

## Das Yocto Projekt - Ein Überblick

Ziel dieses Kurses ist es, die erforderlichen Grundlagen zu vermitteln, das Yocto Projekt zu verwenden, allerdings wird vorausgesetzt, dass Sie schon einiges an Wissen zum Thema Embedded GNU/Linux, dem GNU/Linux Kernel und Freunden mitbringen. Wir werden uns ansehen wie sowohl ein BSP oder Framework Betreuer das Yocto Projekt verwenden würde als auch ein Entwickler, der vielleicht gar nichts von dem YP zu wissen braucht bzw. auch nichts davon wissen will.

Dieses viertägige Training kombiniert Hands-on Übungen mit Theorie, um das Yocto Projekt vorzustellen. Es beantwortet häufig gestellte Fragen wie:

- Muss man wirklich für jedes Embedded GNU/Linux Projekt eine andere Version der Toolchain/ der Libraries/Pakete und einen anderen Workflow haben?
- Kann man sicher stellen dass die Entwicklungsumgebung für alle Entwickler/Zulieferer identisch ist und man die Software die heute baut auch noch identisch in 10+ Jahren bauen kann?
- Kann das YP helfen herauszufinden welche Software Lizenzen alle benutzen Pakete benutzen?
- ... und vieles mehr

Die Hands-on Sessions werden auf einer Zielhardware (z.B. Beagle Bone Black Rev. C - <http://beagleboard.org/BLACK>) durchgeführt. Die Kursteilnehmer können sich ein vorgefertigtes Docker Image mit Ubuntu 14.x und alle Beispiele nach dem Training herunterladen, was es Ihnen ermöglicht mit dem Kursmaterial im eigenen Labor zu arbeiten.

Bitte beachten Sie, dass dies kein Einführungskurs in Embedded GNU/Linux so wie [1], den wir Ihnen gerne anbieten. Sie sollten schon wissen wie Embedded GNU/Linux funktioniert und wie man einen GNU/Linux Kernel und Kernel Drivers konfiguriert und baut.

## Wer sollte teilnehmen?

Sie verwenden bereits GNU/Linux für Ihre Projekte und haben wahrscheinlich schon von dem Yocto Projekt gehört, aber sich noch nicht richtig drüber getraut. Sie wissen nicht ob und wie Sie Ihren täglichen Workflow in dem YP unterbringen können und finden mit dem YP vieles recht kompliziert. Wozu braucht man das alles eigentlich wenn es doch bis jetzt auch (vermeintlich) einfacher ging?

Sie sollten aus dem Training raus gehen und in der Lage sein zu entscheiden ob sie das YP brauchen oder nicht.

Der Workshop richtet sich an Software-, Entwicklungs-, System-Ingenieure, Tester, Administratoren, Techniker und Interessierte am YP, die bereits gut fundierte Embedded GNU/Linux Kenntnisse haben.

## Erforderliche Vorkenntnisse

- Grundkenntnisse wie man ein GNU/Linux System (z.B. Ubuntu) im User Space verwendet
- Grundkenntnisse mit dem Umgang der Command Line Shell
- Grundkenntnisse von User/Kernel Space Programmierung unter GNU/Linux
- Mittelmäßige C-Programmierkenntnisse
- Sie sollten "Embedded Linux: Von Systemarchitektur bis Echtzeit (5 Tage)" besucht haben. (dringend empfohlen! . . . es sei denn Sie sind mit dem darin vorgetragenen Material schon sehr vertraut. Um dem Training folgen zu können sollten Sie Erfahrung mit Embedded GNU/Linux mitbringen (wie man den Kernel baut, userspace, root file-system, tools).
- Es könnte auch hilfreich sein wenn Sie einen Kurs wie "Embedded GNU/Linux Device Drivers and Kernel Internals (5 days)" besucht haben, aber das ist keine erforderliche Vorkenntnis. Es reicht wenn Sie wissen wie man den GNU/Linux Kernel, Kerneltreiber in/out of tree baut und weiß, was ein fdt ist um dem Kernel Kapitel folgen zu können.
- Die oben erwähnten Trainings können on-site und on-line angeboten werden. Hier können Sie auch Angebote dazu einholen.

