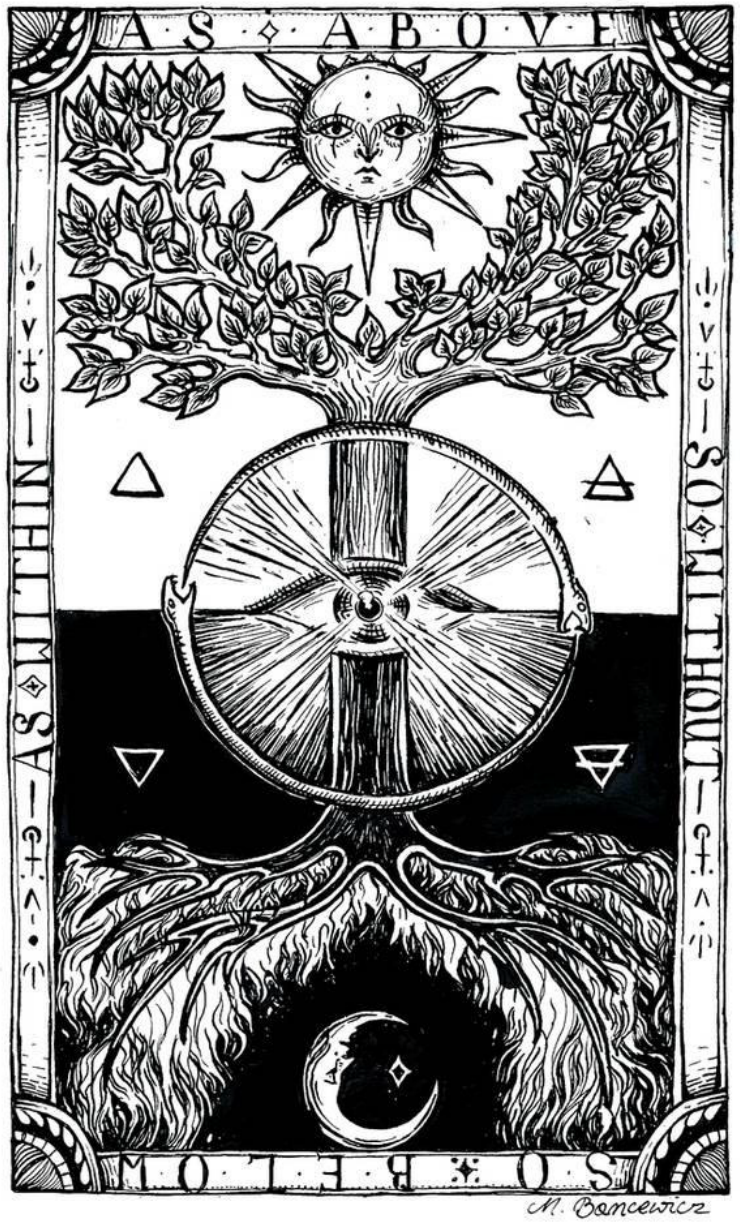
Integrating LoRa nodes into global DTN networks

Carlo Caini*, Samo Grasic^, Alberto Soncini*
*DEI/ARCES, University of Bologna, Italy
^IPNSIG

Introduction

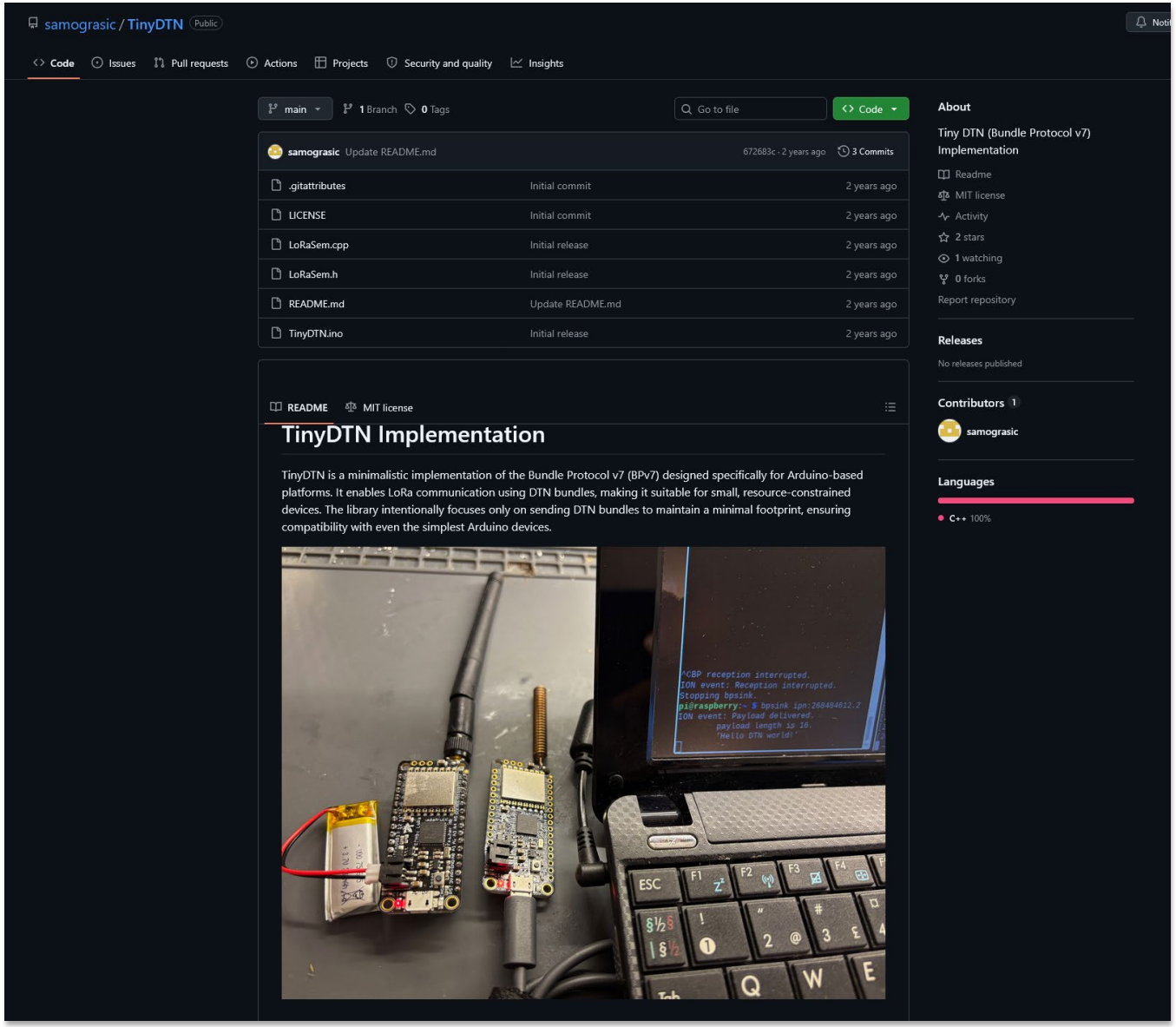
<<That which is above is like to that which is below, and that which is below is like to that which is above.>>



