

IMPLEMENTING LIBP2P OVER BLUETOOTH LOW ENERGY



Antoine Eddi
@aeddi





Libp2p transport over Bluetooth Low Energy



Quick overview:
How does BLE work?



Specs comparison

Bluetooth Classic

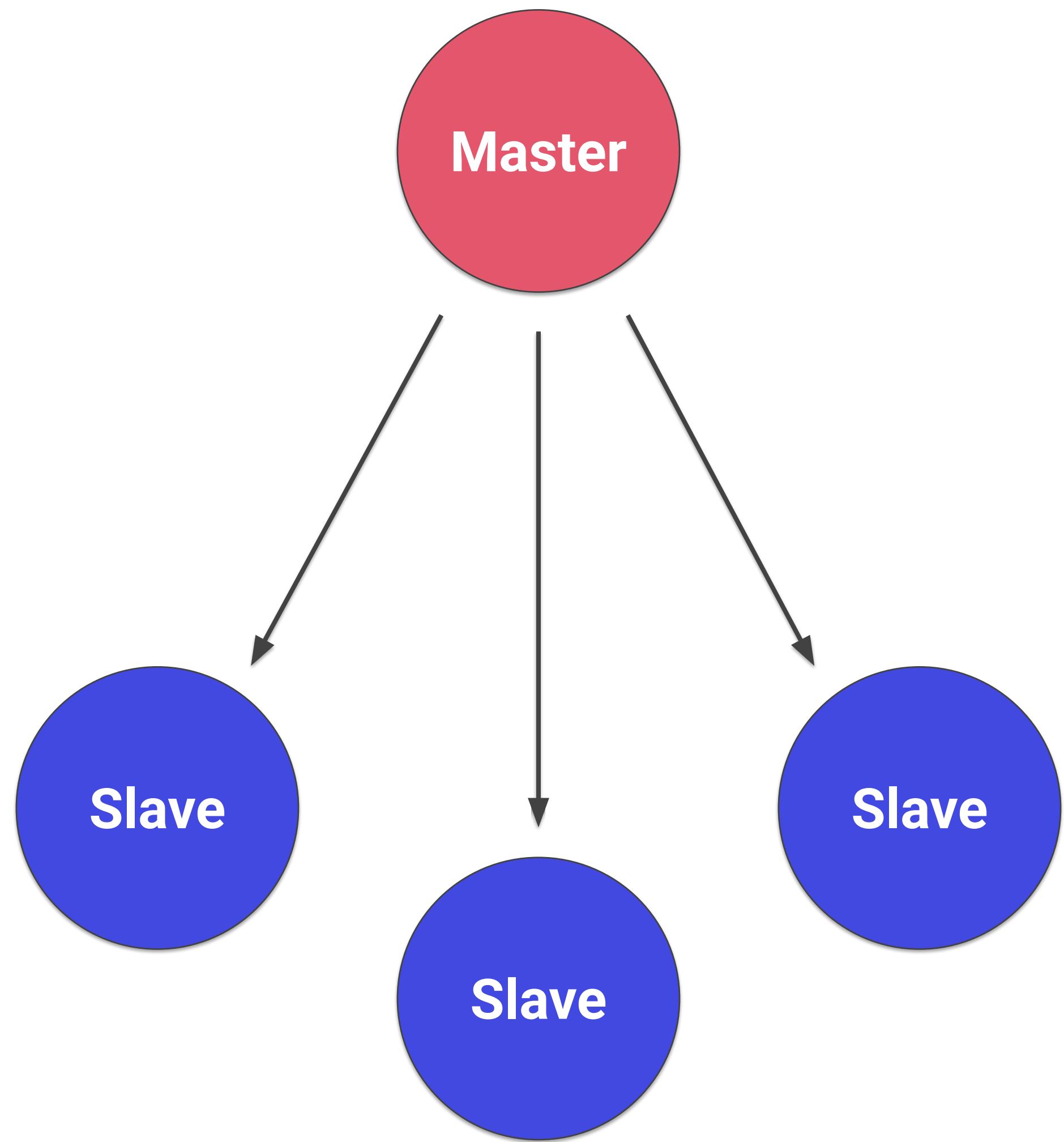
Bluetooth Low Energy

	Bluetooth Classic	Bluetooth Low Energy
Power consumption	1 as reference	0.01 to 0.5 (use case dependent)
Application Troughput	0.7 to 2.1 Mbit/s	0.3 Mbit/s (v4) / 1.36 Mbit/s (v5)
Device pairing	Mandatory	Optional
Max connections	7	Unlimited (implem. dependent)
Role simultaneity	No (master OR slave)	Yes (master AND slave)

