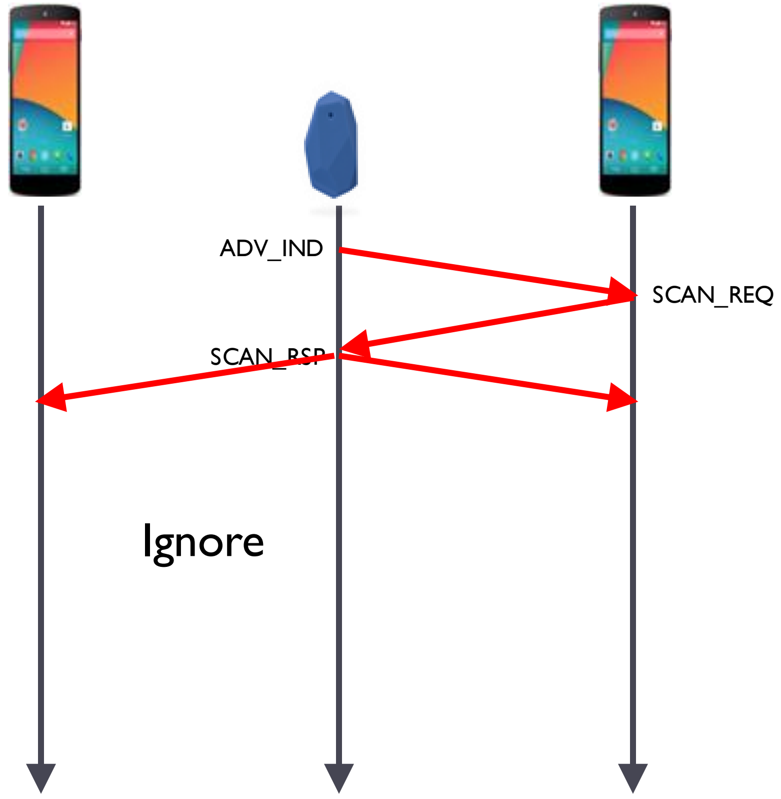
Scannable Tags

- One SCAN_RSP per channel per advertising event



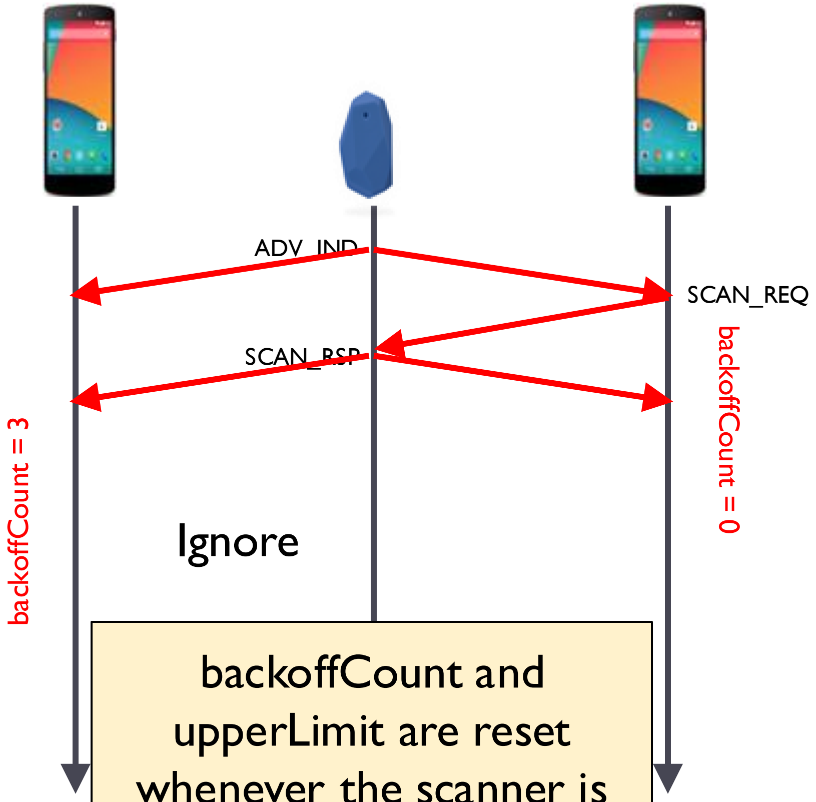
Scannable Tags

- ONLY accept SCAN_RSP if SCAN_REQ was sent to that tag on that channel during that advertising event
- Some collision tolerance
  - Any requesting scanner can receive a SCAN_RSP as long as one SCAN_REQ is received and the tag responds
  - BUT, No SCAN_RSP if all SCAN_REQs collide



SCAN_REQ Collision Avoidance

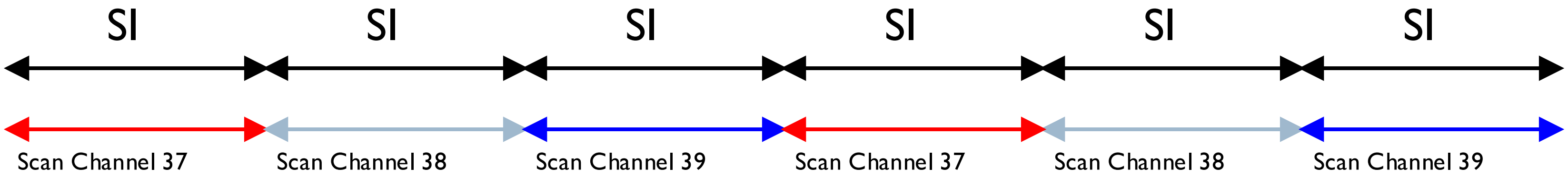
- Scanner backoff procedure
  - Two parameters
    - backoffCount, upperLimit
  - On starting scan
    - upperLimit = 1, backoffCount = 1
  - Decrement backoffCount on receipt of ADV message
    - Only send SCAN_REQ if backoffCount == 0
  - Adapt upperLimit based on success or failure of receipt of SCAN_RSP
    - Reset backoffCount
    - backoffCount = rand (1, upperLimit)



backoffCount and upperLimit are reset whenever the scanner is turned on
(i.e., after an idle time)

BLE Highlights: Low-level Scanning

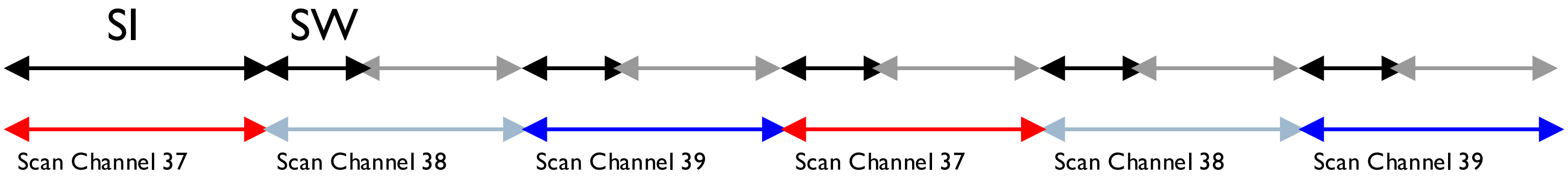
- Scanners
- Scan for tags on sequential channels (37, 38, 39)
- Scan Interval (SI)
  - Time spent on a channel



BLE Highlights: Low-level Scanning

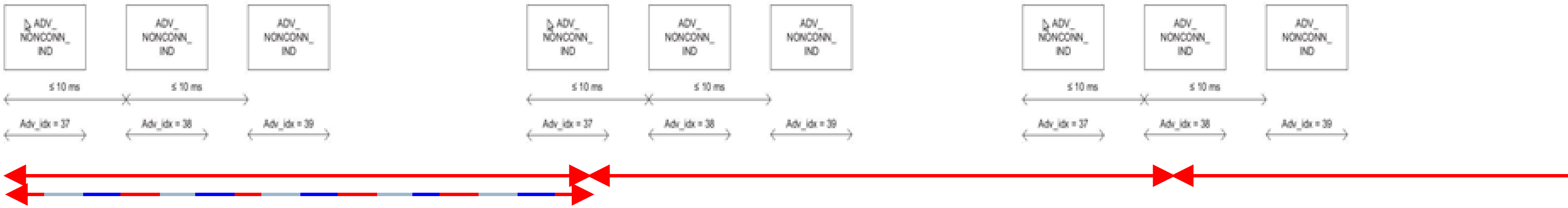
- Scan Time
  - Scan Int == Scan Window ⇒ Always on

- Scanners
- Scan for tags on sequential channels (37, 38, 39)
- Scan Interval (SI)
  - Time spent on a channel
- Scan Window (SW)
  - Time spent scanning at beginning of Scan Interval



BLE Highlights: Application-level Scanning

- Scanners
- Application Scan Time
  - > Tag Advertising Interval

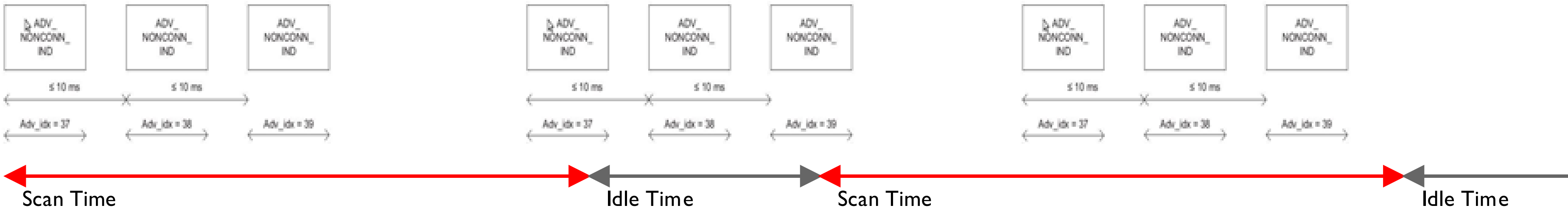


Application Scan Time

BLE Highlights: Application-level Scanning

- Scan Time
  - 100% on Idle Time = 0
- (Continuous scanning)
  - 10% on Idle Time = 10 * Scan Time

