

# draft-gomez-lpwan-fragmentation-header-03

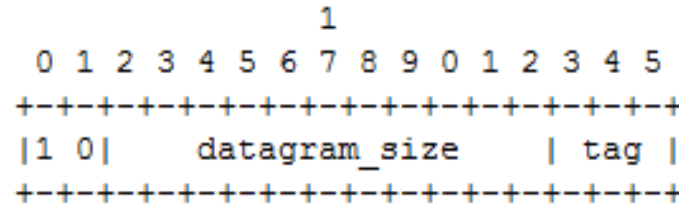
Carles Gomez, Josep Paradells  
Universitat Politècnica de Catalunya / Fundació i2cat

Jon Crowcroft  
University of Cambridge

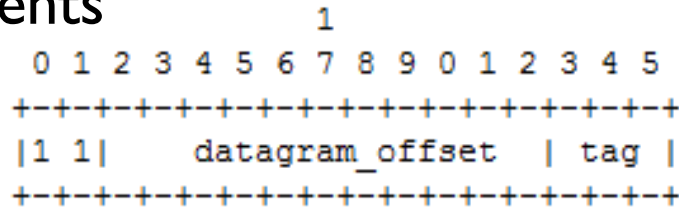
# Updated content (I/III)

- Fragmentation header
  - From 3-byte to 2-byte format

- First fragment



- Subsequent fragments



# Updated content (II/III)

- Format now not bound to 6LoWPAN dispatch
  - To be aligned with LPWAN work on header compression
  
- Name
  - Old: Optimized 6LoWPAN Fragmentation Header for LPWAN (6LoFHL)
  - New: LPWAN Fragmentation Header (LFH)

# Updated content (III/III)

- Adaptation layer fragmentation header overhead (bytes)

	IPv6 datagram size (bytes)							
	11		40		100		1280	
L2 payload (bytes)	4944	LFH	4944	LFH	4944	LFH	4944	LFH
10	----	4	----	10	----	26	----	320
15	0	0	24	8	64	16	799	198
20	0	0	19	6	59	12	794	144
25	0	0	14	4	34	10	399	112
30	0	0	9	4	24	8	269	92

# Discussion: 1-byte format ?

# Option A

- Possible format
  - 1 bit: fragmentation header or not
  - 7 bits: fragment number
  - No tag, no 'more fragments' bit
- Is this feasible at all ?
  - LoRaWAN: yes (enough to number all fragments for a 1280-byte packet)
  - Sigfox: yes (uplink), no (downlink)

# Option A: issues

- Incomplete packets
  - E.g. received sequence of fragments 1, 2, 1, 2, 3, 4
    - If two packets carried by 4 fragments each had been sent, the first one is incomplete
- Additional delay
  - Receiver does not know when all fragments of a packet have been received
    - Must wait for a time that, given message rate constraints, may be significant
- Apparently correct reassembly
  - E.g. received sequence of fragments 1, 2, 3, being in reality 1-A, 2-B, 3-B

# Option B

- Possible format
  - 1 bit: fragmentation header (or not)
  - 1 bit: more fragments (or not)
  - 6 bits: fragment number
  - No tag
- Is this feasible at all ?
  - LoRaWAN: yes (enough to number all fragments for a 1280-byte packet)
  - Sigfox: no



# Option B: issues

- No incomplete packets issue
  - The ‘more fragments’ bit allows to identify incomplete packets
- No additional delay
  - Receiver knows whether all fragments of a packet have been received
- Apparently correct reassembly
  - E.g. received sequence of fragments 1, 2, 3, being in reality 1-A, 2-B, 3-B

# Summary

- LoRaWAN
  - Can use option B
  - 1-byte, but ‘apparently correct reassembly’ issue
- Sigfox
  - Can use option A for the uplink (only)
  - 1-byte, but ‘incomplete packets’, ‘apparently correct reassembly’, and ‘additional delay’ issues