

# 1

## Yocto Project

11 october 2016

- Introduction
- Embedded systems
- GNU/Linux introduction
- Toolchains
- Build systems
- Yocto Project

- MPSIT
- Yocto Project
- embedded linux
- lecture: Tuesday 4pm-6pm, room PR103
- labs: Tuesday 6pm-8pm, room PR606
- <http://ocw.cs.pub.ro/courses/mpsit>

- Alexandru Radovici, Razvan Rughinis: support
- Alexandru Vaduva: labs
- Alexandru Vaduva: lectures

- Yocto Project
- Embedded Linux
- bootloader
- Linux kernel
- toolchain
- Buildroot
- Openembedded
- open source
- u-boot
- gcc
- glibc
- binutils
- hardware-agnostic
- target machine
- host machine
- BSP

- Embedded Linux Development with Yocto Project, Otavio Salvador, Daiane Angolini, 2014
- Embedded Linux Projects Using Yocto Project Cookbook, Alex González, 2015
- Learning Embedded Linux Using the Yocto Project, Alexandru Vaduva, 2015
- Using Yocto Project with BeagleBone Black, H M Irfan Sadiq, 2015
- Yocto for Raspberry Pi, Pierre-Jean Texier, Petter Mabäcker, 2016

- **2 points** – lecture & lab involvement
- **2 points** – tests during lectures
- **5 points** – assignment
- **3 points** – final exam

- at the beginning of lectures from the previous lectures
- tests will be announced before
- make sure you check the **News** section
- 10 minutes, short questions



- first part: multiple answer questions – 20 minutes
- second part: paper test – 30 minutes
- covers all lectures
- passing grade is not mandatory

- Need to be agreed upon
- Proposed projects:
  - Testing framework for a Yocto Project Linux distribution
  - Haptic feedback framework
  - House automation graphical interface
  - House automation web interface
  - Optimize boot time & size for a Yocto Project Linux distribution
  - Security for a Yocto Project Linux distribution
  - Exploit and enhance the security holes of a Yocto Project Linux distribution

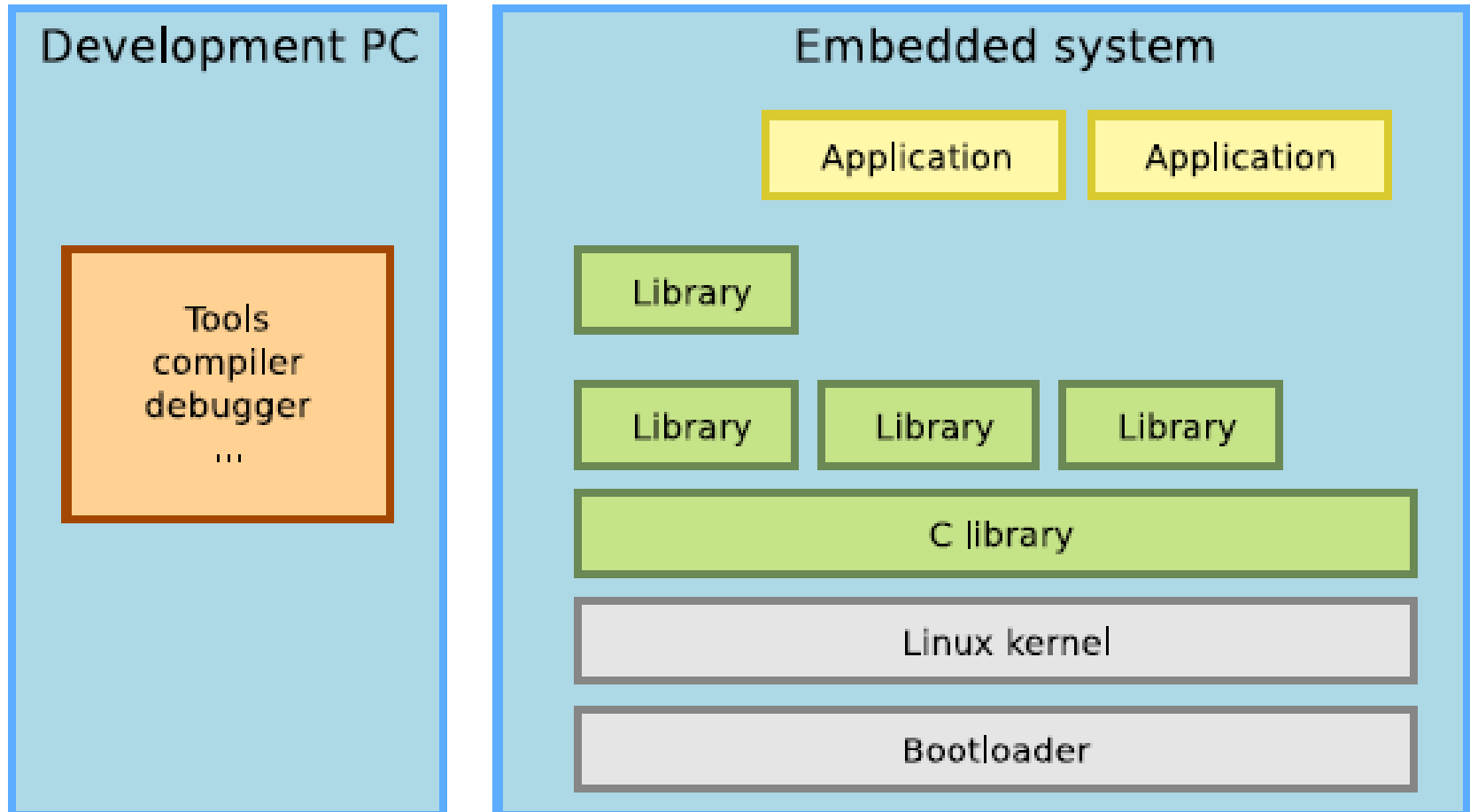
- Educational, practice
- Linux kernel patches
- Yocto Project patches
- Other open source activities
- may be equivalent with lecture & lab activity and test points

- Introduction
- Embedded systems
- GNU/Linux introduction
- Toolchains
- Build systems
- Yocto Project