- Scanners
- Application Scan Time
  - > Tag Advertising Interval
- Application Idle Time



BLE Highlights: MAC Behavior

- No Carrier Sense
  - Tag does not listen for a clear channel before sending any message
- Minimal Contention Avoidance
  - Jitter length of advertising interval + rand [0, 10 ms]
  - Backoff for sending SCAN_REQ
- Other parameters

Inter-frame spacing	150us (from spec)
Channel switching delay	274us (from Nordic)
Scan Interval	11.25ms (from spec)
Scan Window	11.25ms (continuous scanning)

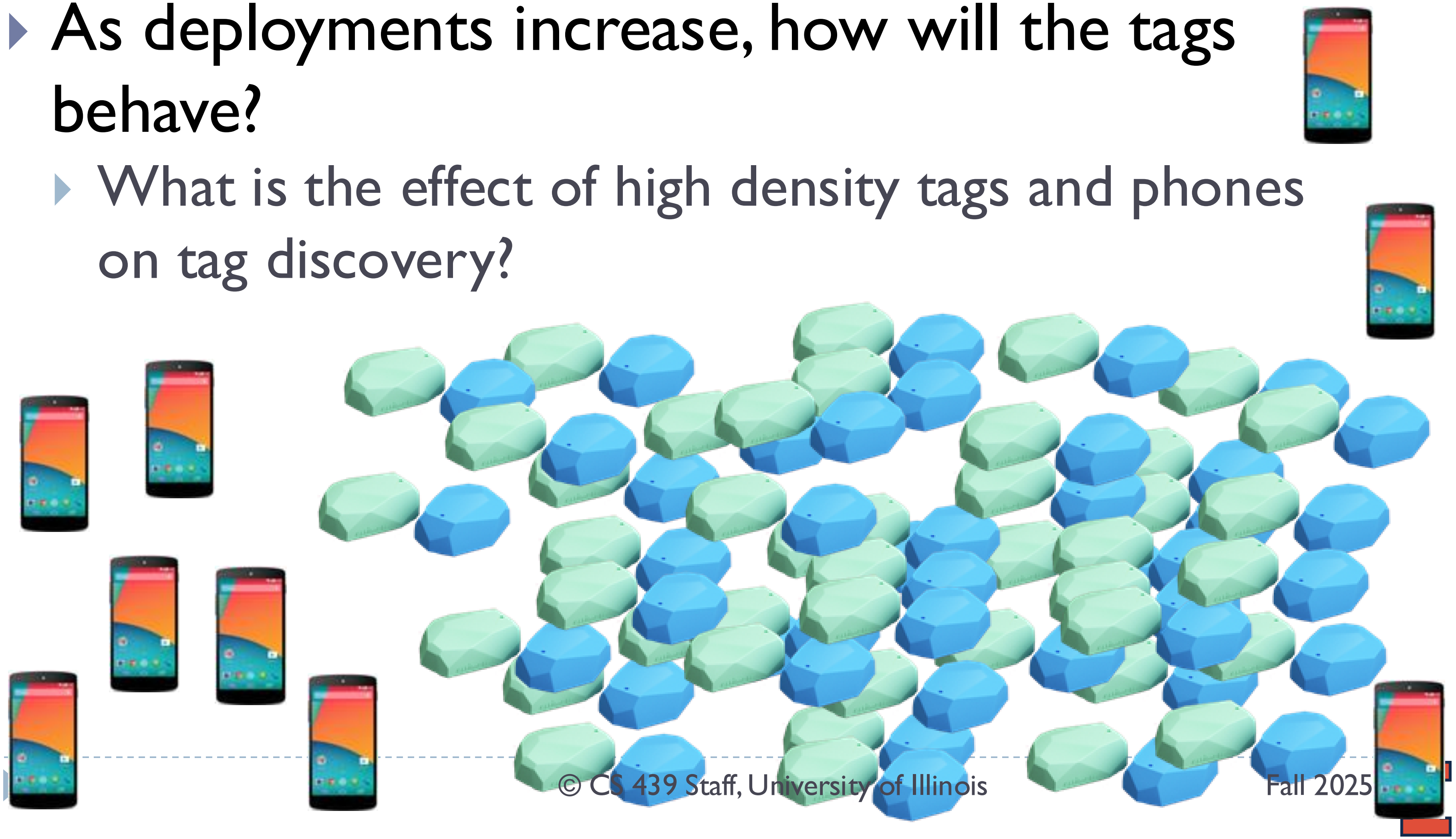
BLE in the Real World

- BLE beacons (or tags)
  - Location-specific information
  - Deployed in public places
    - Stores, airports, museums
  - Accessed via phones with BLE

- Performance questions
  - How long does it take to detect a nearby tag?
  - Can we detect a tag within 5 sec with 95% success?

BLE in the Real World - Density

- As deployments increase, how will the tags behave?
  - What is the effect of high density tags and phones on tag discovery?



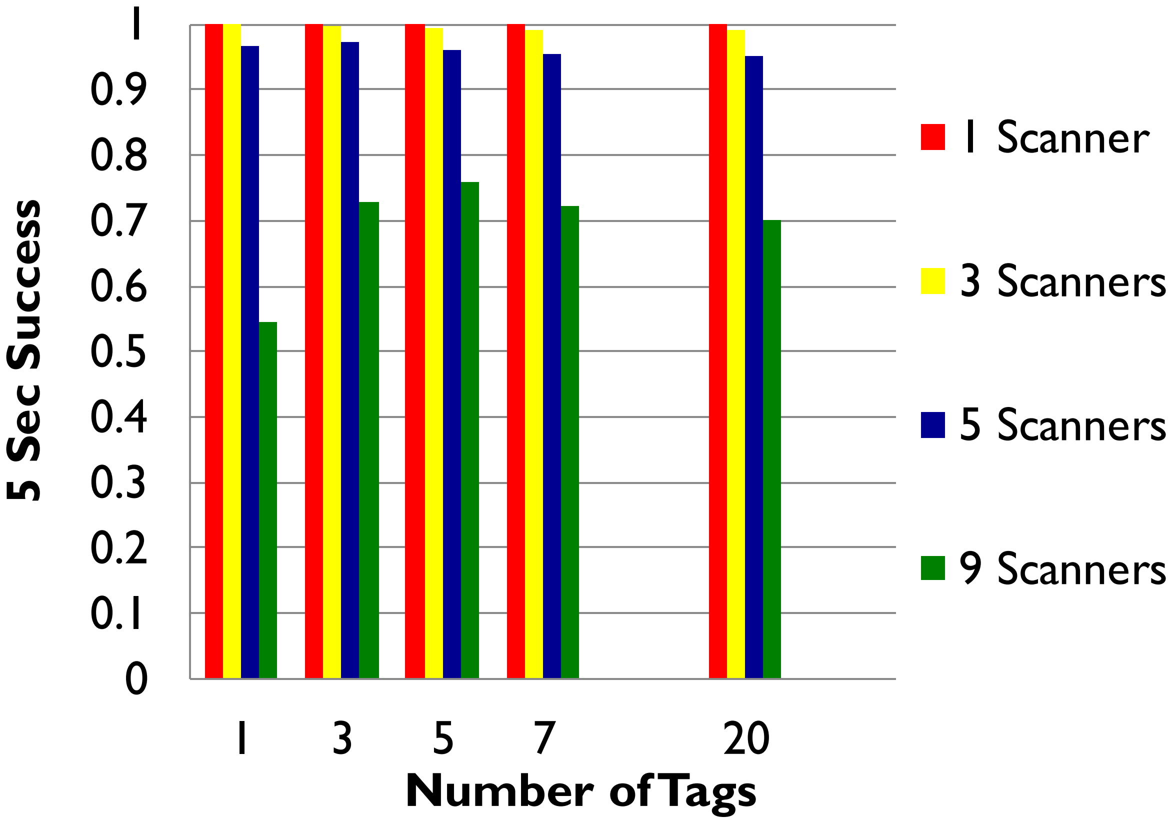
Evaluating Tag Behavior

- Environmental Impact
  - At what density of tags or phones does the system break down?