Introduction

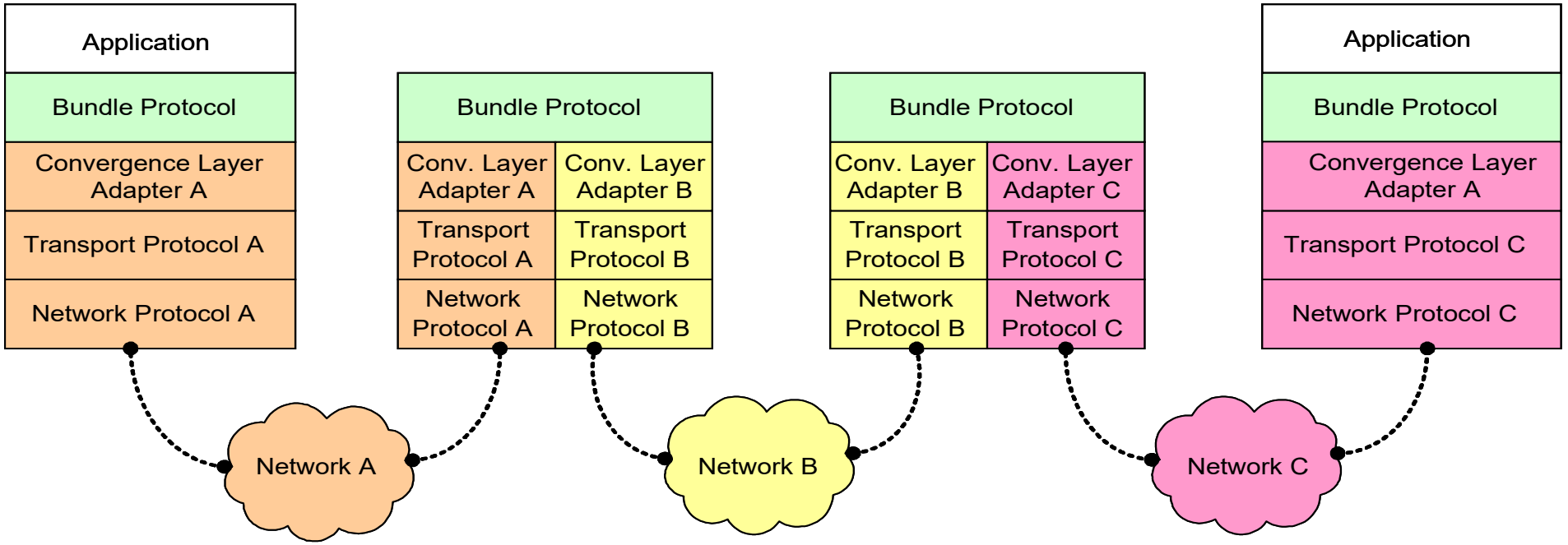
TinyDTN is a Proof of Concept exploring the possibility of implementing DTN architecture on resource-constrained hardware.

