Mastering Yocto Project

If you are working with embedded Linux you will be aware that the Yocto Project is a key component for generating device firmware. This intensive four day training course will teach you how to make effective use of Yocto Project to create highly functional, reliable system images for your target device. It will show you how to select packages, how to customise system behaviour, and how to create your own meta layers. With the relevant information under your belt you will have your embedded Linux system up and running in no time!

The theory is backed up with **hands-on labs** where you will get a chance to try out all the concepts covered in the presentations. You will build a fully-functional system using a BeagleBone Black development board.

Audience

This course is ideal for anyone who needs to know about the Yocto Project and how to apply it to their work, and for anybody who is evaluating it for a future project

Prerequisites

Either a good background in embedded devices, *or* a reasonable proficiency in Linux command-line tools. Delegates with neither will find the learning curve rather steep

Duration

4 days

Course materials

All students will receive:

• Printed and electronic copies of the presentations and lab notes

• Worked solutions to the lab sessions

• A copy of the trainer's book, "Mastering Embedded Linux Programming"

About the trainer

Chris Simmonds has been using Linux in embedded systems for over 15 years. He has been running training courses and workshops in embedded Linux since 2002 and has delivered hundreds of sessions to many well-known companies in the UK, Europe, USA, South America and SE Asia. He is the author of the book "Mastering Embedded Linux Programming", and is a frequent presenter at open source and embedded conferences, including Embedded Linux Conference and Embedded World. You can see some of his work on the "Inner Penguin" blog at www.2net.co.uk

Enquiries and bookings

Please email training@2net.co.uk or call +44 (0)7788 130719

Mastering Yocto Project

Course outline

The Yocto Project

- Embedded Linux build systems
- Installing Yocto Project and building images for the BeagleBone Black
- Simple customisations: adding packages and selecting image features

Packages and package managers

- Package managers: RPM, IPK and deb
- Types of package: development, debug, release
- Deploying packages to a target

Meta layers

- Extending Yocto Project through layers
- Adding third party layers
- Creating your own meta layer
- Dependencies and layer priorities

Bitbake and recipes

- Bitbake variables and assignment
- Controlling bitbake
- Debugging: what to do when things go wrong

Bitbake tasks in detail

- What happens when you run Bitbake?
- How are tasks defined?
- How are tasks run?

Writing your own recipes

- Creating a package (application) recipe
- Declaring dependencies with other packages
- Building packages using autotools

Image recipes

- Images based on core-image
- Creating images using package groups
- Configuring locales

Extending recipes

- Overriding a recipe vs appending with bbappend
- Creating and adding a patch to a recipe
- Extending tasks with prepend and append

Kernel recipes

- Selecting a kernel
- Custom kernel configuration
- Config fragments (.cfg) and kernel features (.scc)
- Custom device tree

Creating a BSP meta layer

- Setting the machine configuration
- Customising the kernel recipe

Creating an SDK

- Creating an SDK for application developers
- What goes into the SDK?
- The extended SDK (eSDK)
- Installing and using an SDK

Improving work-flow with devtool

- Modifying local copies of a recipe
- Testing on the target
- Integrating with the main repository