- Metric
  - 5 Sec Success
    - Could the tag be found in 5 sec?
    - Checked every 1 sec over the whole run

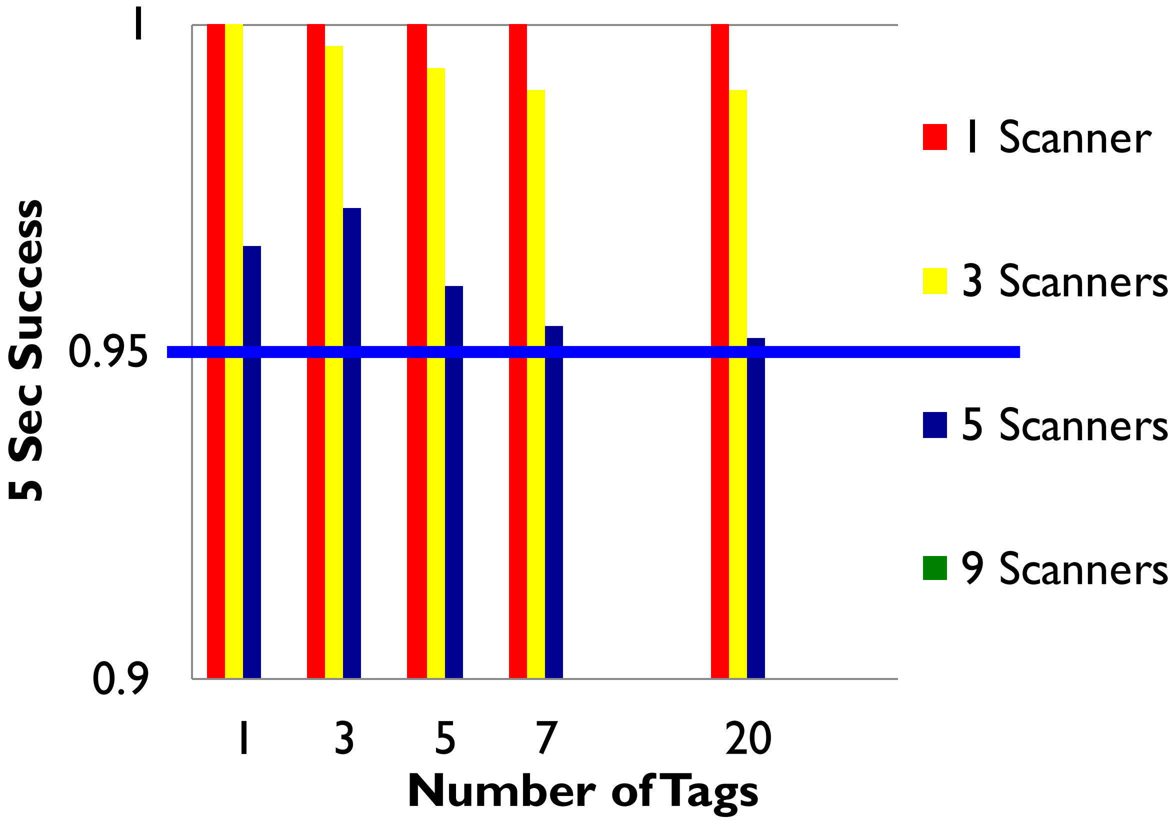
Evaluation: BLE Scan/Response

- 5 second success
  - Multiple chances to find the tag
  - Success decreases significantly as more phones are added
  - Number of phones is more important than number of tags



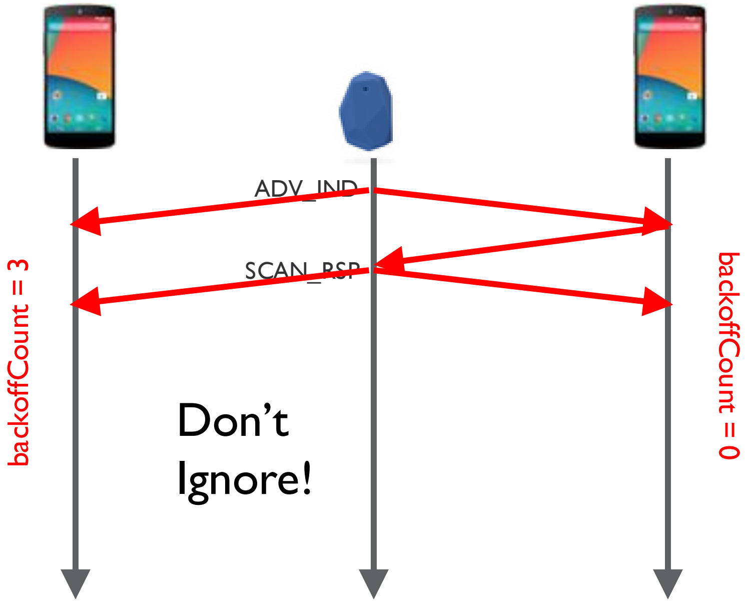
Evaluation: BLE Scan/Response

- 5 second success
  - Below target threshold for more than 5 phones



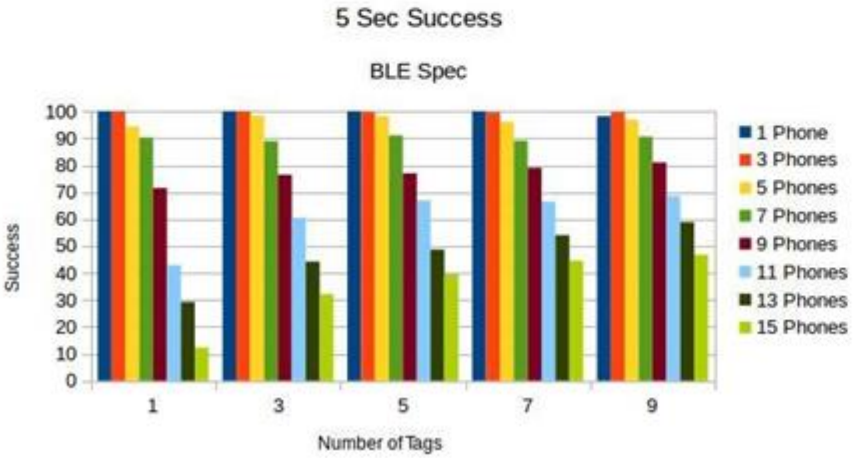
SCAN_REQ: Opportunistic Listening

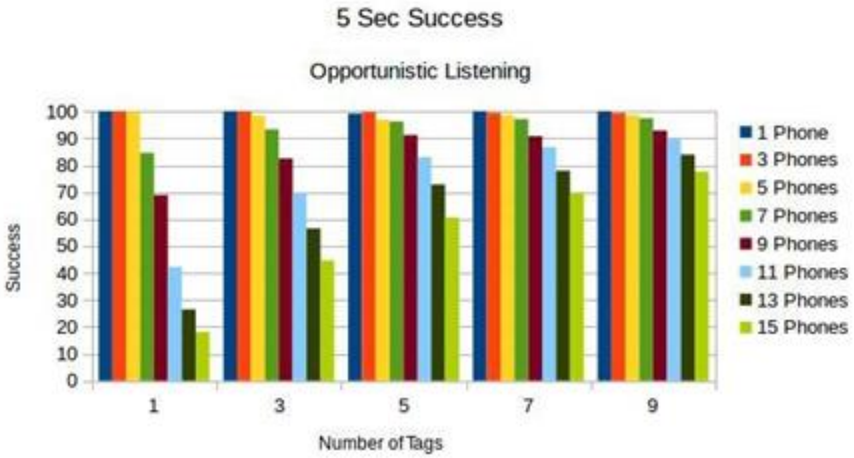
- Accept a SCAN_RSP on a channel if a SCAN_REQ would have been sent, but the backoff procedure indicated not to send it
  - Any requesting or backing off scanner can receive a SCAN_RSP as long as one SCAN_REQ is received and the tag responds
  - Still, No SCAN_RSP if all SCAN_REQs collide



Opportunistic Listening: Simulation Comparison

- Significant increase in success rate as number of phones increases
- Cannot prevent SCAN_REQ collisions

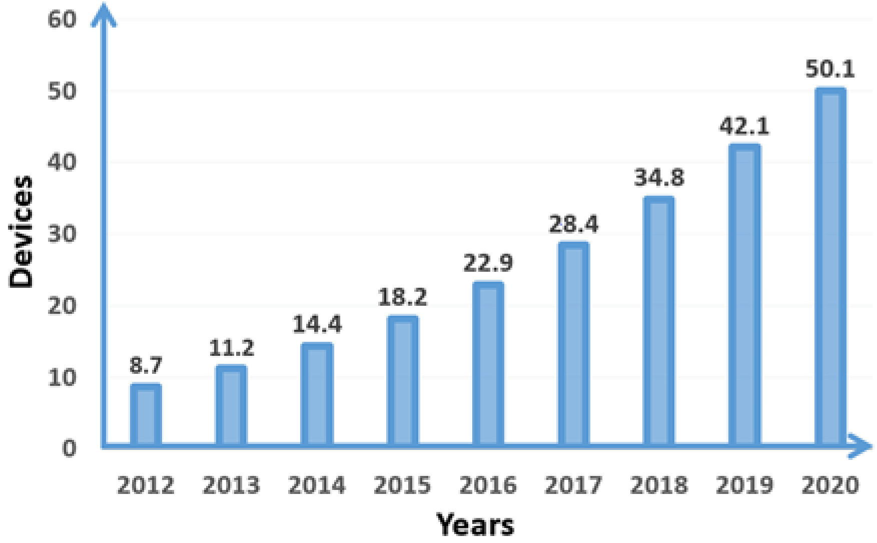




Bluetooth 5.0 Why An Upgrade Was Needed

- 4.0 is too slow
- Low range (especially indoors)
- Power issues
- Issues relating to multiple radios on the same device

