

RIOT

[www.riot-os.org](http://www.riot-os.org)  
Emmanuel Baccelli  
*Inria*  
on behalf of the RIOT Community

# RIOT Agenda

- Why?
- How?
- What is RIOT?
  - Solving IoT technical challenge 1: constrained devices
  - Solving IoT technical challenge 2: interoperability
- Current stand, in a nutshell

# RIOT Agenda

- Why?
- How?
- What is RIOT?
  - Solving IoT technical challenge 1: constrained devices
  - Solving IoT technical challenge 2: interoperability
- Current stand, in a nutshell

# Software Platforms Are Crucial

- Recent calls for OS and data "sovereignty"

Snowden political scandal

→ **strategic & privacy incentives**



Android generates billions of dollars  
→ **business incentives**

# IoT = Future Internet's extremity



Advantage: extremely big business

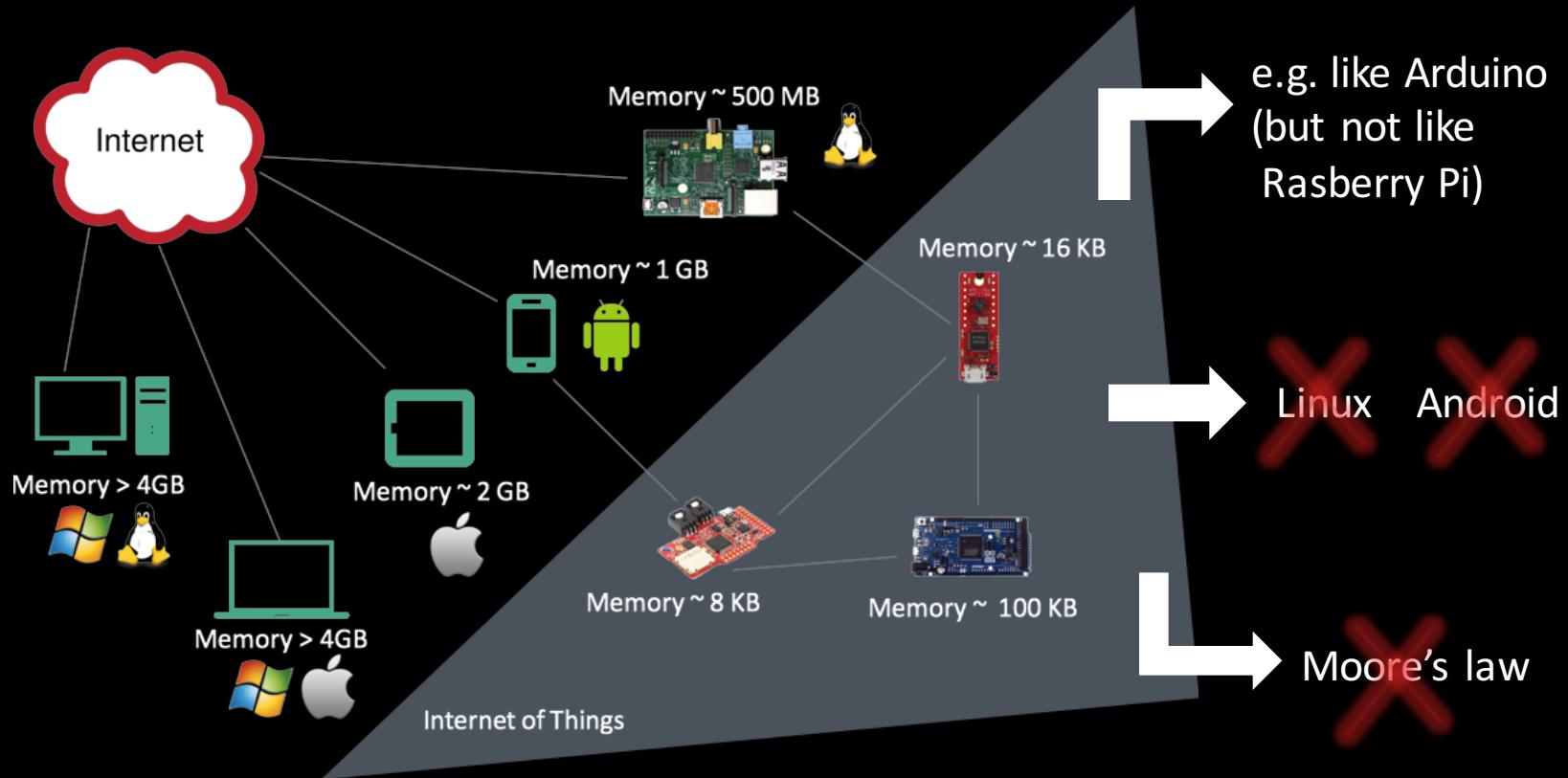
Drawback: extreme challenge for privacy



**Software platforms are  
even more crucial in IoT**

# Software Platforms for IoT?

- Great expectations for IoT, but...  
...no standard platform yet, to program most IoT devices!