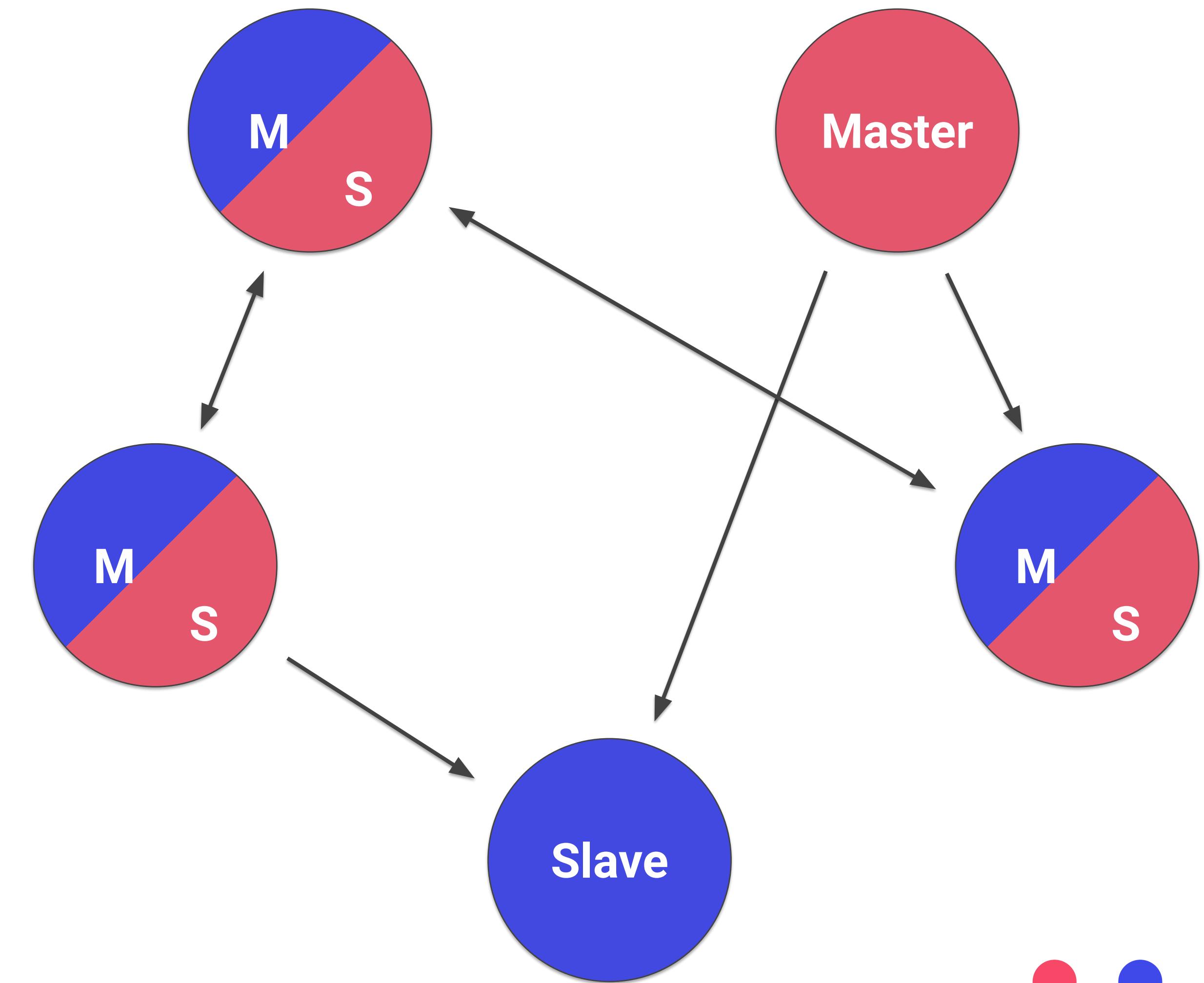


Connection model

Bluetooth Classic



Bluetooth Low Energy



GATT

Generic Attribute Profile

Provides data model and hierarchy based on three types of attributes:

- Service
- Characteristic
- Descriptor (optional)

