## About us!

<table>
<thead>
<tr>
<th>Yocto Project Community Manager</th>
<th>Linaro</th>
</tr>
</thead>
<tbody>
<tr>
<td>[Image of person]</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Yocto Project Training and Dev Day</th>
<th>[Image of person]</th>
</tr>
</thead>
</table>

[Image of person]
Agenda

- Yocto Project® organisation and membership
- Current release process
- Automated testing improvements
- LTS background and overview
- How can you help?
What is the Yocto Project®?

IT'S NOT AN EMBEDDED LINUX DISTRIBUTION, IT CREATES A CUSTOM ONE FOR YOU.

The Yocto Project (YP) is an open source collaboration project that helps developers create custom Linux-based systems regardless of the hardware architecture.

The project provides a flexible set of tools and a space where embedded developers worldwide can share technologies, software stacks, configurations, and best practices that can be used to create tailored Linux images for embedded and IOT devices, or anywhere a customized Linux OS is needed.
Yocto Project® organisation

- An open source, collaborative project
  - Hosted by the Linux Foundation in 2010
  - Project architect is Richard Purdie.
  - Uses OpenEmbedded (started in 2003).

- Yocto Project Governing Board
  - Platinum, Gold and Silver member organizations
  - Linux Foundation corporate members
  - Influence the project direction (vote)
  - Contribute financially to its infrastructure and stability.
  - Nominate/vote: Technical Steering Committee (TSC)

https://www.yoctoproject.org/ecosystem/members/
https://www.yoctoproject.org/how-to-join/
Yocto Project® governance model

Governing Board
Oversee business decisions, budgets, and general administration

Technical Steering Committee
Technical oversight to project and upstream

Advocacy Team
Oversee Marketing, Communication, Outreach, Events, and Training

Community
Oversee Community and ecosystem

Advisory Board
Current release process
Predictable release cadence: twice a year since inception
A Yocto Project® typical release content

- A release is the aggregation of:
  - OE-core
  - bitake
  - meta-yocto
  - yocto-docs

- Major component upgrades
  - Includes ABI/API changes
  - Include major version upgrades
  - New features
  - Using Long Term Support kernels

- New Yocto Project features
  - Test infrastructure changes
  - Automation changes
  - Architectures added/removed

- Bug fixes for issues reported to Yocto Project

- Tight integration loop with upstreams
### Yocto Project® Stable release content

<table>
<thead>
<tr>
<th>Acceptable</th>
<th>Unacceptable</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Security and CVE fixes</td>
<td>• General version upgrades</td>
</tr>
<tr>
<td>• Fixes for bugs</td>
<td>• New Features</td>
</tr>
<tr>
<td>• Fixes so codebase works with newly released distros</td>
<td></td>
</tr>
<tr>
<td>• Bug fix only version upgrades (especially where follows upstream policy)</td>
<td></td>
</tr>
</tbody>
</table>

- **Strict backport policy (master first)**
- **Same testing as regular releases**
Stable release challenges

• Typically, alongside the latest release the previous two releases are also maintained
  • *Maintenance window is up to one year.*
• Done thanks to developers volunteering community time
• Manual testing overhead relying on Yocto Project® members resources.
Automated testing improvements
Yocto Project® automated testing improvements

- Continuous integration using Yocto Project autobuilder infrastructure
  - [https://autobuilder.yoctoproject.org/](https://autobuilder.yoctoproject.org/)
  - 25+ builders to validate 12+ numbers of host distros
  - Fairly large project expense in 2019 to modernize builders

- Major improvements (last two years) to reduce needs for manual testing
  - ptests on x86 and arm for ~60 pieces of software
  - LTP and LTP posix tests
  - Test reproducibility of our toolchain and minimal images
  - gcc, binutils and glibc testsuites on all arm/mips/powerpc/x86 32/64 bit
  - Over 1.9 million tests in 8 hours
LTS background and overview
Yocto Project® Long term support background

- Six month release cadence aligns well with pace of open source development
  - However *too frequent* for some use cases and markets
- Many companies using Yocto Project releases end up doing *some* LTS on their own
- Over the years it became a common complain at developers’ events and BoF.
  - YP TSC presented to the governing board a request to establish a formal LTS process at ELCE 2019
  - LTS announced in March 2020
  - [https://www.yoctoproject.org/yocto-project-long-term-support-announced/](https://www.yoctoproject.org/yocto-project-long-term-support-announced/)
Yocto Project® LTS proposal in a nutshell

● An LTS every two years
  ○ Maintained for two years (initial plan)
  ○ Yocto Project 3.1 is the first LTS
  ○ Regular stable release maintenance is reduced to seven months

● Funded by the Yocto Project membership
  ○ 20 hours a week of maintainer duties
  ○ Steve Sakoman is the Yocto Project maintainer for v3.1

● The TSC is responsible for LTS releases, processes and maintainer.
Yocto Project® LTS releases

- **Components to be covered**
  - Bitbake, OpenEmbedded-core
  - meta-yocto-bsp, meta-poky, yocto-docs

- **Not covered**
  - meta-openembedded, vendor layers, …

- **Only on a subset of supported native build platforms**

- **Rely on Yocto Project AB resources and automated testing**
  - Follow the same testing process as the original release
  - Only run virtualized tests

- **Evaluate use of CVE automated tools (cve-check)**
## Yocto Project® LTS: *is / is not*

<table>
<thead>
<tr>
<th>IS</th>
<th>IS NOT</th>
</tr>
</thead>
</table>
| • A major change for the Yocto Project, which requires significant resources from the membership | • A replacement for commercial offering from OSVs which offer very long term support (5y, 10y, ...)
| • An initiative to create a place for the community to collaborate on LTS | • A guarantee of "all" CVE fixes, we only agree to integrate what is submitted and passes testing
| • A guarantee that we have resources available to integrate and test changes in a timely manner and to have fast turn around of point releases | • Is not able to check your own additions and customisations |
Release cycle overview
How to contribute to Yocto Project® LTS

- **Follows standard Yocto Project / OpenEmbedded contributions guidelines.**
  - Submit patches to appropriate mailing list, tagged with [dunfell]
  - Submitted patches should be on **master**, except when incorporating a CVE/bug fix for an old version.

- **LTS maintainer**
  - assemble and test ‘group of patches’
  - submit working set on the list once a week for review
  - submit pull request to Richard Purdie

- **The branch** stable/dunfell-nut always contains the ‘work in progress’ patches.
How can you help?
How can you help?

- Contribute, share your bug fixes, CVEs and security patches
- Test LTS branch with your distro and BSP
- Report bugs
- Continue to test and develop the main release branch
- Join us and support the Yocto Project