Our project began where TinyDTN left off, aiming for a full implementation of the pioneered ideas



DTN architecture and Bundle Protocol

- Two key aspects differentiate DTN from standard Internet architecture.
  - The end-to-end path is divided into multiple DTN hops
    - Challenges confined in the hop where they appear
  - BP allows intermediate nodes to store data for extended periods of time
    - Essential to cope with link intermittency

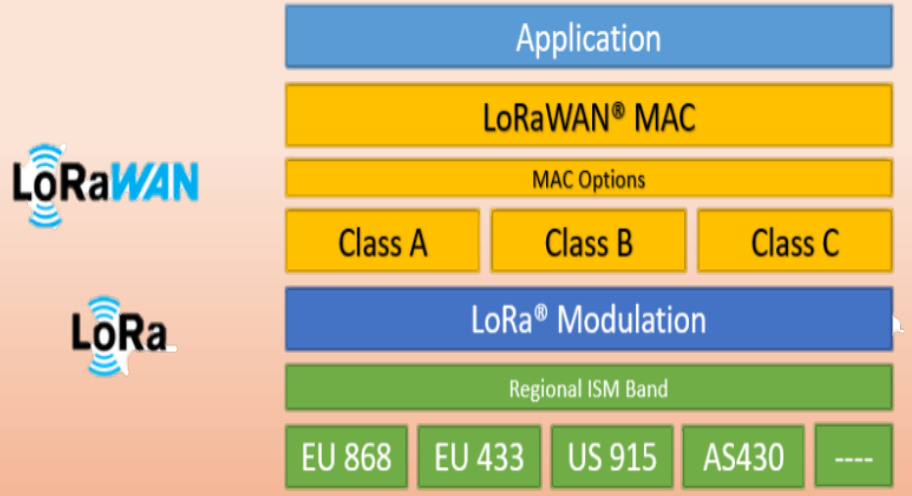


LoRa

LoRa is a proprietary radio communication technology widely used in wireless sensor networks, thanks to its extended range of a few kilometers and low power consumption.

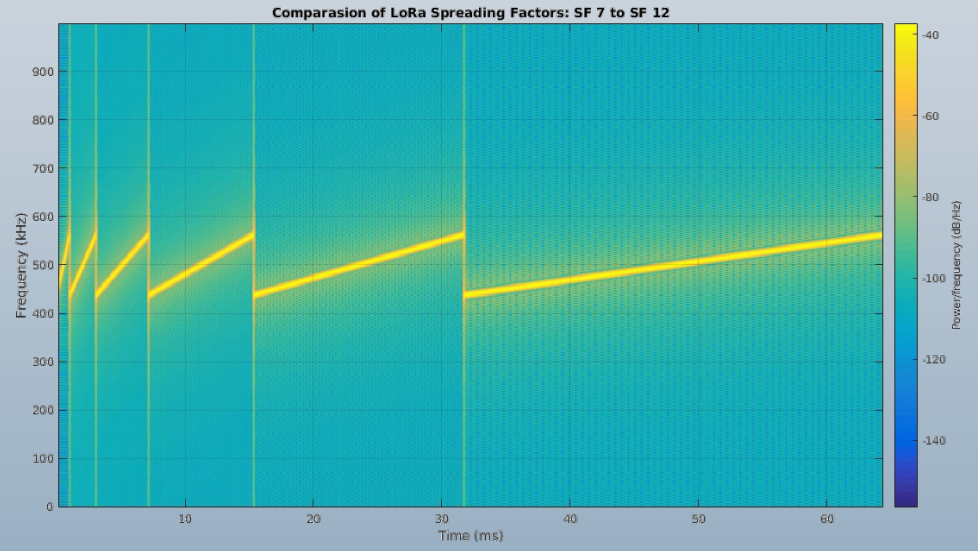
LoRa is NOT LoRaWAN

LoRaWAN (LoRa Wide Area Network) uses LoRa at the physical layer.



LoRa

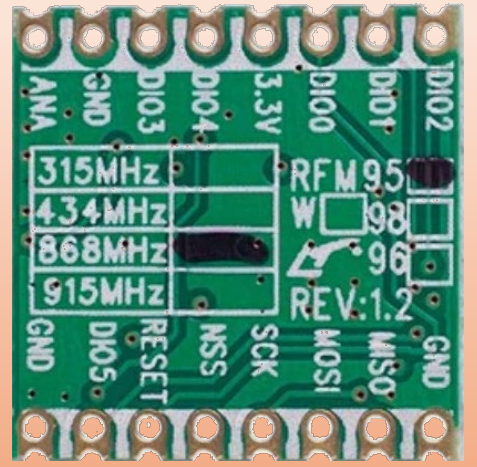
The achievable data rate depends on many factors, e.g. the Spreading Factor (SF), the Coding Rate (CR), the bandwidth (BW), etc., but is always quite modest (from about 0.25 to 27 kbit/s). MTU is also proportional to data rate.



LoRa operates in a few unlicensed ISM (Industry, Scientific, Medical) bands.

LoRa modules are usually tuned for operation in or near a specific band.

We used modules for EU868 (863-870 MHz), which is the most common LoRa band in Europe.

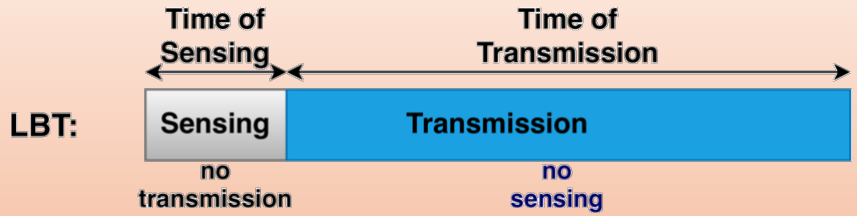


LoRa

Devices operating in EU868 are subject to a Time on Air regulatory limitation of 1% Duty Cycle over a 1-hour sliding window.

