IP protocols over LPWANs

Work going on at IETF

Presentation to Journées non-thématiques RESCOM Jan 2019, Grenoble

Dominique Barthel, Quentin Lampin
(Orange Labs Meylan)

Orange Expert
Future
networks

Agenda

Problem statement

Header Compression

Fragmentation

Draft status, Implementations

Conclusions and Future Work

Problem statement

Problem statement (1/2)

All networks in the real world are IP-based, but Low Power Wide Area Networks (LPWANs).

6LoWPAN addressed IEEE 802.15.4 networks (WPANs)

- Meshed
- Payload about 100 bytes

6lo addresses “low-power” networks other than 15.4

focuses on the work that facilitates IPv6 connectivity over constrained node networks with the characteristics of:

- limited power, memory and processing resources
- hard upper bounds on state, code space and processing cycles
- optimization of energy and network bandwidth usage
- lack of some layer 2 services like complete device connectivity and broadcast/multicast

6lo adapts 6LoWPAN work to technologies similar to IEEE802.15.4

Problem statement (2/2)

LPWANs are next level of constrained-ness

- Payload of 10-100's of bytes, variable MTUs
- Asymmetric transmission allowance
- Star topology

LPWAN Working Group formed June 2016

Reference technologies: LoRaWAN, Sigfox, NB-IoT, WiSUN

- see RFC8376 for a short description of them

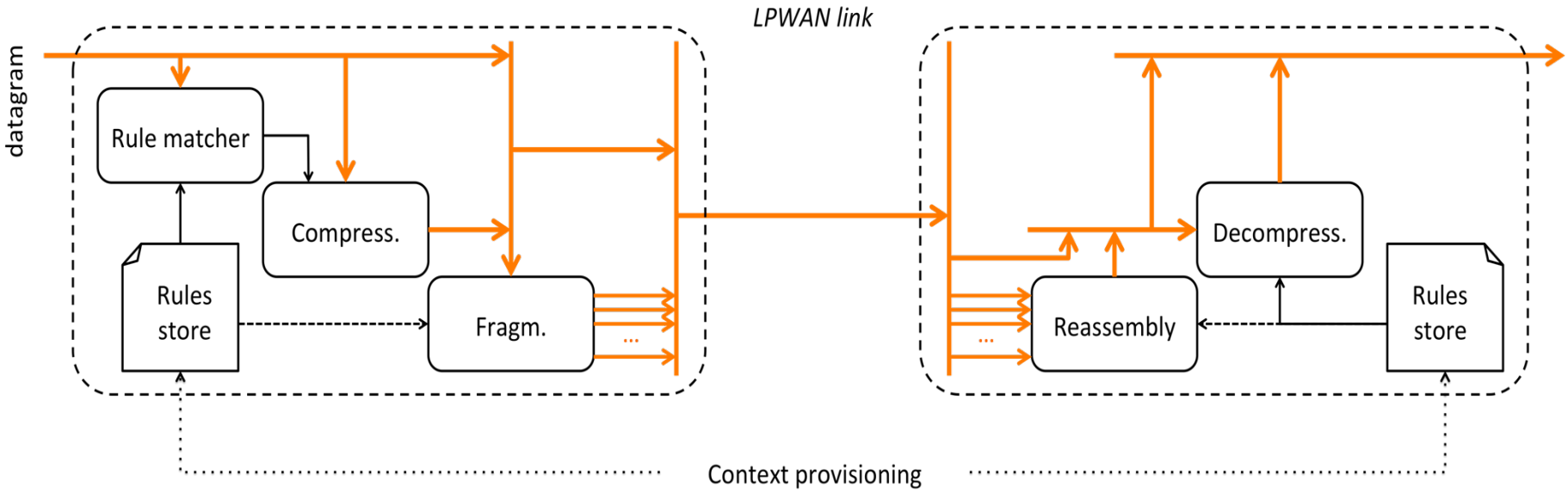
Header Compression prior art

RoHC (RFC5795), 6LoWPAN HC (RFC6282), 6lo GHC (RFC7400)

Fragmentation prior art

6LoWPAN (RFC4944)