*Devices in billions



Bluetooth Progression

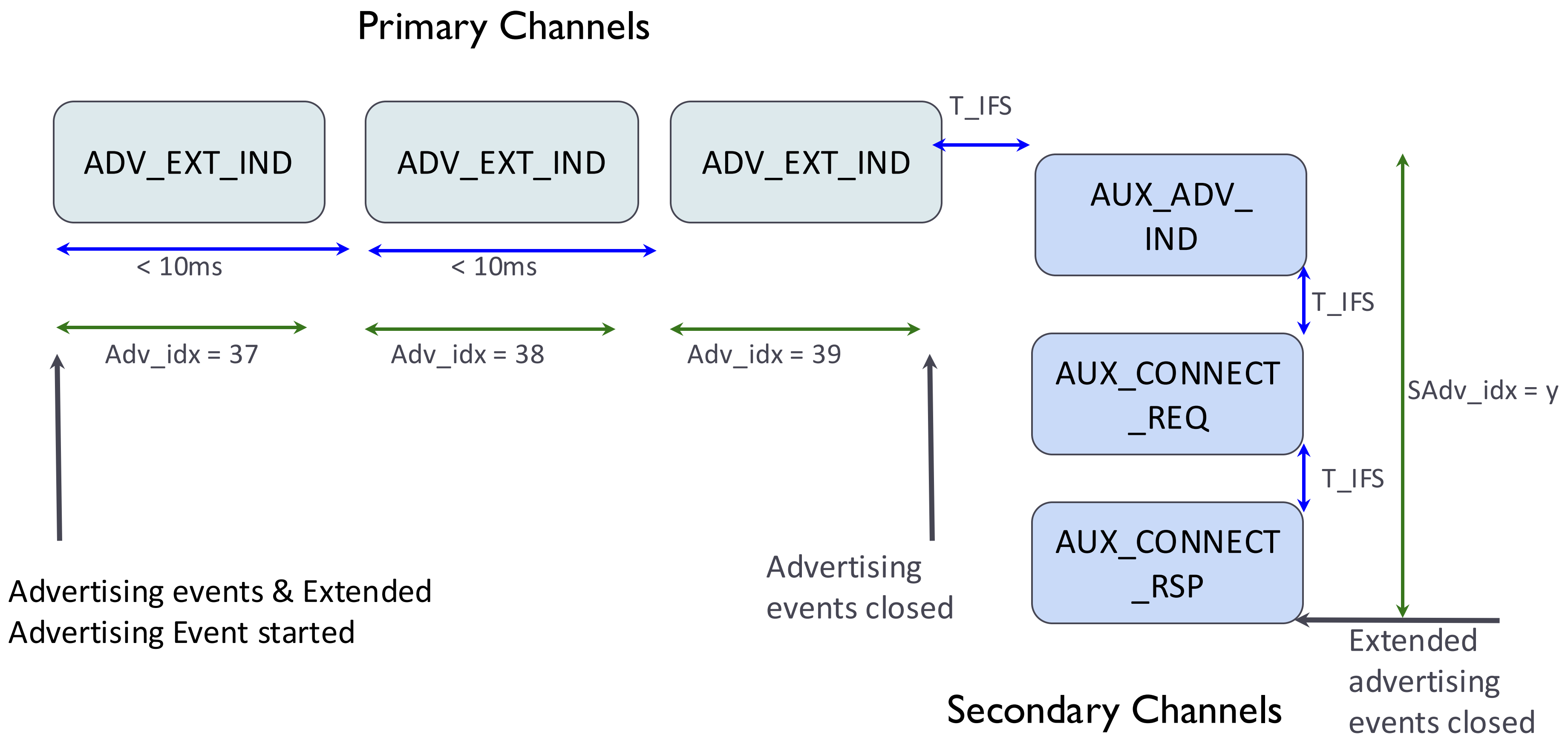
Bluetooth 1.0	Bluetooth 2.0	Bluetooth 3.0	Bluetooth 4.0	Bluetooth 5.0
Initial version	Significantly Increased Speed	High-Speed Bluetooth	Bluetooth Low Energy	IoT Bluetooth

4.0 is too slow
Low range (especially indoors)
Power issues
Issues relating to multiple radios on the same device

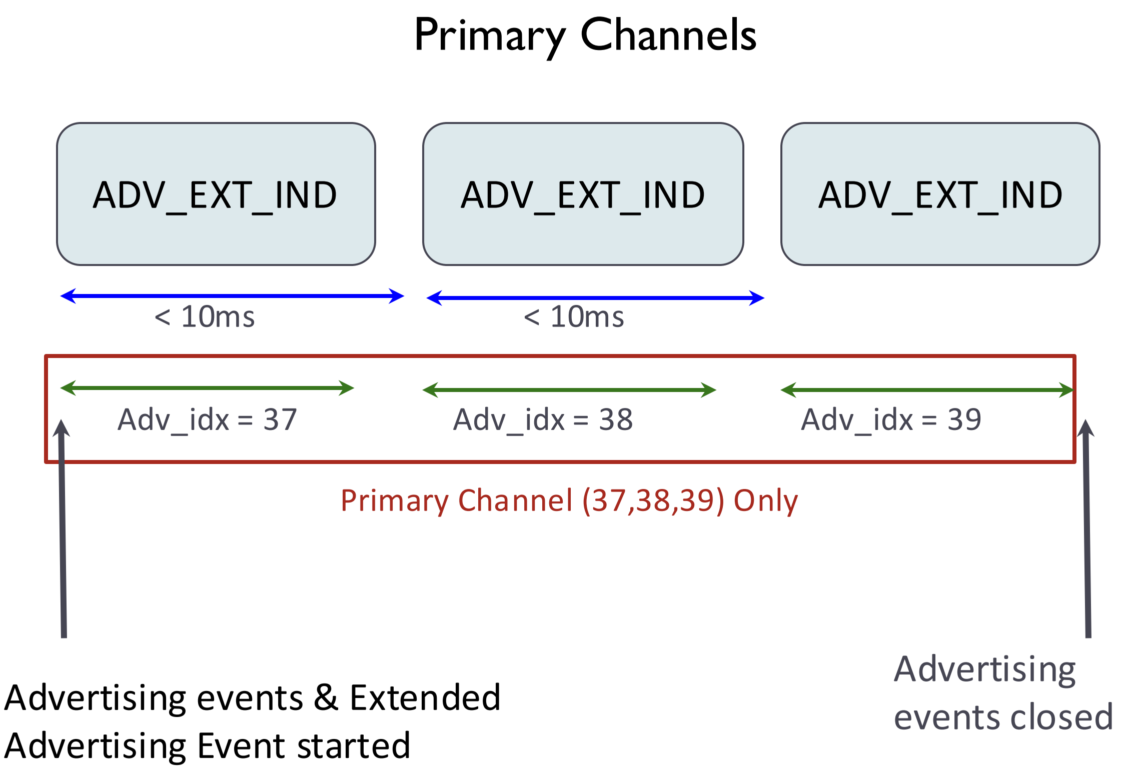
Bluetooth 5.0 Improvements

BLE 4.0	BLE 5.0
Advertising Congestion/Interference	Use of Secondary channels Increase payload size -> less transmission

Bluetooth 5.0: Extended Advertising

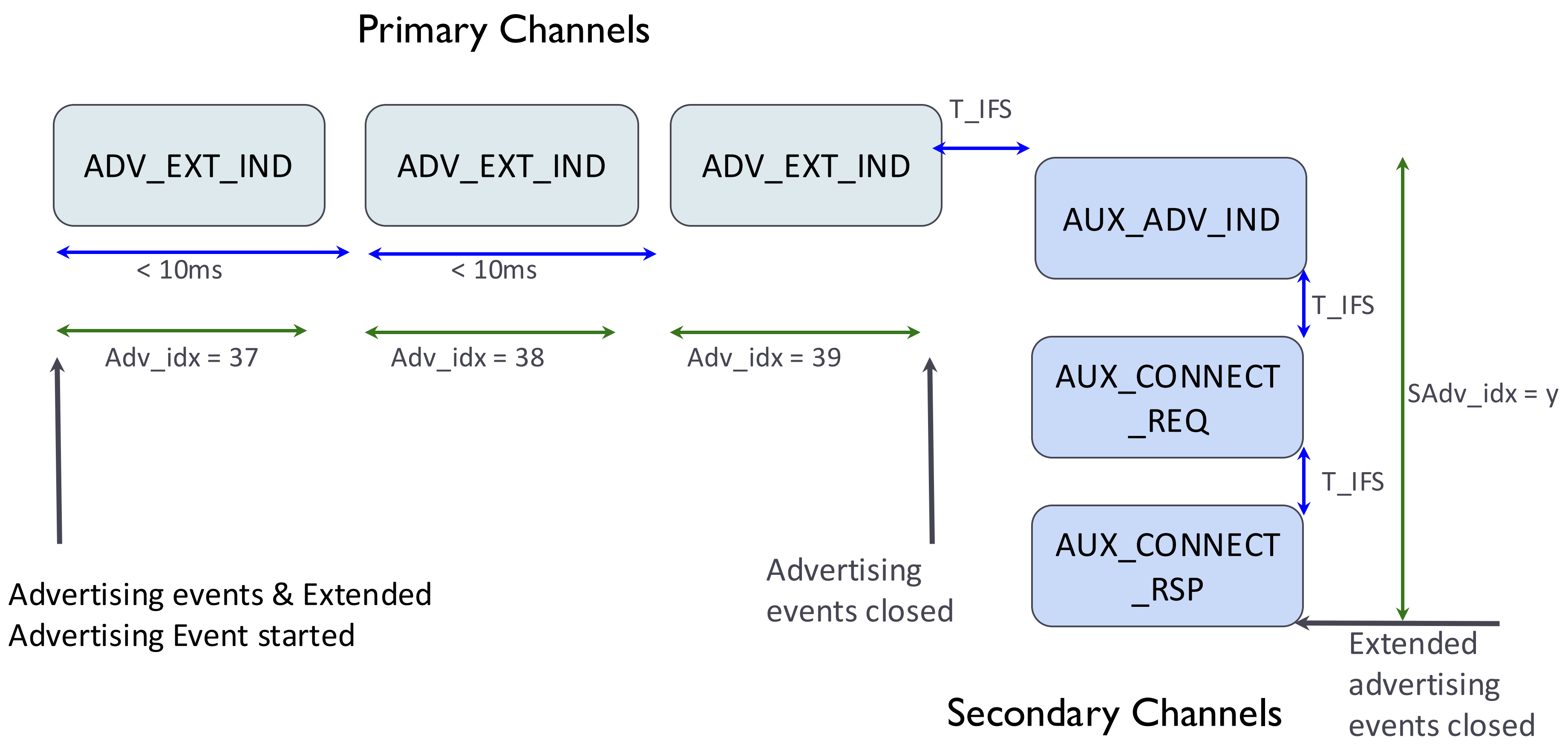


Bluetooth 5.0: Extended Advertising


Primary Channels
ADV_EXT_IND
ADV_EXT_IND
ADV_EXT_IND
< 10ms
< 10ms
Adv_idx = 37
Adv_idx = 38
Adv_idx = 39
Primary Channel (37,38,39) Only
Advertising events & Extended Advertising Event started
Advertising events closed

