+33 (0)1 48 10 09 46
formation@ambient-it.net
140 Avenue Jean Lolive 93500 Pantin - France

Updated on 02/04/2026

Register

Yocto Project and OpenEmbedded development training

2 days (14 hours)

Overview

Yocto Project is the industry standard for creating custom embedded Linux systems. Unlike a traditional distribution, it offers a complete framework for generating reliable, lightweight, and reproducible distributions, regardless of the target hardware.

Our Yocto Project & OpenEmbedded training will enable you to master the BitBake build tool, understand the Layers architecture, and design your own Linux distribution tailored to industrial constraints. You will learn how to configure the environment, write your own compilation recipes, and generate system images and the SDKs needed by application developers.

At the end of the training, you will be able to independently initialize a Yocto project, manage software dependencies, and maintain your embedded OS over time.

Like all our training courses, this one is based on a recent LTS (Long Term Support) version of Yocto and favors a practical approach using a QEMU simulator (transferable to a real target).

Like all our training courses, this one will introduce you to **the latest stable version** of the technology and its new features.

Objectives

- Understand the architecture of Yocto, Poky, and OpenEmbedded.
- Master the BitBake build system and configuration.
- Create and organize layers to isolate specific clients.
- Write and modify recipes (.bb and .bbappend) to integrate software.

- Generate a complete system image and its associated SDK.

Target audience

- Embedded system engineers
- Linux/Kernel developers
- Industrial software architects

Prerequisites

- Good knowledge of the Linux/Unix environment (command line)
- Understanding of the compilation process (GCC, Make)
- Basic knowledge of Python and Bash

Technical prerequisites

- Please note: Yocto requires a powerful machine for compilation.
- High-performance CPU (i7/Ryzen 7 or higher recommended)
- Minimum 16 GB of RAM (32 GB recommended)
- Minimum 150 GB of free disk space
- OS: Linux (Ubuntu LTS recommended) natively or via a high-performance virtual machine

Yocto Project & OpenEmbedded training

[Day 1 - Morning]

Introduction and First Steps with Yocto

- Overview of Embedded Linux: Buildroot vs. Yocto vs. Debian
- Vocabulary: Poky, OpenEmbedded-Core, BitBake
- Directory Structure and Metadata
- Configuring the Build Environment (Host)
- The local.conf file and global variables
- Hands-on workshop: Installing the environment and compiling a first minimal image for QEMU.

[Day 1 - Afternoon]

Organization into Layers

- Layer philosophy: modularity and reuse

- Managing layer priorities
- Using bitbake-layers
- Creating a custom "Meta-Layer" for the project
- Exploring classes and machine configuration files
- Hands-on workshop: Creating a meta-client layer and integrating it into the build.

[Day 2 - Morning]

Mastering Recipes

- BitBake Syntax and Structure of a .bb File
- The task lifecycle (fetch, unpack, configure, compile, install)
- Writing a recipe for C/C++ software (Makefile/CMake)
- Extending or modifying existing recipes with .bbappend
- Dependency management (RDEPENDS / DEPENDS)
- Hands-on workshop: Integrating a custom "Hello World" application via a new recipe.

[Day 2 - Afternoon]

Images, SDK, and Industrialization

- Customizing the final image (adding packages, users, configuration)
- Understanding BSPs (Board Support Packages) for hardware
- Generating the cross-compilation SDK for application developers
- License management and source archiving (compliance)
- Using Devtool for iterative development
- Hands-on workshop: Generating a customized final image and exporting the SDK.

Target companies

This training is intended for both individuals and companies, large or small, wishing to train their teams in a new advanced IT technology or to acquire specific business knowledge or modern methods.

Positioning at the start of training

The positioning at the start of the training complies with Qualiopi quality criteria. Upon final registration, the learner receives a self-assessment questionnaire that allows us to assess their estimated level of proficiency in different types of technologies, as well as their expectations and personal objectives for the upcoming training, within the limits imposed by the selected format. This questionnaire also allows us to anticipate certain connection or internal security issues within the company (intra-company or virtual classroom) that could be problematic for the monitoring and smooth running of the training session.

Teaching methods

Practical training: 60% practical, 40% theory. Training materials distributed in digital format

to all participants.

Organization

The course alternates between theoretical input from the trainer, supported by examples and discussion sessions, and group work.

Assessment

At the end of the session, a multiple-choice questionnaire is used to verify that the skills have been correctly acquired.

Certification

A certificate will be issued to each trainee who has completed the entire training course.