

RIOT - 6TiSCH and ICN

IETF94 - Hackathon

Thomas Eichinger and Oliver "Oleg" Hahm

October 31, 2015



Agenda

- 1 Start the RIOT
- 2 Topic: 6TiSCH
- 3 Topic: ICN

Agenda

- 1 Start the RIOT
- 2 Topic: 6TiSCH
- 3 Topic: ICN

An Operating System for the IoT

"If your IoT device cannot run Linux, then use RIOT!"

- RIOT requires only a few kB of RAM/ROM, and a small CPU
- With RIOT, code once & run heterogeneous IoT hardware
 - 8bit hardware (e.g. Arduino)
 - 16bit hardware (e.g. MSP430)
 - 32bit hardware (e.g. ARM Cortex-M, x86)



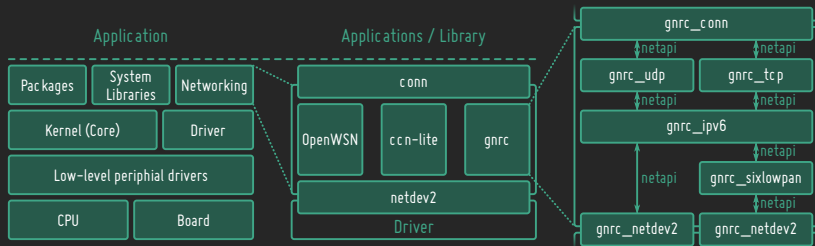
RIOT in a Nutshell

- Microkernel architecture (for **robustness**)
- The kernel itself uses 1.5K RAM @ 32-bit
- Tickless scheduler (for **energy efficiency**)
- Deterministic $O(1)$ scheduling (for **real-time**)
- Low latency interrupt handling (for **reactivity**)
- Modular structure (for **adaptivity**)
- Preemptive multi-threading & powerful IPC

Open Standards, Open Source

- Free, open source (LGPLv2.1) operating system for constrained IoT devices
- Write your code in **ANSI-C** or **C++**
- Compliant with the most widely used POSIX features like pthreads and sockets
- No IoT hardware needed for development
- Run & debug RIOT as native process in Linux

The Structure



Get in touch!

- News: https://twitter.com/RIOT_OS
- For developers: questions: devel@riot-os.org
- For cooperation: questions: riot@riot-os.org
- Support & discussions on IRC: irc.freenode.org, [#riot-os](https://irc.freenode.org/#riot-os)

Agenda

- 1 Start the RIOT
- 2 Topic: 6TiSCH
- 3 Topic: ICN

Why using 6TiSCH in IoT Scenarios?

- IEEE802.15.4e: amendment to the Medium Access Control (MAC) portion, TSCH mode.
- significantly increases robustness against external interference and persistent multi-path fading, while running on legacy IEEE802.15.4 hardware.
- 6TiSCH: IPv6-enabled protocol stack on top of IEEE802.15.4e TSCH

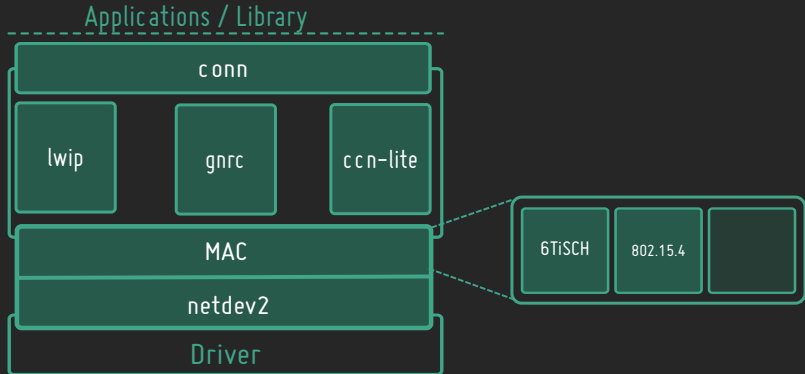
OpenWSN

"The OpenWSN project is an open-source implementation of the network protocol stack based on IETF standards and IEEE 802.15.4e, that can function on a variety of hardware and software platforms."