Service thermostat

Characteristic current-temp

- Type: hex
- Value: 0x24
- Perm: Read

Descriptor temp-unit

- Type: string
- Value: celsius

Characteristic desired-temp

- Type: hex
- Value: 0x42
- Perm: Read/Write

Descriptor temp-unit

- Type: string
- Value: celsius



Transport implementation

Using go-libp2p and Darwin / Android API



Stack

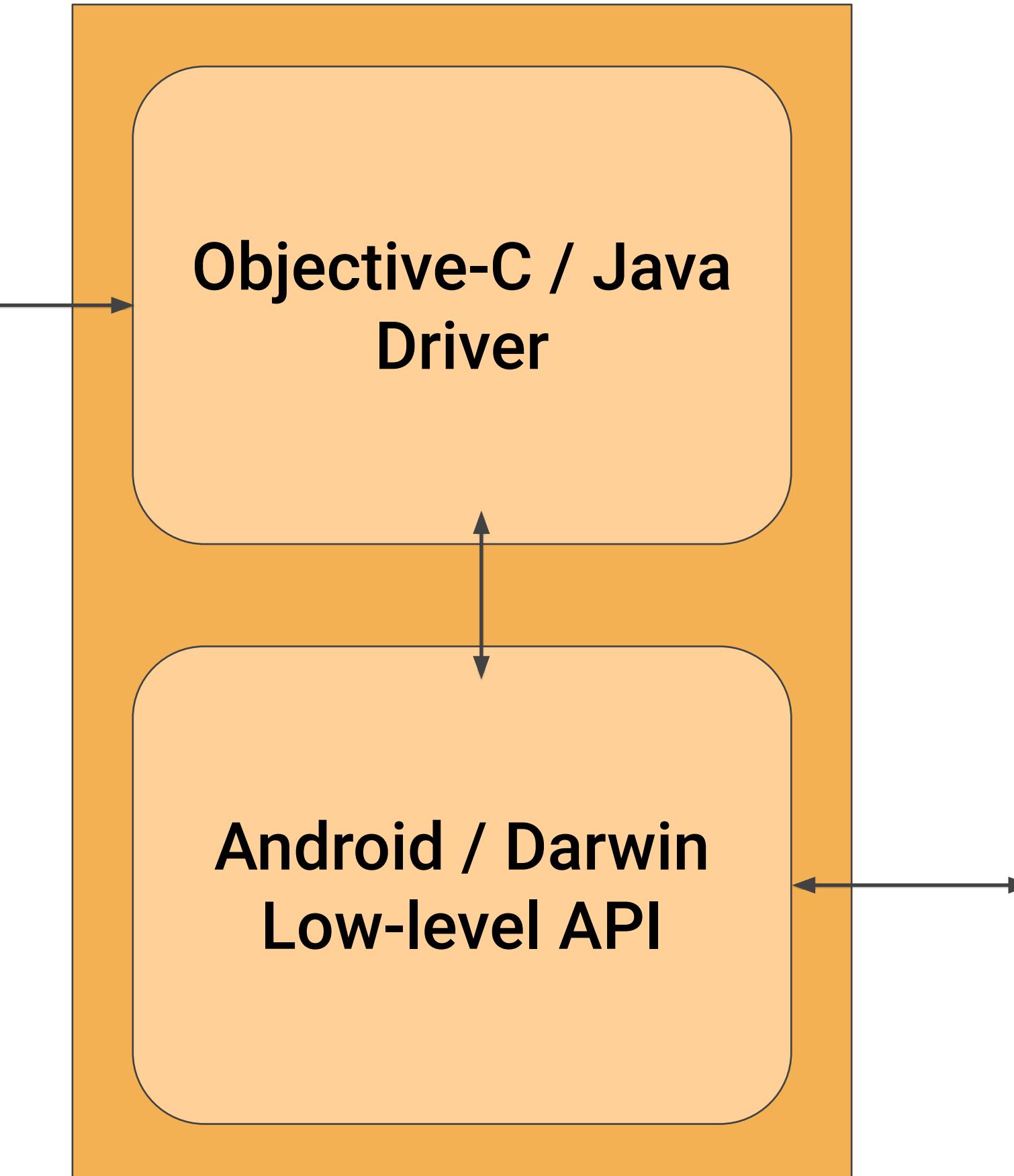
Go-libp2p transport

```
type Transport interface {
    Dial(
        context.Context, ma.Multiaddr, peer.ID
    ) (CapableConn, error)
    CanDial(ma.Multiaddr) bool
    Listen(ma.Multiaddr) (Listener, error)
    Protocols() []int
    Proxy() bool
}

type Listener interface {
    Accept() (CapableConn, error)
    Close() error
    Addr() net.Addr
    Multiaddr() ma.Multiaddr
}

type CapableConn interface {
    mux.MuxedConn
    network.ConnSecurity
    network.ConnMultiaddrs
    Transport() Transport
}
```

