OpenSDK OE Builds on 96Boards

Andrey Konovalov (LHG)
OpenSDK OE Builds on 96Boards

- Use the same OE build system as Reference Platform Builds do

- meta-lhg layer contains components developed inside LHG

- The builds are running in jenkins. Only one image (per every supported board) is built: Weston with Chromium browser included into the image

- The images produced by these builds are published at snapshots.linaro.org

- Boards supported: DragonBoard-410c, HiKey, Beagle X15
OpenSDK OE Builds: key components

- HW accelerated graphics: BSP layers (meta-96boards, meta-qcom, meta-ti)

- gstreamer, mesa: Linaro’s meta-backports layer
  Current versions: gstreamer 1.10.3, mesa 13.0.4

- OPTEE: Linaro’s meta-linaro/meta-optee
  Current version: OPTEE 2.3.0

- Chromium + OpenCDM: LHG’s fork of meta-browser layer
  Current version: chromium_v45

- AES descriptor TA: meta-lhg
Meta-lhg structure

https://github.com/linaro-home/meta-lhg

- **meta-lhg layer:**
  The OpenEmbedded layer for LHG specific components.
  lhg-westeros-wpe-image and rpb-westonchromium-image recipes, optee-aes-decryptor recipe, chromium and portmap bbappends (needed by OpenCDM), nss_3.19 needed by the (old) chromium version used

- **meta-lhg-integration:**
  This layer contains fixes or workarounds needed to integrate components external to LHG into public LHG OE builds.
  Mostly weston and ti-sgx-ddk-um (powervr userland driver) bbappends needed to enable this driver in weston, and couple other bbappends
LHG contributes to meta-rpb and meta-96boards
(or why meta-lhg is that small)

About a year ago meta-lhg was considerably bigger than it is now. Its content was reviewed, the components which are useful outside LHG have been moved to proper layers.

E.g.
conf/distro/include/egl.inc in meta-rpb
  - selects the proper EGL/GLES provider
meta-96boards/recipes-graphics/mali-userland in meta-96boards
  - added public Mali450 userland driver
Next Steps

- chromium is be updated to a more recent version, the OpenCDM plugin included
- fix HW accelerated playback in gstreamer with waylandsink (rendering issue)
- ...
OpenSDK pre-built images

- Jenkins jobs:
    May miss most recent changes or have workarounds for the issues which couldn’t be resolved in short term.
    This is the choice for those who need it working out of the box.
  - https://ci.linaro.org/view/All/job/lhg-oe-build-next/ aka “staging”
    Uses the tips of the relevant trees

- Pre-built images:
  - “stable” - http://snapshots.linaro.org/openembedded/pre-built/lhg/morty/
  - “staging” - http://snapshots.linaro.org/openembedded/pre-built/lhg/morty-staging/
Building OpenSDK images

See https://github.com/linaro-home/lhg-oe-manifests (the README.md)
Check the “Required Packages” section in Yocto Project Reference Manual.
This is essentially:

repo init -u https://github.com/linaro-home/lhg-oe-manifests.git -b morty
repo sync
source setup-environment
bitbake rpb-westonchromium-image

Two common issues faced when doing an OE build first time:

- Doing it on a VM with 1GB of RAM
- Download (do_fetch) failures when downloading from behind a proxy

An rpb-westonchromium-image build needs approx. 10GB of (virtual) RAM and
100GB of free disk space
OpenSDK OE Builds on 96Boards

Andrey Konovalov (LHG)
OpenSDK OE Builds on 96Boards

- Use the same OE build system as Reference Platform Builds do
- meta-lhg layer contains components developed inside LHG
- The builds are running in jenkins. Only one image (per every supported board) is built: Weston with Chromium browser included into the image
- The images produced by these builds are published at snapshots.linaro.org
- Boards supported: DragonBoard-410c, HiKey, Beagle X15
OpenSDK OE Builds: key components

- HW accelerated graphics: BSP layers (meta-96boards, meta-qcom, meta-ti)
- gstreamer, mesa: Linaro’s meta-backports layer
  Current versions: gstreamer 1.10.3, mesa 13.0.4
- OPTEE: Linaro’s meta-linaro/meta-optee
  Current version: OPTEE 2.3.0
- Chromium + OpenCDM: LHG’s fork of meta-browser layer
  Current version: chromium_v45
- AES descriptor TA: meta-lhg
Meta-lhg structure

https://github.com/linaro-home/meta-lhg

- **meta-lhg layer:**
  The OpenEmbedded layer for LHG specific components.
  lhg-westeros-wpe-image and rpb-westonchromium-image recipes, optee-aes-decryptor recipe, chromium and portmap bbappends (needed by OpenCDM), nss_3.19 needed by the (old) chromium version used

- **meta-lhg-integration:**
  This layer contains fixes or workarounds needed to integrate components external to LHG into public LHG OE builds.
  Mostly weston and ti-sgx-ddk-um (powervr userland driver) bbappends needed to enable this driver in weston, and couple other bbappends
LHG contributes to meta-rpb and meta-96boards
(or why meta-lhg is that small)

About a year ago meta-lhg was considerably bigger than it is now. Its content was reviewed, the components which are useful outside LHG have been moved to proper layers.

E.g.
conf/distro/include/egl.inc in meta-rpb
  - selects the proper EGL/GLES provider
meta-96boards/recipes-graphics/mali-userland in meta-96boards
  - added public Mali450 userland driver
Next Steps

- chromium is be updated to a more recent version, the OpenCDM plugin included
- fix HW accelerated playback in gstreamer with waylandsink (rendering issue)
- ...

OpenSDK pre-built images

- Jenkins jobs:
    May miss most recent changes or have workarounds for the issues which couldn’t be resolved in short term.
    This is the choice for those who need it working out of the box.
  - https://ci.linaro.org/view/All/job/lhg-oe-build-next/ aka “staging”
    Uses the tips of the relevant trees

  The both builds are based on current yocto release, Morty atm.

- Pre-built images:
  - “stable” - http://snapshots.linaro.org/openembedded/pre-built/lhg/morty/
  - “staging” - http://snapshots.linaro.org/openembedded/pre-built/lhg/morty-staging/
Building OpenSDK images

See https://github.com/linaro-home/lhg-oe-manifests (the README.md)
Check the “Required Packages” section in Yocto Project Reference Manual.
This is essentially:

```bash
repo init -u https://github.com/linaro-home/lhg-oe-manifests.git -b morty
repo sync
source setup-environment
bitbake rpb-westonchromium-image
```

Two common issues faced when doing an OE build first time:

- Doing it on a VM with 1GB of RAM
- Download (do_fetch) failures when downloading from behind a proxy

An rpb-westonchromium-image build needs approx. 10GB of (virtual) RAM and
100GB of free disk space
Thank You

#BUD17
For further information: www.linaro.org
BUD17 keynotes and videos on: connect.linaro.org
Thank You

#BUD17

For further information: www.linaro.org
BUD17 keynotes and videos on: connect.linaro.org