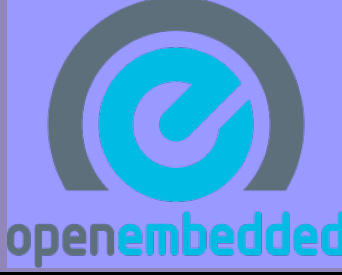
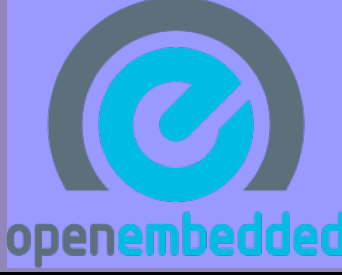
Yocto Crash Course



Welcome to the Yocto Crash Course – Or, Building Your Own Embedded Linux OS for Fun and (possibly) Profit

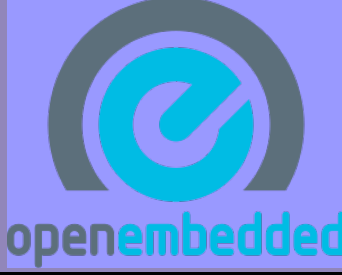
Prepared / Presented by
Stephen Arnold, Principal Scientist VCT Labs

Build Host Reqs and Potential Issues



- Officially Supported Distributions
  - Debian/Ubuntu, CentOS, Fedora, OpenSUSE
- Other “unsupported” Distributions
  - Gentoo x86, Arch, Slackware, etc
- Gentoo amd64, VMs, and chroots
  - libpseudo fails on Gentoo x86_64 multilib
  - Build in a VM or chroot environment
- Common Build errors: "command not found..."
  - “hidden” build deps
    - bc, lzop, u-boot-tools, dtc
    - Can depend on kernel config
  - Connectivity issues

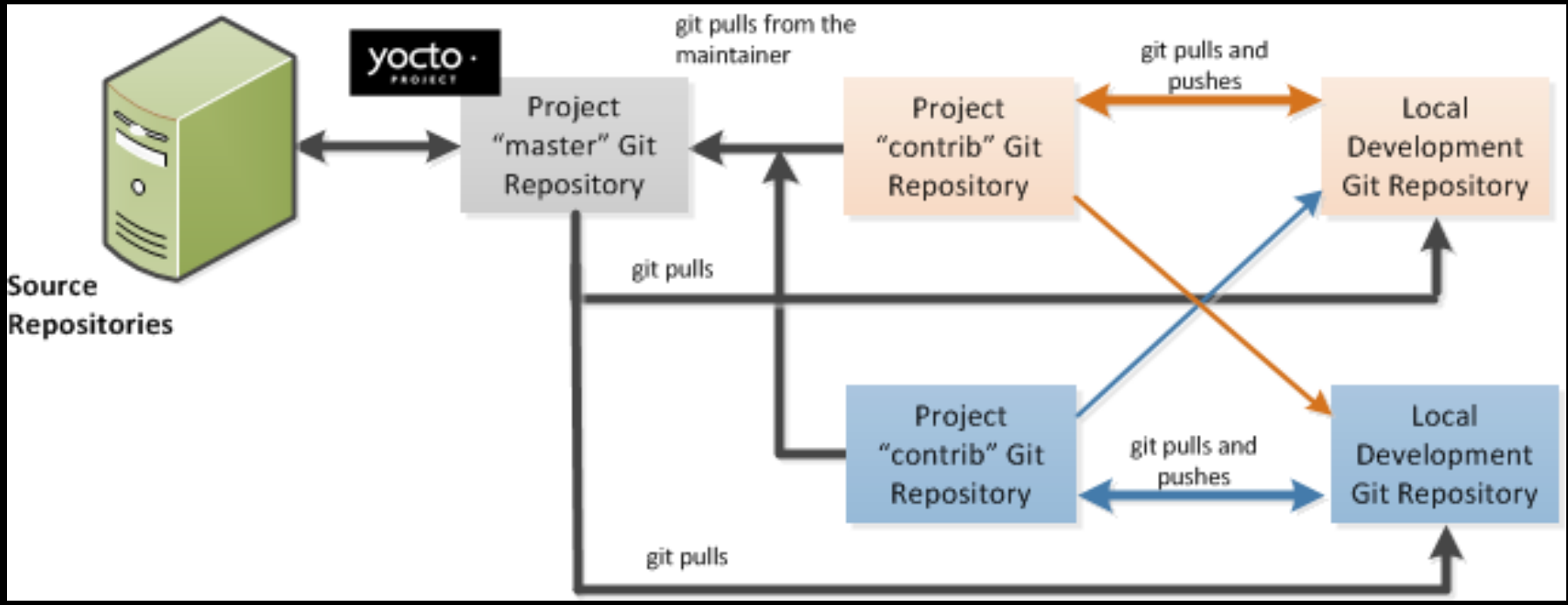
OpenEmbedded Workflow & Layers



- Yocto is a BSP (and a project) and OE (oe-core) is a collection of layers
- Yocto project BSP and Poky distribution
  - Self-contained set of “demo” machines and OE layers
  - Default distribution built is “poky”
  - Default kernel is linux-yocto
  - Can build beaglebone, edgerouter, qemu images
- What do you really need?
  - A BSP layer for your machine
  - The core metadata layers (ie, poky or oe-core)
  - Additional application layers (as needed)
  - Custom layers (*optional)

