LF Real-Time Operating Systems: Evolution and Trends

Kate Stewart. Senior Director of Strategic Programs
@_kate_stewart
Linux Foundation Real-Time OS Projects: Real-Time Linux & Zephyr

Both launched around the same time - 2015Q4/2016Q1
The Linux Operating System Has Grown into the Most Important Software Platform in the World

Every market Linux has entered it eventually dominates
Why do we care about Real-Time?

Real-Time Linux is primarily being used in/by:

▪ Financial institutions, brokerages, stock exchanges
▪ Audio/Video devices
▪ Aviation, flight simulators, medical, military
▪ Control systems, manufacturing
▪ Some telecom applications

Emerging use in IoT, automotive and safety-critical products
What is Real-Time Linux?

The standard Linux kernel only meets soft real-time requirements.

▪ Provides basic POSIX operations for userspace time handling but has no guarantees for hard timing deadlines
▪ Ingo Molnar's Real-Time Preemption patch and Thomas Gleixner's generic clock event layer with high resolution support (together referred to as the PREEMPT-RT patch), the kernel gains hard Real-Time capabilities.

The PREEMPT-RT patch is in use throughout many industries.

▪ Its aim towards mainline integration made it an interesting option for hard and firm Real-Time applications from professional audio to industrial control.
▪ PREEMPT-RT work has been going on for ~15 years and is very comparable to the best Real-Time Operating Systems
Goals of Real-Time Linux Project

- Announced 2015Q4, development started 2016Q1.
- Full mainline integration of PREEMPT-RT
- Sustaining patch maintenance and development
- Foster the Real-Time developer community
- Improve Testing infrastructure
A bit of history...

- 1991: First version of Linux published
- 1994: Academic discussions on Real-Time Linux start
- 2004: LKML discussions on Real-Time Linux start
- 2006: First production version of PREEMPT-RT released.
- 2012: Sustainability of PREEMPT-RT concerns raised.
- 2014: PREEMPT-RT went into hobbyist mode...
- 2015: OSADL funded basic maintenance... but maintenance was not enough.
- 2016: Real-Time Linux project started to work on upstreaming of PREEMPT-RT.
2017 Linux Kernel Development Report

- 4,300 Contributors From 450 Organizations
- 10,000 Lines of Code Added Daily
- 2,000 Lines of Code Modified Daily
- 2,500 Lines of Code Removed Daily
- 8.5 Changes Per Hour
Upstreaming: catching the moving target

2016 → 2018Q4: ~1000 patches merged upstream
- Rework of CPU hotplug
- Rework of the timer wheel
- Preparatory work all over the place

2019: finish upstreaming
- Printk series
- FPU rework
- Memory management swap patches
- Soft IRQs
- Bit Spinlocks
Initial patches for released kernel version created based on what is accepted upstream. Status today:

4.16: Handed off to stable
   ▪ EOL by now

4.19: Version of patches released
   ▪ Switched over from working on 4.18
# Community Now Supports Stable Versions

<table>
<thead>
<tr>
<th>Developer</th>
<th>Current Focus Areas</th>
</tr>
</thead>
</table>
| Steven (VMware)        | Maintenance: **Linux v4.14-rt** & fixing regressions  
                         Working with Julia on a process to coordinate backports among different kernel releases.                                                  |
| Julia (NI)             | Maintenance: **Linux v4.9-rt** & fixing regressions                                                                                                 |
| Daniel (Siemens)       | Maintenance: **Linux v4.4-rt** & fixing regressions. Come up with more generic tools.                                                                 |
| Tom (Intel)            | Maintenance: **Linux v3.18-rt** & fixing regressions.                                                                                               |
| Mark (Linaro)          | rt-stable trees are now being tested in kernelci.org                                                                                                  |
Continuous Integration sources for Real-Time publicly available:

- https://github.com/ci-rt/test-description
- https://github.com/ci-rt/jenkins-cirt-libs

And yes, there is documentation:

- overview about all ci-rt repos
- setup documentation
After Upstream? Stability & Sharing Knowledge

Immediate focus:

- Ensuring the stability of RT in the upstream kernel
- Addressing the side effects of the RT merge and helping upstream developers to understand the limitations and scope
- Addressing documentation gaps
- Enhancing and running the test infrastructure

Additional longer term focus is needed on developing required training so the knowledge is spread out and continuous upstream work can be guaranteed.
Who to thank...