Global architecture



Header Compression

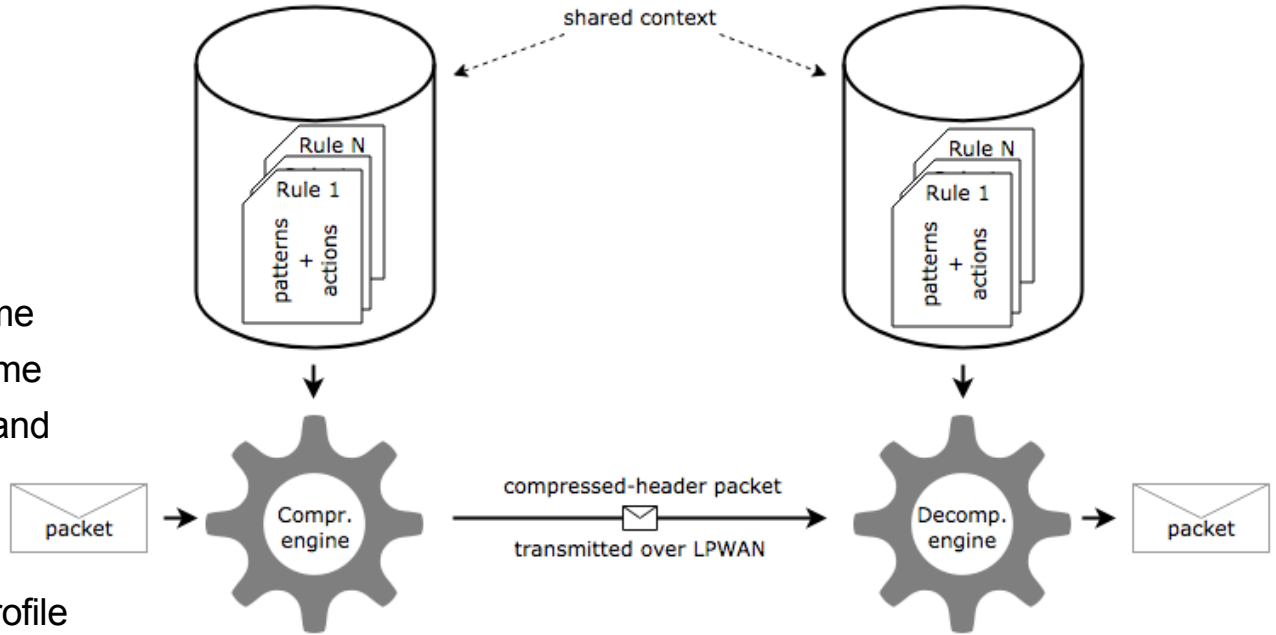
Header Compression principles

Compression is Rule-based

- Similar to RoHC

Static Context (= rule set)

- No dynamic context update
- Really-constrained devices have unchanged traffic pattern over time
- Traffic pattern known ahead of time
- Context provisioned into device and network
  - Pre-provisioned, or via a configuration protocol
- Rule set generated per device profile



Header pattern matching and compression

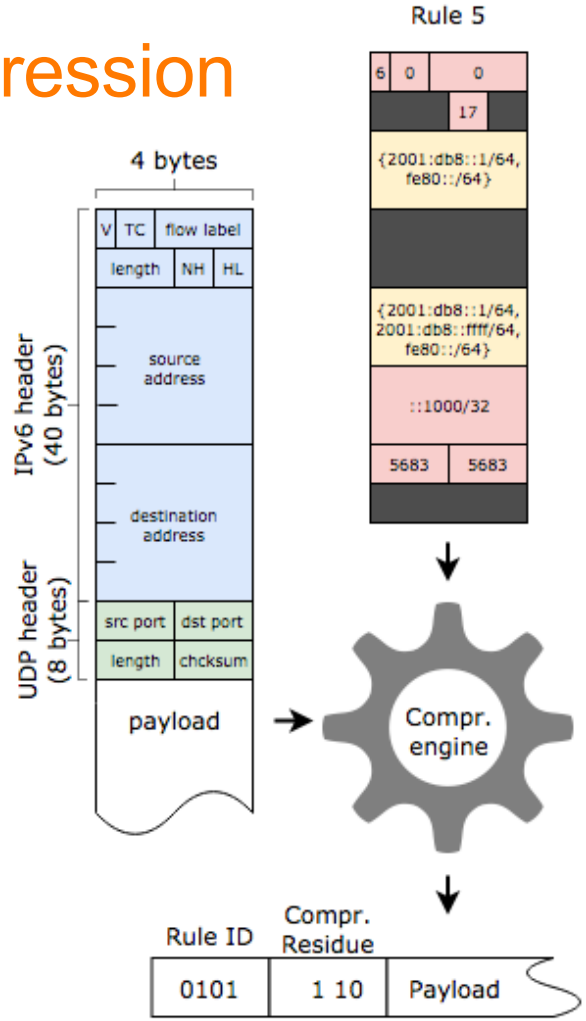
Find rule, using Matching Operator per field