- Hardware dependent components
  - Bootloader
  - Kernel
  - Toolchain
  
- Hardware independent components
  - Busybox
  - Userspace applications

- Introduction
- Embedded systems
- **GNU/Linux introduction**
- Toolchains
- Build systems
- Yocto Project

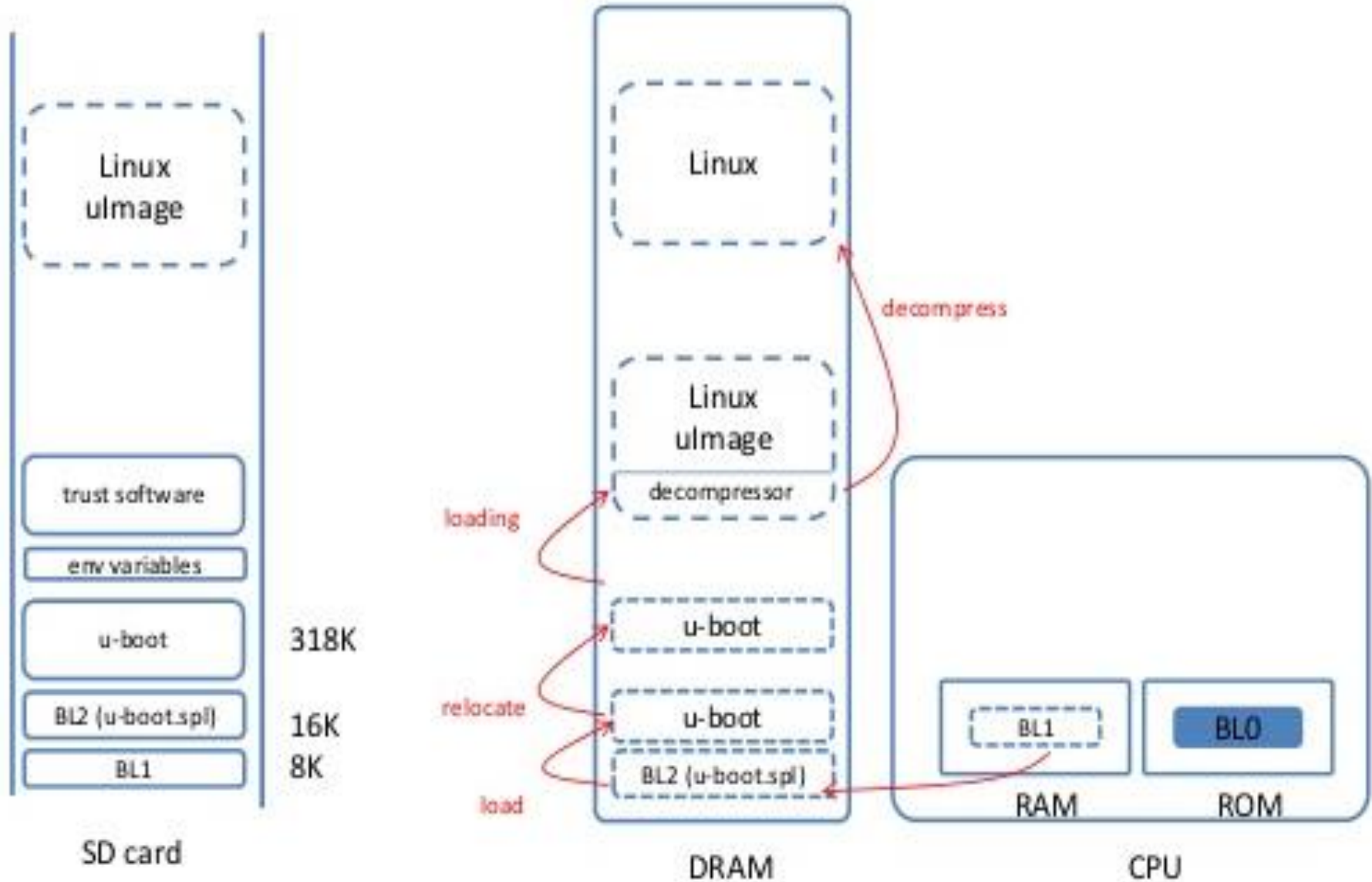


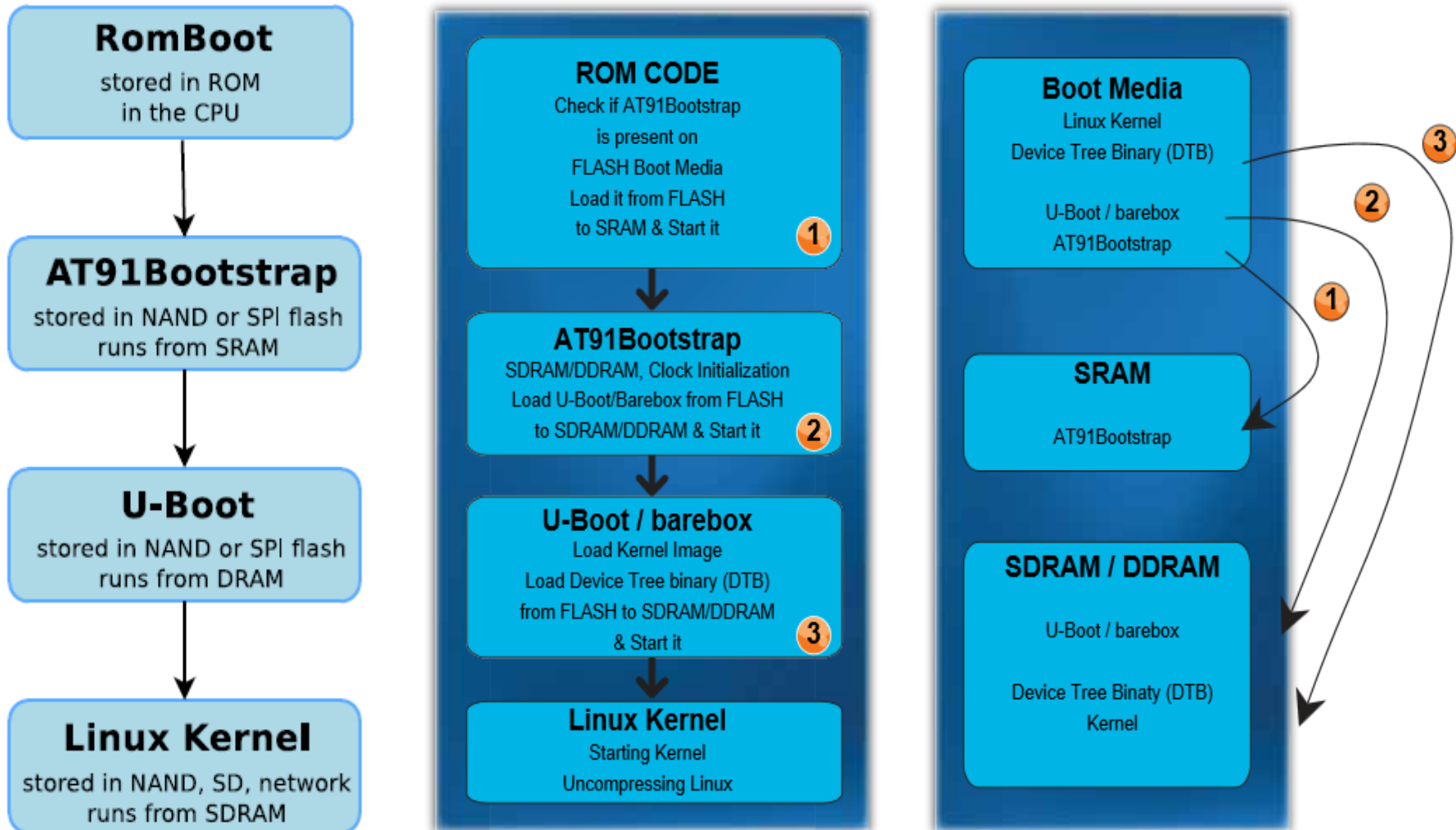


- Ubuntu 14.04
- Git (sudo apt-get install git-core)
- U-boot (<http://git.denx.de/u-boot.git>)
- Linux kernel  