# Goals for an IoT software platform

- ✓ long-term IoT **software robustness & security**
- ✓ trust, transparency & **protection of IoT users' privacy**
- ✓ **faster innovation** by spreading IoT software dev. costs
- ✓ **less garbage** with less IoT device lock-down

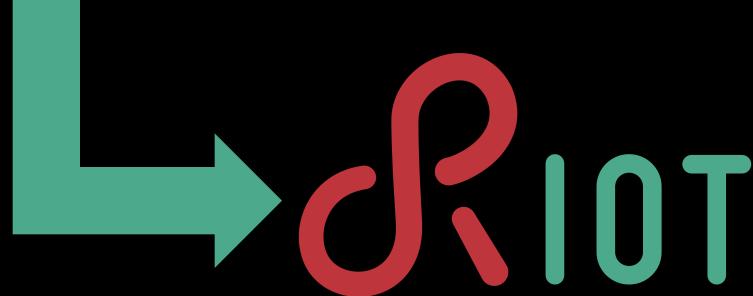
# RIOT Agenda

- Why?
- How?
- What is RIOT?
  - Solving IoT technical challenge 1: constrained devices
  - Solving IoT technical challenge 2: interoperability
- Current stand, in a nutshell

# How can we achieve our goals?

- Experience (e.g. with Linux) shows we are likely to succeed with a platform that is:

- open source
- free
- driven by a grassroots community



# RIOT Principles

- Contributors worldwide
  - People from industry, academia, hobbyists/makers
  - Community self-organizes, follows open processes
- Large-scale distributed source code management



Geopolitical neutrality

# RIOT Principles

- Free & open source software platform
  - core distributed with non-viral copyleft license



(aims for less forks = more coherent code + community)

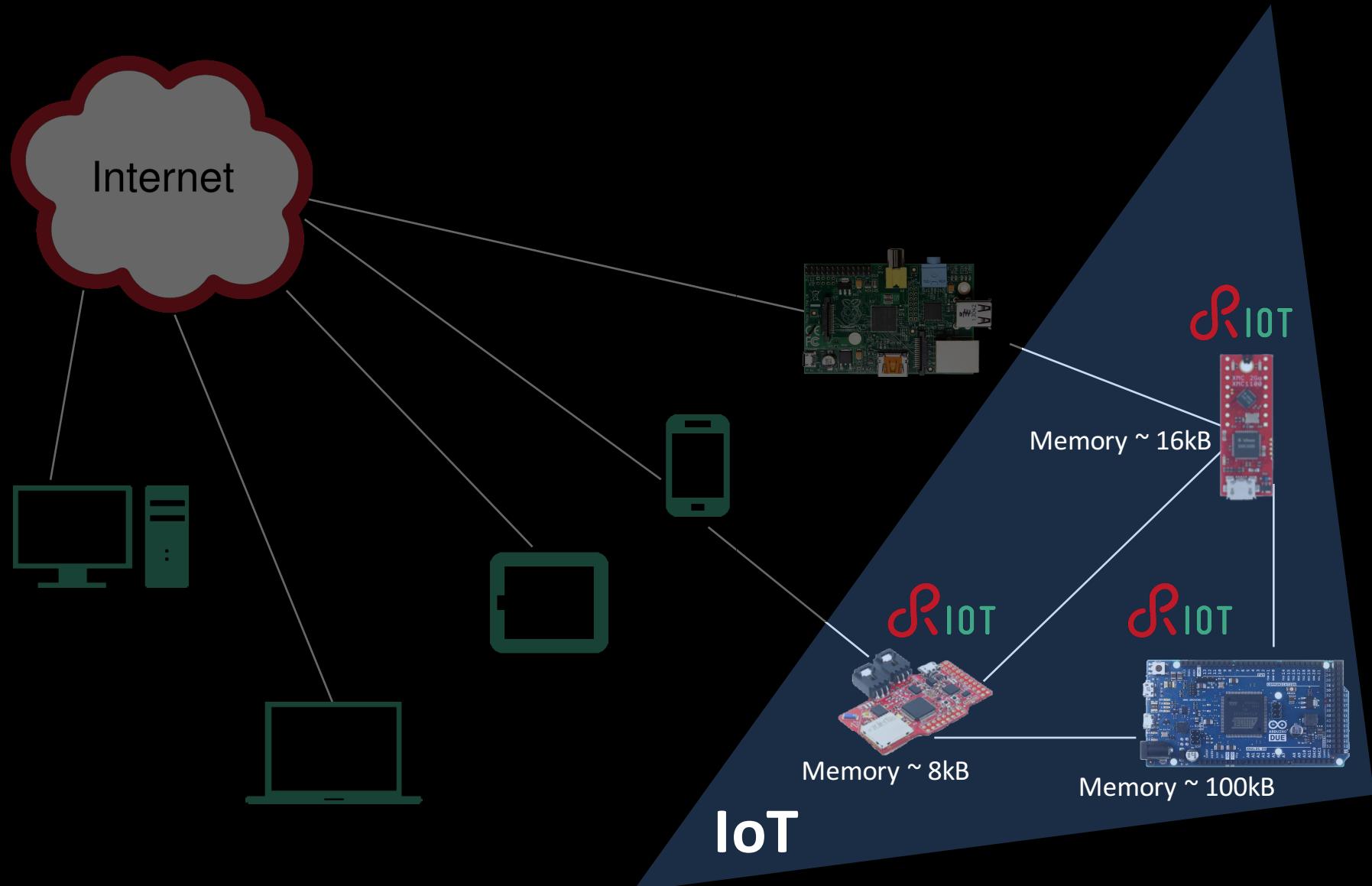
## Indirect business models

(like business with Linux)

# RIOT Agenda

- Why?
- How?
- What is RIOT?
  - Solving IoT technical challenge 1: constrained devices
  - Solving IoT technical challenge 2: interoperability
- Current stand, in a nutshell

# RIOT : an OS that fits IoT devices



# RIOT : an OS that fits IoT devices

- RIOT is the combination of:
  - needed **memory & energy efficiency** to fit IoT devices
  - functionalities of a **full-fledged operating system**
    - Advanced, consistent APIs across 32-bit, 16-bit, 8-bit hardware
    - Full-featured, extensible network stacks

# RIOT Agenda

- Why?
- How?
- What is RIOT?
  - Solving IoT technical challenge 1: constrained devices
  - Solving IoT technical challenge 2: interoperability
- Current stand, in a nutshell