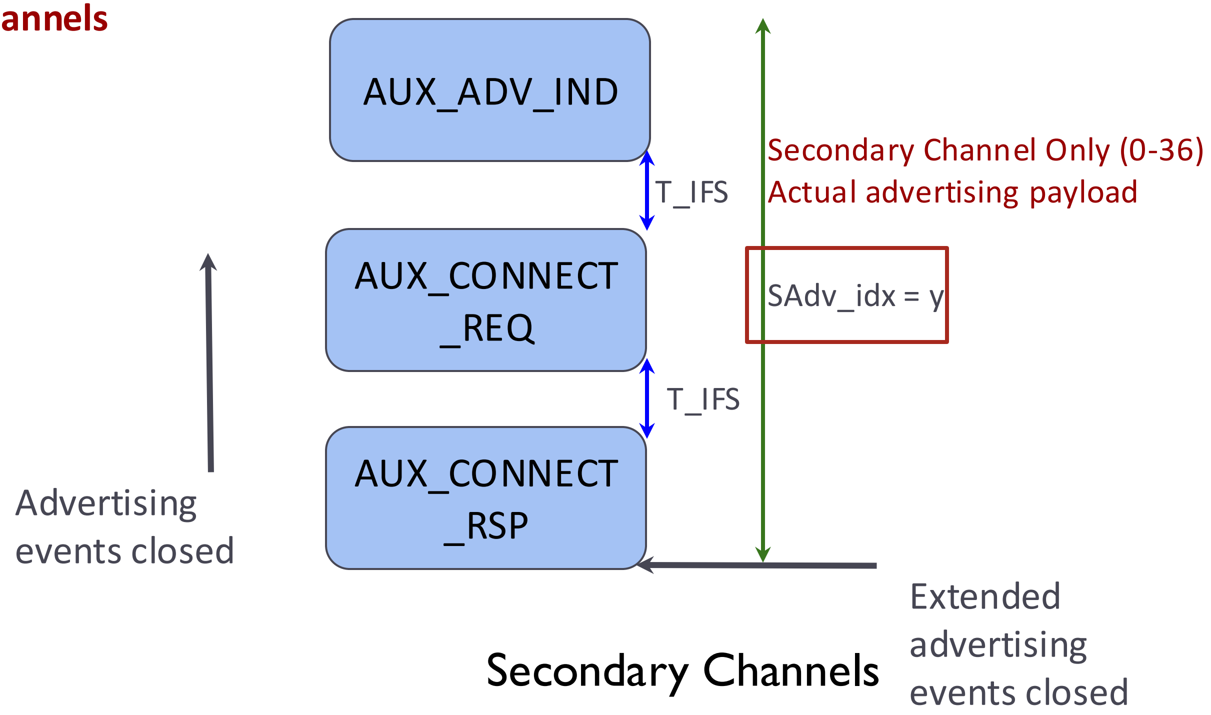
ADV_EXT_IND (extended advertising indicator):
A pointer with
1. Channel indices
2. Timing info
3. PHY layer setting