Native drivers



GATT profile

Service Berty

Characteristic MultiAddr

- Value: /ble/518f1bed-dd5a...
- Perm: Read

Characteristic PeerID

- Value: QmaJYi21cafzJpPW...
- Perm: Read

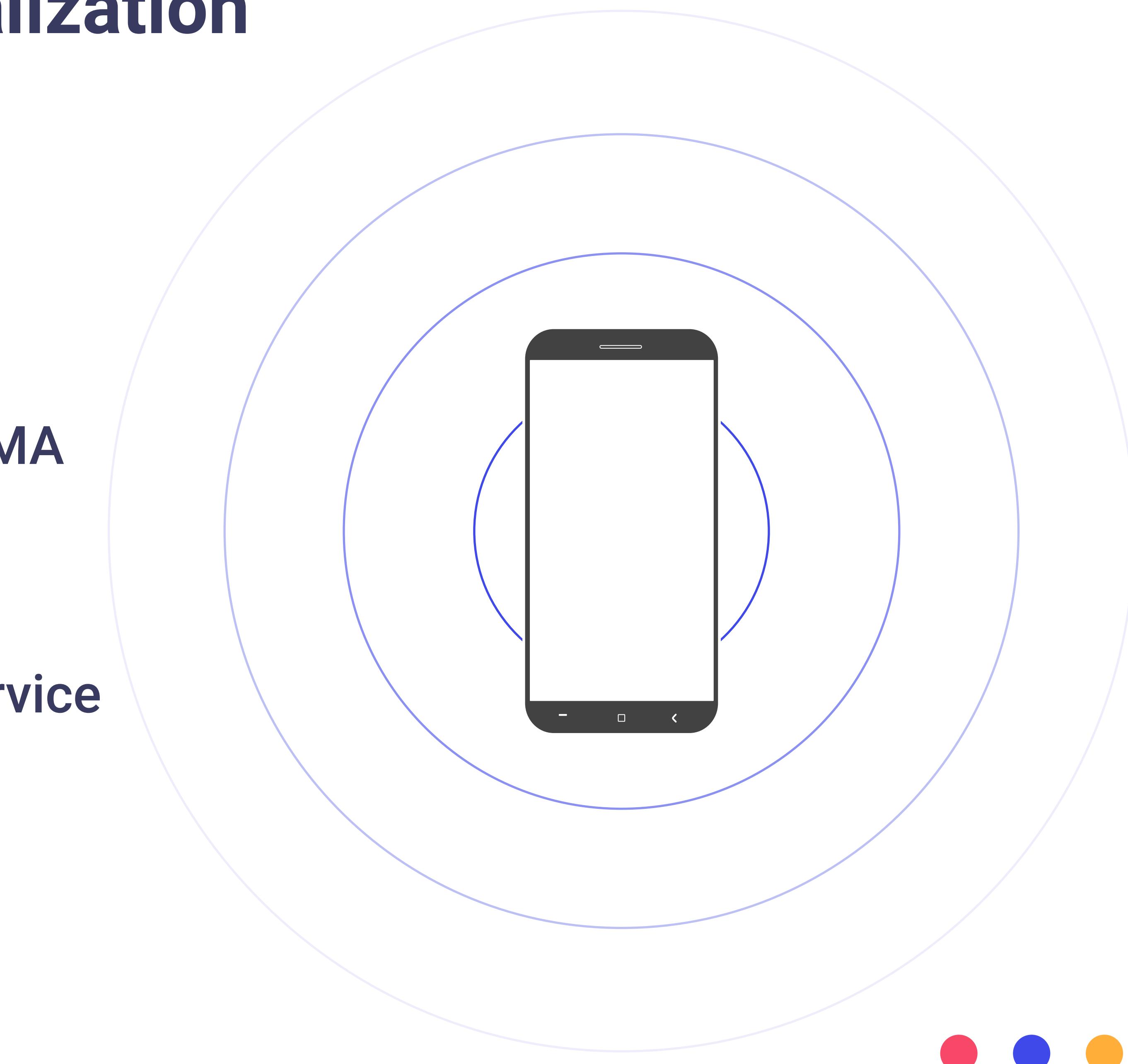
Characteristic Writer

- Perm: Write

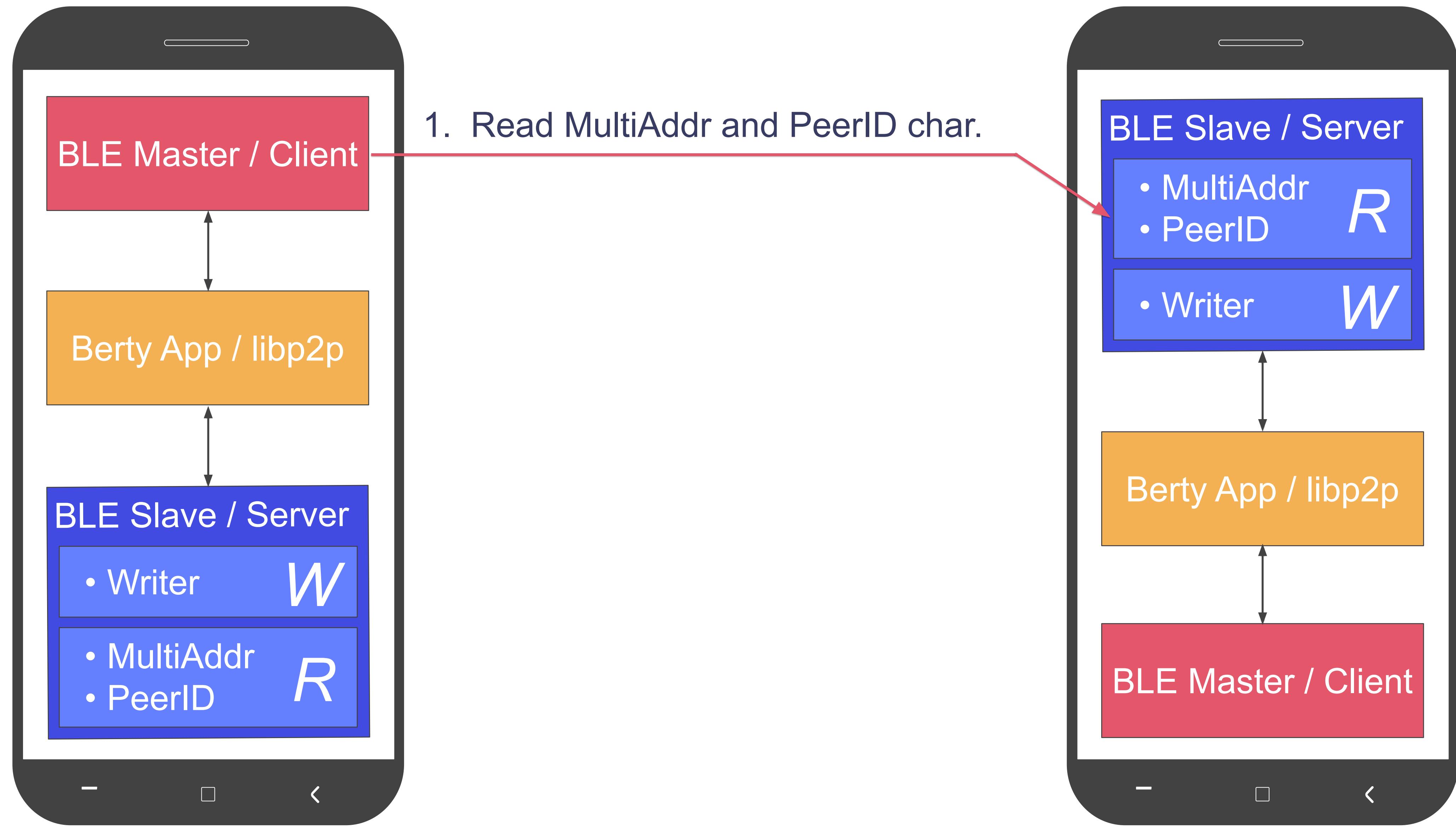


Transport and driver initialization

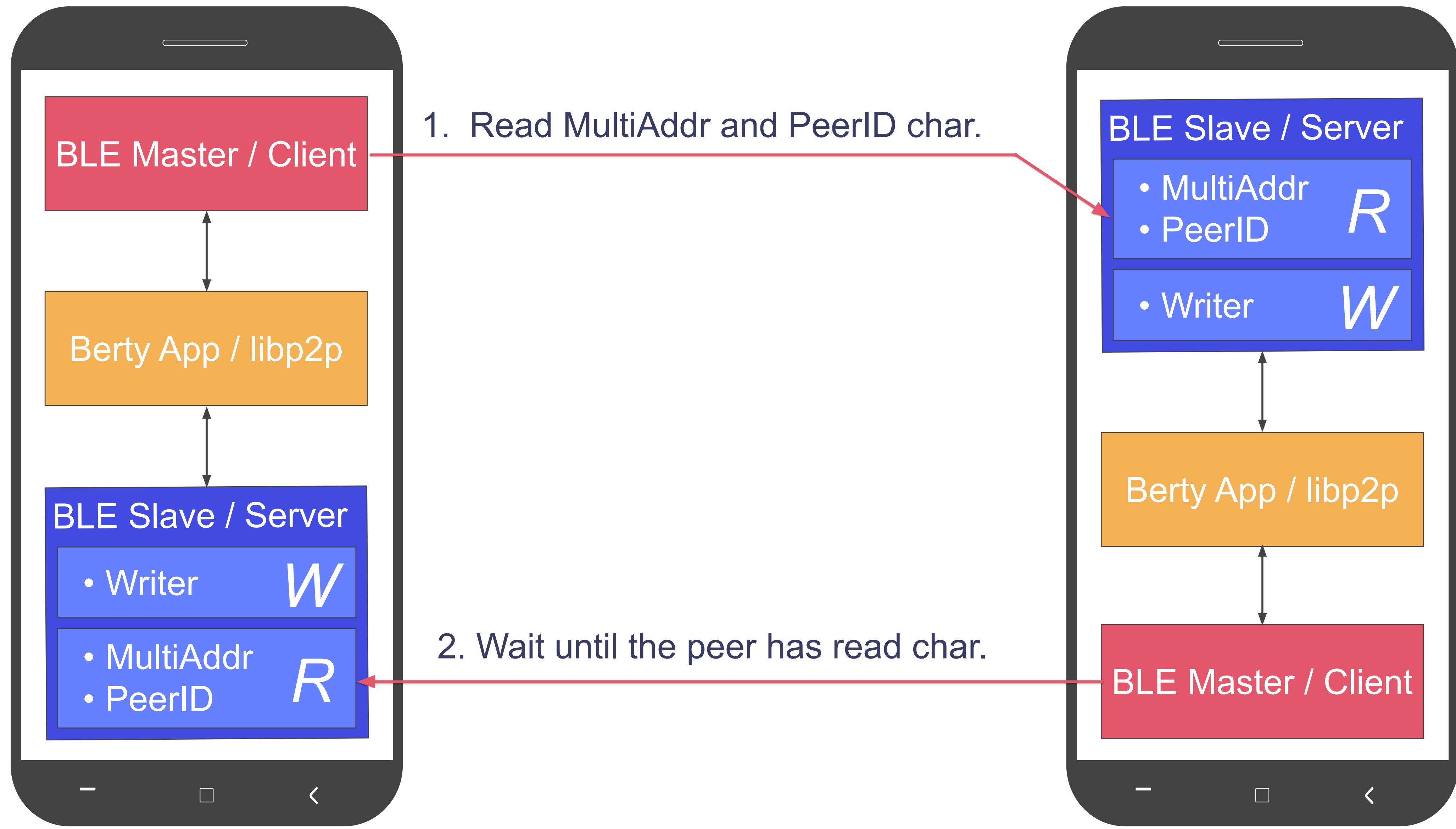
- Create a new BLE transport
- Call Listen() on it using an UUID as MA
- Start the native driver
- Advertise Berty's service
- Scan for peer advertising Berty's service



