To Boldly Go Where Linux Cannot
With Zephyr and Eclipse IoT

Frédéric Desbiens
Program Manager, IoT and Edge Computing

August 2019
Characteristics of an IoT Solution

- **Long lifespan**
  Spans multiple years, if not decades

- **Heterogenous**
  Nobody can deliver an end-to-end solution alone

- **Constraints**
  Power, compute, environmental and many others

- **Connectivity**
  Connectivity is a given, but stability and reliability are not
## Will it (Linux) fit?

<table>
<thead>
<tr>
<th></th>
<th>Adafruit Feather nRF52</th>
<th>Reel Board</th>
<th>BBC micro:bit</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>CPU</strong></td>
<td>Cortex-M4F @ 64 MHz</td>
<td>Cortex-M4 @ 64 MHz</td>
<td>Cortex-M0 @ 16 MHz</td>
</tr>
<tr>
<td><strong>RAM</strong></td>
<td>64 KB</td>
<td>256 KB</td>
<td>16 KB</td>
</tr>
<tr>
<td><strong>Storage</strong></td>
<td>512 KB</td>
<td>1024 KB</td>
<td>256 KB</td>
</tr>
</tbody>
</table>
Constrained devices

The top three CPU architectures for constrained devices used by respondents are ARM-based, with significant use of niche 8-bit, 16-bit and 32-bit MCUs.
Top device operating systems

- FreeRTOS
- Contiki/Contiki-NG
- MBed OS
- RIOT OS
- QNX

Other standouts (75%+) include:
- Contiki/Contiki-NG

dominates constrained devices (along with its Amazon derivation)
Non-Linux operating systems over time
How to pick an OS/RTOS?

> Functional
  • Hardware support
  • Connectivity
  • Power supply
  • Secure boot and device authentication

> Non-functional
  • Lock-in to upstream or vendor’s Cloud
  • Licensing and IP
  • Security updates and process (CVEs)
  • Safety Certification
  • Open source: number of contributors
The Zephyr RTOS

- Modular and configurable
- Cooperative and pre-emptive threading
- Integrated device driver interface
- Memory protection
  - Stack overflow
  - Kernel object and device driver permission tracking
  - Thread isolation
- Bluetooth® Low Energy
  - Controller and host
  - Mesh
- Native networking stack
The Zephyr project

- Open Source
- Permissive license (Apache 2.0)
- Vendor Neutral Governance
- Long Term Support (LTS) branch
- Ready for Security Certification
The Eclipse Foundation - By the Numbers

370+ Projects

275+ Members

1550+ Committers

195M+ Lines of Code

30 Staff Members

10+ Working Groups
We provide a collaborative environment for the world’s leading Java ecosystem players to advance open source enterprise Java technologies for the cloud.

We enable industry leaders to collaborate on an end-to-end IoT architecture that is secure, flexible, and fully based on open source and open standards.

We provide leading automotive OEMs, their suppliers, and partners with a sustainable, transparent, and vendor-neutral platform to collaborate on open technologies and standards.

The Eclipse IDE is the critical development environment for more than 4 million active users. Our community is innovating on the next generation of cloud native developer tools.
Eclipse IoT Community

3.9M
lines of code

38
projects

350+
contributors

40
member companies
## Protocols & Standards

<table>
<thead>
<tr>
<th>Protocol or standard</th>
<th>Projects</th>
</tr>
</thead>
<tbody>
<tr>
<td>MQTT</td>
<td>Paho, Mosquitto</td>
</tr>
<tr>
<td></td>
<td>Tahu</td>
</tr>
<tr>
<td></td>
<td>Californium</td>
</tr>
<tr>
<td></td>
<td>Wakaama, Leshan</td>
</tr>
<tr>
<td></td>
<td>Cyclone</td>
</tr>
<tr>
<td></td>
<td>TinyDTLS</td>
</tr>
<tr>
<td></td>
<td>Unide</td>
</tr>
<tr>
<td></td>
<td>ThingWeb</td>
</tr>
<tr>
<td></td>
<td>OM2M</td>
</tr>
<tr>
<td></td>
<td>Milo</td>
</tr>
<tr>
<td>Sparkplug</td>
<td></td>
</tr>
<tr>
<td>CoAP</td>
<td></td>
</tr>
<tr>
<td>LWM2M</td>
<td></td>
</tr>
<tr>
<td>DDS</td>
<td></td>
</tr>
<tr>
<td>DTLS</td>
<td></td>
</tr>
<tr>
<td>PPMP</td>
<td></td>
</tr>
<tr>
<td>W3C Web of Things</td>
<td></td>
</tr>
<tr>
<td>oneM2M</td>
<td></td>
</tr>
<tr>
<td>OPC-UA</td>
<td></td>
</tr>
</tbody>
</table>
IoT Working Group Member Organizations