(<https://git.kernel.org/pub/scm/linux/kernel/git/torvalds/linux.git>)
- C library (<http://sourceware.org/git/glibc.git>)
- Poky (<http://git.yoctoproject.org/git/poky>)

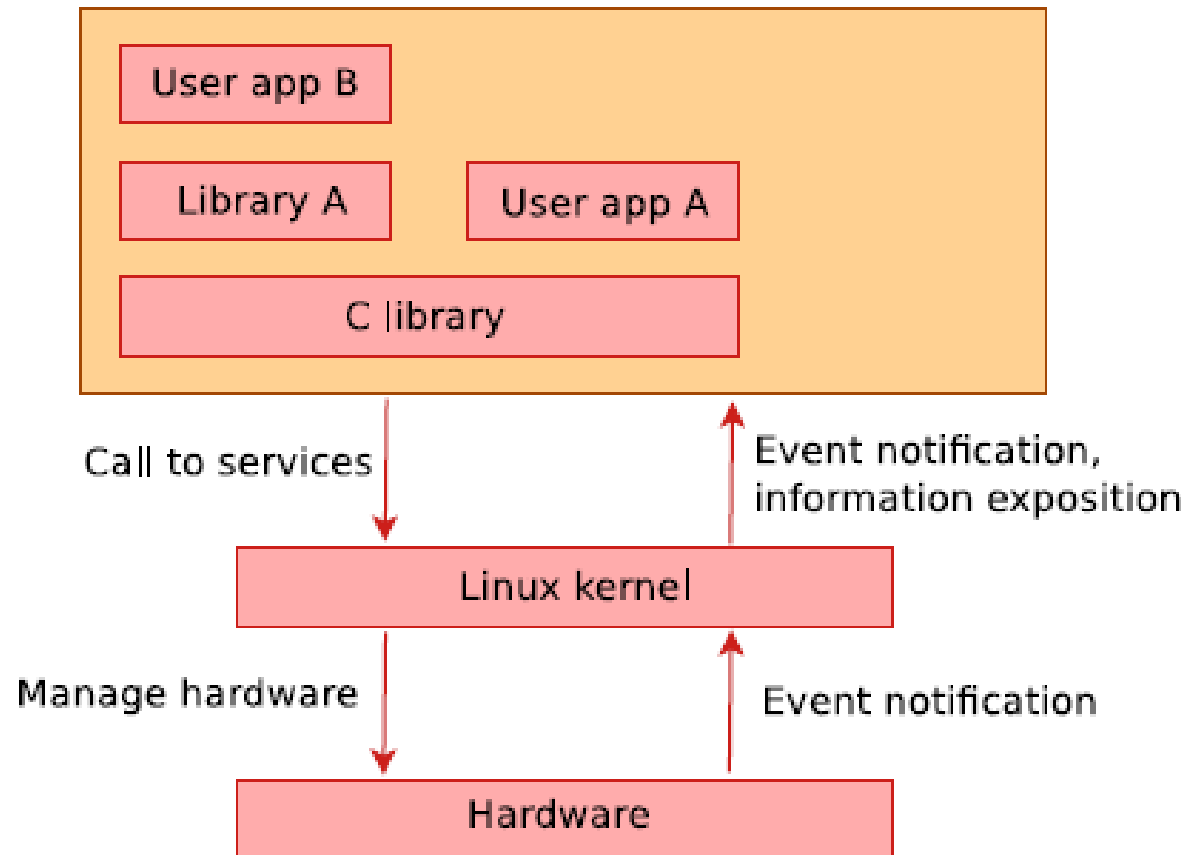
- Initialize: Bootstraps CPU, serial console, boot string
- Hardware detection and initialization
- Load the kernel image
- Load the hypervisor
- Support for:
  - Different architectures, CPU, boards and devices
  - Different file formats
  - Secure boot and trusted software