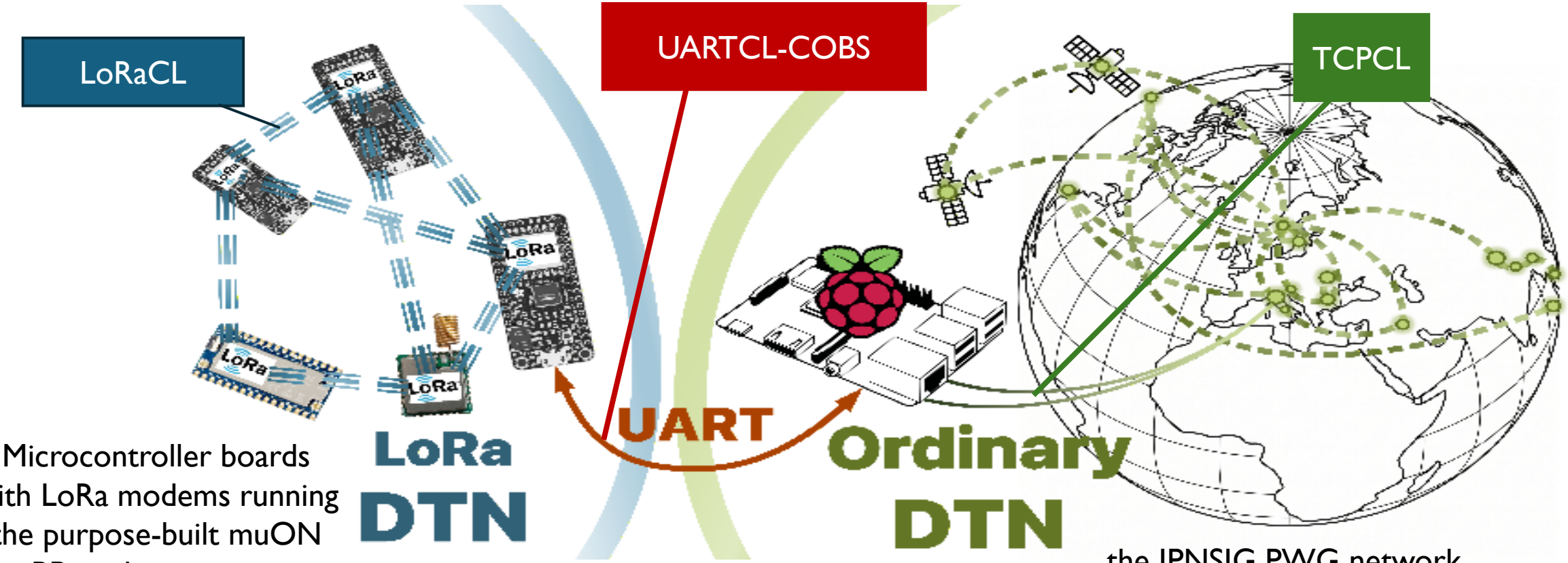
This aspect means LoRa connectivity is inherently intermittent, making DTN technology integration quite appealing.

Lora is also subject to limitations on max Tx power and requires adopting a Listen Before Talk strategy due to the nature of the channel (shared among all users in a wide range).



System Model

A LoRa-based DTN network is connected bi-directionally to an ordinary DTN network



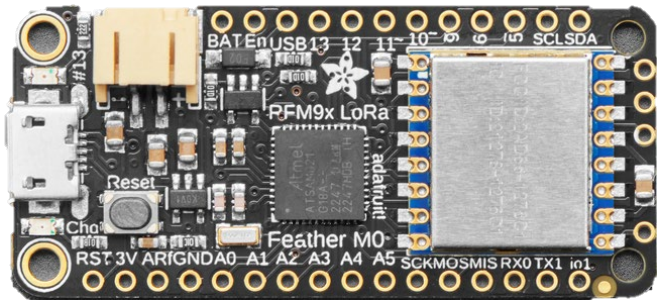
Microcontroller boards with LoRa modems running the purpose-built muON BP implementation

the IPNSIG PWG network

A closer look at the test hardware

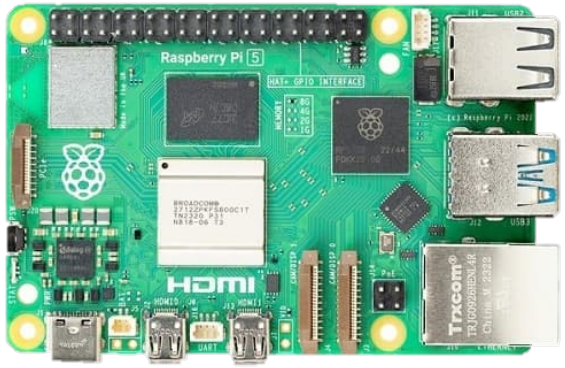
Adafruit feather m0 LoRa

ATSAMD21G18 @ 48MHz
256K of FLASH
32K of RAM
Hardware UART, I2C, SPI
SX127x LoRa Modem