- Equal
- MSB(x) match
- List match
- Special case
- Ignore

Apply Compression Action to each field

- Elide
- Send Least Significant Bits
- Send index within pre-defined set
- Recompute at receive end
- Transmit in extenso

Also supports variable length fields

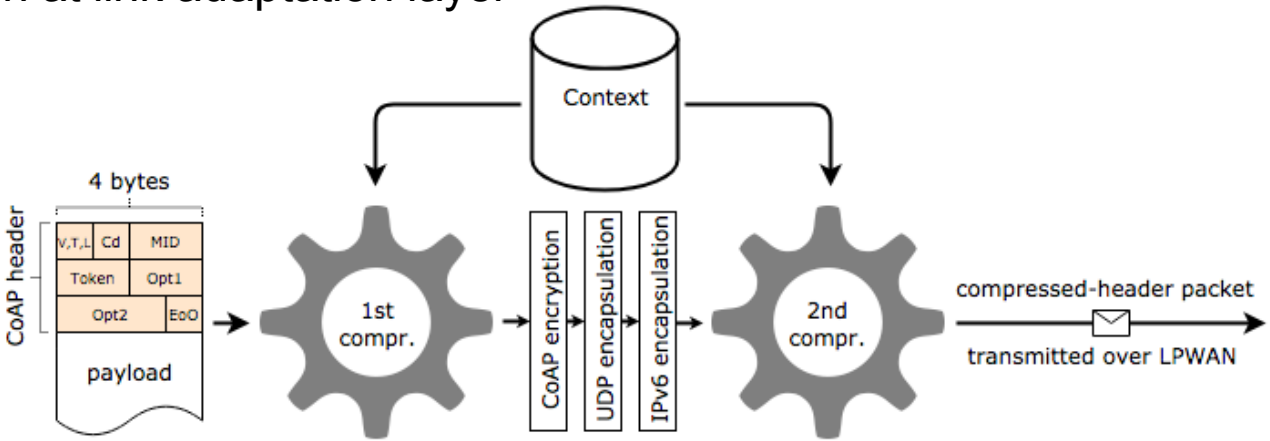


Double compression

Needed for efficient compression when encryption is present

For example, OSCORE

- Outer header compression at application layer
- Inner header compression at link adaptation layer