# IoT Challenge 1: Constrained Devices

ENERGY

Milliwatt instead of Watt

CPU

Megahertz instead of Gigahertz

Memory

Kilobytes instead of Gigabytes

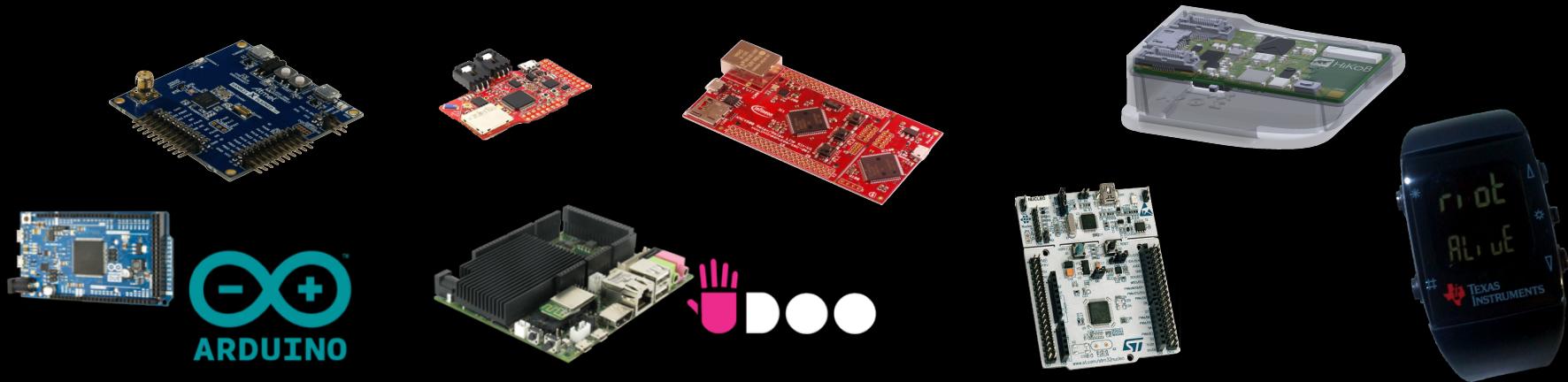


# How RIOT solves Challenge 1

- Micro-kernel architecture (contrary to Linux)  
→ minimal requirements around 1kB RAM
- Tickless scheduler → energy efficiency
- Deterministic O(1) scheduler → real-time
- Low latency interrupt handler → reactivity

# How RIOT solves Challenge 1

- Consistent, powerful API on 8-bit, 16-bit, 32-bit  
→ preemptive multithreading, IPC...

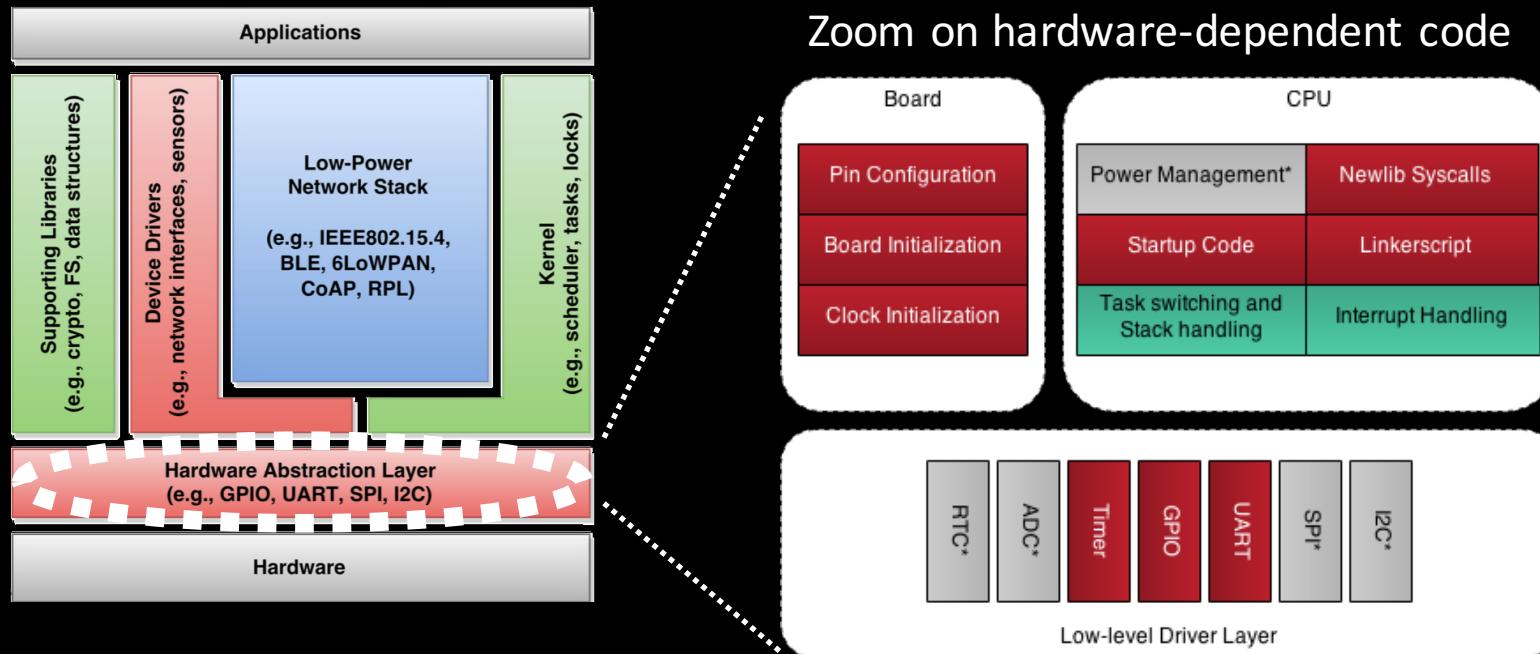


- Modular structure, adaptive to diverse hardware  
→ support for 50+ different IoT boards/devices and counting

MCUs: ARM Cortex M, TI MSP430, AVR... Radios: AT86RF233...