- Thomas and developers working on this effort!
- Members of Real-Time Linux Project who fund the project:
To learn more:

- Real-Time Project Home page: https://wiki.linuxfoundation.org/realtime/start

- Real-Time Linux Microconference at LPC (Nov. 2018)
  - etherpad notes from session
  - Videos of discussions
But what happens when Linux is “too big”?

Emerging IoT applications calling for similar hard and firm Real-Time guarantees.

Sensors as well as other power & size constrained devices need Real-Time as well.
IoT RTOS Landscape
Zephyr Project:

- Open source real time operating system
- Vibrant Community participation
- Built with safety and security in mind
- Cross-architecture with growing developer tool support
- Vendor Neutral governance
- Permissively licensed - Apache 2.0
- Complete, fully integrated, highly configurable, modular for flexibility, better than roll-your-own
- Product development ready with LTS
- Certification ready with Auditable

Open Source, RTOS, Connected, Embedded Fits where Linux is too big
Architecture

- Highly Configurable, Highly Modular
- Cooperative and Pre-emptive Threading
- Memory and Resources are typically statically allocated
- Integrated device driver interface
- Memory Protection: Stack overflow protection, Kernel object and device driver permission tracking, Thread isolation
- Bluetooth® Low Energy (BLE 4.2, 5.0) with both controller and host, BLE Mesh
- Native, fully featured and optimized networking stack

Fully featured OS allows developers to focus on the application
Zephyr Supported Architectures
Sample of Board Support

119 BOARDS TODAY, WITH MORE ON WAY...
http://docs.zephyrproject.org/boards/boards.html
Very Active Diverse Community!

1.13 release statistics:
- 13 weeks cycle, with 2 weeks of merge window
- 1,834 changes (patch commits)
- 140 developers identified
- 25 companies participated
- 20 changes / day (0.8/hour)

Repositories:
- 2016/2: 5
- 2018/11: 15

Authors:
- 2016/2: 80
- 2018/11: 389

Commits:
- 2016/2: 2,806
- 2018/11: 25,448

Boards:
- 2016/2: 4
- 2018/11: 119 in progress
Zephyr in RTOS Landscape 11/26

Total Contributors

<table>
<thead>
<tr>
<th>Rank</th>
<th>RTOS</th>
<th>#</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>mbed OS</td>
<td>468</td>
</tr>
<tr>
<td>2</td>
<td>Zephyr</td>
<td>382</td>
</tr>
<tr>
<td>3</td>
<td>nuttX</td>
<td>276</td>
</tr>
</tbody>
</table>

Total Commits

<table>
<thead>
<tr>
<th>Rank</th>
<th>RTOS</th>
<th>#</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>nuttX</td>
<td>37,492</td>
</tr>
<tr>
<td>2</td>
<td>Zephyr</td>
<td>25,321</td>
</tr>
<tr>
<td>3</td>
<td>mbed OS</td>
<td>20,387</td>
</tr>
</tbody>
</table>

Commits to Master (last 30 days)

<table>
<thead>
<tr>
<th>Rank</th>
<th>RTOS</th>
<th>#</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Zephyr</td>
<td>849</td>
</tr>
<tr>
<td>2</td>
<td>mbed OS</td>
<td>355</td>
</tr>
<tr>
<td>3</td>
<td>nuttX</td>
<td>222</td>
</tr>
</tbody>
</table>
Comparison and characterization work just starting by Zephyr members who need it for products.

NXP whitepaper about some Real-Time benchmarks based on common API’s for comparing with Linux

Initial results show performance benefit* in:
- Context Switching
- Locking and unlocking Mutexes

Zephyr is ready for more 3rd party characterization and Real-Time testing.

*Linux used did not have PREEMPT-RT patches applied.
Summary: watch this space in 2019! :-)

Real-Time Linux PREEMPT-RT patch set is almost upstream
- Developers are starting to plan tasks for after merging

Zephyr will be having first LTS release in early 2019
- Members have key applications that need Real-Time
- Already seeing Zephyr used in products, before first LTS

Both projects are setting the foundation for safety critical products
Thank you for your interest.

Questions?