Yocto Project® Intro to Yocto validation

Armin Kuster

Yocto Project Virtual Summit Europe, October 29-30, 2020
Intro

• This would not be possible without the gracious contributions of the Yocto Project Members.
• They have help fund the infrastructure and resources to make this all happen.
• Lots of community support
Intro

• Agenda
  – Continuous Integration
  – Quality Assurance
  – Some extras
Continuous Integration
CI - overview

• Our CI is called “AutoBuilder”
  - Buildbot technology
  - Initial version 2012
  - Version 2 - 2018
    • Buildbot separation
      - Easier to maintain
      - Easier to update
CI - overview

• Public
  – https://autobuilder.yoctoproject.org/

• Workers are x86 and Arm64

• Wide range of Hosts
  – Centos       Fedora
  – Ubuntu       Debian
  – OpenSuse     Tumbleweed
CI - overview

• Two system configurations
  – Controller
    • [https://git.yoctoproject.org/git/yocto-autobuilder2](https://git.yoctoproject.org/git/yocto-autobuilder2)
  – Worker
    • Per branch configurations
    • Per supported distros
    • [https://git.yoctoproject.org/git/yocto-autobuilder-helper](https://git.yoctoproject.org/git/yocto-autobuilder-helper)
CI: A-Quick

- 41 builds
- All Machines
- Multilib
- musl
- single selftest
- Runtime test
  - Fast Ptest
CI: A-Quick

- Alt config
  - Systemd
  - pam
  - Different kernel version
CI: A-full

• All of A-quick
  – 14 more build configurations

• All selftest

• All ptest
  – ltp

• buildperf

• Member layers
CI: Use cases

- Package updates
- New host OS validation
- Stable releases
- LTS release
- AUH
- Kernel updates
CI: artifacts

• Builds
  - https://autobuilder.yocto.io/pub/non-release/
  - https://autobuilder.yocto.io/pub/releases/

• Official Releases
  - http://downloads.yoctoproject.org/releases/
CI: artifacts

• Mirrors
  - Shared State
    • http://sstate.yoctoproject.org/PATH
  - Sources
    • http://downloads.yoctoproject.org/mirror/sources/
CI: artifacts - Historical

- Errors
  - https://errors.yoctoproject.org

- Buildhistory
  - https://git.yoctoproject.org/git/poky-buildhistory

- Wiki build logs
  - https://wiki.yoctoproject.org/wiki/BuildLog
QA

• There are two QA categories:
  – CI
    • A-quick
    • A-full
  – Formal
    • Independent verification
    • Covers Manual tests
    • Reports generated
CI - QA

• First tests added in 2013

• Framework based on “unittest”

• Test groupings:
  – Selftests
  – Runtime
  – (e)sdk
  – Buildperf
QA - selftests

• Two kinds
  - Bitbake-selftest
    • 353 + testcases
  - Oe-selftest
    • 400 + testcases

• Tests run on the host
QA - selftests

• Used to help validate new OS’s

• Standard three test states:
  – Pass/Fail
  – Skip
    • Auto detection
QA - runtime

• Run on a MACHINE
  – Virtual
  – Real hardware

• Executive commands
  – Check ps
  – Parse logs
QA - runtime

• Special case tests
  – Ptest
    • Tests per package
    • Executor ptest-runner
    • Currently 135 tests
  – LTP
• Split results
  – pass/fail/skip
QA - runtime

- (e)SDK
  - 25 tests

- Results
  - Json
    - tmp/logs/oeqa
  - uploaded
    - [http://git.yoctoproject.org/cgit/cgit.cgi/yocto-testresults/](http://git.yoctoproject.org/cgit/cgit.cgi/yocto-testresults/)
CI - QA

• Buildperf
  - Collects data on how long things take to run and resource usage.
  - https://wiki.yoctoproject.org/wiki/TipsAndTricks/MiningPerformanceData
  - http://git.yoctoproject.org/cgit/cgit.cgi/yocto-buildstats/
## General

<table>
<thead>
<tr>
<th>Hostname</th>
<th>perf-centos7.yoctoproject.org</th>
<th>perf-centos7.yoctoproject.org</th>
</tr>
</thead>
<tbody>
<tr>
<td>Branch</td>
<td>master</td>
<td>master</td>
</tr>
<tr>
<td>Commit number</td>
<td>57867</td>
<td>57867</td>
</tr>
<tr>
<td>Commit</td>
<td>5d47cdf448b</td>
<td>5d47cdf448b</td>
</tr>
<tr>
<td>Number of test runs</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