# How RIOT solves Challenge 1

- Efficient HAL: minimized hardware-dependent code



Task Switching, Stack Handling, Interrupt Handling:  
done for ARM Cortex M3, M4 and M0 is on the way

(GPIO, UART, SPI, Timers: done for STM, Atmel, NXP...)

# Well-known tools are usable!

- Compliance with common system standards
  - ✓ POSIX sockets, pthreads
  - ✓ standard C, C++ application coding

→ Much shorter development life-cycles

- ✓ Run & debug as native process in Linux
- ✓ Use of well known debug tools enabled



**GDB**  
The GNU Project  
Debugger

Valgrind

WIRESHARK

# RIOT Agenda

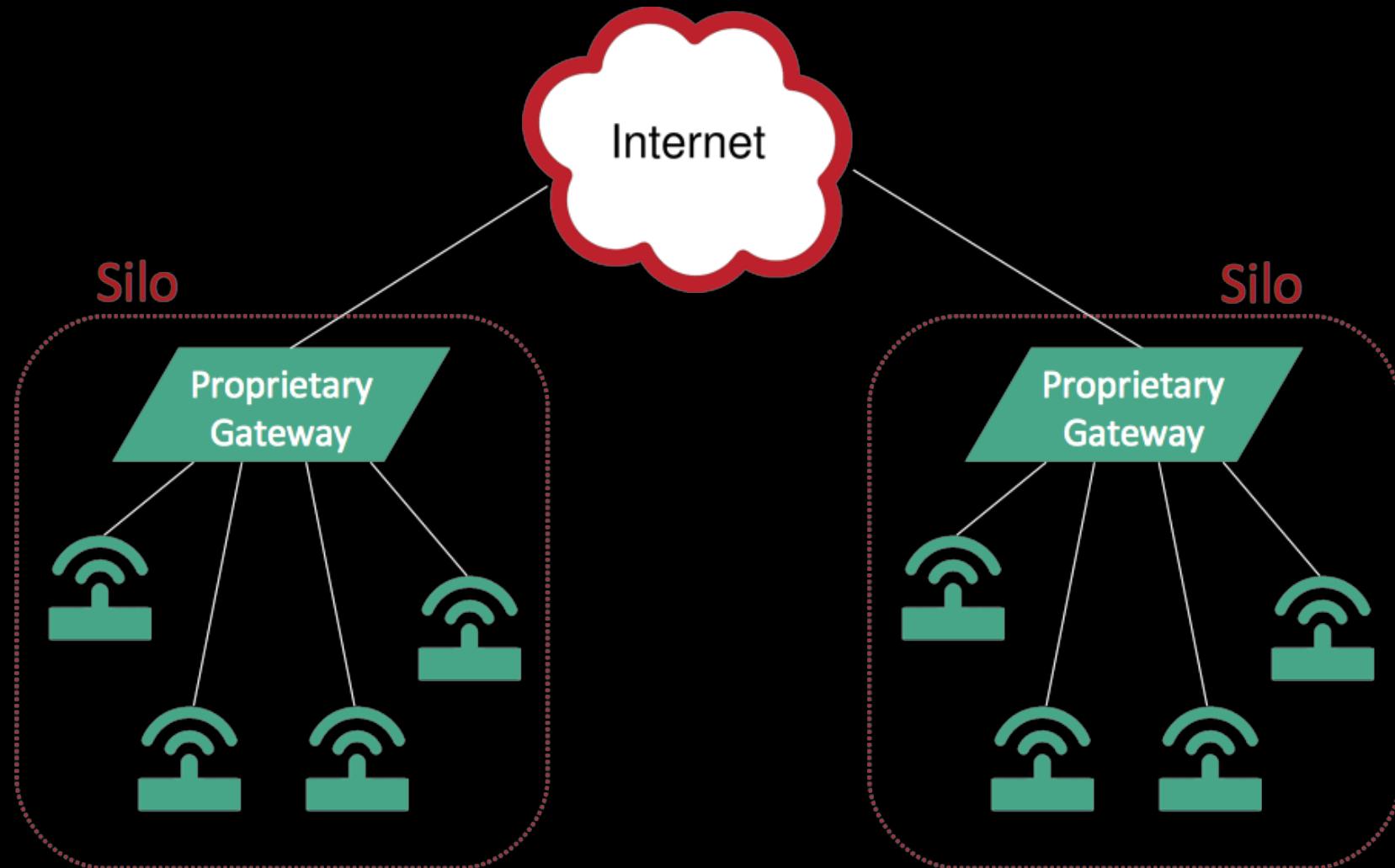
- Why?
- How?
- What is RIOT?
  - Solving IoT technical challenge 1: constrained devices
  - Solving IoT technical challenge 2: interoperability
- Current stand, in a nutshell