## Zeitlicher Ablauf

Dienstag bis Freitag von 09:00 Uhr bis 17:00 Uhr mit Kaffee- und Mittagspausen

## Kursinhalt

### Einführung

- Geschichte von Unix/Linux, Lizenzen, Standards, mit freier Software arbeiten, Spelunking, Die Unix Philosophie

### Besonderheiten

- Linux auf dem Desktop im Vergleich zu Linux auf einem Eingebetteten System, Cross-/Native Toolchains, Schon Vorgebaut, Build-Systeme, C-Bibliotheken, . . .

### Eval Board

- Wie booted es? Sehen Sie es booten

### Yocto Einführung

- Was ist das YP? Warum sollte man das YP verwenden?, Poky, BitBake, OE-Core, Metadata

### Der Yocto Autobuilder

- Was ist der YP Autobuilder?, Docker container (pull, start container)

### Der YP Workflow

- Einleitung
- Konfiguration (Benutzer, Metadaten, Maschine, Distribution Policy)
- Quellen
- Bauen (Quellen fetchen, Patchen, Konfigurieren/Kompilieren/Installieren, Pseudo, Beispiele für Rezepte, Output-Analyse/Packaging, Erzeugen von Images, Erzeugen von SDKs)
- Anpassen von Images (Intro, local.conf, IMAGE FEATURES, kundenspezifische .bb files, kundenspezifische packagegroups)

### BitBake

- Geschichte
- Syntax
- Debugging (Debug Level, Rezepte/Images/Packagegroups finden, BitBake Umgebung/Tasks/Logging, build/spezifischen Task erzwingen, cleansstate, stamp für ungültig erklären, devshell, Abhängigkeits-Explorer, alle BitBake Instanzen töten, grafische Wrapper für BitBake )

### Layers

- Intro, bitbake-layers tool, yocto-layer tool

### BSP

- Intro, Systementwicklung Workflow, BSP-Entwicklerhandbuch (bsp-tool)

### Kernel

- Intro, Systementwicklung Workflow, Kernel-Entwicklerhandbuch (defconfig, defconfig + Konfigurationsfragment, in tree kmod, out of tree kmod, fdt, . . .)

### Application Development Toolkit

- Intro, Cross-Development Toolchain, Sysroot, ADT- Eclipse Yocto Plug-in, Der QEMU Emulator, User Space Werkzeuge
- Installieren von ADT & Toolchains
- Einen Cross-Toolchain tarball verwenden (Intro, den Cross-Toolchain tarball Bauen/Installieren, Cross-Toolchain + Makefile/Autotools/Autotools lib + App, recipes)
- Den ADT Installer verwenden (Intro, Web Server, Image + ADT Installer bauen, ADT installer konfigurieren und ausführen, Den ADT Installer über einen Webserver zur Verfügung stellen, Den ADT installer am Host ausführen)

### Debugging

- Intro, gdb, gdb Remote-Debugging, (gdb Remote) Debugging mit Eclipse, (remote) Ausführen mit Eclipse

### Profiling/Tracing

- Intro, perf, gprof, gcov, strace, ftrace, systemtap, oprofile, LTTng + Eclipse (Datenvisualisierung)

### Paketverwaltung

- mit Paketen arbeiten, IPK, die Schaffung eines Paket-Feeds, Installation eines Pakets mit opkg auf der Zielhardware

### Lizenzierung

- Intro, eine benutzerdefinierte Lizenz hinzufügen, Open-Source-Lizenz-Compliance

### Devtool

- Intro, devtool, Demonstration wie man einen meta-layer für ein richtiges Projekt meta-cfengine erstellt

## Ihr Trainer

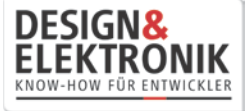
Robert Berger hat seit 1993 Praxis- und Führungserfahrung in der Industrie bei Design und Entwicklung von Software für "Eingebettete Systeme" mit und ohne harten Echtzeitanforderungen. Er arbeitet seit dem Beginn des 21. Jahrhunderts mit GNU/Linux auch im Desktop/Server-Bereich, aber hauptsächlich für "Eingebettete Systeme" (Automotive, Industrielle Steuerungen, Robotik, Telekommunikation, Konsumelektronik,...). Man trifft ihn regelmäßig bei internationalen Events wie z.B. "embedded world", "Embedded Systems Conference" oder "Embedded Linux Conference" als Experten und Vortragenden.