Raspberry Pi 5 Model B

4x 64-bit ARM Cortex-A76 @ 2.4GHz
8 GB LPDDR4X SDRAM
Linux-based OS (Debian variant)
Gigabit Ethernet
802.11ac Wi-Fi



muON – microcontroller Overlay Network

- muON is a lightweight BP implementation for resource-constrained devices such as microcontrollers.
- muON aims to fulfill three objectives:
  - Enable DTN technology on microcontrollers with limited system resources.
  - Simplify rapid research and development of embedded tools leveraging DTN architecture.
  - Make DTN easily available (and *hackable*) to the general public.
- muON is released under GPLv3 terms and is available now on GitHub.
  - The code is currently WIP and new features are under active development.

muON – microcontroller Overlay Network

- muON is written in C++ with a modular design
    - First developed for the SAM D architecture and Arduino framework, it may easily be ported to other architectures and frameworks, such as STM32, ESP32, RP2XX0, etc.
- The essential BP features already implemented in muON are:
    - the ability to generate and read bundles fully conforming with the standard
    - the ability to store bundles locally
    - a convergence layer for LoRa (LoRaCL)
    - a convergence layer for UART (UARTCL-COBS)
    - Priorities (at present in a non standard way)

The LoRa Convergence Layer

- LoRaCL enables transmission of bundles on LoRa packets.
- Fully implemented in muON
- LoRa payload size is both very small and variable with the modulation settings
  - the obvious solution of limiting the bundle size to the LoRa payload was rejected, as very constraining.
- A better solution is to organize the transmission of one bundle, a "Bundle Transfer", in one or more XFER_MESSAGEs, each of which is then encapsulated into a LoRa packet
  - a mechanism largely borrowed from TCPCL and QUICCL

The LoRaCL messages

- Two services
  - Unreliable
    - No feedback about the reception of the sent bundle
  - Notified
    - If all segments of the transfer are received, the transfer is ACK-ed by a BDL_XFER_ACK
    - Otherwise, the failure is signaled by a XFER_REFUSE
      - The bundle can be possibly resent by BP
- In both cases, bundles can be larger than LoRa packet payloads!

MT	Name	Description
00	XFER_SEGMENT	Contains a segment of bundle data
01	BDL_XFER_ACK	Acknowledges a Bundle Transfer in the notified service
10	XFER_REFUSE	Refuses a bundle transfer initiated by the peer. / Signal a failure in the notified service
11	MSJ_REJECT	Reports an unrecoverable error

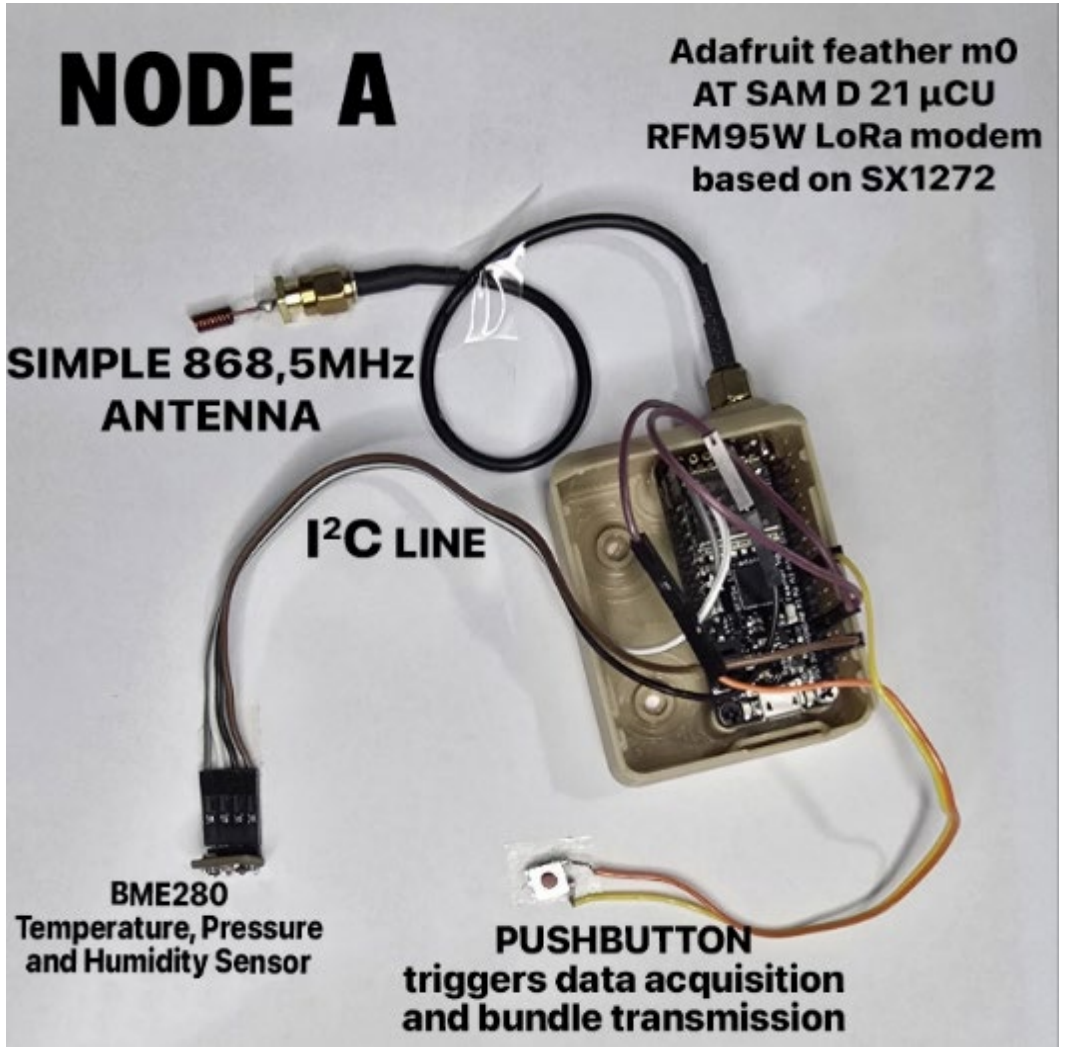
The management of LoRa link intermittency