Bluetooth 5.0: Extended Advertising

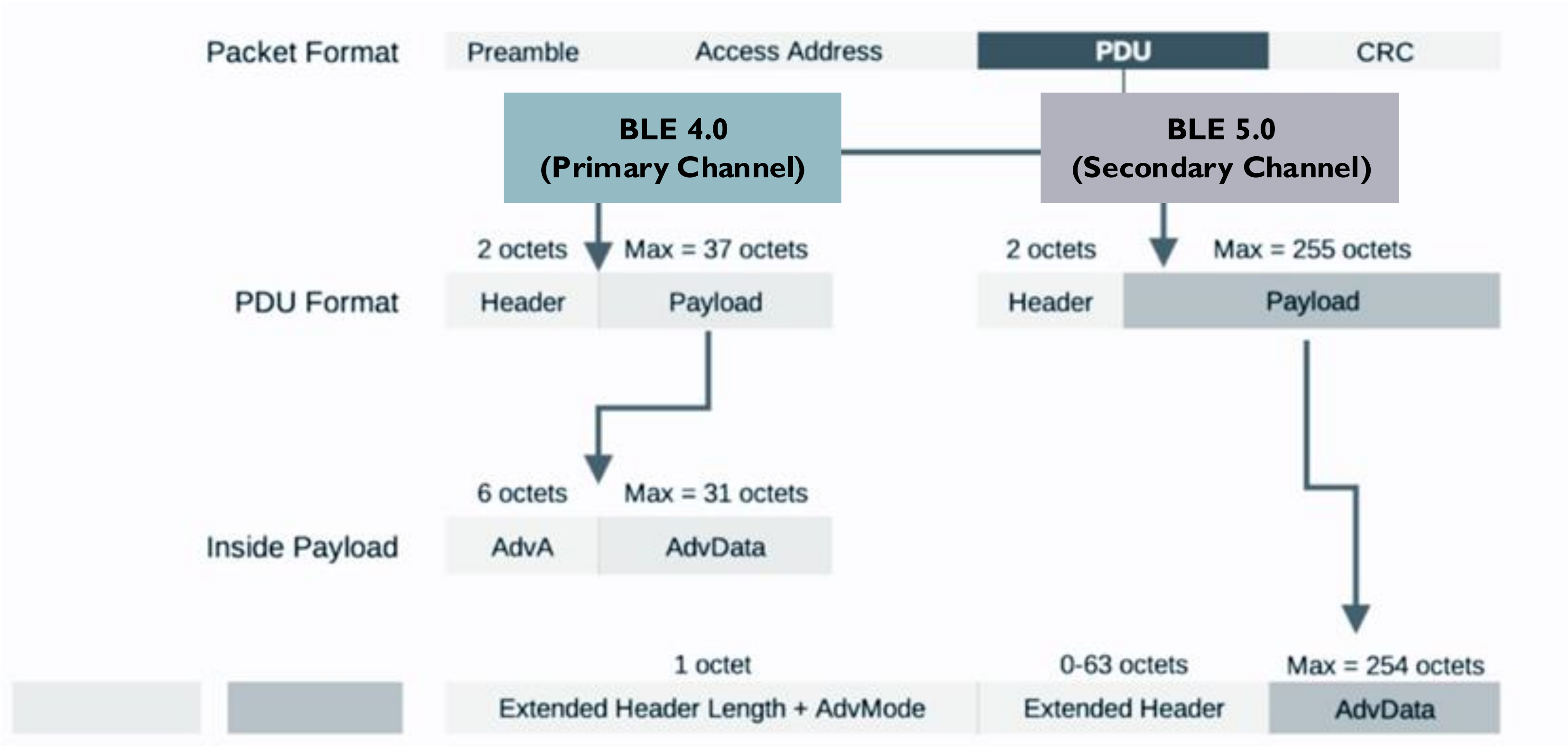


Bluetooth 5.0: Extended Advertising

No data transmission on primary channels
During extended advertising

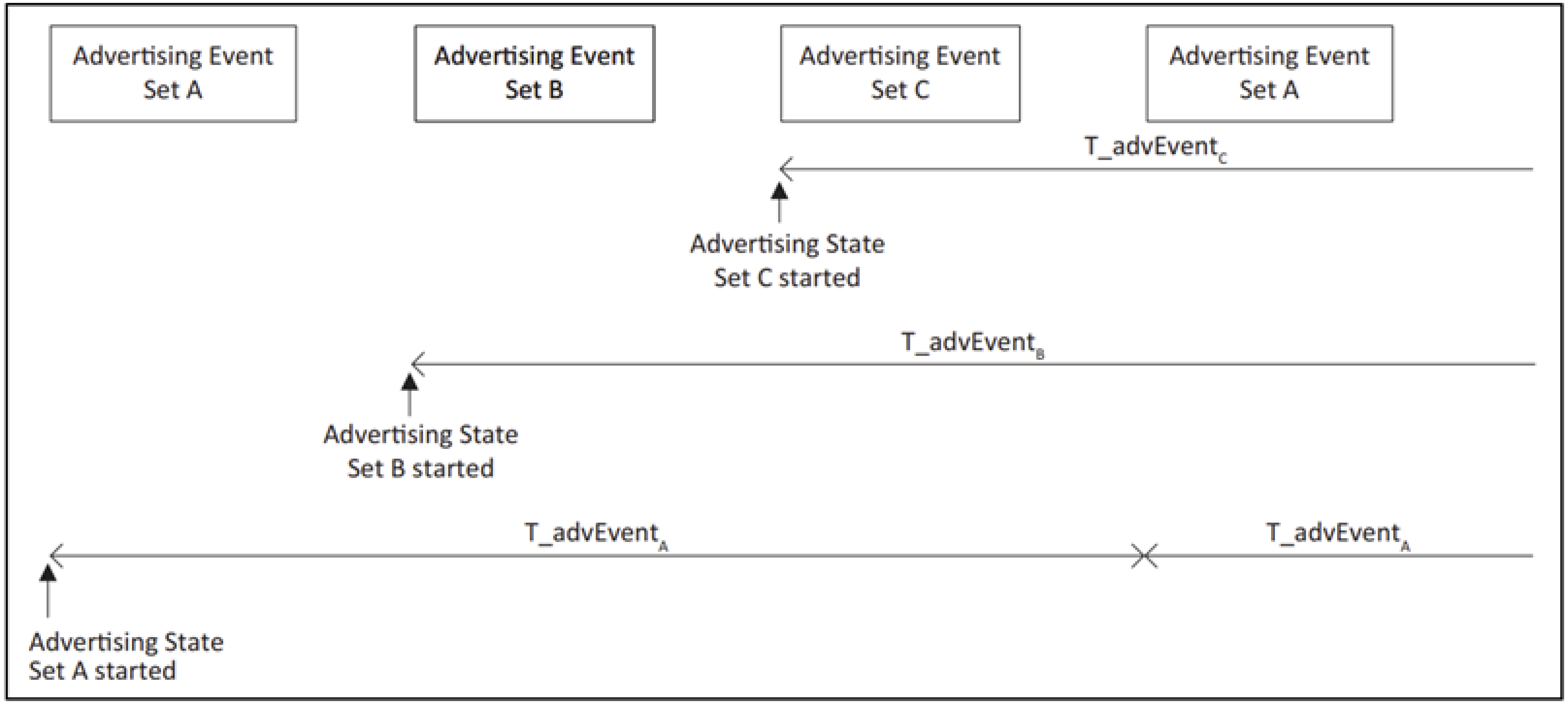

AUX_ADV_IND
T_IFS
AUX_CONNECT_REQ
T_IFS
AUX_CONNECT_RSP
Secondary Channel Only (0-36)
Actual advertising payload
SAdv_idx = y
Advertising events closed
Extended advertising events closed
Secondary Channels

Bluetooth 5.0: Payload Increase

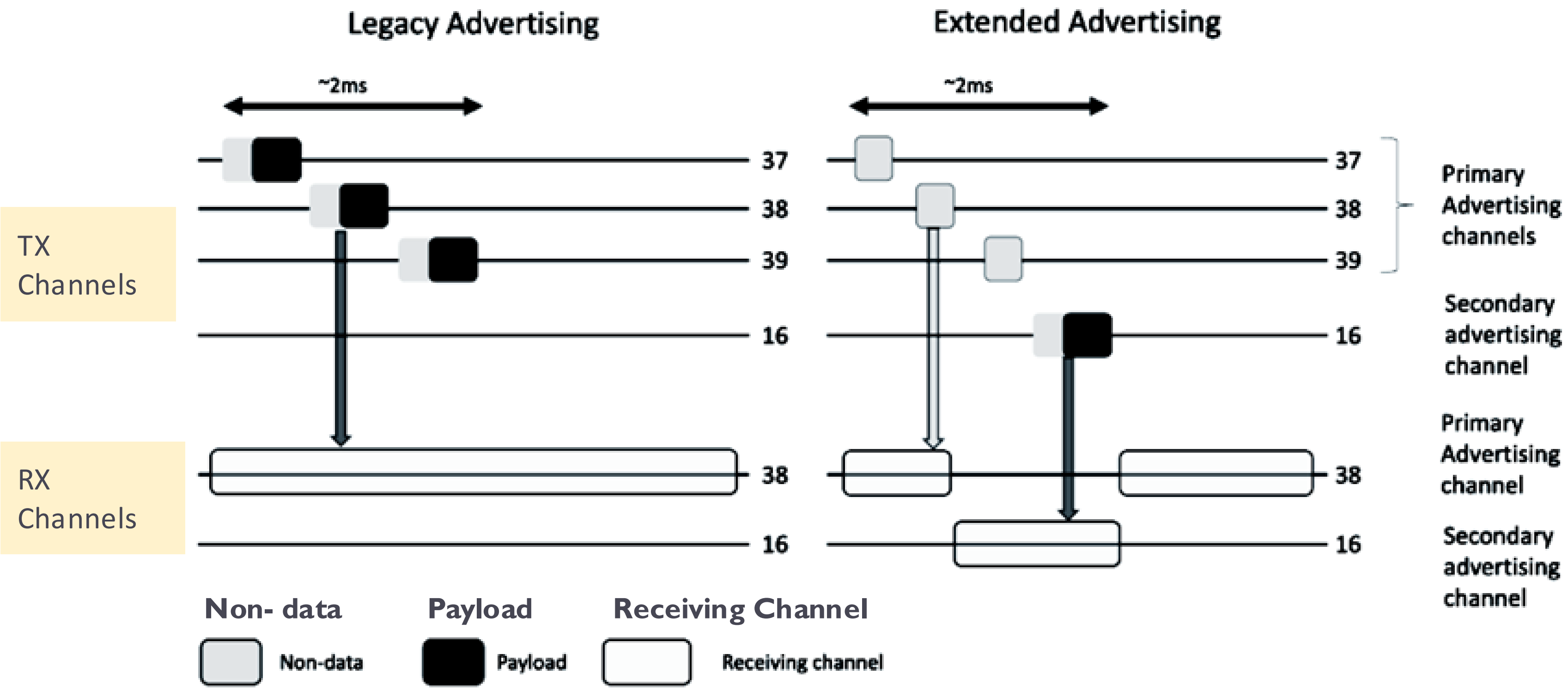


Bluetooth 5.0: Multiple Advertising Sets

- Multiple, independent advertising sets simultaneously
- Enhances flexibility and efficiency of advertising



