Resources & References:

Yocto Project
www.yoctoproject.org

Yocto Project Git Repository
https://git.yoctoproject.org/cgit/cgit.cgi/

hello.bb
ftp://ftp.gnu.org/gnu/hello/hello-2.9.tar.gz

OpenEmbedded Layers:
http://layers.openembedded.org

Toradex Git Repository
https://git.toradex.com/meta-toradex-nxp.git

Toradex BSP Files
https://developer.toradex.com/files/toradex-dev/uploads/media/Colibri/Linux/Images/

Toradex Colibri Arm Family
https://www.toradex.com/computer-on-modules/colibri-arm-family

Doulos Yocto Overview
https://www.doulos.com/yocto

Doulos Yocto Training
www.doulos.com/yoctotraining

Doulos Yocto Training Live Online
www.doulos.com/yoctotrainingol

Q&A Log for Session 1

Q&A log for Session 2 »

Audience Question:
Q: What is the difference between recipe and class?
A: A recipe is a method for compiling a particular package. A class is similar to a library in C - it contains tasks which can be inherited and used by multiple recipes.

Audience Question:
Q: Can we have the record of this webinar please?
A: Yes, we will email you a link to a streamed video recording of the presentation as soon as possible within the next week - along with the compiled Q&A log and links to useful resources.

Audience Question:
Q: Are there any advantages in any of the package types (RPM, DEB, IPK). I prefer IPK but is there any reason to prefer either of the other two?
A: Partly familiarity I guess but also IPKs use less metatdata than rpms - about 30% on average so generally would be more efficient.

Audience Question:
Q: Are there also CMake classes from which we can inherit? To support project that would build with CMake.
A: Yes, there is a cmake class which can be inherited.

Audience Question:

Q: Do all these different classes come "as standard" in Yocto, or do they have to be added/installed somehow?

A: These all come with the default installation. And you can create your own if you wish!

Audience Question:

Q: To customise / opt-in / opt-out / ... any package, is it still " make menuconfig "?

A: No, this is done by configuration files at different levels - build directory, filesystem image, distro...

Audience Question:

Q: Will Toaster be included in this webinar? It's a GUI to help visualize building, right?

A: It is a web-based GUI for monitoring and controlling builds yes. I don't think it is mentioned in any detail in this presentation.

Audience Question:

Q: How to ensure that OE core does not add unintended software etc there might be malware which OE core adds which we might not know?

A: You have complete control over what is built, and the build system provides full manifests of packages etc. The OE core packages included in Yocto should be safe.

Audience Question:

Q: What host system do you use to work with Yocto?

A: You should use one of the common Linux distros with Yocto, although there are build appliances available which can be used in Windows - see https://www.yoctoproject.org/learn-items/cross-platform-enablement-for-the-yocto-project-with-containers/

Audience Question:

Q: Why is the poky git repository not also checkout with branch rocko? The slide where all the Toradex repos where cloned, the poky was the only one without the option -b rocko?

A: In this case, perhaps the intention was to install the complete repo and then checkout the relevant branch when building.

Audience Question:

Q: Is the repo tool "limited" to git or does it also support other version control tools like svn?

A: I believe repo is intended to work with git only.

Audience Question:

Q: If you choose to work with Yocto, there should be, beside the effort to do so, an advantage in terms of time. So how many times can I actually save, if we think about a hardware architecture which will change every five years. And what are the disadvantages?

A: This is a hard question to answer but in my experience there would be a substantial saving in effort and also gains in terms of certainty - you can be sure you are building the same versions of everything for the new hardware, down to the git commit. The disadvantages are the learning curve and the amount of storage required.

Audience Question:

Q: With bitbake, can you uninstall files of some package from rootfs at dev-time?

A: Yes, this is possible - almost anything is possible. You could write what is called an append recipe to modify the installation for a given package.

Audience Question:
Q: When using the recipe tool, were the md5sum/sha256sum generated automatically or do they have to be edited?
A: They are generated automatically.

Audience Question:
Q: Full build means, I got my final image and after that I want to copy into HDD?
A: Generally, yes - the image output directory will contain everything you need to write to the SD card, HDD or whatever to boot your system. Exactly what is generated varies from machine to machine.

Audience Question:
Q: What is Yocto? Is it git or svn?
A: Yocto is a build framework for embedded Linux systems, not a source code control tool.