Fragmentation

ACK-on-Error fragmentation mode

Reliable fragmentation mode

- With retransmissions and final integrity checking

Intermediate ACKs only sent when fragments are lost

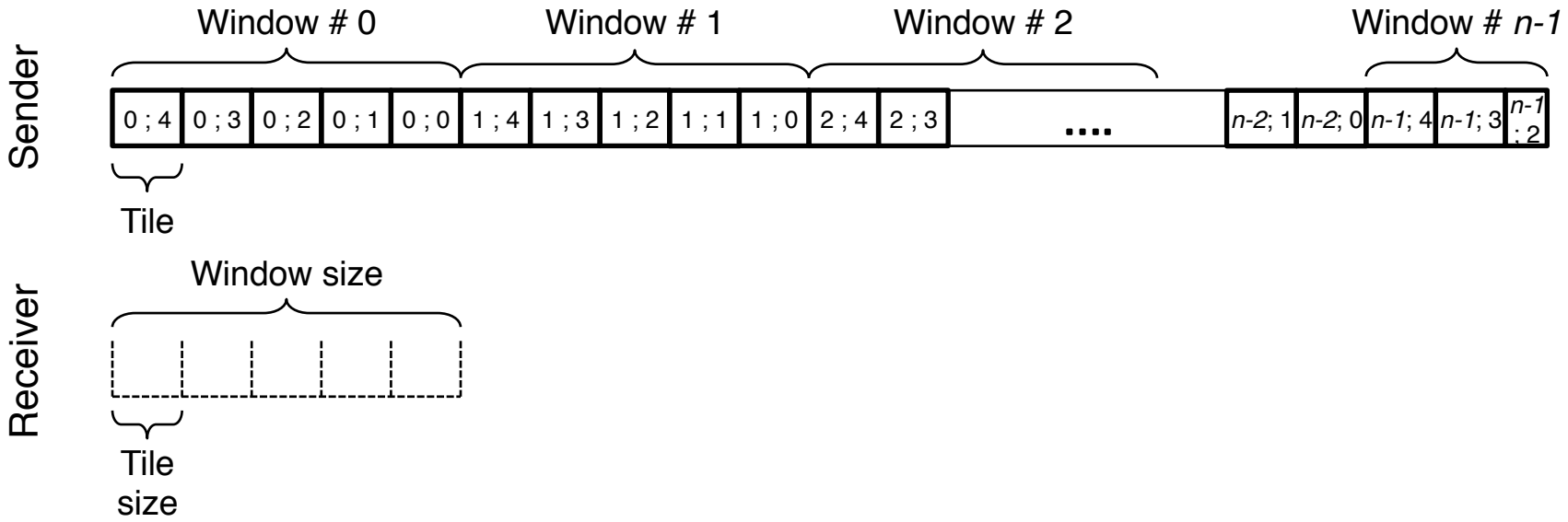
- Saves on downlink transmission for uplink fragmented packets