- OpenWSN stack = reference open source implementation of 6TiSCH protocols
- RIOT already supports OpenWSN stack, as an external package
- But OpenWSN support could be improved
 - only limited interaction between OpenWSN and RIOT threads
 - reduce code duplication



Concept for 6TiSCH integration



Goals and Challenges

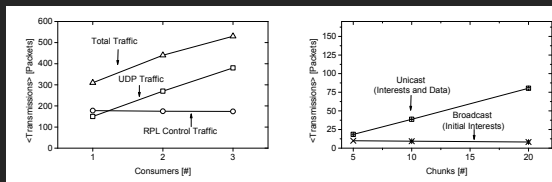
- Goals
 - RIOT 6TiSCH enabled node joining OpenWSN node's TSCH schedule and vice versa
 - Pinging between OpenWSN node and RIOT 6TiSCH enabled node
- Challenges
 - Get to know RIOT's netdev2 and netapi API
 - What's needed from OpenWSN stack to comply with 6TiSCH-minimal?
 - Comply with 6TiSCH's strict timing constraints

Agenda

- 1 Start the RIOT
- 2 Topic: 6TiSCH
- 3 Topic: ICN

Why using ICN in IoT Scenarios?

- Less complexity and smaller memory footprint in comparison to a full IPv6/6LoWPAN stack
- Increased dependability and decreased energy consumption by in-network caching



Module	ROM	RAM
RPL + 6LoWPAN	53412 bytes	27739 bytes
CCN-Lite	16628 bytes	5112 bytes

E. Baccelli, C. Mehlis, O. Hahm, T.C. Schmidt, M. Wählisch: Information Centric Networking in the IoT: Experiments with NDN in the Wild, (ACM ICN 2014), September 2014, Paris, France.

ccn-lite - Lightweight CCN Implementation

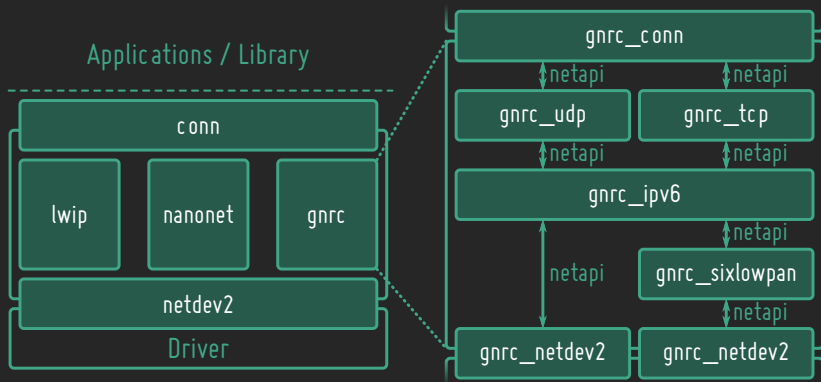
"CCN-lite is a lightweight (no bells & whistles), and functionally interoperable implementation of the Content Centric Networking protocol CCNx of XEROX PARC."

- tiny code base: core has less than 2.000 LoC, pure C, runs over UDP and raw Ethernet
- multiple Platforms: the same code runs in (UNIX) user space, Linux kernel, OMNeT++, Android, Arduino, and Docker
- multiple packet formats: ccnb, NDN, CCNx1.0, IoT-TLV, Cisco-TLV

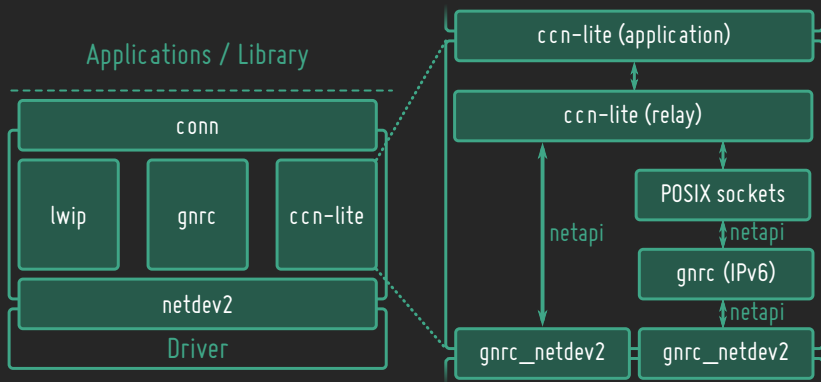


www.ccn-lite.net

GNRC stack



Concept for ccn-lite integration



Goals and Challenges

- Goals
 - CCN over L2
 - CCN over netapi
- Challenges
 - Get to know RIOT's netapi
 - Deal with different address types (e.g. IPv6, Ethernet 48bit addresses, EUI64)
 - Come up with a generic solution for different transports
- Other Todos
 - make ccn-lite configurable via the shell
 - use a static memory pool for ccnl_malloc and friends
 - use RIOT's xtimer for ccn-lite timers
 - implement select for BSD sockets in RIOT



www.riot-os.org