# IoT Challenge 2: Interoperability

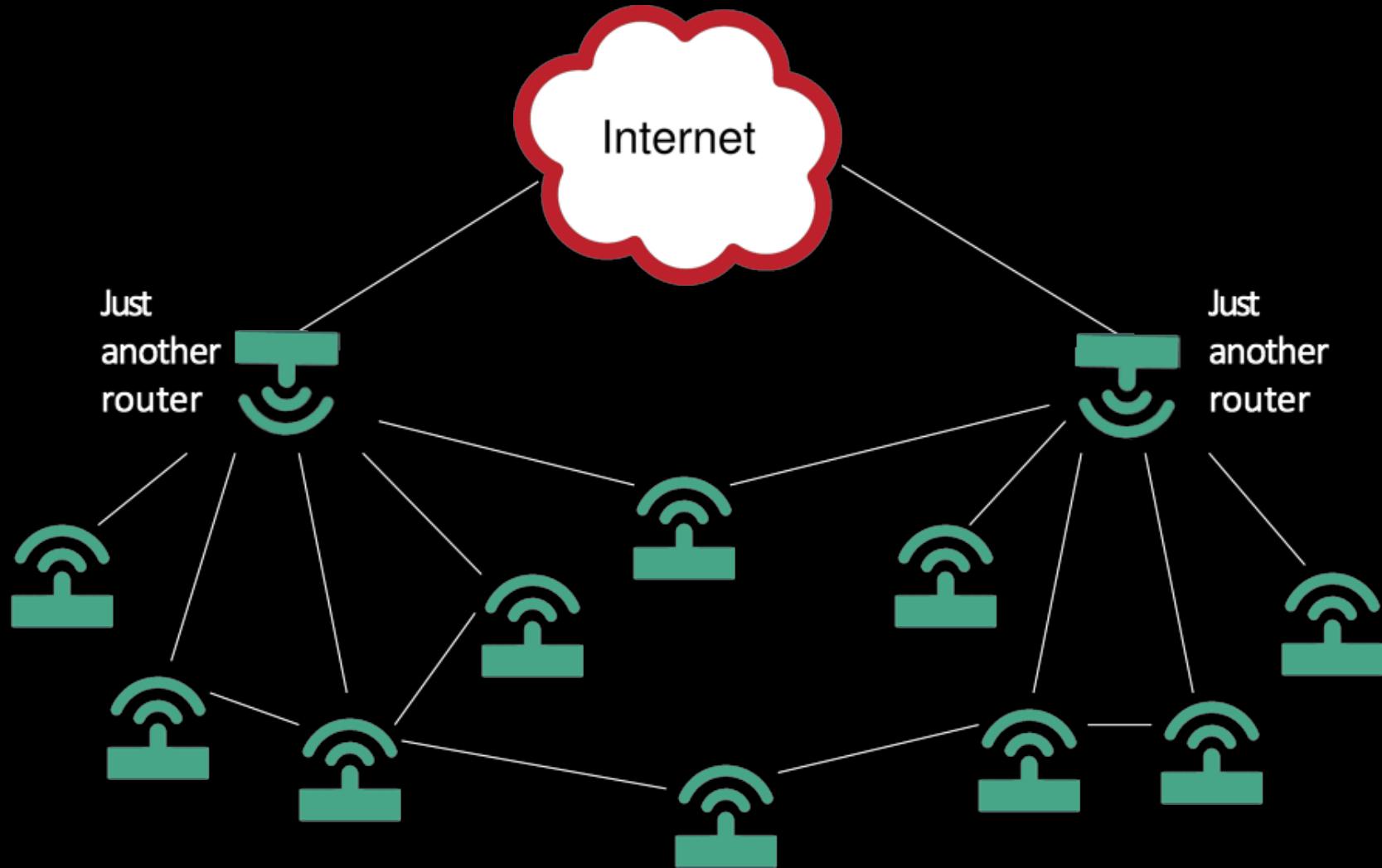
- System-level interoperability
  - Hardware-independent IoT software
  - Usability of third-party, well-known tools
- Network level interoperability
  - End-to-end connectivity per default
  - Device-to-device connectivity