Supports variable MTUs

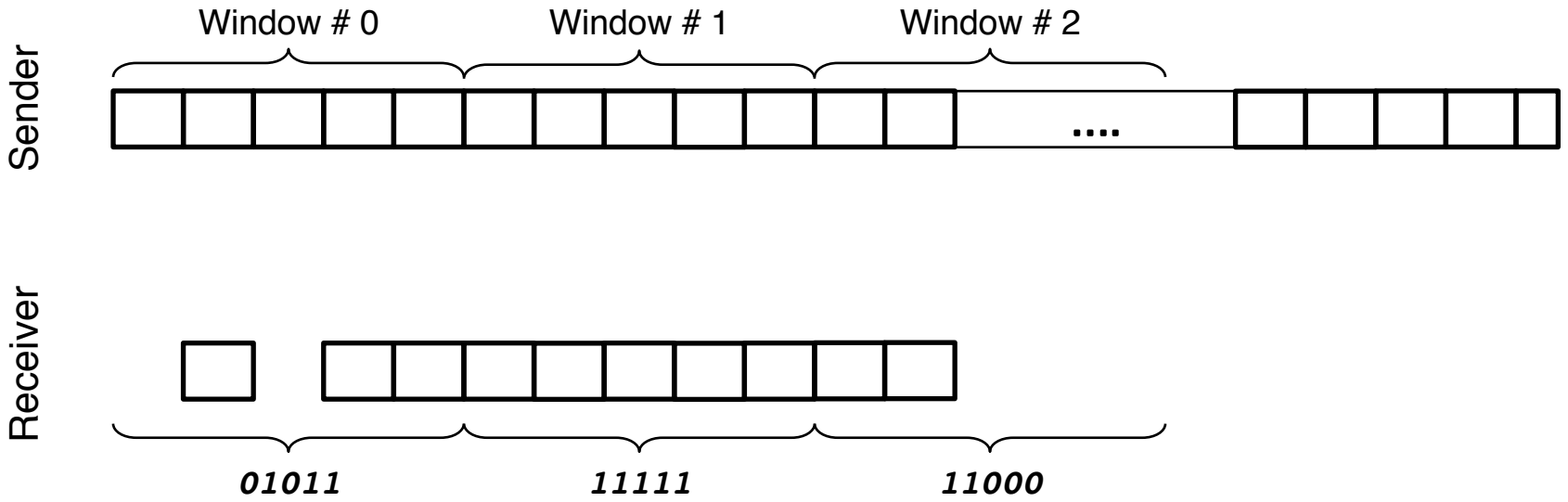
ACK bits grouped into bitmaps

- Saves on ACK header overhead
- Controls the size of the ACK message

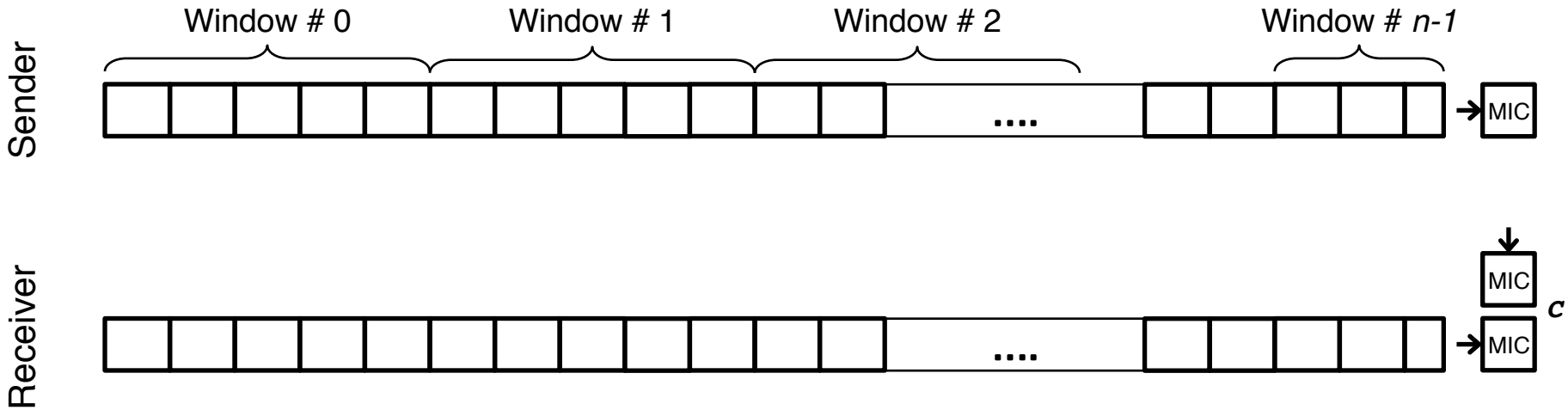
Tiles, windows of tiles



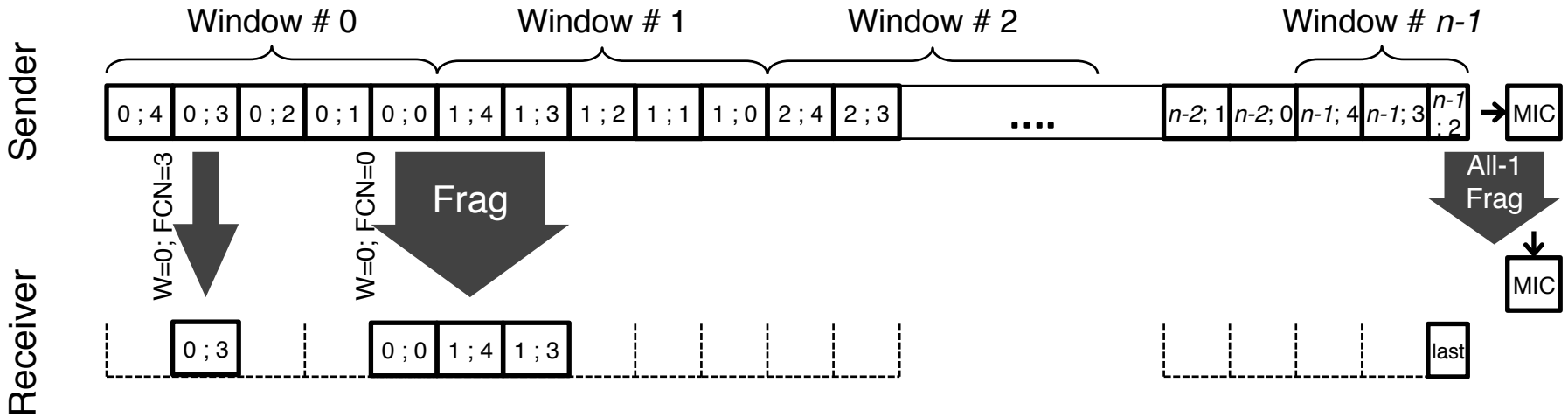
Bitmaps


Window # 0
Window # 1
Window # 2
Sender
....
Receiver
01011
11111
11000

