



# Yocto Project® Scale your Development with Extensible-SDK

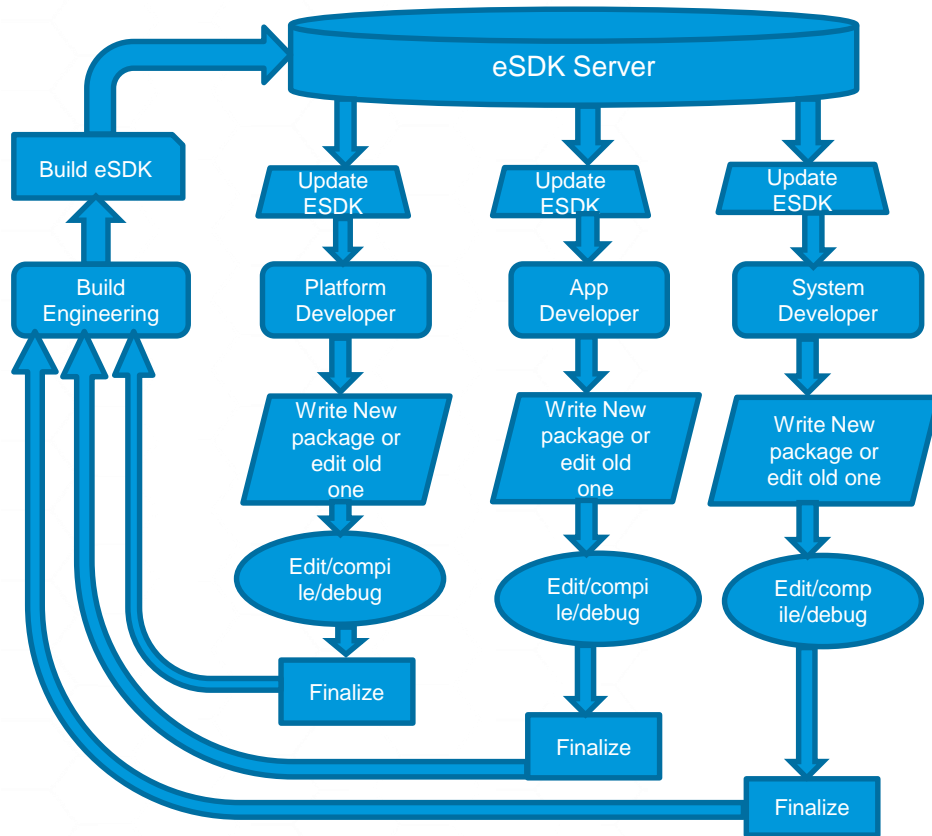
KHEM RAJ  
@himvis

**Yocto Project *Virtual* Summit Europe, October 29-30, 2020**

# Agenda

- **Introduction**
- **Building extensible SDK**
- **Deploying**
- **Using**
- **Updating and maintaining**

# 3000 Foot View





# Building

# Building eSDK

- Extensible SDK – Canned build environment
- Some Configs

```
SDK_VENDOR = "-yoesdk"
```

```
SDK_VERSION := "${DISTRO_VERSION}"
```

```
SDK_NAME_PREFIX = "yoe"
```

```
SDK_NAME = "${SDK_NAME_PREFIX}-${SDKMACHINE}-${IMAGE_BASENAME}-${TUNE_PKGARCH}-${MACHINE}"
```

```
SDKPATH = "/opt/${SDK_NAME_PREFIX}/${SDK_VERSION}"
```

```
SDKEXTPATH = "~/${SDK_NAME_PREFIX}_sdk/${DISTRO_VERSION}"
```

```
SDK_INCLUDE_NATIVESDK = "1"
```

# Building eSDK

- **Configuration**

```
# Options are full or minimal  
SDK_EXT_TYPE ?= "full"
```

```
SDK_INCLUDE_TOOLCHAIN ?= "${@'1' if d.getVar('SDK_EXT_TYPE') == 'full' else '0'}"
```