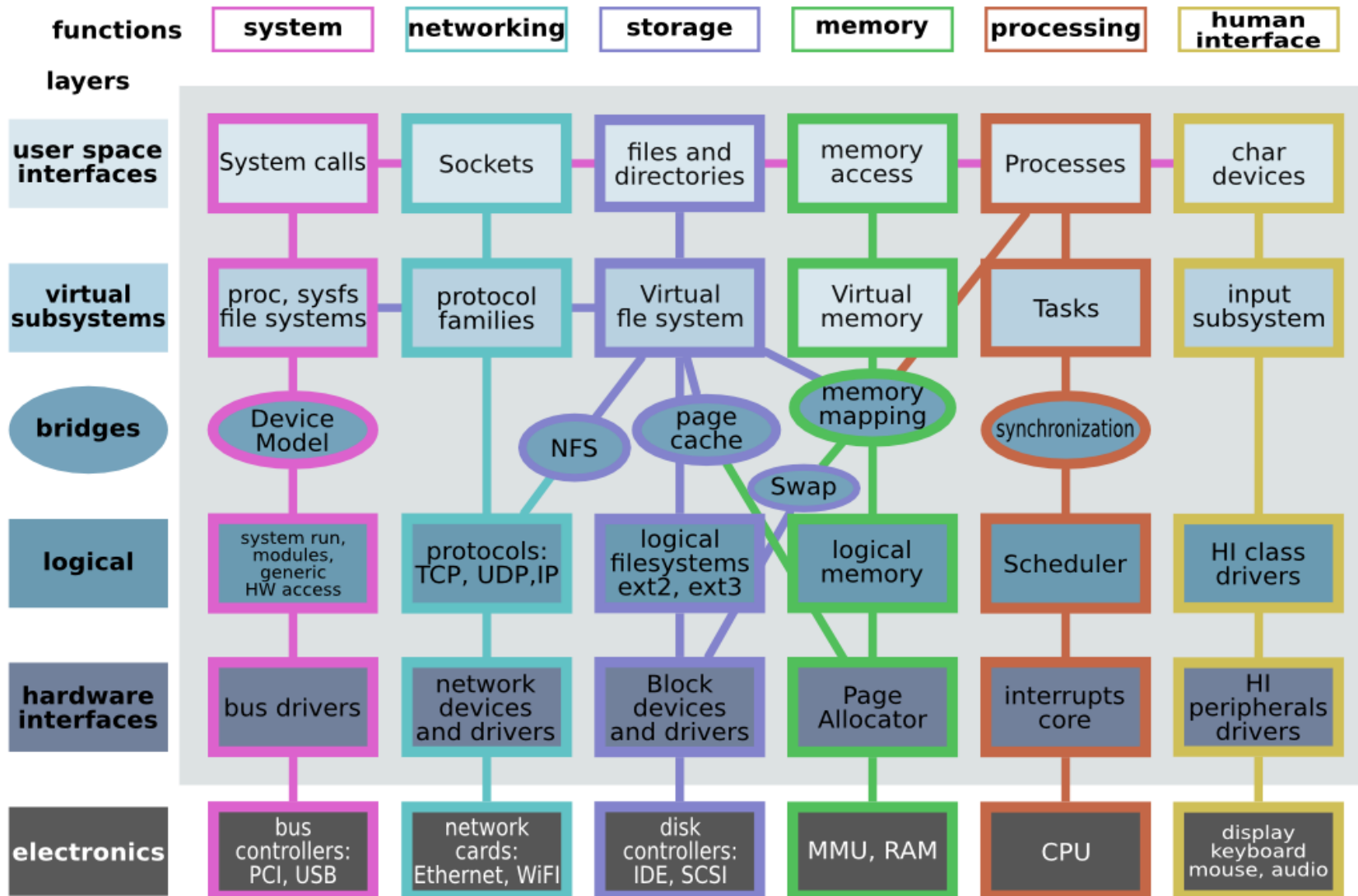
- Usually involve the use of vendor provided low level components
- Incremental initialization of resources
- Every stage is done in a different context





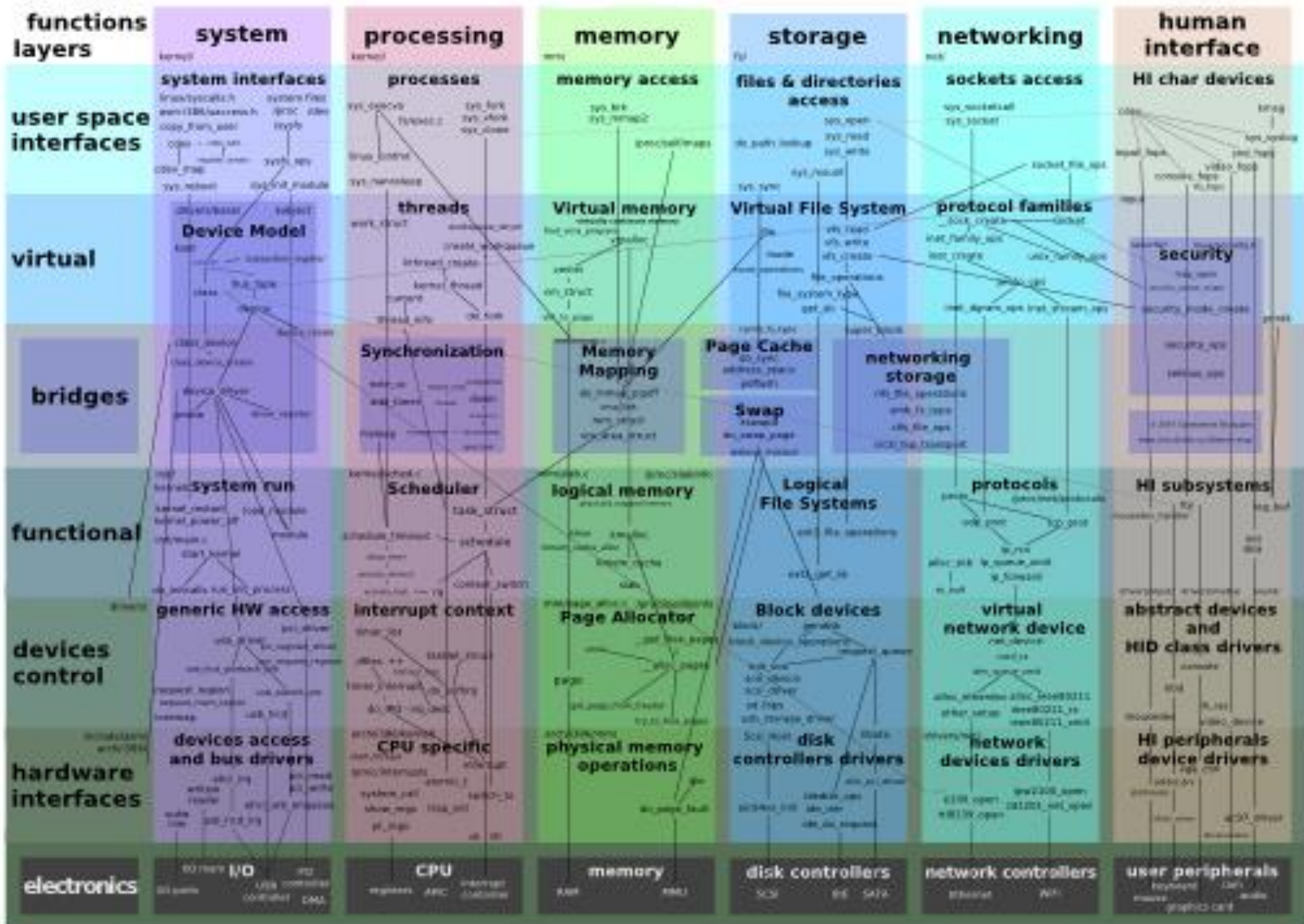
- The core of the OS
- Modules and sub-systems that provide the OS functions
- Monolithic type of kernel
- What else do we know about it?