Audience Question:
Q: If you import a layer from openembedded, do ALL recipes get picked up into your image? Else how do you control which of the recipes in a layer get picked up?
A: By default, all recipes would be available, but they won't be included unless you specifically list them at the local build dir, image or distro configuration level.

Audience Question:
Q: You always create package (for example .ipk), or is it also possible to directly copy file to rootfs?
A: The build system uses packages internally, but files can be individually copied to the rootfs if required.

Audience Question:
Q: I read about a utility called Docker to manage a Yocto build, is Docker available?
A: Docker is a wider Linux project for virtualisation and isolated execution environments. Yocto provides a project called CROPS which uses virtual instances to allow cross platform builds, is this what you are thinking of?

Audience Question:
Q: In SDK can we have eclipse ide for that target??
A: The SDK can be used with Eclipse yes.

Audience Question:
Q: Can you build SDK that will run with windows, with cygwin or WSL or mingw?
A: Not to my knowledge - it is intended for Linux hosts.

Audience Question:
Q: Can you extract img-file at build computer (to see file structure)?
A: It depends on what format the image file is in - it might be possible.

Audience Question:
Q: While working with multiple layers, how to avoid conflicts like glibc, linuxheaders etc?
A: There is no magic solution for this - it is a case of working through the dependency issues. Generally, if you're using layers targeting the same Yocto version then there shouldn't be any issues like this.

Audience Question:
Q: How would one choose which recipes to include in a specific image like `core-image-minimal`?
A: You can easily write an append recipe for an image recipe and modify the packages included.

Audience Question:

Q: What is the Yocto training cost?

A: Can I suggest you get in touch with our team please and we'll look at your requirements. Thanks for your interest! www.doulos.com/contacts

Audience Question:

Q: Is it possible to capture a snapshot of the configuration used so that in 5 years for example the same image could be built (with same version of different packages, kernel, ...)?

A: Yes - all of the configs can be stored in a source code control system e.g. git for complete reproductability.

Audience Question:

Q: You may also simply observe sysroot directory and see what the tree of the target will look like.

A: Yes, that is also true.

Audience Question:

Q: What is the long-term support of a Yocto release?

A: Just annoucned: https://www.yoctoproject.org/yocto-project-long-term-support-announced/ - initial 2 years for LTS versions.

Session 2 Q&A log on next page

Session 2 Q&A log

Audience Question:
Q: Why is the WiFi module disabled?
A: This is just an example of how different features may be enabled or disabled on different versions of a product.

Audience Question:
Q: What is the difference between Yocto and buildroot?
A: That is difficult to answer in a few words (in fact I think we have a different webinar on the comparison!). The webinar on this subject will probably run within the next few months - we'll keep you posted about that. But broadly: buildroot can be used to generate a working system but you are limited by the packages available through the build system. Also building for new platforms effectively means starting from fresh. Yocto, as we will see, provides much more flexibility and is designed to be extendable and for building for multiple platforms.

Audience Question:
Q: How is Yocto different from Raspbian which comes on Raspberry pi?
A: Raspbian is an example of a Linux distro for an embedded system which is not designed to be extendable (beyond installing packages into a given filesystem). Yocto is for generating completely custom Linux distros for embedded systems.

Audience Question:
Q: Does Yocto support cmake as well as traditional gnu make?
A: Yes, there are classes for autotools (as shown) and cmake build systems, which will automate the configuration and build process for you.

Audience Question:
Q: So theoretically, Yocto could be used to build Raspbian?
A: Yes - as long as you knew what packages were required then you could create the relevant recipes etc. to do this.

Audience Question:
Q: A quick question in the BB recipe during install phase ... is it possible to install ipk package?
A: Yes, IPK packages are supported.

Audience Question:
Q: Can we run "make" in Yocto? Since "make" is not available in buildroot.
A: In Yocto, the build tool is called bitbake as described here. This could call make for your package if that is what is required to build it.

Audience Question:
Q: Are Poky and Bitbake both Yocto build systems? What is the difference?
A: The naming is confusing. Bitbake is the build tool (equivalent to make). Poky is used as a 'nickname' for the entire buildsystem which is bitbake plus a load of other stuff. But poky is also the name of the reference Linux distro created by Yocto by default. So....confusing.... I tend not to refer to the build system as poky - I call it the Yocto build system!

Audience Question:
Q: Can we use completely custom sources? e.g custom kernel, u-boot... etc.
A: Yes, recipes can get sources from anywhere....local files, git repos, tarball on an FTP.... so you can do anything.

