Meta Layer

What is a layer?

- A layer is a logical collection of related recipes.
- Types of Layers: oe-core, BSP Layer, application layer
- Layer name starts with meta-, but this is not a technical restriction.
- Eg. meta-mycustom

Why create a meta layer?

- Despite most of the customization can be done with the local.conf configuration file, it is not possible to:
    - Store recipes for your own software projects
    - Create your own images
    - Consolidate patches/modifications to other people's recipes
    - Add a new custom kernel
    - Add a new machine
- Most important point: Do not edit POKY/UPSTREAM Layers, as it complicates future updates
- Advantage: This allows you to easily port from one version of Poky to another

Depending on the type of layer, add the content:

- If the layer is adding support for a machine, add the machine configuration in conf/machine/
- If the layer is adding distro policy, add the distro configuration in conf/distro/
- If the layer introduces new recipes, put the recipes you need in recipes-* subdirectories of the layer directory.

Recipe directories inside layers

- By convention, recipes are splitted into categories
- The most difficult part is deciding in which category your recipe will go
- By checking what was already done in the official layers should give you a good idea of what you should do

Layer Priority:

- Each layer has a priority, which is used by bitbake to decide which layer takes precedence if there are recipe files with the same name in multiple layers
- A higher numeric value represents a higher priority.

Creating Layer

- There are two ways to create your own layer.
    1. Manually
    2. Using script

Manually:

- Step 1 : Create a directory for the layer. For example: 'meta-mylayer'
- Step 2 : Create a conf/layer.conf
    - You can simply copy meta-oe's one and just change "openembedded-layer" to something appropriate for your layer; you may also want to set the priority as appropriate.
- Step 3 : Update bblayers.conf file with the new layer

Creating layer using tool

- You can create your own layer using the bitbake-layers create-layer command
    - $ bitbake-layers create-layer --help
- The tool automates layer creation by setting up a subdirectory with a layer.conf configuration file, a recipes-example subdirectory that contains an example.bb recipe, a licensing file, and a README
    - $ bitbake-layers create-layer ../source/meta-mylayer
- Default priority of the layer is 6
    - $ bitbake-layers add-layer ../source/meta-mylayer
    - $ bitbake-layers show-layers

Layer Configuration File layer.conf

- # The configuration and classes directory is appended to BBPATH
    - BBPATH .= ":${LAYERDIR}"
- # The recipes for the layers are appended to BBFILES
    - BBFILES += "${LAYERDIR}/recipes-*/*/*.bb ${LAYERDIR}/recipes-*/*/*.bbappend"
- # The BBFILE_COLLECTIONS variable is then appended with the layer name
    - BBFILE_COLLECTIONS += "skeleton"
    - BBFILE_PATTERN_skeleton = "^${LAYERDIR}/"
- # The BBFILE_PRIORITY variable then assigns a priority to the layer.
    - BBFILE_PRIORITY_skeleton = "1"
- # This should only be incremented on significant changes that will
- # cause compatibility issues with other layers
    - LAYERVERSION_skeleton = "1"
    - LAYERDEPENDS_skeleton = "core"
    - LAYERSERIES_COMPAT_skeleton = "zeus"

yocto-check-layer

- The yocto-check-layer script provides you a way to assess how compatible your layer is with the Yocto Project
- You should use this script if you are planning to apply for Yocto Project Compatible Program
    - $ source oe-init-build-env
    - $ yocto-check-layer your_layer_directory

Challenge Find out in meta folder which one is used a lot: += or _append