

# Embedded Android + Automotive

**Android Automotive** is a version of Android tailored for In-Vehicle Infotainment (IVI) – navigation, vehicle status, audio, and more. The source code for Android Automotive is available as part of the Android Open Source Project (AOSP), on which this course is based.

The first part of the course gives a detailed background on the Android operating system, following the same curriculum as the 2net **Embedded Android** course. In this section, you will learn how to build Android from source and flash the system images using fastboot. You will learn about the Android security framework and SELinux policy. There are modules on how SurfaceFlinger renders graphics for various displays, and on the all-important topic of debugging and profiling the final system. All code and examples are based on **Android version 8.1 “Oreo”** and covers the new hardware abstraction layers introduced in **Project Treble**.

Once the ground work is in place, the later topics show how the services and features that are part of Android Automotive provide the framework for an IVI system. You will begin by building Android Automotive from AOSP and then see how it integrates with vehicle networks, such as CAN bus, via the Vehicle HAL. You will also find out about the new Exterior View System which ensures the video stream from the exterior cameras is presented to the driver within 2 seconds of booting the device.

Roughly half of the course is taken by hands-on lab sessions during which you will apply the theory to create an embedded Android device.

## Duration

5 days

## Prerequisites

Good knowledge of C/C++ and Java

Familiarity with Linux development and command-line tools

## Course materials

All students will receive:

- A printed copy of the presentations and lab notes
- Worked solutions to the problems, plus electronic copies of the course materials

## Audience

This course is ideal for system architects, engineers and project leaders who want to learn the details of Android Automotive

## About the trainer



Chris Simmonds has a wealth of experience in customizing Android for embedded systems. He has been running training courses and workshops in Android since 2010 and has trained teams at many well-known organisations 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](http://www.2net.co.uk)

## Enquiries and bookings

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

# Embedded Android + Automotive

## Course outline

### Introduction to Android

- Android architecture: the big picture
- The impact of Project Treble
- The Android Open Source Project (AOSP)
- Building Android from source

### Creating a new Android product

- Creating a new product
- Setting basic hardware characteristics
- Adding the product to the “lunch” menu

### The kernel

- Android-specific changes to Linux
- Google common kernel and vendor kernels
- Building Linux from source
- Understanding and modifying device trees

### Booting Android

- Learning how the bootloader starts Android
- Flashing images using fastboot
- Customizing the boot scripts

### The Android build system

- Building a from packages and modules
- Android makefiles (Android.mk)
- Android blue print files (Android.bp)

### The Hardware Abstraction layer

- Changes to the HAL since Project Treble
- The Vendor INterface: VINTF
- Types of HAL
  - legacy HAL
  - passthrough and Same Process
  - binderized HAL
- HIDL – the HAL Interface Definition

### Testing: CTS and VTS

- Testing the application interface using the Compatibility Test Suite, CTS
- Testing the Vendor HAL using the Vendor Test Suite, VTS

### The Android framework

- Understanding Android services
- The role of binder and AIDL in defining interfaces to services
- Using JNI to call native code

### Graphics

- How SurfaceFlinger orchestrates image rendering
- Internal and external displays
- Streaming multi-media

### Debugging and profiling

- Debugging native code
- Profiling CPU usage using perf
- Profiling power usage

### Security

- Android permissions
- Understanding and modifying SELinux policy files
- Security threats and “rooting”

### Android Automotive

- Android and IVI (In-Vehicle Infotainment)
- The architecture of Android Automotive
- Building Android Automotive from AOSP

### The Car API

- Car apps and the Car Service

### The Vehicle HAL

- Interface to vehicle buses (e.g. CAN)
- Defining and exporting Vehicle Properties
- Subscribing to vehicle events

### The Exterior View System

- Meeting the 2 second boot requirement
- The Exterior View System (EVS)
- The EVS camera and display HALs