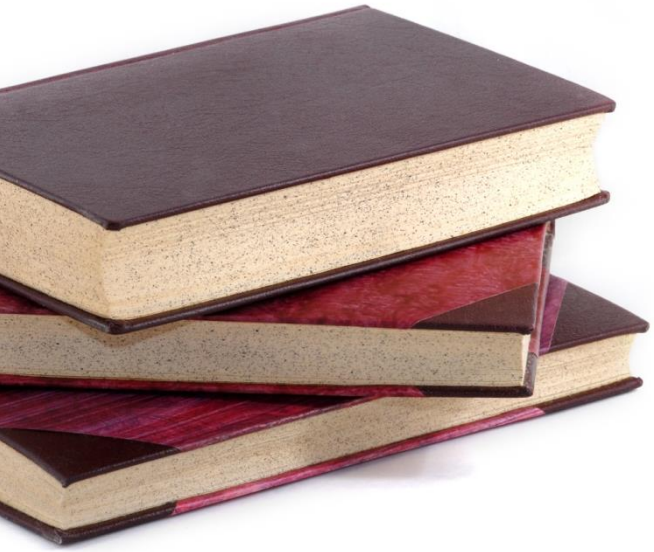
MPSIT

Lecture 6



6

Yocto Project external tools

15 noiembrie 2016

Outline

- Wic
- Swabber
- LAVA
- Build Appliance
- Matchbox
- Extra

Recap: Internal tools

- Hob: deprecated due to lack of vision towards the product
- Toaster: useful web interface and strong features behind
- Autobuilder: QA support and build testing results need to be visible for the whole community
- Devtool: system development enhancement tool, used to automate recipe generation process

Outline

- Wic
- Swabber
- LAVA
- Build Appliance
- Matchbox
- Extra

Short introduction

- Tools have nothing in common with each other
- Swabber: access detection on host
- Wic: resolves a limitation of Bitbake
- LAVA: automation testing framework
- Build Appliance: build and boot Yocto Project Linux image using a non-Linux development system
- Matchbox: X11 base environment for constrained platforms

Wic

- Standalone framework
- Extension of Bitbake
- Feature not desired as part of Bitbake
- Partitioning mechanism and description language
- Bitbake way of working:
  - genext2fs -b $ROOTFS_SIZE ... ${IMAGE_NAME}.rootfs.ext3
  - tune2fs -j ${DEPLOY_DIR_IMAGE}/${IMAGE_NAME}.rootfs.ext3
- Wic proposed another more complex solution

Bitbake implementation

- image_types.bbclass
  - IMAGE_CMD_bar = "some shell commands"
  - IMAGE_CMD_baz = "some more shell commands"
- foo-default-settings.inc
  - IMAGE_CLASSES += "image_types_foo"
- image.bbclass
  - inherit ${IMAGE_CLASSES}
  - image_types_foo.bbclass
  - IMAGE_FSTYPES

Complex filesystems

- meta-fsl-arm/imx23evk.conf
  - include conf/machine/include/mxs-base.inc
  - SDCARD_ROOTFS ?= "${DEPLOY_DIR_IMAGE}/${IMAGE_NAME}.rootfs.ext3"
  - IMAGE_FSTYPES ?= "tar.bz2 ext3 uboot.mxsboot-sdcard sdcard"
- IMAGE_CLASSES +="image_types_fsl“
- image_types_fsl.bbclass
  - inherit image_types
  - IMAGE_CMD_uboot.mxsboot-sdcard = "mxsboot sd ${DEPLOY_DIR_IMAGE}/u-boot-${MACHINE}.${UBOOT_SUFFIX} ${DEPLOY_DIR_IMAGE}/${IMAGE_NAME}.rootfs.uboot.mxsboot-sdcard"

Wic history

- Name derived from OpenEmvedded Image Creator (oeic)
- Based on Meego Image Creator (mic) framework
  - https://github.com/01org/mic
- Redhat kickstart project syntax: clearpart, part & bootloader
  - http://fedoraproject.org/wiki/Anaconda/Kickstart
- Configuration inside .wks files
  - known as Openembedded kickstart file
  - wic list images
- Implemented using pykickstarter library