- To comply with regulatory limits, LoRaCL adopts a token bucket strategy
  - ToA budget replenished at a rate of 1ms every 100ms; it can never exceed 36s (1/100 of hour).
- Before each bundle transmission, LoRaCL calculates the expected transmission time.
  - If this is larger than the available ToA budget, LoRaCL signals the temporary unfeasibility to the BP Agent, which can then decide how to proceed, e.g.:
    - to postpone the transmission
    - to ask the routing engine to look for a different proximate node…

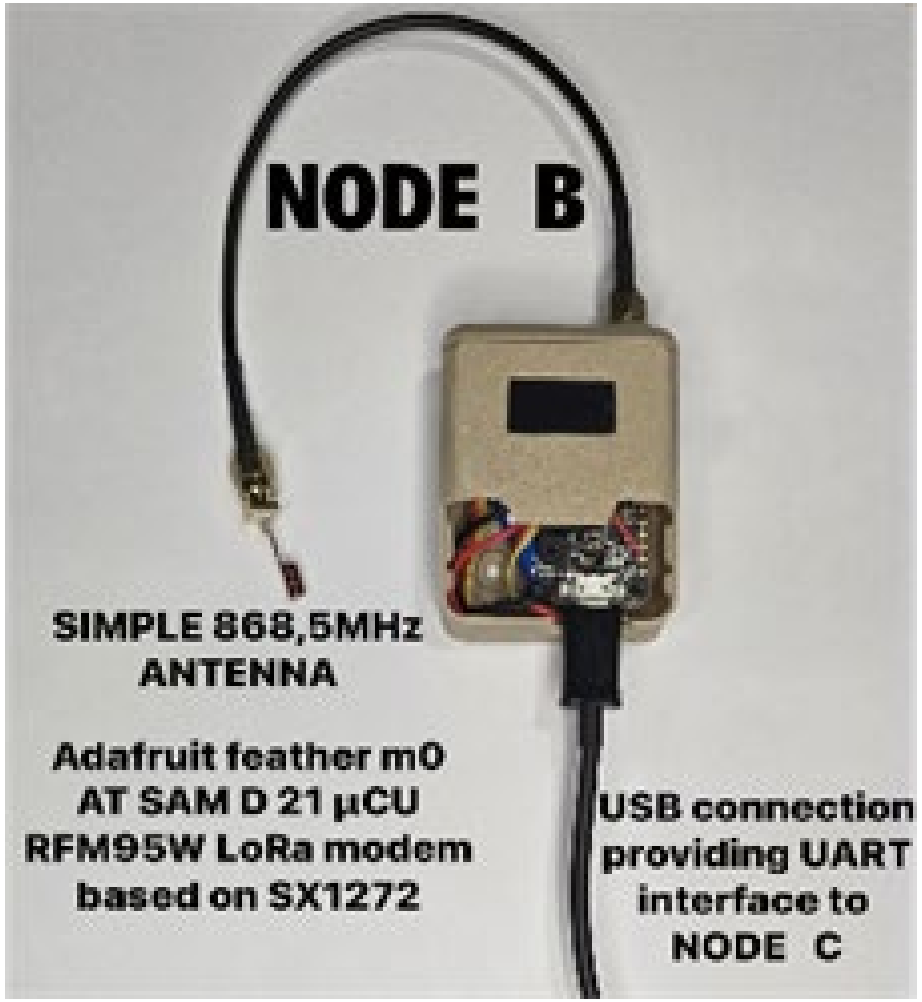
UARTCL-COBS

- UARTCL-COBS has been designed as an asymmetric point-to-point link responsible for encapsulating a CBOR-encoded bundle in a byte stream, providing protection against corruption, and delivering it over a UART connection.
- In our scenario it is used only between the two Gateways
  - since these devices are running two different BP implementations (muON and IONe), this protocol must be implemented in both.
- UARTCL-COBS is a best-effort delivery duct
  - It provides no ACK nor retransmission capabilities and only guarantees that corrupted frames will be discarded
  - eventual recovery operations are delegated to the overlaid BP Agent.
  - This allows UARTCL to remain lightweight, avoid state machine complexity and use processing power conservatively within the ample allowances of a UART link.