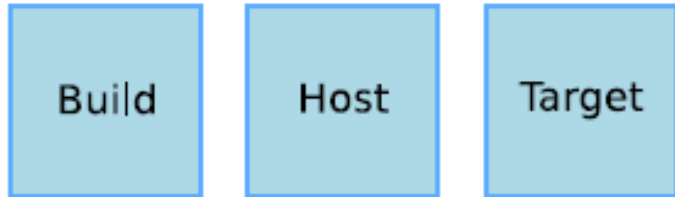
© 2007-2009 Constantine Shulyupin <http://www.MakeLinux.net/kernel/diagram>



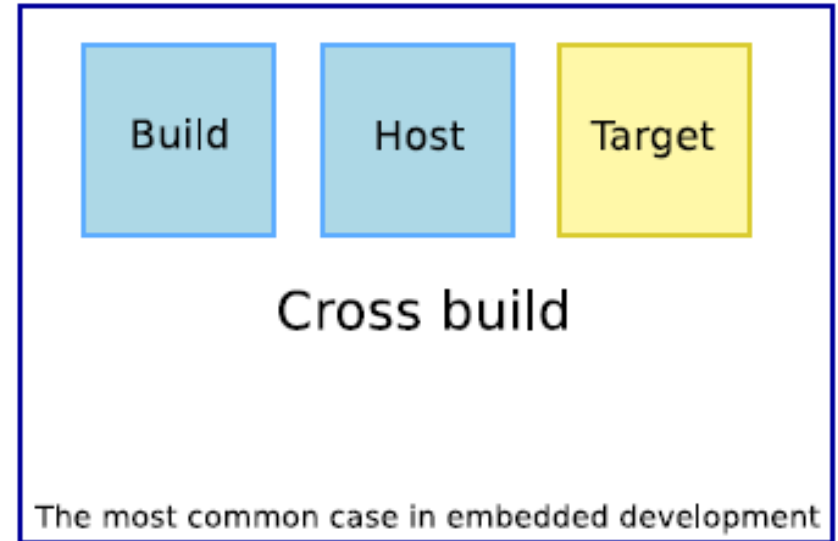


- Introduction
- Embedded systems
- GNU/Linux introduction
- **Toolchains**
- Build systems
- Yocto Project

- The usual development tool on a GNU/Linux distribution
- On embedded we work with cross-toolchains
- Target is slower and more restrictive than the host
- Is defined by a couple of components:
  - GCC compiler
  - GDB debugger (optional)
  - C/C++ libraries
  - Binutils
  - Kernel headers



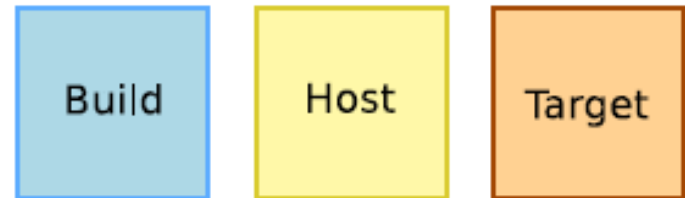
Native build



Cross build



Cross-native build



Canadian build

- C library from the GNU project
- Used by Yocto Project
- Designed for performance, standards compliance and portability
- Actively maintained
- Found on all GNU/Linux host systems

- Choosing the C library
- Choosing the configuration of the toolchain
- Choosing the version of different toolchain
- Which ABI should be used (most of them are incompatible)
- Should the toolchain support software floating point?
- Does the hardware support floating point operation?
- Should the toolchain support IPv6 or other specific features?

- Introduction
- Embedded systems
- GNU/Linux introduction
- Toolchains
- **Build systems**
- Yocto Project

- Buildroot
- OpenEmbedded
- Yocto Project
- Lots of proprietary ones



- Free software
- Embedded Linux build systems
- Build from scratch
- Use cross-compilation
- Actively developed and maintained
- Widely used in industry
- Very well documented

- Focus on simplicity
- Special use cases handled via extension scripts rather than Buildroot itself
- Re-use of existing technology
- Minimalistic by default
- Open community, no vendor or corporate management

- Main product is a root filesystem image
  - But also: bootloader, kernel image, toolchain etc.
- Main formats supported: ext2/3/4, ubifs, iso9660 etc.
- No binary packages, no package management system
  - Belief that partial updates are harmful
  - Updates are not possible via packages
  - Updates require a full system update

- Yocto Project and OpenEmbedded share the same build engine, called Bitbake
- Bitbake is inspired by Gentoo Portage build tool
- Evolved from Buildroot and it exploits its weaknesses
- Started in 2003
- Independent community driven project

- Introduction
- Embedded systems
- GNU/Linux introduction
- Toolchains
- Build systems
- **Yocto Project**

- Support for the major architectures
  - Add support for a few machines besides qemu ones
- Only provides core recipes, uses layers to get support for more packages and machines
- Custom modification stay in separate layer
- Build system as flexible as possible
- Open community governed by the Yocto Project Advisory Board

- Builds distribution, the main output being a package feed
  - The package management system is optional on the target
  - It is possible to install and update only part of the system
- Generates not only a root filesystem image but a full OS distribution. Supports many more filesystems extensions
- With the help of images classes or wic it can create disk images
- Able to generate and SDK for further application development and other purposes



WIND RIVER



ENEAA software

JUNIPER  
NETWORKS

Mentor  
Graphics

montavista™



THE  
LINUX  
FOUNDATION

Yocto Project presentation @ OSCON 2012  
© 2012 The Linux Foundation. All rights reserved.  
All logos are trademarks of their respective owners.