Configuration file

- clearpart: clears disk of partitions
- part: the reverse operation
- Bootloader: bootloader install and part specific information handling (booting done as described in configuration file)

```
common.wks.inc
directdisk-bootloader-config.cfg
directdisk-bootloader-config.wks
directdisk-gpt.wks
directdisk-multi-rootfs.wks
directdisk.wks
mkefidisk.wks
mkgummidisk.wks
mkhybridiso.wks
qemux86-directdisk.wks
sdimage-bootpart.wks
systemd-bootdisk.wks
```

```
def pre():
    free-form python or named 'plugin' commands

  clearpart commands
  part commands
  bootloader commands
  named 'plugin' commands

  def post():
    free-form python or named 'plugin' commands
```

Plugins

- Allow for wic functionality to be extended and specialized
- Source plugins provide mechanism to customize image generation process
- Created as a subclass of SourcePlugin
- Implementation added to scripts/lib/wic/plugins/source/
- Can also be added from external layers
- At the end call plugin.get_source_plugin_methods() to get the needed method

Outline

- Wic
- Swabber
- LAVA
- Build Appliance
- Matchbox
- Extra

Swabber

- Build system contamination detection system
- Last commit since September 2011
- Still available onto Yocto Project website
  - https://www.yoctoproject.org/tools-resources/projects/swabber
- It is still presented as under development

Swabber usage

- git clone http://git.yoctoproject.org/git/swabber
- cd swabber
- make
- ./update_distro <mydistrodir>
  - Lets choose Ubuntu-distro-test

```
Ubuntu-distro-test/
├── distro
├── distro.blob
├── md5
└── packages
```

- strace -o logs/foo.log -e trace=open,execve -f <my command>
- ./swabber -c all -l logs -o required.txt -r extra.txt -p <project_dir> -d mydistrodir/ ~ /tmp

Swabber tree

```
tree swabber/
swabber/
├── BUGS
├── canonicalize.c
├── canonicalize.h
├── COPYING
├── detect_distro
├── distros
│   ├── Fedora
│   │   └── whitelist
│   ├── generic
│   │   ├── blacklist
│   │   ├── filters
│   │   └── whitelist
│   ├── Ubuntu
│   │   ├── blacklist
│   │   ├── filters
│   │   └── whitelist
│   └── Windriver
│       └── whitelist
├── dump_blob.c
├── lists.c
├── lists.h
├── load_distro.c
├── Makefile
├── packages.h
├── README
├── swabber.c
├── swabber.h
├── swabprof.c
├── swabprof.in
├── swab_testf.c
├── update_distro
├── wandering.c
└── wandering.h

5 directories, 28 files
```

Swabber output

- cat required.txt

```
# Automatically generated by swabber on Wed Nov 17 16:52:03 2010
# Distro: Ubuntu-10.04.1-64

binutils 2.20.1-3ubuntu7
# at-config:
#     /usr/lib/libopcodes-2.20.1-system.20100303.so
#     /usr/lib/libbfd-2.20.1-system.20100303.so
#     /usr/bin/as
# at-make:
#     /usr/lib/libopcodes-2.20.1-system.20100303.so
#     /usr/lib/libbfd-2.20.1-system.20100303.so
#     /usr/bin/ld
#     /usr/bin/as

dash 0.5.5.1-3ubuntu2
# at:
#     /bin/sh

sed 4.2.1-6
# at:
#     /bin/sed
```

