

# CoAP over GATT

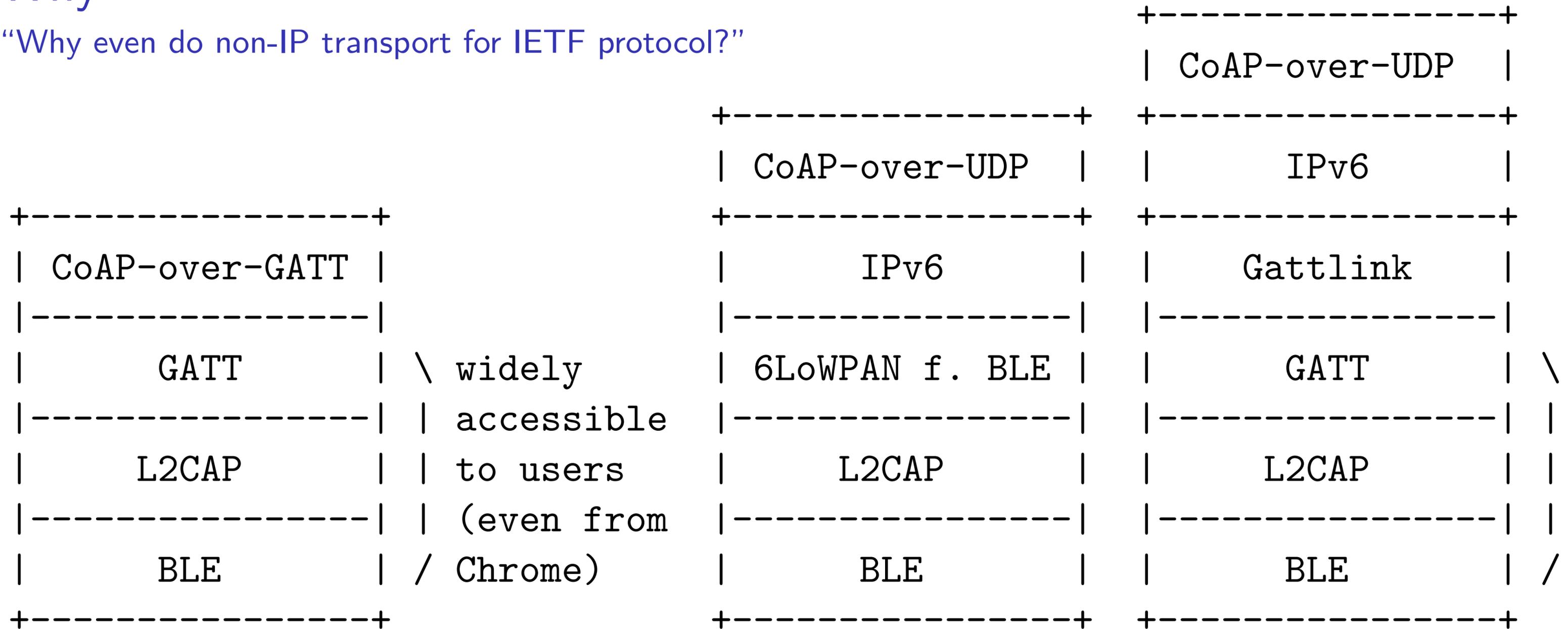
`draft-amsuess-core-coap-over-gatt`

Christian Amsüss

IETF116 Yokohama, CoRE, 2023-03-28

# Why

“Why even do non-IP transport for IETF protocol?”



You Are Here

BLE IPSP

Goldengate

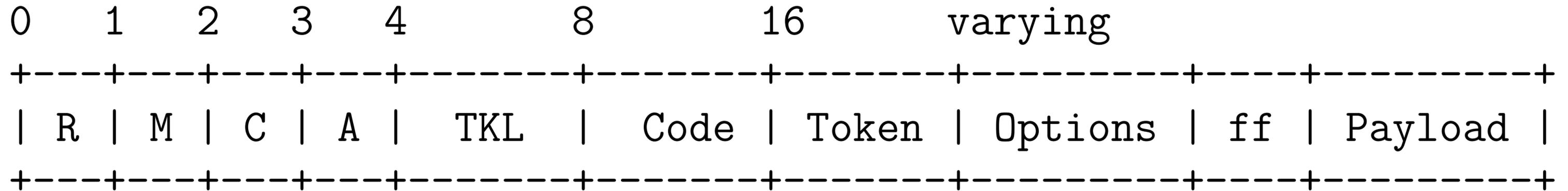
Same reasons as for CoAP-over-WebSockets: Can't escape limited APIs.

## 2022: Gathered implementation experience with -02

- BLE's reliability doesn't work exactly like I originally thought.
- Neither do BLE implementations.
- Lack of role reversal was limiting.
- Discovery is lacking.