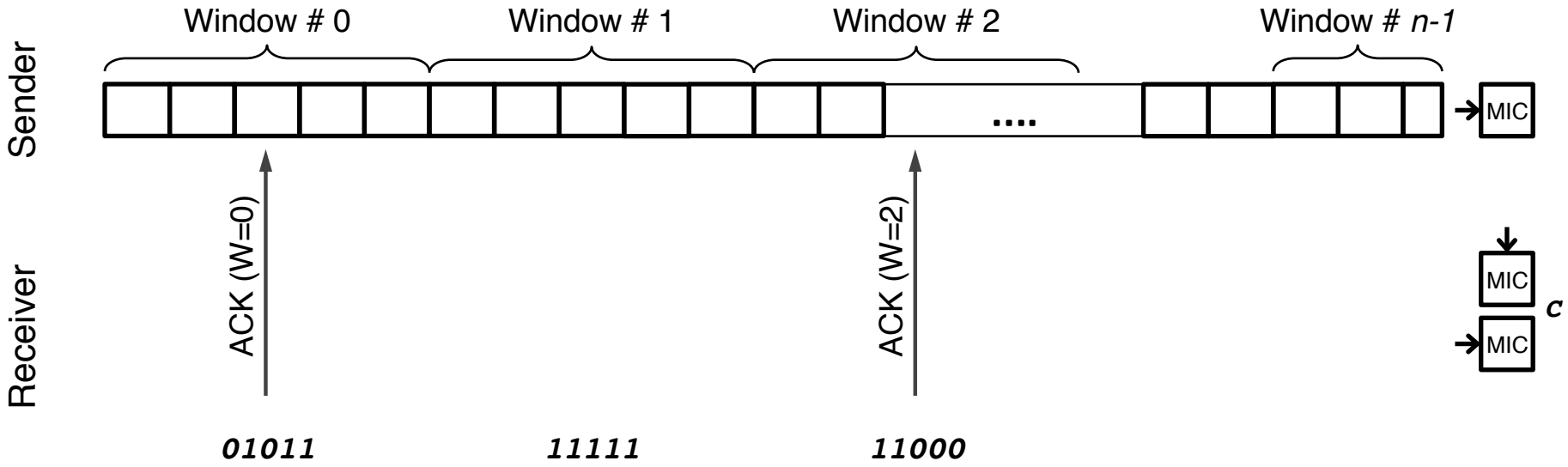
Message Integrity Check (MIC)



Fragment messages



ACK messages



Window size adjusts trade-off between size of ACK message and number of ACK message.

ACK-on-Error algorithm (simplified)

Sender

Sends all tiles, expects ACK after sending All-1 Frag.
Resends tiles reported missing by ACK, sends ACK REQ.
Iterates until ACK reports MIC matches.