Strategic members

BOSCH
Invented for life

EUROTECH

Red Hat
Reading data from a I2C sensor

Setup pinmuxes
- Set Tristate to 0
- Set GPIO 14 to input
- Set GPIO 165 to input
- Set GPIO 236 to output & disable
- Set GPIO 237 to output & disable
- Set Tristate to 1
- Change pinmode for pin 28
- Change pinmode for pin 27
- Set GPIO 212 to input
- Set GPIO 213 to input

Sensor interaction
- Define smbus data structs
- Fill in values
- Ioctl write configure sensor
- Allocate read buffer
- Ioctl read

~ 200 loc

A better way
- Call sensor constructor
- Call read function

~ 5 loc
Eclipse MRAA and Eclipse UPM

- In the previous example
  - MRAA handles pinmuxing and memory allocations
  - UPM handles the sensor

- **MRAA**
  - Standard IO interface for IoT hardware
  - Abstracts GPIO, UART, Analog (AIO), 1-Wire, PWM, Firmata, SPI, IIO, \(^{2}\text{C}\) and LED APIs

- **UPM**
  - Standardized sensor and actuator APIs.
  - Light, Pressure, Humidity, Temperature...

- Both written in C/C++ and support multiple operating systems (Zephyr, Linux) and CPU architectures (x86, ARM, MIPS)

- Java, Javascript and Python bindings
> Suppose you have:
  - A factory with over a thousand machines fitted with sensors...
  - A digital building with thousands of sensors over multiple floors...
  - A fleet of hundreds of public transit vehicles with several sensors and cameras

> How do you manage those IoT devices?
Eclipse Wakaama and Eclipse Leshan

- Implementations of the Open Mobile Alliance's Lightweight M2M (OMA LWM2M)
  - On the top of CoAP
  - Supports UDP and SMS for the transport layer
  - Simple Object based resource model
  - Transport layer security based on DTLS
  - Basic M2M functionalities: LWM2M Server, Access Control, Device, Connectivity, Firmware Update, Location, Connectivity Statistics

- Eclipse Wakaama
  - Written in C; code part of your application
  - Leverages Eclipse tinydtls

- Eclipse Leshan
  - Modular Java libraries
  - Based on Eclipse Californium (CoAP) and its Scandium sub-project (DTLS implementation)
Eclipse hawkBit

> A domain independent back-end framework for rolling out software updates to constrained edge devices as well as more powerful controllers and gateways connected to IP based networking infrastructure.

> Used in several large scale software as a service platforms.

> Can leverage LWM2M
IoT Functional Concerns

- **SECURITY**
- **MODELS**
- **TOOLS**

**Communication**
- Field protocols
- IoT protocols

**Hardware Abstraction Layer (HAL)**

**OS / RTOS**

**Remote Management**

**Connecting**
- Data Management & Messaging
- Connectivity
- Network Management
- Application Runtime
- OS / RTOS

**Remote Management**

**Connection**
- Message Routing

**Application Enablement**
- Analytics
- Data Management
- Device Management
- Device Registry
- OS / PaaS
Where Eclipse IoT Projects Fit

**CONSTRAINED DEVICES**
- Hardware Abstraction Layer (HAL)
- Communication
  - Field protocols: LoRa, NB-IoT, Zigbee...
- OS / RTOS
  - FreeRTOS, Zephyr...

**EDGE NODES / GATEWAYS**
- Application Runtime
  - Java, Jakarta EE, Node.js...
- Data Management & Messaging
  - TAHU, ioFog
- Connectivity
  - Field protocols
    - IoT protocols: LoRa, NB-IoT, Zigbee...
  - Network Management
    - IoT protocols
- OS / RTOS
  - Linux, Windows...

**IOT CLOUD PLATFORM**
- Application Enablement
  - ditto
- Analytics
- Data Management
- Device Management
  - Device Registry
  - Device Enablement
- OS / PaaS
  - Eclipse Che
  - Mita

**TOOLS**
- MODELS
  - Vorto
  - unide
- SECURITY
- MODELS
- unide
- SECURITY

COPYRIGHT © 2019, ECLIPSE FOUNDATION, INC. | MADE AVAILABLE UNDER THE ECLIPSE PUBLIC LICENSE 2.0 (EPL-2.0)
> Learn about our projects by visiting iot.eclipse.org/projects

> Try our technology

> Subscribe to the Eclipse IoT newsletter

> Follow and engage with us on social media: @EclipselIoT

> Attend an Eclipse community event or join our Virtual IoT Meetup
  - Eclipse Con Europe 2019
    Ludwigsburg, Germany - October 21 - 24, 2019
Thank you!