## -03: Updated mechanism

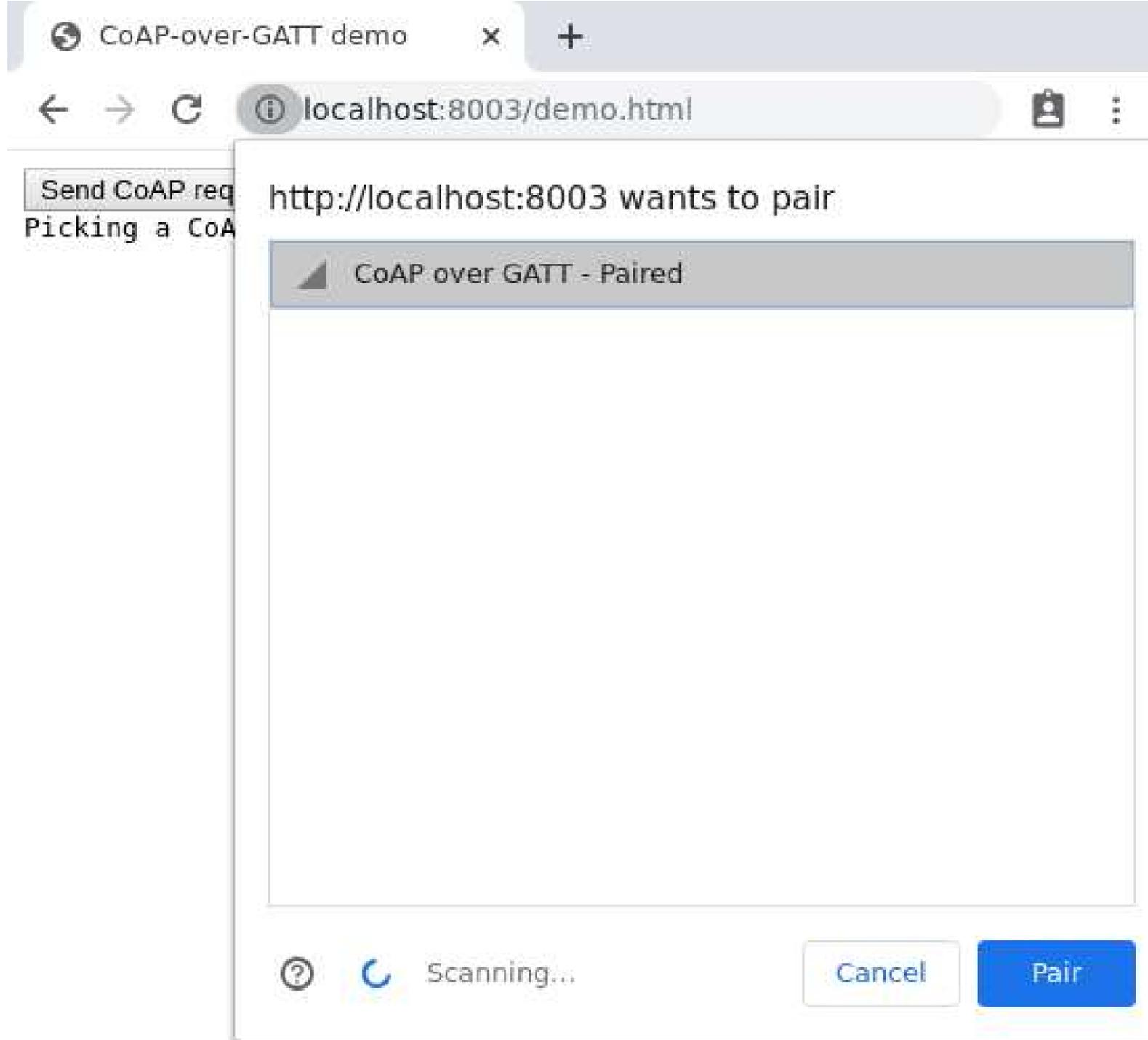
based on GATT's properties (ordered, unilateral reliability – highly unlike UDP)



- Re-introduced token.
- 1-bit message ID only takes care of incomplete operations.
- R-bit prevents accidental message reflection.
- Mechanism for requesting confirmation (a bit like empty ACKs).

Similar to CoAP-over-UDP, but adapted to BLE. Single-byte header.

# For -04: More about discovery



What if we already know our peer...

...and there are many  
CoAP-over-GATT devices around?

“know”: by URI, by AS/audience, by  
advertised resources?

For further exploration, possibly in a later add-on

- Multicast by CoAP messages in beacons?
- SCHC retconning of non-CoAP attributes. (“Turn the temperature into a CoAP addressable resource”).

# Addressing

It could be simple...

```
coap+gatt://00-11-22-33-44-55/.well-known/core
```

...but it is not, so we need `protocol-indication` (no updates here).

# Roadmap

- Update implementations, re-run tests.
- Keep using CoAP-over-GATT as driver for transport-indication (could also pick -over-WebSockets or t2trg-slipmux)
- Explore this in the WG?

# Thanks

Comments?

Questions?