Seine Spezialitäten sind vorwiegend Schulungen, aber auch Beratungen (auf Deutsch oder Englisch) weltweit und zwar von kleinsten Echtzeitsystemen (FreeRTOS) bis hin zu Systemen mit mehreren Prozessoren/Cores und Embedded GNU/Linux (User-, Kernel-Space, Device Drivers, Hardware Interfacing, Debugging, Multi-core, Yocto Project,...) mit Schwerpunkt Free- und Open-Source-Software. Robert ist ein Weltenbummler. Er ist CEO & Embedded Software-Spezialist bei Reliable Embedded Systems (St. Barbara/Österreich), und wenn er keine Kundenbesuche macht, ist er bei seiner Familie in Athen/Griechenland.

# Training for Professionals

## Das Yocto Projekt - Ein Überblick

16.-19. Juli 2018 | inHouse-Workshop | WEKA FACHMEDIEN



## Anmeldecoupon

Ausfüllen, abschicken, teilnehmen.

Alle mit \* gekennzeichneten Felder sind Pflichtfelder.  
Sie erhalten eine Anmeldebestätigung per Mail.

### Kontakt:

Elena Sämann  
Tel.: + 49 (0) 89 / 255 56 – 1359  
Fax: + 49 (0) 89 / 255 56 – 0359  
Email: ESaemann@weka-fachmedien.de

Nachname *	Vorname *	Anrede *
Firma	Abteilung	Jobtitel
Straße/Hausnr.*		
PLZ *	Ort *	
Tel./Fax	Email *	

Hiermit melde ich mich verbindlich an: \*

Das Yocto Projekt – Ein Überblick

Teilnahme vom 16.-19.07.2018

Datum / Unterschrift \*

## Das Yocto Projekt - Ein Überblick

### Teilnahmegebühren

Gebühr bis 11.06.2018	€ 1.990,00
Gebühr ab 12.06.2018	€ 2.290,00

#### Teilnahmebedingungen:

Es gelten die Allgemeinen Geschäftsbedingungen unter [www.training-for-professionals.de](http://www.training-for-professionals.de).  
Die Preise verstehen sich zzgl. der gesetzl. MwSt. (19%). In den Teilnahmegebühren enthalten sind die Teilnahme an den gebuchten Tagen, Unterlagen und Teilnahmezertifikat, sowie Erfrischungen und Mittagsbuffet. Bei Stornierung der Anmeldung bis 22 Tage vor Trainingsbeginn erheben wir eine Bearbeitungsgebühr in Höhe von € 100,00 (zzgl. gesetzl. MwSt.), bei Absage ab 21 Tage vor Trainingsbeginn oder Nichterscheinen wird die gesamte Teilnahmegebühr fällig. Eine Vertretung des angemeldeten Teilnehmers ist jederzeit möglich. Der Veranstalter behält sich vor, bei Nichterreichen einer Mindestteilnehmerzahl, den Workshop abzusagen. Hierdurch entsteht kein Anspruch des Teilnehmers auf Schadensersatz. Bei Anmeldung von mind. 2 Personen einer Firma, erhält die zweite Person und jeder folgende Teilnehmer derselben Firma 10% Rabatt auf die Teilnahmegebühr.

Veranstaltungsort: WEKA FACHMEDIEN GmbH, Richard-Reitzner-Allee 2, 85540 Haar bei München

Faxen Sie den ausgefüllten Coupon an +49 (0) 89 / 255 56 – 0611 oder  
buchen Sie direkt im Internet unter [www.training-for-professionals.de](http://www.training-for-professionals.de)

Stand: 21.11.2017