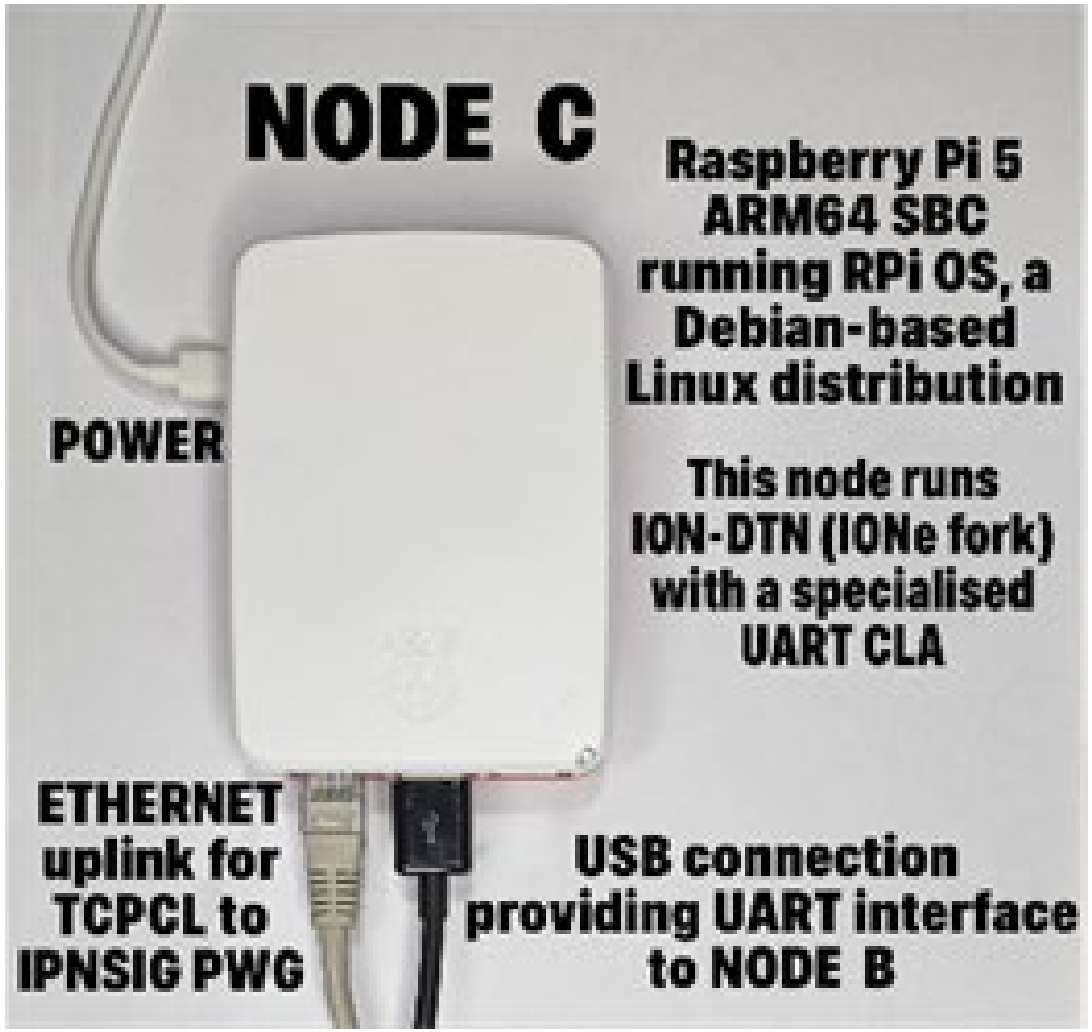
Testbed Setup – NODE A



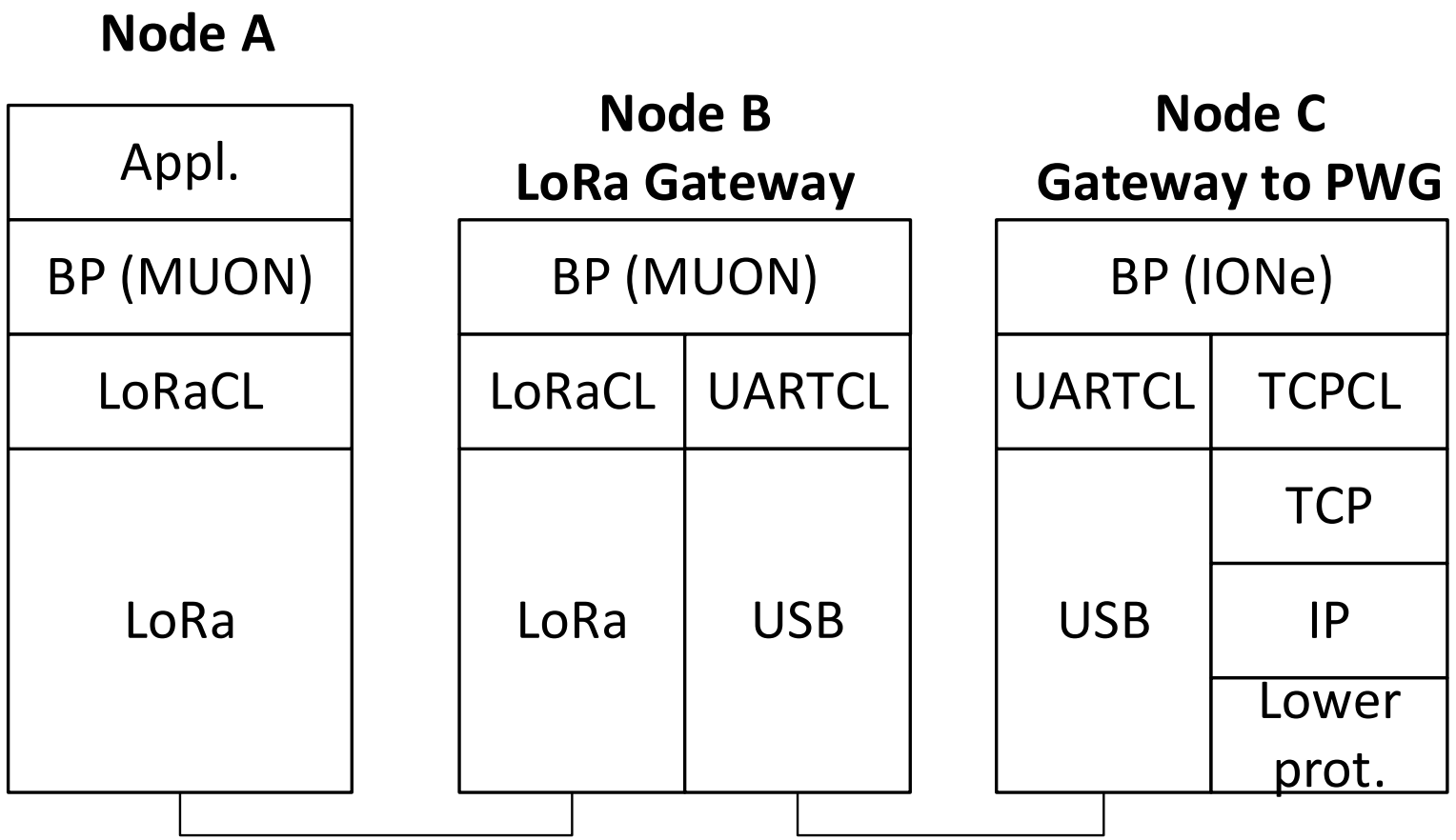
Testbed Setup – NODE B



Testbed Setup – NODE C

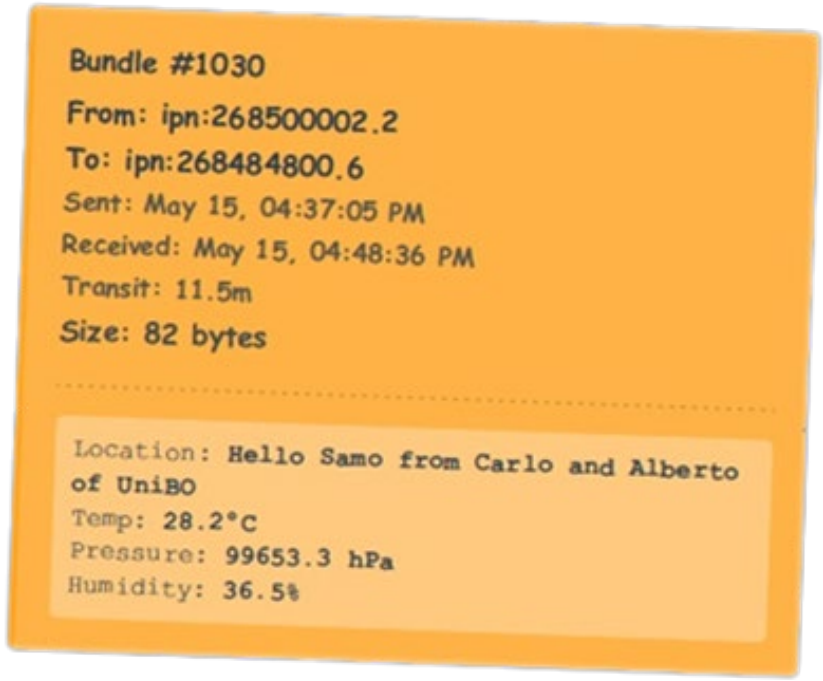


Testbed Setup – Protocol Stacks



Tests

- Sending JSON-encoded sensor data generated on node A via DTN to a "Bundle Board" application, running on a remote PWG node.
- Sending a bundle from node C to node A containing a command for toggling a LED built-in to the microcontroller board.
- Sending longer messages to confirm LoRaCL segmentation of bundles larger than the LoRa payload.



Conclusions

- Our research proved a network of LoRa-linked microcontroller nodes can become a DTN network and that such a network can be successfully interfaced with an ordinary DTN network, as is the IPNSIG PWG.
- The key to transforming microcontrollers into DTN nodes is muON, a lightweight BPv7 implementation specifically designed for this task.
- We focused on its two convergence layers
  - LoRaCL, which allows bundles to be transmitted between LoRa nodes
  - UARTCL-COBS, necessary to transfer bundles between a LoRa node and the PC acting as a gateway towards the ordinary DTN network
- To make integration possible, UARTCL-COBS has also been added to a fork of IONe, an experimental version of the ION package.
- Tests proved the feasibility, as well as the convenience, of integrating LoRa microcontrollers with a large global DTN network.

Future developments

- muON ports to more microcontroller families
- SuperGateway nodes which perform duties of both nodes B and C:
  - Compatible implementations of TCPCL and UDPCL have been completed, are currently in testing and will be released shortly
  - SuperGateway nodes need more hardware features (BT, WiFi,…), thus requiring uC such as those from the ESP32 family
- Further quantitative testing and characterisation of LoRaCL

Thank you for your kind attention!

We are delighted to provide live demonstrations at your request.