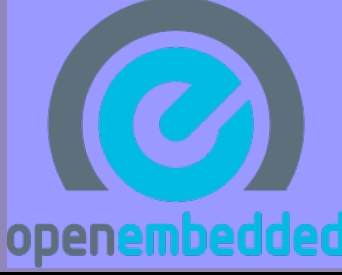
OE Workflow & Layers cont.

Yocto Project Workflow



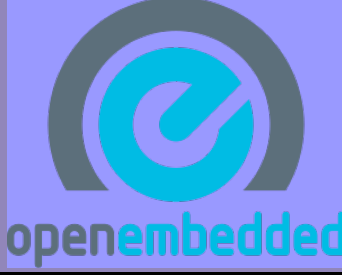
*http://www.yoctoproject.org/docs/current/dev-manual/dev-manual.html#workflows

Inside the OE Environment



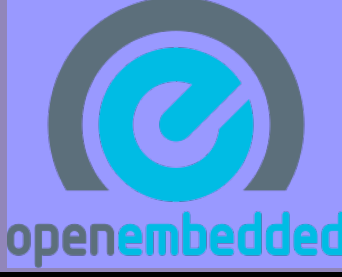
- User Configuration, Metadata, Machine Configuration
  - Distro Layers: poky, ångstrom, custom, “distro-less”
  - BSP Layers
    - yocto reference bsp
  - Software Layers
    - meta-beagleboard-extras
    - meta-fsl-demos
    - meta-openstack
  - Kernel Recipes and Versions
    - linux-yocto (meta-yocto-bsb)
    - linux-mainline (meta-ti, meta-beagleboard)
    - linux-ti-staging (meta-ti)
    - 3.18 to 3.8+ (and older)

Inside the OE Environment cont.



- Image Features and Package Configuration
  - Grep is your friend / read the comments
    - IMAGE/EXTRA_IMAGE_FEATURES
    - PACKAGECONFIG (sort of like USE flags)
- Recipes and Sources
  - File Types (recipes, bbclass, includes, configs)
  - Upstream Releases/Repos, Local Projects
    - Source tarballs
    - git/svn/hg/cvs
  - Source Mirror(s)
    - Make a local mirror for downloads

Inside the OE Environment cont.

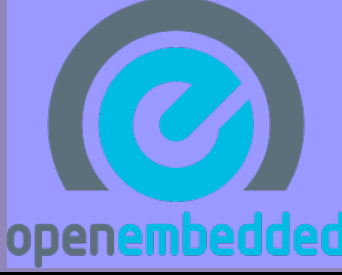


- BitBake Tips and Tricks
  - Recipes and Tasks
    - Use the -c argument to bitbake to execute one task
    - Use the -b argument to ignore recipe build depends
    - Use the -D argument to get more debug output
  - Source Fetching, Patching, Configuration, and Compilation
    - Use “-c fetchall” to prefetch sources for a build target
  - Package Splitting, Image Generation, SDK Generation
    - One recipe, many packages
  - Custom Recipes and Layers

    http:///layers.openembedded.org/layerindex/branch/master/layers/

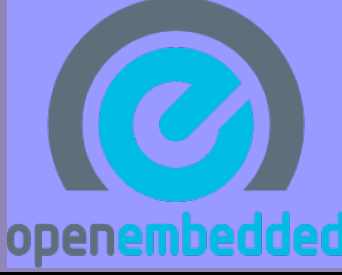
    https://github.com/sarnold/meta-alt-desktop-extras

Hands-On Poky



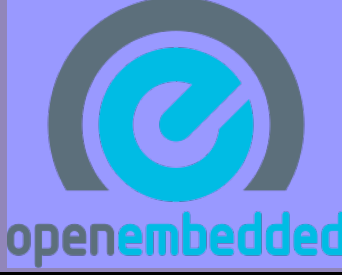
- Qemux86 extra-quick quick start:
  - Clone poky repo
  - Source OE environment script
  - Configure local.conf
  - Source environment script again
  - Build target image and deploy
- Beaglebone
  - Make new build directory
  - Configure local.conf
- RaspberryPi
  - Clone meta-raspberrypi BSP
  - Make new build directory
  - Configure local.conf