Audience Question:

Q: If I have Yocto what kind of auto provisioning tool can I use?

A: I'm not aware of any support for this in the Yocto Project itself but I know that there are various systems which can be used 'on top' of Yocto to manage auto updates in the field etc. Open source projects include:

swupdate - https://sbabic.github.io/swupdate/swupdate.html

RAUC - https://rauc.readthedocs.io/en/latest/

Audience Question:

Q: What does layer mean?

A: A layer in this context is a collection of recipes - for a given vendor, or a given family of software for example.

Audience Question:

Q: For new designs, is there Linux kernel 5 support? Do you recommend using 5 or stay with 4?

A: It is always a good idea to be as up to date as you can to ensure you get the best versions of everything. You may want to consider the kernel versions in long term support as well: see:

https://www.kernel.org/category/releases.html

Audience Question:

Q: Can machine name be anything of my choice?

A: The machine name has to match a configuration file "a machine conf file" in one of the layers in use. This tells the build system what you are building for.

Audience Question:

Q: Does Yocto need to be supported and originated from the semiconductor vendor? Or, can the vendor have a typical Buildroot setup for the reference design boards sdks, and Yocto is ported to the ref boards by third parties? What is the process here?

A: Most if not all of the SoC vendors provide Yocto support at some level, providing support for their reference platforms. They may well offer support in Buildroot as well but for many Yocto is the only option.

Audience Question:

Q: Can Yocto be used for real time applications?

A: Yocto can be used for any applications, so yes.

Audience Question:

Q: How can we encrypt rootfs in Yocto build for iMX6UL?

A: The iMX6UL can offload encryption operations to CAAM when using dm-crypt. This NXP application note covers the steps for an iMX8 but can be adapted for iMX6 - https://community.nxp.com/external-link.jspa?url=https%3A%2F%2Fwww.nxp.com%2Fdocs%2Fen%2Fapplication-note%2FAN4581.pdf. However, the standard OpenEmbedded/Yocto recipes do not provide the option to automatically encrypt the rootfs. This needs to be done either manually (see https://www.yoctoproject.org/pipermail/yocto/2017-September/038113.html) or by adding an extra layer, such as meta-secure-core (needs TPM) or meta-digi.

Audience Question:

Q: Where is the core-image-minimal recipe located? And does local.conf affect it?

A: Core-image-minimal is in meta/recipes-core/images/core-image-minimal.bb. And yes, you can change the local conf file to add packages etc.

Audience Question:

Q: Why would you pick one over the other (poky, angstrom)?

A: I guess each one has its own strengths and weaknesses - I don't know enough about angstrom to know the differences. This choice is provided in the Toradex layers, it is not a generic Yocto option.

Audience Question:

Q: Can we use dm-crypt in Yocto build for entire rootfs encryption?

A: The standard Yocto build system does not provide this feature but meta-digi (https://github.com/digi-embedded/meta-digi) and meta-secure-core (with TPM, https://github.com/jiazhang0/meta-secure-core/tree/master/meta-encrypted-storage) do.

Audience Question:

Q: Do you know any technique to authenticate root file system that is available in Yocto project?

A: The Yocto build can help sign fit images (which can include an initramfs) using uboot-sign.bbclass. Alternatively, meta-digi has more extensive support for full encryption and authentication (including initramfs): http://cms.digi.com/resources/documentation/digidocs/embedded/dey/2.4/cc6/yocto-trustfence_t_secure-boot-set-up

Audience Question:

Q: Are .rpm, .deb, and .ipk mutually exclusive or would/could you use them in the same build?

A: They're not mutually exclusive and it is possible to create up to three 'package feeds' which are the repositories of rpms, ipks etc used to populate the final filesystem.

Audience Question:

Q: How do these tools avoid version-dependency-mismatch hell?

A: The way that the buildsystem works is that all of the recipes etc. are parsed as a first step and all of the dependencies are resolved (hopefully) – if not you are given logging information to show where the conflicts are so you can resolve them by e.g. writing an append recipe for an upstream package to modify which version it requests. Generally, if you don't match Yocto version numbers supported by different layers it should be fine.

Audience Question:

Q: Where can I find rules for writing bitbake recipe?

A: The Yocto documentation is very thorough and includes a bitbake reference manual which details all of the rules: https://www.yoctoproject.org/docs/3.1.1/bitbake-user-manual/bitbake-user-manual.html