Outline

- Wic
- Swabber
- LAVA
- Build Appliance
- Matchbox
- Extra

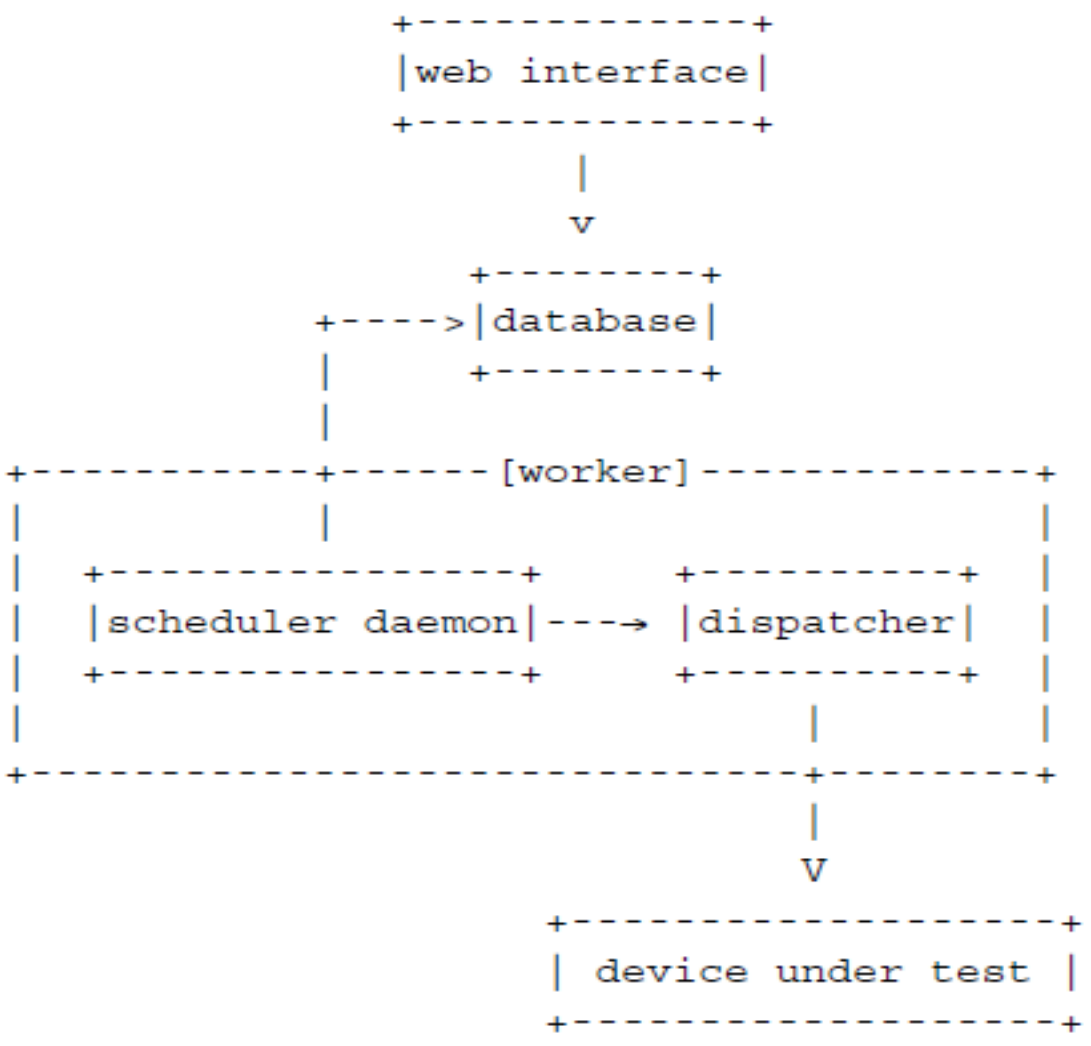
LAVA

- Linaro Automation and Validation Architecture
- Written in Python and YAML
- Continuous integration system for virtual or physical hardware deployment
- Very scalable variety of tests:
  - Boot the board
  - Board interaction
  - External hardware interaction

Functionalities

- https://www.youtube.com/watch?v=GDOgtzd0dDs
- https://validation.linaro.org/
- Scheduled automatic testing
- System recovery after crash
- Regression testing, continuous integration testing, platform enablement testing
- Support for both local and cloud solution
- Performance and power consumption measurement

Architecture



Outline

- Wic
- Swabber
- LAVA
- Build Appliance
- Matchbox
- Extra

Build Appliance

- A VM that allows you to build and boot a custom embedded Linux image with the Yocto Project using a non-Linux development system
- build-appliance-image
- Not recommended as a day-to-day production development environment
- Primary role for testing and experimenting
- https://www.yoctoproject.org/downloads/tools