## Test result summary

<table>
<thead>
<tr>
<th>Test Case Description</th>
<th>Exec</th>
<th>Build Time</th>
<th>Tempdir Size</th>
<th>Roofs Size</th>
</tr>
</thead>
<tbody>
<tr>
<td>test1: Build core-image-sato</td>
<td></td>
<td>1:04:57</td>
<td>+0.00 s</td>
<td>+0.0 %</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>+0.0 0 S</td>
<td>+0.0 %</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>+44.93 GiB</td>
<td>+44.93 %</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>+256.16 MiB</td>
<td>+256.16 %</td>
</tr>
<tr>
<td>test2: Run core-image-sato do_rootsfs with sstate</td>
<td></td>
<td>2:17:4</td>
<td>+0.00 s</td>
<td>+0.0 %</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>+0.0 0 S</td>
<td>+0.0 %</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>+74.4 GiB</td>
<td>+74.4 %</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>+204.8 MiB</td>
<td>+204.8 %</td>
</tr>
<tr>
<td>test3: Bitbake parsing (bitbake -p)</td>
<td></td>
<td>15.3 s</td>
<td>+0.00 s</td>
<td>+0.0 %</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>+0.0 0 S</td>
<td>+0.0 %</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>+10.4 s</td>
<td>+10.4 %</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>+0.0 0 S</td>
<td>+0.0 %</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>+2.2 s</td>
<td>+2.2 %</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>+0.0 0 S</td>
<td>+0.0 %</td>
</tr>
<tr>
<td>test4: eSDK metrics</td>
<td></td>
<td>264.46 MiB</td>
<td>+0.00 s</td>
<td>+0.0 %</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>+0.0 0 S</td>
<td>+0.0 %</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>+1:08.3 s</td>
<td>+1:08.3 %</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>+0.0 0 S</td>
<td>+0.0 %</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>+1.51 GiB</td>
<td>+1.51 %</td>
</tr>
</tbody>
</table>
Formal QA

- Donated resources from Intel and Windriver
  - Intel does Intel h/w
    - QA lead
  - Windriver does Non-Intel h/w
    - Beaglebone
    - PPC
    - EdgeRouter
Formal QA

• We do Formal QA when the Project has a need to do an official release
  - Milestones
    • Typical cycle
    • https://wiki.yoctoproject.org/wiki/Yocto_3.2_Schedule#Yocto_Project_3.2_release_milestone_structure
  - New stable
  - Dot releases
Formal QA

- Provides as an independent checks & balances
- Re-Run Automated tests
- Run the Manual tests on real hardware including our official Yocto BSP’s
Formal QA

- Additional regression analysis
  - Resulttool
    - Comparison between test runs
      - Ptests
      - Ltp
      - Toolchain
Formal QA

- Official QA results reported
  - Uploaded to git
    - http://git.yoctoproject.org/cgit/cgit.cgi/yocto-testresults-contrib/
  - Defects opened on regressions
  - Sent to mailing list
    - Yocto@
Hello all,
This is the full report for yocto-3.2_M3.rc2:
https://git.yoctoproject.org/cgit/cgit.cgi/yocto-testresults-contrib/tree/?h=intel-yocto-testresults

====== Summary ======
No new defects are found.
No high milestone defects.

Bugs verified:
1. BUG id:14052 - The beaglebone can not bootup
   Verified Fixed.
2. BUG id:14055 - failure in oe-core manual test: test_dependency_explorer_is_launched
   Verified Fixed (with patch applied).

Existing bugs observed in this release:
1. valgrind ptest failed (BUG id:14051)
   Updated with additional failures observed.
2. parted ptest failed (BUG id:14050)
3. pango ptest failed (BUG id:14049)

====== Bugs ======
https://bugzilla.yoctoproject.org/show_bug.cgi?id=14052
https://bugzilla.yoctoproject.org/show_bug.cgi?id=14055
https://bugzilla.yoctoproject.org/show_bug.cgi?id=14051
https://bugzilla.yoctoproject.org/show_bug.cgi?id=14050
https://bugzilla.yoctoproject.org/show_bug.cgi?id=14049

Thanks,
Sangeeta
Formal QA

• Yocto Project TSC
  – Reviews
  – Approves
Hey old Man! Anything else?
Extra

• The base framework is contained in core

• meta/lib/oeqa
  – Buildperf
  – Runtime
  – Selftest
  – sdk
  – Manual
• Framework can be extended
  - Any layer can add their own
  - Use the same directory structure
  - Inherit the appropriate classes

• Examples
  - meta-openembedded
  - meta-intel
  - meta-security
• Run automated tests on a MACHINE
  - INHERIT += “testimage”
  - Bitbake {image} -c testimage
• On real hardware
  - TEST_TARGET_IP
  - TEST_SERVER_IP
Thanks for your time