Poky Source Tree

- bitbake – Holds Python scripts for bitbake
- documentation – Yocto Project documentation
- meta – Contains oe-core metadata
- meta-poky – Configuration for Poky reference distribution
- meta-skeleton – Template recipes for BSP and kernel development
- meta-yocto-bsp – BSPs like Beaglebone, EdgeRouter
- scripts – Environment and development tool scripts
- LICENSE – Licensing information

Build System Configuration (conf)

Upon running source poky/oe-init-build-env, the build folder contains conf directory with:

1. local.conf – Configures build settings.
2. bblayers.conf – Specifies layers used during the build.

Important Variables in local.conf

- MACHINE: Specifies the target machine (e.g., qemux86-64)
- DL_DIR: Defines where downloaded sources are stored
- TMP_DIR: Specifies the build output location

Parallel Processing

- BB_NUMBER_THREADS – Number of parallel tasks for bitbake
- PARALLEL_MAKE – Number of processes GNU make can run in parallel

Commands to check values:

```
bitbake -e core-image-minimal | grep ^BB_NUMBER_THREADS=
```

```
bitbake -e core-image-minimal | grep ^PARALLEL_MAKE=
```

Build System Workflow

1. Define architecture, policies, patches, and configuration.
2. Fetch and download source code.
3. Extract sources into a local work area.
4. Apply patches.
5. Configure and compile software.
6. Install software into a temporary staging area.
7. Generate binary package feed.
8. Create filesystem image and SDK.

Images Generated by Poky Build

Located in tmp/deploy/images/machine/:

1. **Kernel Image** – Determines kernel file naming.
2. **Root Filesystem Image** – Built with specified types.
3. **Kernel Modules** – Tarballs with built kernel modules.
4. **Bootloaders** – If applicable, bootloaders are provided.

Example Commands:

```
bitbake -e core-image-minimal | grep ^KERNEL_IMAGETYPE=
```

```
bitbake -e core-image-minimal | grep ^IMAGE_FSTYPES=
```

Saving Disk Space

Yocto Build System can consume significant disk space. To save space, add the following to local.conf:

```
INHERIT += "rm_work"
```

Excluding specific recipes:

```
RM_WORK_EXCLUDE += "core-image-minimal"
```

Metadata

Poky = Bitbake + Metadata

Metadata is a collection of:

- Configuration files (.conf)
- Recipes (.bb and .bbappend)
- Classes (.bbclass)
- Includes (.inc)

Recipes

Non-Yocto: A recipe is a set of instructions that describe how to prepare or make something, especially a dish.

Yocto: A recipe is a set of instructions read and processed by the bitbake.

Extension of Recipe: .bb

A recipe describes:

- Where to get source code
- Which patches to apply
- Configuration options
- Compile options (library dependencies)
- Install
- License

Examples of Recipes:

- dhcp_4.4.1.bb
- gstreamer1.0_1.16.1.bb

Configuration Files

Configuration files hold:

- Global definition of variables
- User-defined variables
- Hardware configuration information

They guide the build system to construct images for a specific platform.

Extension: .conf

Types of Configuration Files:

- Machine Configuration Options
- Distribution Configuration Options
- Compiler Tuning Options
- General Common Configuration Options
- User Configuration Options (local.conf)

Classes

Class files abstract common functionality and share it among multiple recipe (.bb) files.

To use a class file, inherit the class in the recipe.

Example: inherit classname

Extension: .bbclass

Class files are usually placed in the classes directory inside the meta* directory.

Examples of Classes:

- cmake.bbclass – Handles cmake in recipes
- kernel.bbclass – Handles building kernels
- module.bbclass – Provides support for out-of-tree Linux Kernel Modules

---

Layers

A collection of related recipes or recipe containers (folders).

Naming Convention: meta-<layername>

Poky includes layers such as:

- meta
- meta-poky
- meta-selftest
- meta-skeleton
- meta-yocto-bsp

Why Use Layers?

- Isolate metadata according to functionality (BSPs, GUI, etc.).
- Simplifies future customization and reuse.

Example:

- meta-poky – Distro metadata
- meta-yocto-bsp – BSP metadata

Checking Layers Used by Poky

Command:

```
bitbake-layers show-layers
```

Get More Layers:

- Yocto Project Layers
- OpenEmbedded Layer Index

Image

An image is the top-level recipe that defines how the root filesystem is built and what packages it includes.

Command to list available image recipes:

```
ls meta*/recipes*/images/*.bb
```

Packages

Yocto Definition: A package is a binary file with extensions like .rpm, .deb, or .ipkg.

A single recipe can produce multiple packages.

Example:

```
vi meta/recipes-multimedia/libtiff/tiff_4.0.10.bb
```

```
PACKAGES += "tiffxx tiff-utils"
```