---

## Silicon Vendors



---

## OSVs



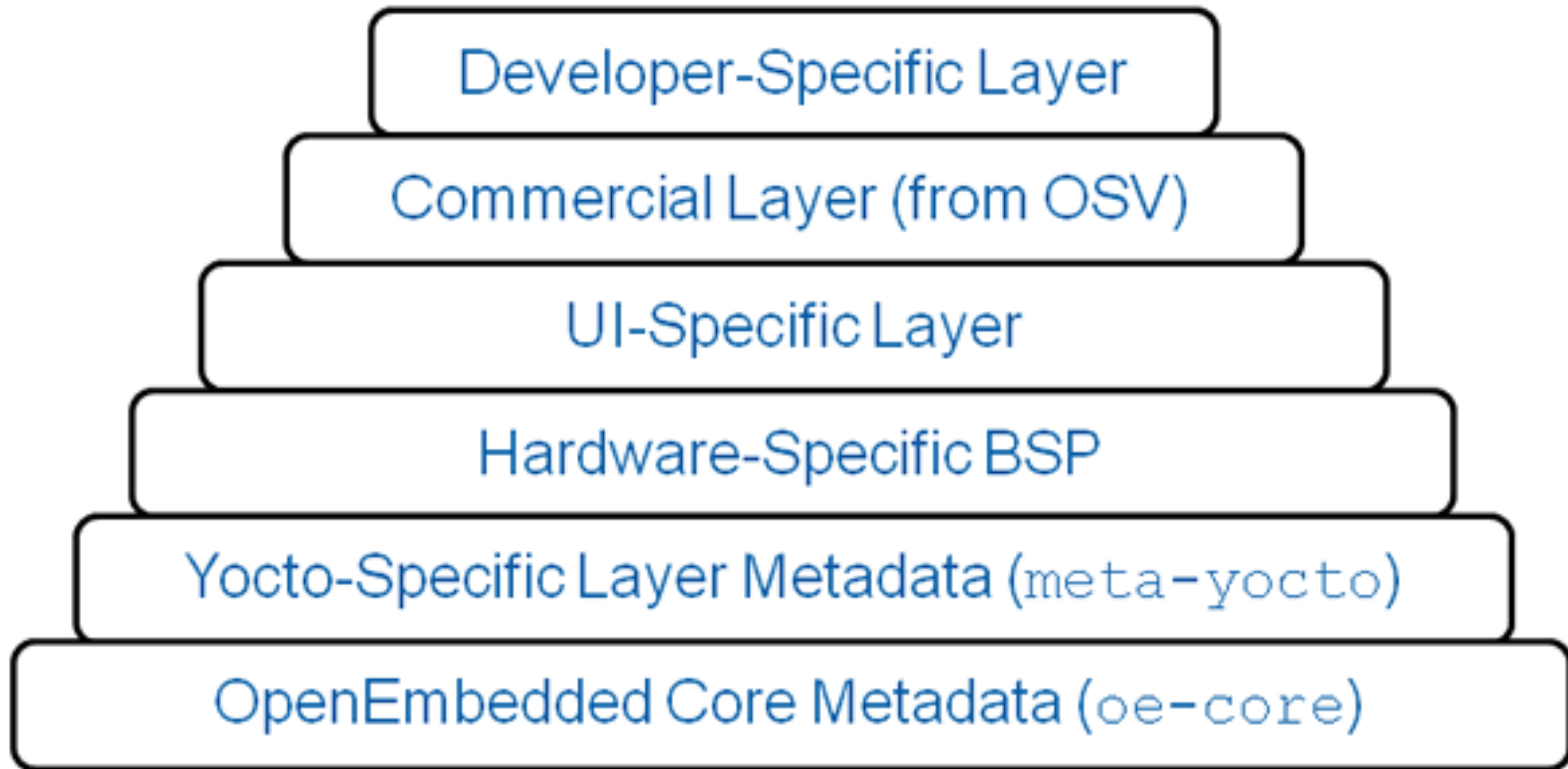
---

## Embedded Tools, Consulting Services, Users...



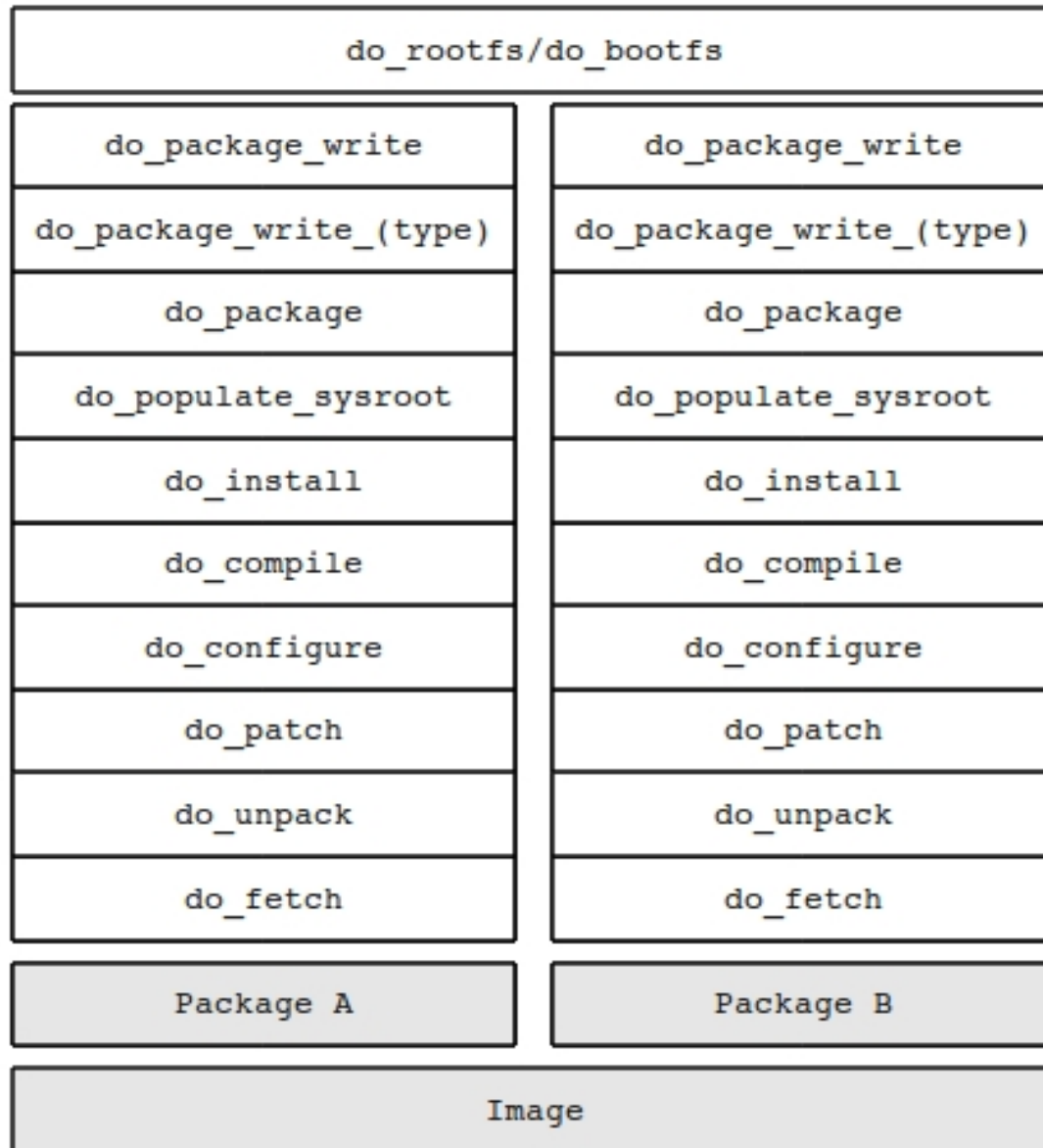
- Append files
- Bitbake
- Build directory
- Classes
- Configuration file
- Cross development toolchain
- Image
- Layer
- Metadata
- Oe-core
- Package
- Poky
- Recipe
- Source directory
- Tasks