# IoT Interoperability Challenge:

## The IoT today looks mostly like this



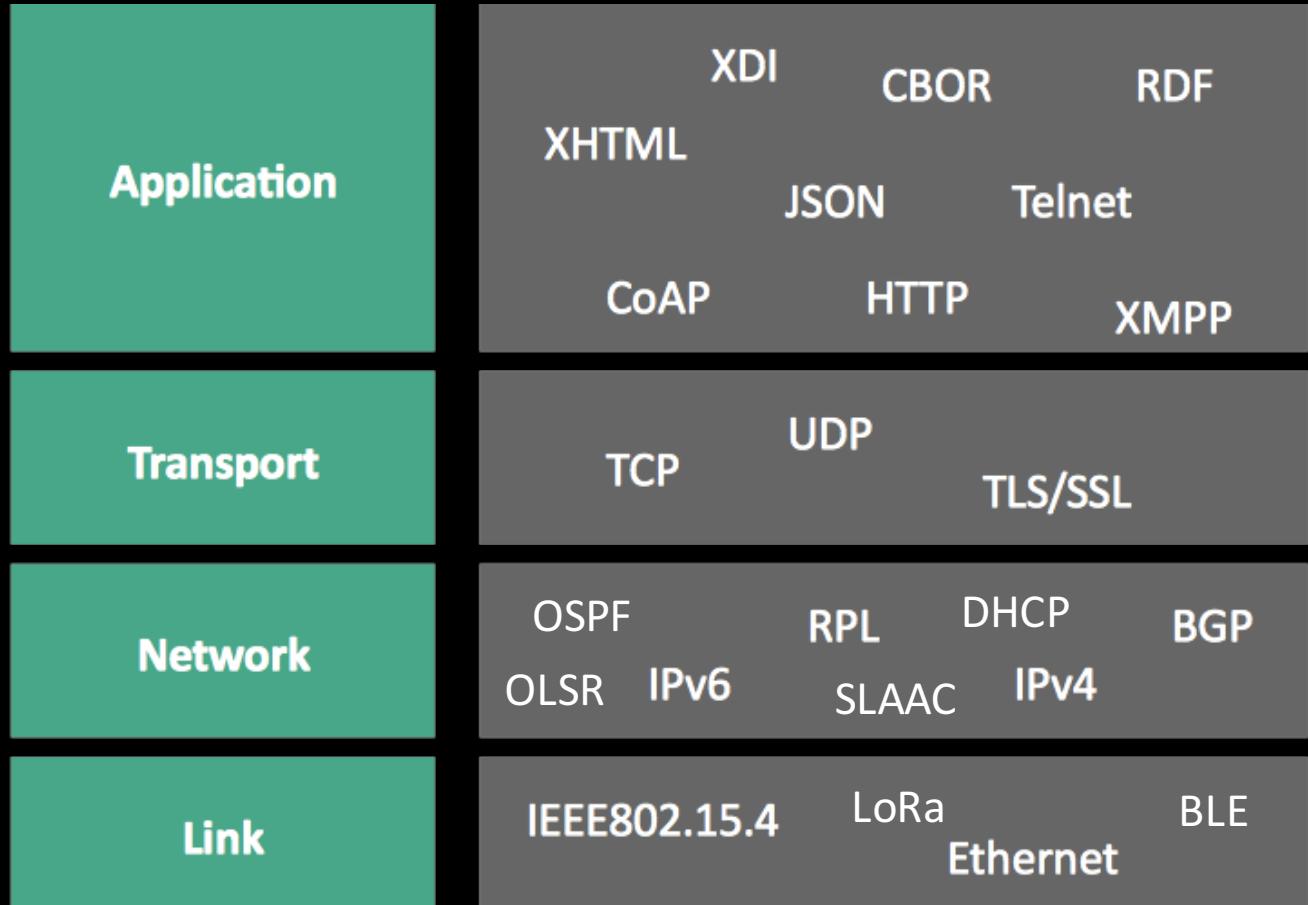
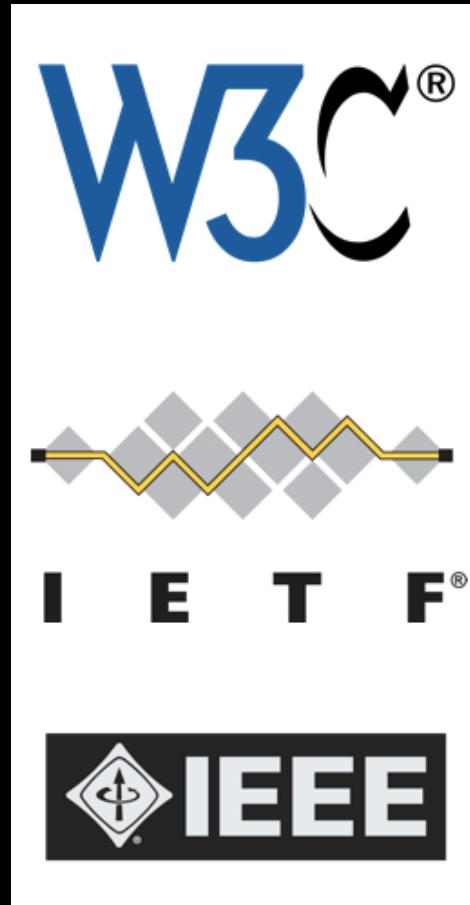
# IoT Interoperability Challenge: The IoT we want looks more like that



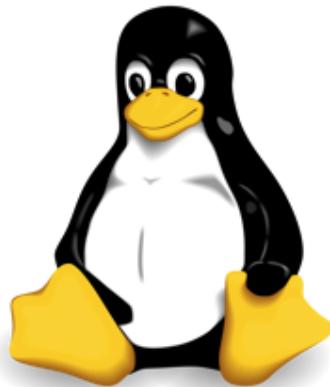
# The IoT we want is... the Internet!



# Internet Interoperability: Based on Open Standards



# Internet Interoperability: Accelerated with Open Source



NEINX



OpenWrt  
Wireless Freedom



PostgreSQL

OpenSSL<sup>TM</sup>  
Cryptography and SSL/TLS Toolkit

# Usual solutions for Interoperability: Challenged by IoT...

... because of resource constraints on IoT devices

- Memory, CPU, energy

... because of low-power communication characteristics

- Lossy / duty cycles
- Super-small frames
- Spontaneous wireless architecture

→ **Adapted standard IoT protocols needed**

# Standard IoT protocols? On the way!

## Work in progress at IETF, IEEE, W3C, OMA...

### New specs for **link layer** technologies

- Low-power radios, PLC, BACnet
- IEEE 802.15.4, Z-Wave, BLE, LoRa (and IEEE 802.11)
- More to come...

### New specs for **network layer** protocols

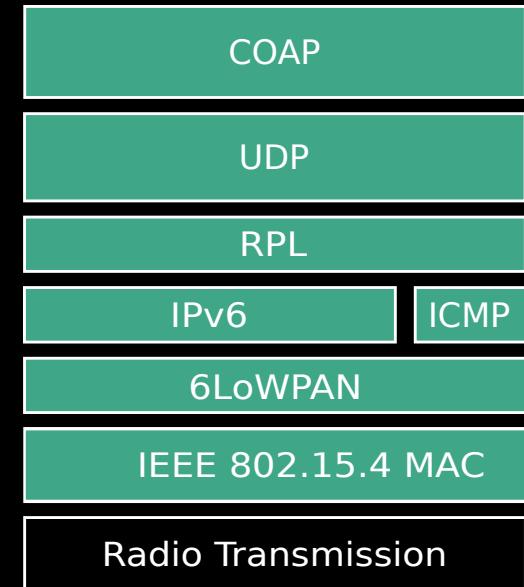
- Fitting IoT requirements and interoperable with IP
- 6TiSCH, 6LoWPAN, RPL, OLSRv2, AODVv2
- More to come...