Adding an Upstream BSP



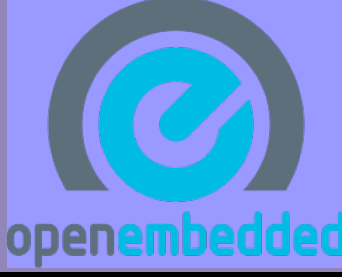
- RaspberryPi layer
  - https://github.com/agherzan/meta-raspberrypi
  - See the README for build requirements
  - Should build with poky, oe-core, ångstrom
- BeagleBoard / TI layers
  - http://git.yoctoproject.org/cgit/cgit.cgi/meta-ti (official)
  - https://github.com/beagleboard/meta-beagleboard (somewhat stale, forks may be more current)
- Freescale Build Scripts
  - http://git.yoctoproject.org/cgit/cgit.cgi/meta-fsl-arm
  - Uses repo manifest and build script for setup

Kernels & Package Feeds



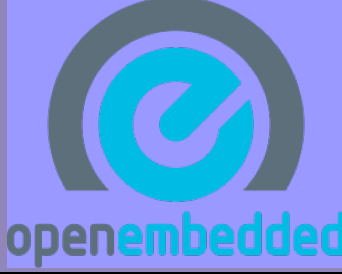
- Kernel Selection
  - Defaults to linux-yocto
  - Use PREFERRED_PROVIDER/VERSION to change
    - PREFERRED_PROVIDER_virtual/kernel = "linux-mainline"
    - PREFERRED_VERSION_linux-mainline = "3.17%"
- Package Feeds
  - Ipk Feed Support
    - PACKAGE_CLASSES = "package_ipk"
    - Point apache doc root at build tree deploy root – tmp/deploy
    - Point feed URL at tmp/deploy/ipk
  - RPM and Deb Feeds
    - Exercise left for the reader...

Customizing Your Build



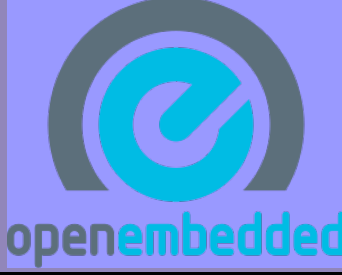
- Kernel Version and Configuration
  - RaspberryPi – override PREFERRED_VERSION
  - BeagleBone – above plus override COMPATIBLE_MACHINE
  - Small number of global config options
- New / Modified Kernel Recipe
  - Make or modify an existing linux-yocto_3.X.bbappend
    - Create/obtain patches and config fragments
    - Append new files to SRC_URI
    - Update the md5sums
  - Create your own linux-custom_X.X.bb kernel recipe
    - See linux-yocto-custom.bb
  - Inherit vs. Include
    - .bbclass and .inc files

Customizing Your Build cont.



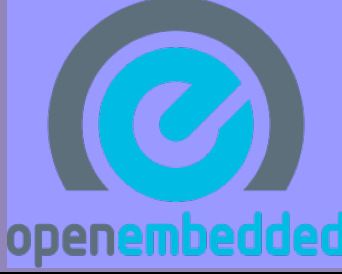
- Image Recipes
  - Inherit/include and IMAGE_* options
  - IMAGE_INSTALL packagegroups and packages
- Package Recipes
  - Inherit/include and PACKAGECONFIG
  - IMAGE/MACHINE_FEATURES drive package options
- Modifying and Adding Packages
  - .bbappend is your friend
  - The scripts directory and docs are also your friends
    - create-recipe, yocto-layer, runqemu, and more
- devshell and TERM config settings
  - TERMCMD and TERMCMDRUN
  - http://www.openembedded.org/wiki/Devshell

Deployment and Debugging



- Deploy Tips and Hacks
  - Image types: rpi-sdimg, ext3, tar.bz2, tar.gz, jffs2
  - Where does U-boot look for the kernel?
  - Use “-c deploy” for incremental kernel testing
  - Create custom deploy tasks (eg, kernel configme task)
  - Local .ipk package feeds
    - Image build updates package index
    - Can add/update packages as needed
- SDK Tools
  - bitbake targets: meta-toolchain vs. populate_sdk
  - IMAGE tweaks: see local.conf EXTRA_IMAGE_FEATURES
- GDB / GDB Server vs. Eclipse / TCF Agent
  - Choose your FEATUREs and tools

Graphical User Interfaces



- Toaster
  - Install django-1.6 and south-0.8.4
  - Enable in local.conf:
    - `INHERIT += "toaster"`
    - `INHERIT += "buildhistory"`
    - `BUILDHISTORY_COMMIT = "1"`
  - `$ cd <poky-dir> && source oe-init-build-env`
  - `$ source toaster start (stop)`
  - `$ bitbake core-image-minimal`
  - `$ xdg-open http://localhost:8000`
  - Default DB is sqlite3
  - Make sure you have a valid timezone set
  - https://wiki.yoctoproject.org/wiki/Setting_up_a_local_instance_of_Toaster