- Upstream
- Hob

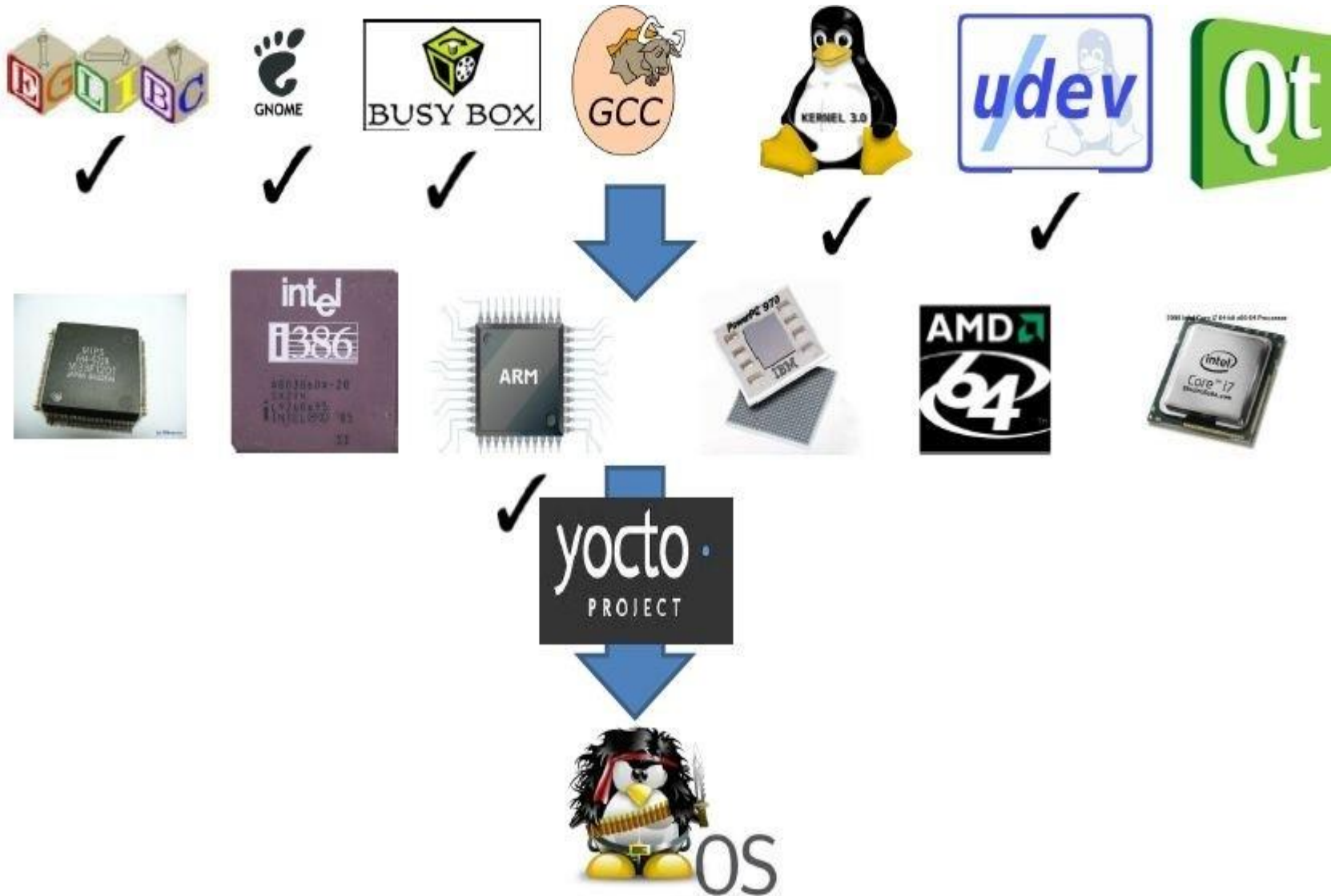
- The reference distribution for the Yocto Project
- Includes the Bitbake build system and OpenEmbedded core metadata
- Used to bootstrap your own custom Linux distribution
- After downloading a number of recipes and configuration files are visible for Bitbake to process
- Similar to OpenEmbedded & Buildroot