### New specs for **application layer** protocols

- Fitting IoT requirements and interoperable with web
- CoAP, LwM2M, CBOR
- More to come...

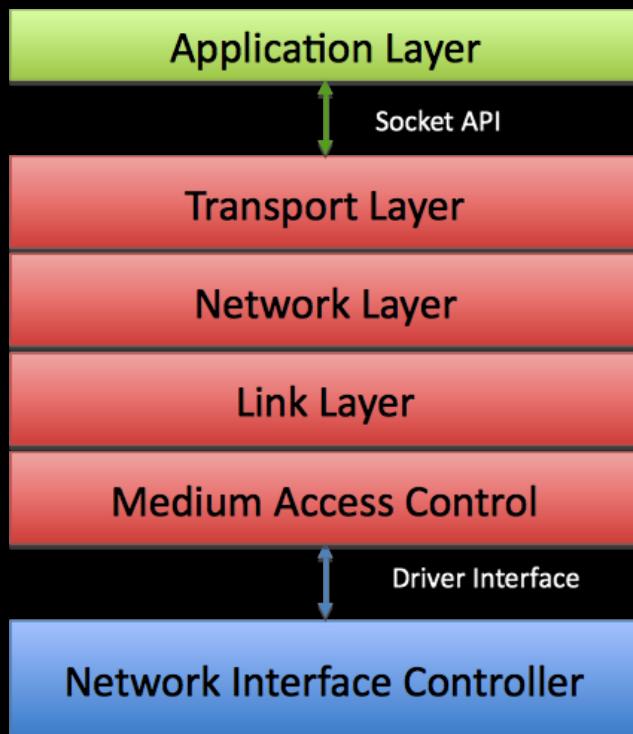
### New network paradigms

- Content-centric networking for IoT
- More to come...

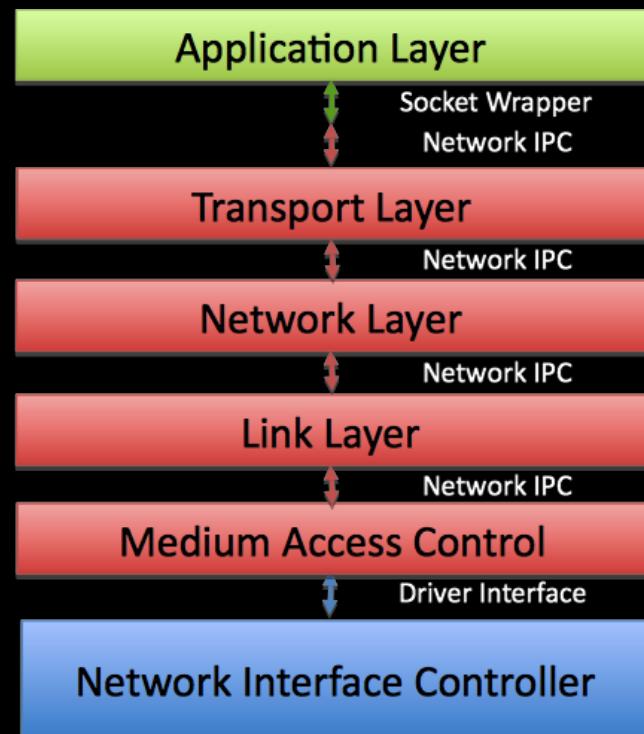


# How RIOT solves Challenge 2

→ Network stack ultra-flexibility and modularity

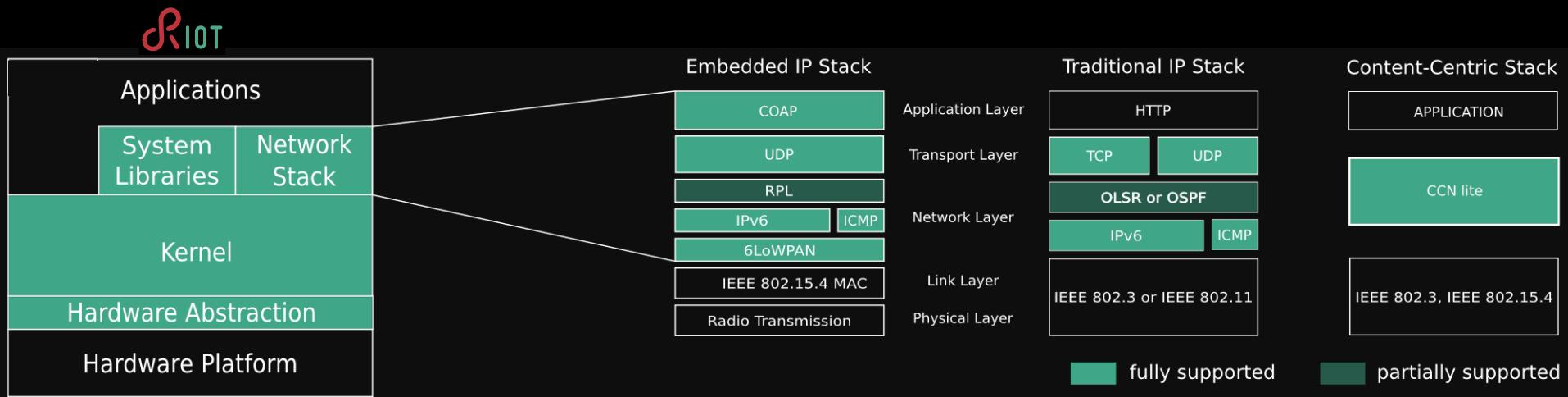


Traditional stack



RIOT stack (GNRC)

# How RIOT solves Challenge 2



- ✓ 6LoWPAN stack, supporting IoT wireless tech.
- ✓ Standard IPv6 stack
- ✓ BSD-like ports for third-party modules/stacks:
  - OpenWSN, CCN-lite, Emb6, lwIP, tinyDTLS...