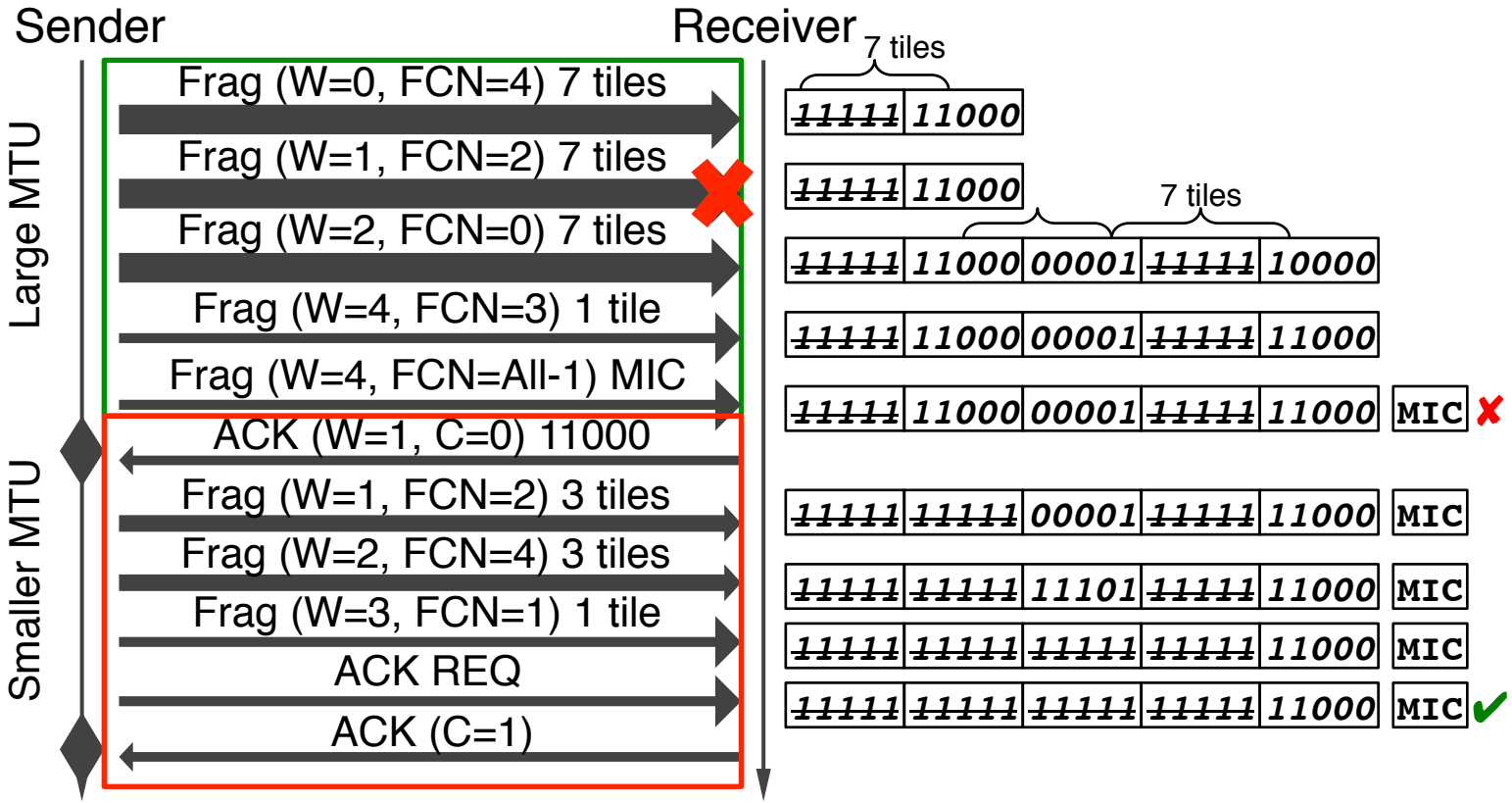
Receiver

Assembles all tiles received.
On receiving All-1 Frag or ACK REQ, returns ACK for lowest-numbered window with missing tile, expects more tiles.
Iterates until MIC matches.

Loosely-coupled behavior. Simple state machines.

ACK policy may be made more tightly-coupled if appropriate.

Fragmentation example



Standardization status and implementations

Standardization status

Description	Status	Name/Link
LPWAN technologies description	Published	RFC8376
SCHC compression, fragmentation, application to UDP/IPv6	Submitted for publication	draft-ietf-lpwan-ipv6-static-context-hc
CoAP header compression	Work in progress	draft-ietf-lpwan-coap-static-context-hc
Application to the reference LPWAN technologies	Work in progress	draft-petrov-lpwan-ipv6-schc-over-lorawan draft-zuniga-lpwan-schc-over-sigfox draft-minaburo-lpwan-nbiot-hc draft-authors-lpwan-schc-802154

Implementations

Proprietary implementation by Acklio

- Spin-off of IMT-Atlantique, Rennes

acklio
Unifying LPWA Networks

Open Source implementation

openSCHC on GitHub

Python3/Micropython code

Contributions welcome

- Implementation
- Test
- Documentation

Send students this way!

Conclusions and Future Work

Conclusions and Future Work

Conclusions

Write IoT applications as Web applications

- Breaking the LPWAN SILOs
- Leverage existing skills
- Speed up market adoption, lower costs

Future work

Rule representation

Compression rule generation

- Automatic generation
- Many selective rules vs. fewer more encompassing rules

Fragmentation parameters optimization for given LPWAN technology/situation

Design of IoT device management architecture using existing IP-based protocols

- CORECONF: RESTCONF with CoAP, YANG modules compiled to CBOR

Thanks