Bluetooth 5.0: Congestion Management



TX Channels

RX Channels

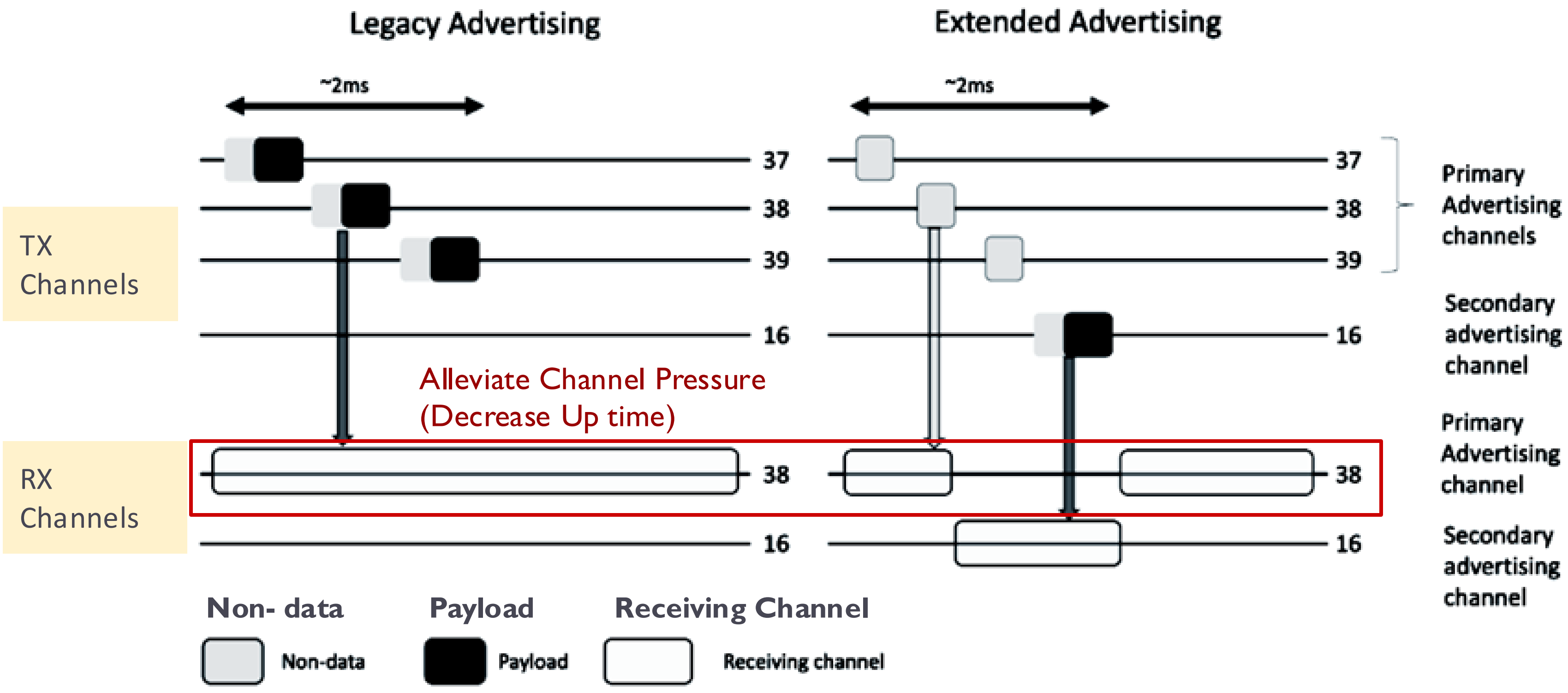
Non- data

Payload

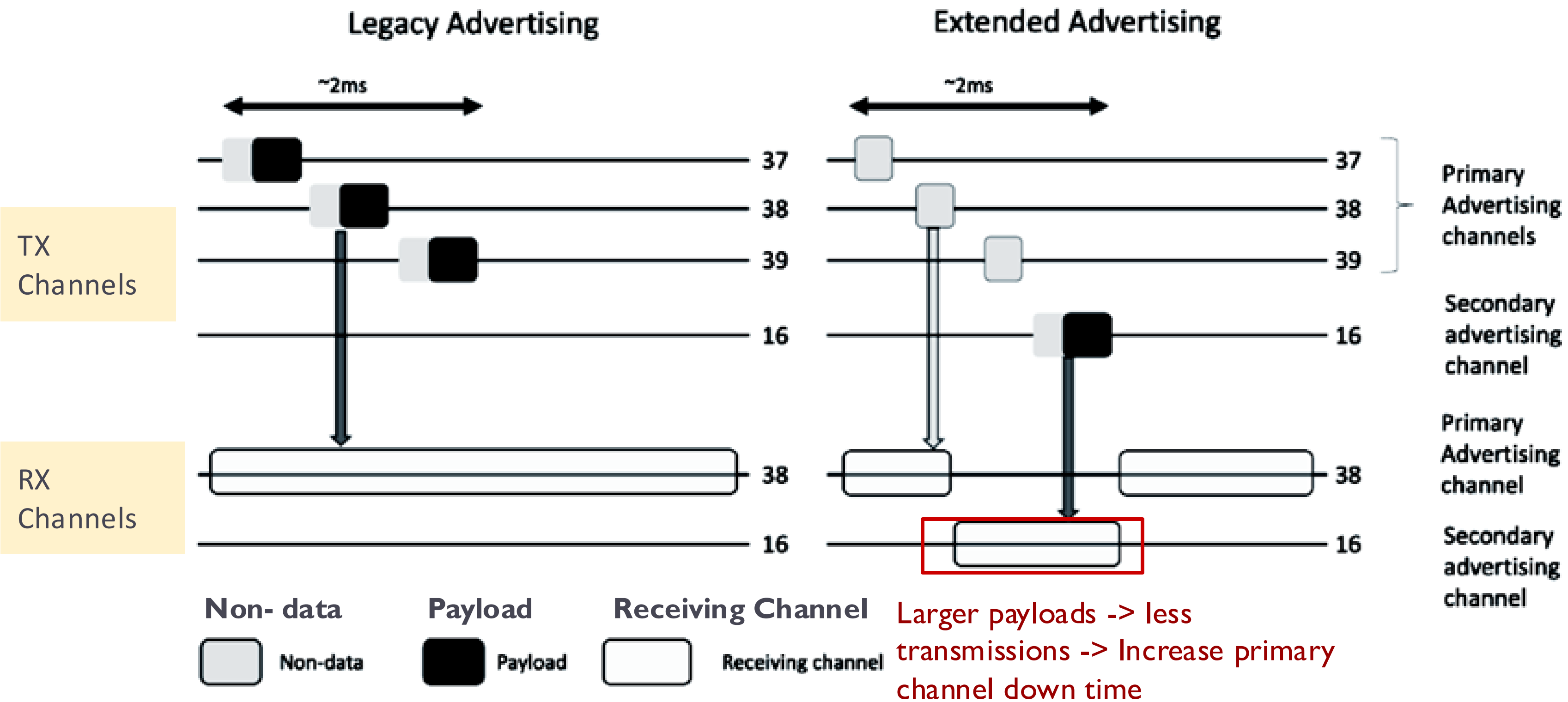
Receiving Channel

Legacy Advertising
~2ms
37
38
39
16
38
16
Extended Advertising
~2ms
37
38
39
16
38
16
Primary Advertising channels
Secondary advertising channel
Primary Advertising channel
Secondary advertising channel
Non-data
Payload
Receiving channel

Bluetooth 5.0: Congestion Management


Legacy Advertising
Extended Advertising
~2ms
~2ms
37
38
39
16
37
38
39
16
TX Channels
RX Channels
38
16
38
16
Primary Advertising channels
Secondary advertising channel
Primary Advertising channel
Secondary advertising channel
Alleviate Channel Pressure (Decrease Up time)
Non- data
Payload
Receiving Channel
Non-data
Payload
Receiving channel

Bluetooth 5.0: Congestion Management



Larger payloads -> less transmissions -> Increase primary channel down time

Bluetooth 5.0 Improvements

BLE 4.0	BLE 5.0
Advertising Congestion/Interference	Use of Secondary channels Increase payload size -> less transmission
Insufficient for High Data Rate Applications	Increased max transfer speed (1Mbps -> 2Mbps)
Inadequate for Long Range Applications	Coded physical layer (up to 400m – 1km) Robust algorithm to strengthen signal

Bluetooth 5.0: Coded PHY

Parameter	LE 1M	LE Coded S2	LE Coded S8	LE 2M
Symbol Rate	1 Msps	1 Msps	1 Msps	2 Msps
Data Rate	1 Mbps	500 kbps	125 kbps	2 Mbps
Error Correction	None	FEC	FEC	None
Range Multiplier	1	~ 2	~ 4	~ 0.8

- Symbols per sec
  - S2: 2 symbols = 1 bit
  - S8: 8 symbols = 1 bit

Bluetooth 5.0: Coded PHY

Parameter	LE 1M	LE Coded S2	LE Coded S8	LE 2M
Symbol Rate	1 Msps	1 Msps	1 Msps	2 Msps
Data Rate	1 Mbps	500 kbps	125 kbps	2 Mbps
Error Correction	None	FEC	FEC	None
Range Multiplier	1	~ 2	~ 4	~ 0.8

Baseline for ble 4.0

Bluetooth 5.0: Coded PHY

Parameter	LE 1M	LE Coded S2	LE Coded S8	LE 2M
Symbol Rate	1 Msps	1 Msps	1 Msps	2 Msps
Data Rate	1 Mbps	500 kbps	125 kbps	2 Mbps
Error Correction	None	FEC	FEC	None
Range Multiplier	1	~ 2	~ 4	~ 0.8

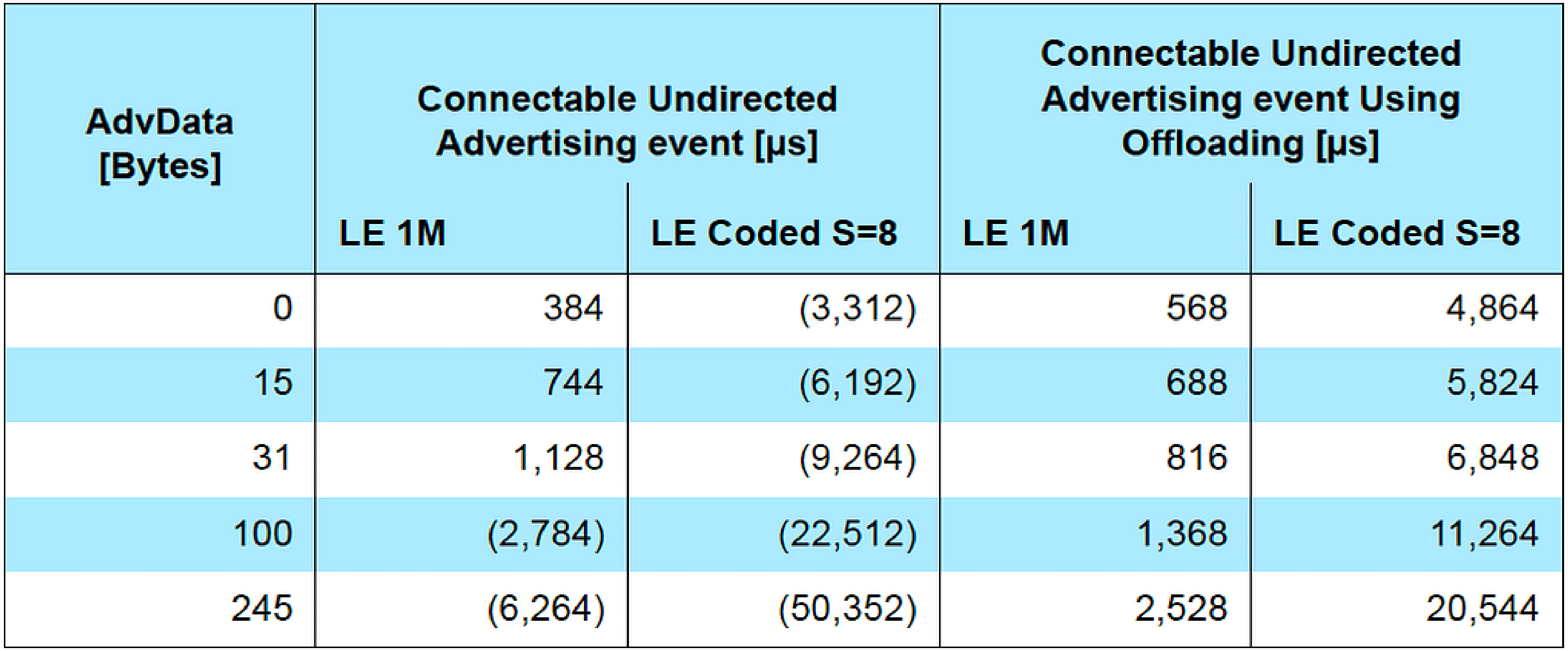
2Mbps max data rate for BLE 5.0 (High data rate application, reduced range)

Bluetooth 5.0: Coded PHY

Parameter	LE 1M	LE Coded S2	LE Coded S8	LE 2M
Symbol Rate	1 Msps	1 Msps	1 Msps	2 Msps
Data Rate	1 Mbps	500 kbps	125 kbps	2 Mbps
Error Correction	None	FEC	FEC	None
Range Multiplier	1	~ 2	~ 4	~ 0.8