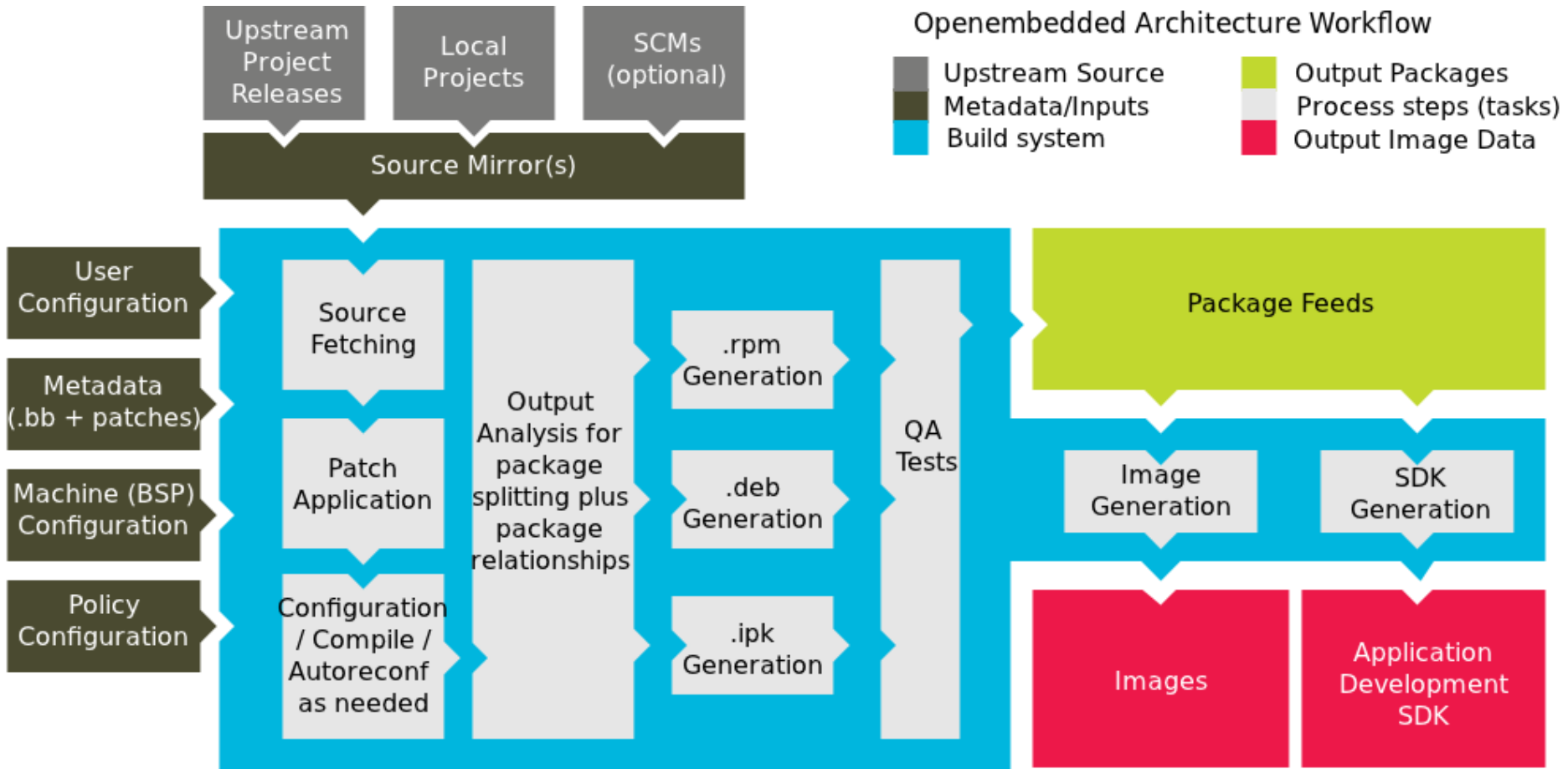
- u-boot – recipes-bsp: instructions necessary to build the u-boot package
- Linux kernel – recipes-kernel: instructions necessary to build the Linux kernel package
- gcc, binutils, glibc – recipes-devtools, recipes-core: instructions necessary to build the toolchain specific packages
- core-image-minimal: reference build image

- Machine: target specific configuration
- Distro: distribution specific configuration
- Layer.conf: layer specific configuration
- Local.conf: build directory specific configuration









- [https://www.yoctoproject.org/sites/yoctoproject.org/files/ypdd\\_2012\\_barcelona\\_intro\\_hands\\_on\\_l  
ab\\_1.pdf](https://www.yoctoproject.org/sites/yoctoproject.org/files/ypdd_2012_barcelona_intro_hands_on_l<br/>ab_1.pdf)
- [http://software.intel.com/sites/default/files/m/2/6/8/b/3/42866-08\\_Yocto\\_overview.pdf](http://software.intel.com/sites/default/files/m/2/6/8/b/3/42866-08_Yocto_overview.pdf)
- <http://www.yoctoproject.org/docs/1.3/dev-manual/dev-manual.html>
- <http://www.yoctoproject.org/docs/current/mega-manual/mega-manual.html>
- <http://layers.openembedded.org/layerindex>
- <http://www.slideshare.net/alexgonzalezgarcia/introduction-to-yocto>
- [https://wiki.yoctoproject.org/wiki/Contribution\\_Guidelines](https://wiki.yoctoproject.org/wiki/Contribution_Guidelines)
- [http://events.linuxfoundation.org/sites/events/files/slides/belloni-petazzoni-buildroot-oe\\_0.pdf](http://events.linuxfoundation.org/sites/events/files/slides/belloni-petazzoni-buildroot-oe_0.pdf)

?