# RIOT Agenda

- Why?
- How?
- What is RIOT?
  - Solving IoT technical challenge 1: constrained devices
  - Solving IoT technical challenge 2: interoperability
- Current stand, in a nutshell

# RIOT in a nutshell

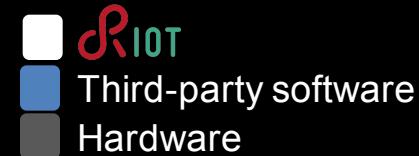
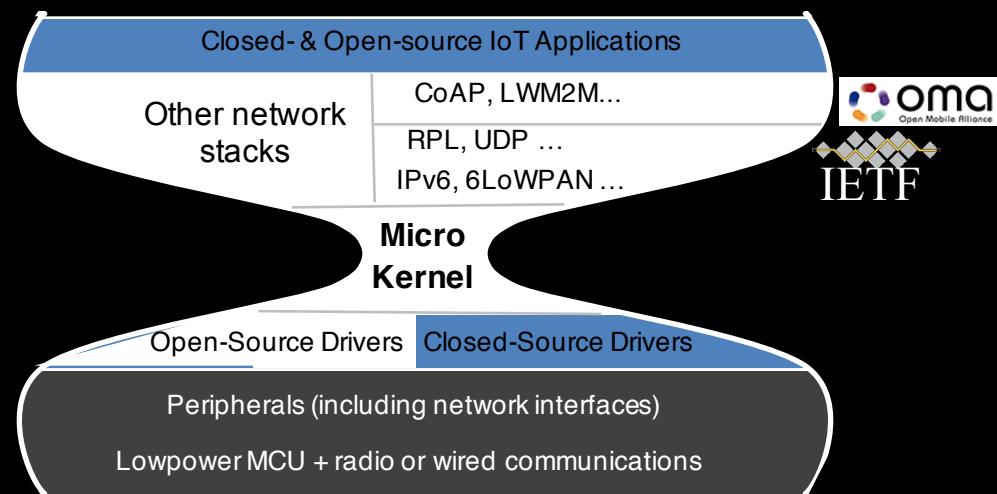
Free, open-source platform for portable IoT software

RIOT offers a platform functionally equivalent to Linux, based on:

open-source,

open-access protocol specs,

community-driven dev.



# RIOT can do more, so RIOT can do less

- Arduino scripts can run as-is on top of RIOT
- mbed applications could run on top of RIOT
- Contiki can run in a single RIOT thread



# RIOT Roots & Evolution

- **2008 – 2012**  
Ancestors of RIOT kernel developed in research projects (FireKernel, uKleos).
- **2013 – 2016**  
Branding of RIOT started, source code moved to Github, major development of the network stack & the OS as such.
- **Speed-evolution of the code-base.**  
110+ contributors.

## Founding institutions

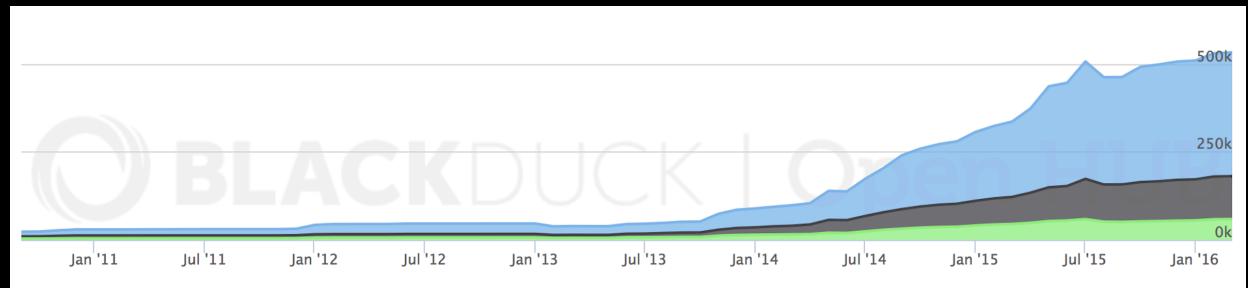
Freie Universität Berlin



**inria**  
INVENTORS FOR THE DIGITAL WORLD



Hochschule für Angewandte  
Wissenschaften Hamburg  
*Hamburg University of Applied Sciences*



# Some supporters/users



... and dozens of independent developers around the world!

# RIOT Summit

## July 15 - 16, 2016

<http://summit.riot.org>

In conjunction with IETF 96 in Berlin



- ✓ bringing together RIOTers, beginners & experts
- ✓ gathering people interested in the IoT in general
- ✓ plenary talks, hands-on tutorials & demos

Call for contributions & sponsors

<http://summit.riot-os.org/category/calls/>

# Thanks for your interest!

News: [https://twitter.com/RIOT\\_OS](https://twitter.com/RIOT_OS)

For cooperation questions: [riot@riot-os.org](mailto:riot@riot-os.org)

For developer questions: [devel@riot-os.org](mailto:devel@riot-os.org)

Support & discussions on IRC: [irc.freenode.org #riot-os](irc://irc.freenode.org/#riot-os)



# Hands-on with RIOT

- ✓ Setup: get RIOT code

```
git clone https://github.com/RIOT-OS/RIOT.git
```

Caveat: You may have to install git and toolchain (things like gcc ...)

See <https://github.com/RIOT-OS/RIOT/wiki/Introduction>

- ✓ Hands-on: compile, flash, run RIOT

Demo: RIOT shell on a Libelium WaspMote (8bit AVR)

- ✓ Hands-on: communication with RIOT (IPv6, 6LoWPAN)

Demo: PING with 2 Atmel SAMR21 boards (32bit ARM M0)