Quadruple distance for coded PHY (Long range application, reduced data rate)

Bluetooth 5.0: Coded PHY



AdvData [Bytes]	Connectable Undirected Advertising event [µs]		Connectable Undirected Advertising event Using Offloading [µs]	
	LE 1M	LE Coded S=8	LE 1M	LE Coded S=8
0	384	(3,312)	568	4,864
15	744	(6,192)	688	5,824
31	1,128	(9,264)	816	6,848
100	(2,784)	(22,512)	1,368	11,264
245	(6,264)	(50,352)	2,528	20,544

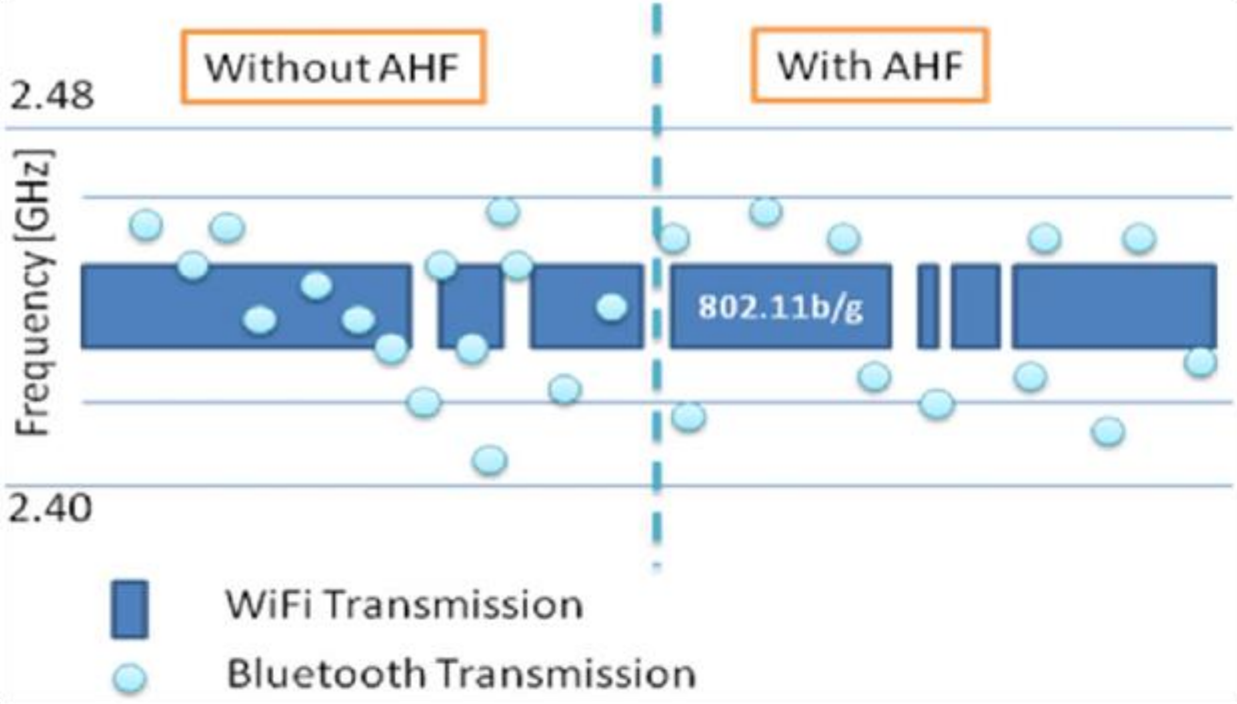
radio on-time in microsec

Bluetooth 5.0 Improvements

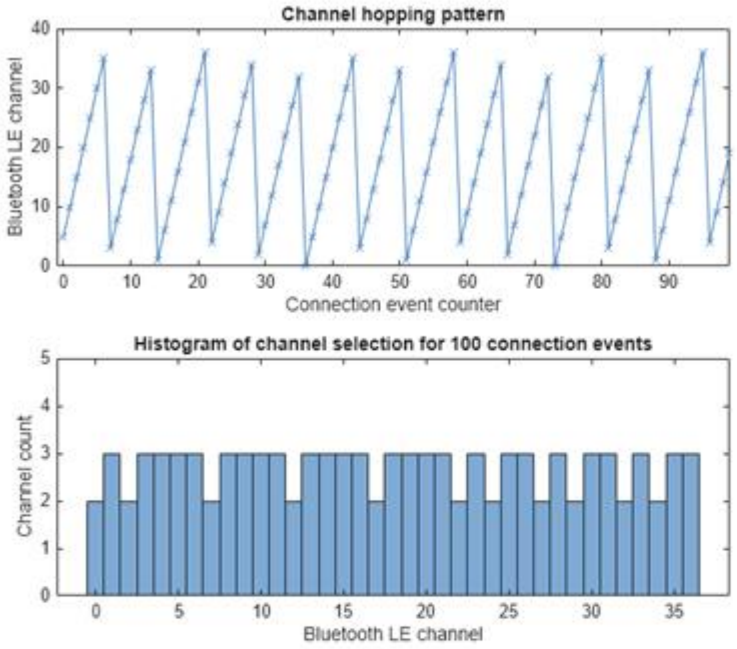
BLE 4.0	BLE 5.0
Advertising Congestion/Interference	Use of Secondary channels Increase payload size -> less transmission
Insufficient for High Data Rate Applications	Increased max transfer speed (1Mbps -> 2Mbps)
Inadequate for Long Range Applications	Coded physical layer (up to 400m – 1km) Robust algorithm to strengthen signal
Limited Advertising Capabilities / Power efficiency	Dynamic advertising sets Improved Channel Selection Algorithm Precise Timing Controls

Bluetooth 5.0: Adaptive Frequency Hopping

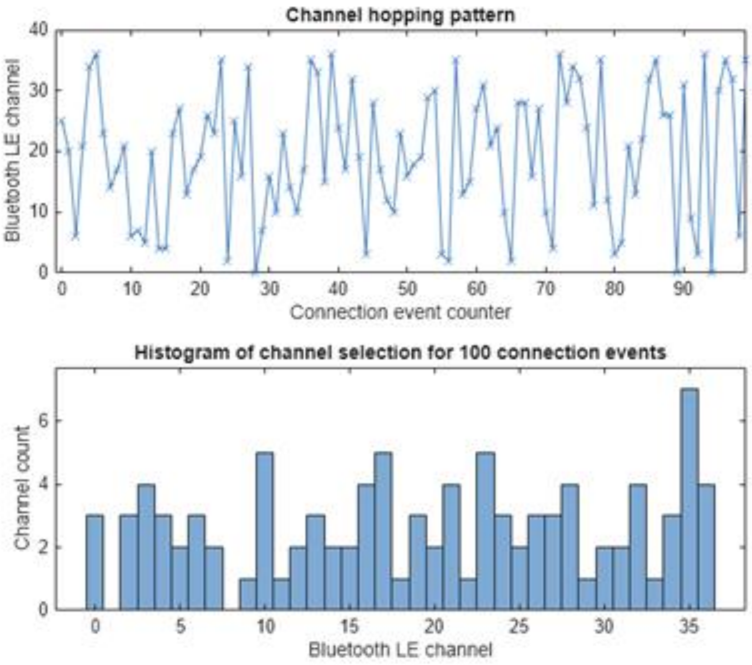
- Adaptive Frequency Hopping
  - Channels (0-36) set to used or unused, algorithmically determine sequence
  - Channel Selection Algorithm (CSA) #1: 12 distinct sets
  - CSA #2 allows for many distinct sequences, reduces collisions



Bluetooth 5.0: Adaptive Frequency Hopping



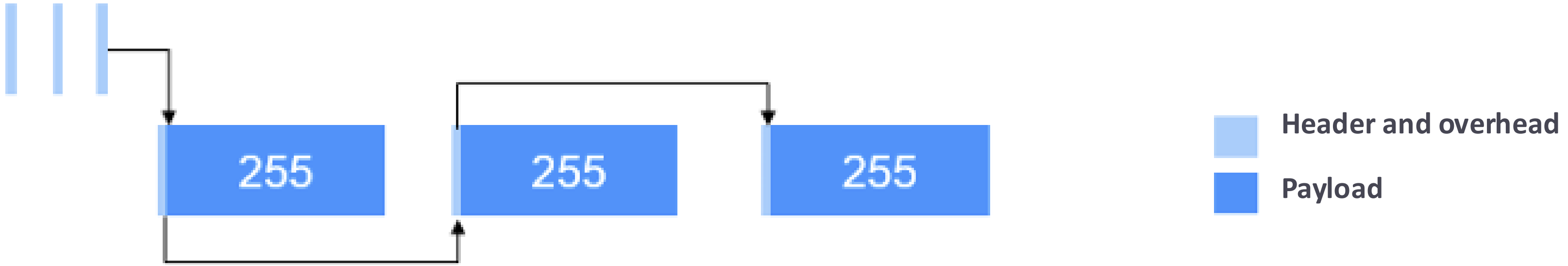
CSA 1 (no randomization)



CSA 2 (pseudo randomization)

Bluetooth 5.0: Packet Chaining

- Controller can chain packets together, using AuxPtr header fields (references to Auxiliary Packets containing payload)
- AuxPtr includes the channel number 0-36, receiver can find it
- Up to 1,650 bytes
- Improves efficiency, data transfer rate, power, etc.



Bluetooth 5.0: Other New Features

- Dual Audio
  - A single source device can stream audio to two different connected audio devices simultaneously
- Mesh Networking
  - Devices can relay packets