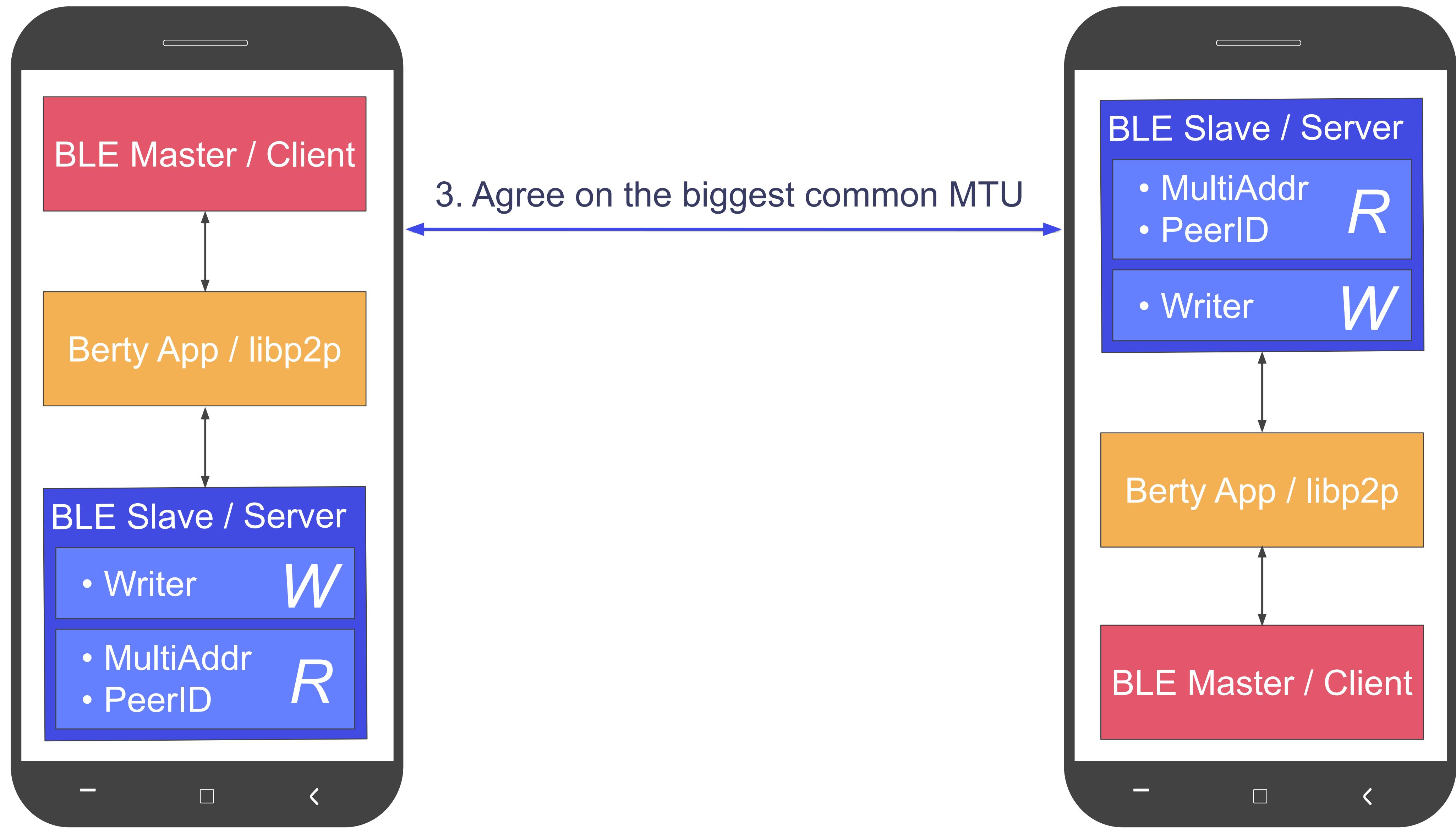
Handshake 1/2



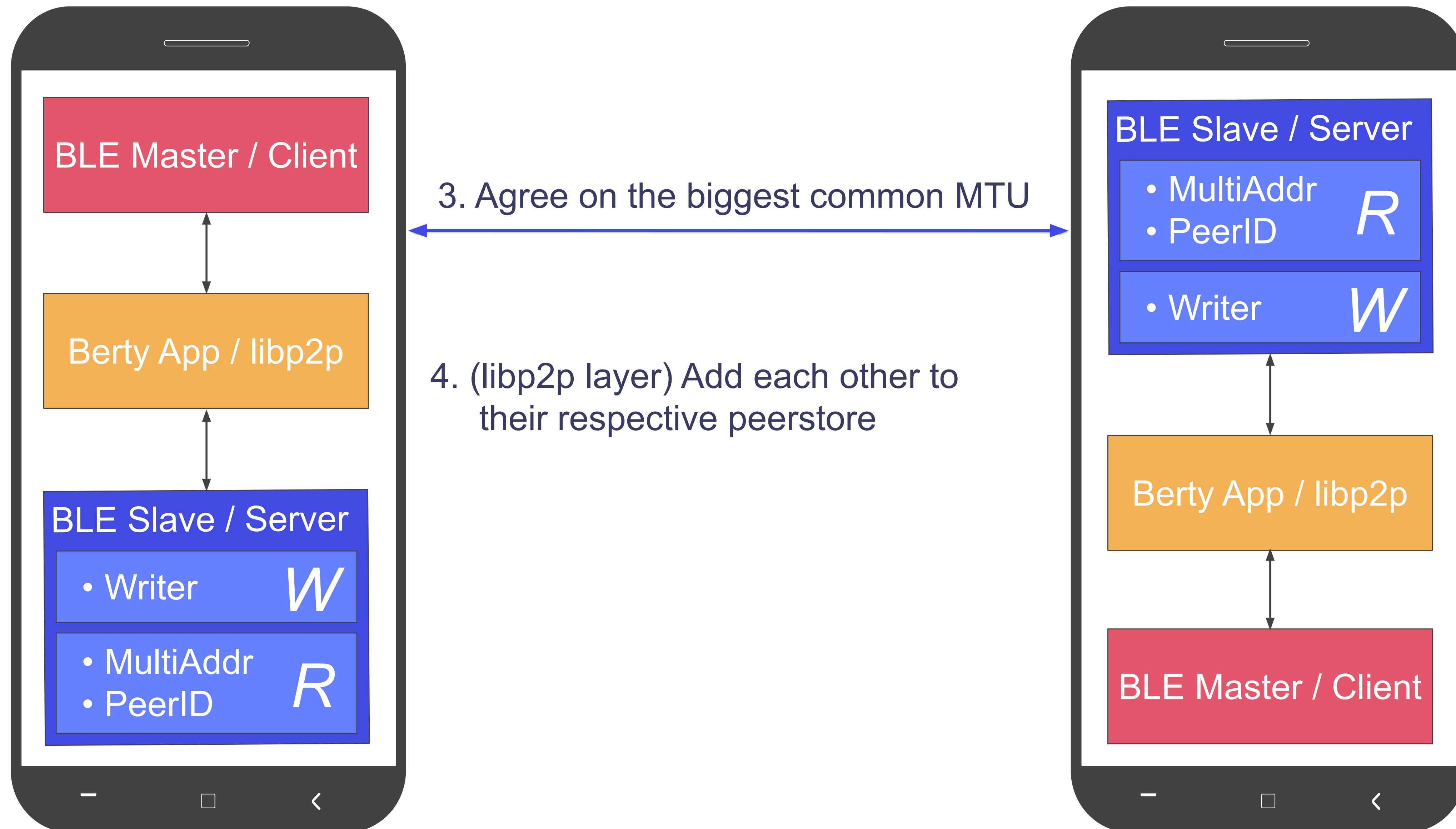
Handshake 1/2



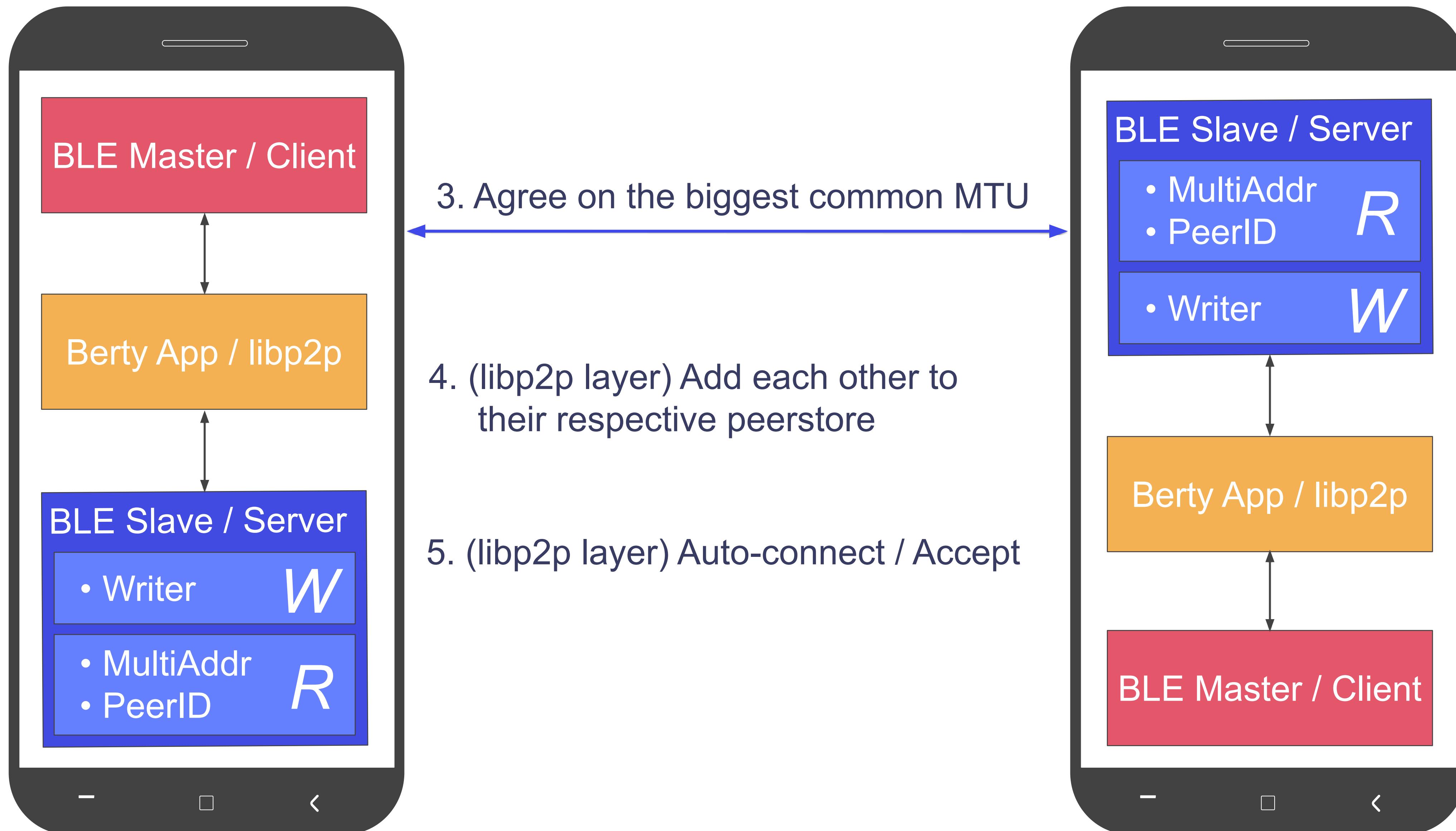
Handshake 2/2



Handshake 2/2



Handshake 2/2



Connection established



Additional notes

- With BLE you can't rely on MAC addresses

Reliability depends a lot on the chip you use

It's slow, possible improvement using L2CAP channel

We plan to implement drivers for Linux and Windows

Our implementation will be open-source soon™

In the meantime, for any question, feel free to reach out:

@aeddi

@sfroment

