• Why?
• How?
• What is RIOT?
Why a software platform for IoT?

• Linux, Android... bare-metal?

• But as IoT software evolves...
  ▪ more complex pieces, e.g. an IP network stack
  ▪ evolution of application logic

• ... non-portable IoT software slows innovation

  90% of IoT soft. should be hardware-independent

  → this is achievable with a good software platform
  (but not if you develop bare-metal)
Goals for an IoT software platform

- **Faster innovation** by spreading IoT software dev. costs
- **Long-term IoT software robustness & security**
- **Trust, transparency & protection of IoT users’ privacy**
- **Less garbage** with less IoT device lock-down
• Why?

• How?

• What is RIOT?
How to achieve our goals?

- Experience (e.g. with Linux) points towards:
  - open source
  - free core
  - driven by a grassroots community

Indirect business models
Geopolitical neutrality
• Why?
• How?
• What is RIOT?
RIOT: an OS that fits IoT devices (if you can’t use Linux!)

Internet

Internet

IoT

Memory ~ 16kB

Memory ~ 8kB

Memory ~ 100kB
RIOT: an OS that fits IoT devices

- RIOT is the combination of:
  - memory & energy efficient design 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
  ✓ Well-known dev. tools, standard C and C++ programming
  ✓ Easy integration of third-party software packages
RIOT offers a platform functionally equivalent to Linux, based on:
- open-source open-access protocol specs
- community-driven development

RIOT = a platform for portable IoT software based on:
- open-source
- open-access protocol specs
- community-driven development

RIOT in one slide

Closed- & Open-source IoT Applications
- CoAP, CBOR, GNRC
- RPL, UDP... IPv6, 6LoWPAN...

Other network stacks
- GNRC

Micro Kernel
- Open-Source Drivers
- Closed-Source Drivers

Peripherals (including network interfaces)
- Lowpower MCU + radio or wired communications

Third-party software

Hardware
Hardware Independent IoT Code

With a simple application over a IPv6/6LoWPAN stack in RIOT, 95% of the code is hardware-independent and/or reusable (and application code is completely portable).
Third-party IoT code & tools

- Packages (similar to BSD ports) for third-party open source code
  - Use code not initially developed for RIOT
  - Use code not even initially developed for IoT!

<table>
<thead>
<tr>
<th>Package</th>
<th>Overall Diff Size</th>
<th>Relative Diff Size</th>
</tr>
</thead>
<tbody>
<tr>
<td>libcoap</td>
<td>639 lines</td>
<td>6.3 %</td>
</tr>
<tr>
<td>libfixmath</td>
<td>34 lines</td>
<td>0.2 %</td>
</tr>
<tr>
<td>lwip</td>
<td>767 lines</td>
<td>1.3 %</td>
</tr>
<tr>
<td>micro-ecc</td>
<td>14 lines</td>
<td>0.8 %</td>
</tr>
<tr>
<td>relic</td>
<td>24 lines</td>
<td>&lt;0.1 %</td>
</tr>
</tbody>
</table>

- Interoperates with common systems standards
  - Run & debug as native process in Linux
  - Use of well-known debug tools enabled
  - Shorter development life-cycles

Only small porting effort needed! (negligible % LoC)
• 115+ contributors from all around the world
• Contributions from industry academia, makers/tinkerers

• 60+ boards: various CPU architectures, radios, sensors...