Terms and concepts

- Host computer: the system on which you are running the Build Appliance
- VMware: the company that provides VMware Player and VMware Workstation applications
- OVF File: The Open Virtual Machine Format file. This file defines the Virtual Machine, including which disk image (vmdk) to use.
- Build Appliance: The Disk File running on either the VMware Player or the VMware Workstation

Setup process

- Download and unzip: http://downloads.yoctoproject.org/releases/yocto/yocto-2.2/build-appliance/Yocto_Build_Appliance.zip
  - Yocto_Build_Appliance.vmxf
  - Yocto_Build_Appliance.vmdk
  - Yocto_Build_Appliance.vmx
- Make sure you have installed any VM provider installed from the ones supported (Vmware Player, Vmware Workstation VirtualBox)

Setup process

- Set the VM accordingly:
  - Generic system configurations
  - Memory & Processor configurations
  - Network configuration in case of proxy usage
- Select Yocto_Build_Appliance.vmx for VMware or Yocto_Build_Appliance.vmdk for VirtualBox
- Start the newly configured VM
- If using Hob it will launch in the environment of not, Bitbake a terminal prompt should appear and you can use Bitbake without restrictions

Shutting down

- Exit Hob by clicking the “x” button
- Confirm exiting Hob to return to a shell prompt
- Enter the shutdown command:
  - shutdown -h now

Outline

- Wic
- Swabber
- LAVA
- Build Appliance
- Matchbox
- Extra

Matchbox

- Open source base environment for X Window System
- Wide-screen space, input mechanism or system resources limitations
- Runs on non-desktop embedded platforms
  - Handhelds
  - Set-top boxes
- Consists of a number of interchangeable applications
- Enhance usability in constrained environment

Components

- Key press simulation library
- Window management library
- Desktop & GUI
- Panels
- Terminal
- Themes
- Media Player

Matchbox repos

Matchbox	
libfakekey	Library for simulating key press events under X11
libmatchbox	Matchbox common functionality library
libmatchboxwm2	Matchbox Window Manager Library 2
matchbox-common	Matchbox shared files
matchbox-config-gtk	Matchbox GUI config tool
matchbox-desktop	Matchbox Desktop v1
matchbox-desktop-2	Matchbox Desktop v2
matchbox-documentation	Matchbox Documentation
matchbox-history	Original Matchbox repository historical record
matchbox-keyboard	Matchbox Keyboard
matchbox-panel	Matchbox Panel v1
matchbox-panel-2	Matchbox Panel v2
matchbox-panel-manager	Matchbox Panel Manager Application
matchbox-stroke	Matchbox touchscreen input method
matchbox-terminal	Matchbox Terminal
matchbox-tests	Matchbox Tests
matchbox-themes-extra	Matchbox themes
matchbox-window-manager	Matchbox Window Manager v1
matchbox-window-manager-2	Matchbox Window Manager v2
mb-applet-input-manager	Matchbox Panel Input Manager Applet
mb-applet-light	Matchbox Panel Screen Brightness Applet
mb-applet-startup-monitor	Matchbox Panel Startup Monitor Applet
mb-applet-volume	Matchbox Panel Volume Control Applet
mb-desktop-xine	Media player for Matchbox
xcursor-transparent-theme	XCursor Transparent Theme
xlib-async	
xsettings-daemon	Small gconf -> xsettings bridge

Outline

- Wic
- Swabber
- LAVA
- Build Appliance
- Matchbox
- Extra

Worth mentioning projects

- Opkg
  - Lightweight package management system - embedded devices
  - Written in C, similar to apt/dpkg, used in OpenEmbedded and OpenWrt
- Pseudo
  - Used directly or as an LD_PRELOAD to allow system administrator operations even though you are an ordinary user
  - Define file`s ownership or permission information
- Eglibc
  - Designed for embedded devices, compatible with glibc
- Cross-prelink
  - Prelinking for performance improvements

Thinks to remember

- Next class we will have the next test
  - Includes courses: 3, 4, 5 and 6
  - Similar with the previous one
- The project presentation will be done in the last week of the semester.
- The week before and the one after the vacation will be used for project related topics

The End