# Building eSDK

- **Build**

```
% bitbake yoe-debug-image -cpopulate_sdk_ext
```

```
% ls -l build/tmp/deploy/sdk
```

```
total 3.0G
```

```
-rw-r--r-- 2 kraj kraj 0 Oct 26 22:51 x86_64-buildtools-nativesdk-standalone-3.2.0-rc.1.host.manifest
```

```
-rwxr-xr-x 2 kraj kraj 30M Oct 26 22:52 x86_64-buildtools-nativesdk-standalone-3.2.0-rc.1.sh*
```

```
-rw-r--r-- 2 kraj kraj 0 Oct 26 22:51 x86_64-buildtools-nativesdk-standalone-3.2.0-rc.1.target.manifest
```

```
-rw-r--r-- 2 kraj kraj 346K Oct 26 22:51 x86_64-buildtools-nativesdk-standalone-3.2.0-rc.1.testdata.json
```

```
-rw-r--r-- 1 kraj kraj 6.1K Oct 26 23:01 yoe-x86_64-yoe-debug-image-riscv32-qemuriscv32-toolchain-ext-3.2.0-rc.1.host.manifest
```

```
-rwxr-xr-x 2 kraj kraj 3.0G Oct 26 23:01 yoe-x86_64-yoe-debug-image-riscv32-qemuriscv32-toolchain-ext-3.2.0-rc.1.sh*
```

```
-rw-r--r-- 1 kraj kraj 9.5K Oct 26 23:01 yoe-x86_64-yoe-debug-image-riscv32-qemuriscv32-toolchain-ext-3.2.0-rc.1.target.manifest
```

```
-rw-r--r-- 2 kraj kraj 381K Oct 26 22:58 yoe-x86_64-yoe-debug-image-riscv32-qemuriscv32-toolchain-ext-3.2.0-rc.1.testdata.json
```

- **Host on a server**



# Installing



# Installing eSDK

- Download on SDK machine
- Install eSDK

```
./build/tmp/deploy/sdk/yoe-x86_64-yoe-debug-image-riscv32-  
gemuriscv32-toolchain-ext-3.2.0-rc.1.sh
```

```
Yoe Linux Extensible SDK installer version 3.2.0-rc.1  
=====
```

```
Enter target directory for SDK (default: ~/yoe_sdk/3.2.0-rc.1): /mnt/b/yoe/3.2.0-rc.1  
You are about to install the SDK to "/mnt/b/yoe/3.2.0-rc.1". Proceed [Y/n]?  
Extracting  
SDK.....done  
Setting it up...  
Extracting buildtools...  
Preparing build system...  
Loading cache: 100% |  
Parsing recipes: 100% |##### | ETA: ----:--  
Initialising tasks: 100% |##### | Time: 0:00:20  
Checking sstate mirror object availability: 100% |##### | Time: 0:00:00  
Loading cache: 100% |##### | Time: 0:00:01  
Initialising tasks: 100% |##### | Time: 0:00:01  
done  
SDK has been successfully set up and is ready to be used.  
Each time you wish to use the SDK in a new shell session, you need to source the environment setup script e.g.  
$ . /mnt/b/yoe/3.2.0-rc.1/environment-setup-riscv32-yoe-linux
```

# Some Gotcha

- **Problem**

```
% devtool
NOTE: Starting bitbake server...
ERROR: No space left on device or exceeds fs.inotify.max_user_watches?
ERROR: To check max_user_watches: sysctl -n fs.inotify.max_user_watches.
ERROR: To modify max_user_watches: sysctl -n -w
fs.inotify.max_user_watches=<value>.
ERROR: Root privilege is required to modify max_user_watches.
...
pyinotify.WatchManagerError: add_watch: cannot watch /mnt/b/yoe/3.2.0-rc.1/conf
WD=-1, Errno=No space left on
device (ENOSPC)
```

- **Solution**

```
% sysctl -n fs.inotify.max_user_watches
524288

% sudo sysctl -n -w fs.inotify.max_user_watches=1048576
```



# Using

# Using eSDK

- **Set eSDK Environment**

```
% . /mnt/b/yoe/3.2.0-rc.1/environment-setup-riscv32-yoe-linux  
SDK environment now set up; additionally you may now run devtool to perform  
development tasks.  
Run devtool --help for further details.
```

# Using eSDK

- **Build image**

```
% devtool build-image
```

- **Run Image ( Qemu Target )**

```
% runqemu nographic
```

- **Flash Image**

- Use normal flash process for board

# Using eSDK

- **Adding a new component (sources)**
  - Create Initial source code ( not yocto metadata )
  - Perhaps use bundled toolchain
  - Create pull request to create new repo
    - 1 Github, your own git server etc.
  - Push all changes "somewhere" ( SCM system, tarball )

# Using eSDK

- **Adding a new package (recipe)**

```
[kraj@apollo /mnt/b/yoe/3.2.0-rc.1]
% . ./environment-setup-riscv64-yoe-linux
% devtool build-image
```

- **Launch Qemu in another window**

- **Add recipe Using devtool**

```
% devtool add https://github.com/jameskbride/cmake-hello-world
% devtool edit-recipe cmakehelloworld
% devtool build cmakehelloworld
% devtool edit-recipe cmakehelloworld
% devtool deploy-target cmakehelloworld qemu
```

# Using eSDK – Prepare layer

- **Create git repo**

```
$ cd /mnt/b/yoel/3.2.0-rc.1/layers/meta-openembedded  
$ git init . && git add . && git commit -m"init"
```

- **Add Upstream remote**

```
git remote add github git@github.com:yoedistro/meta-openembedded
```



# Using eSDK

- **Add recipe to platform**

```
% devtool finish cmakehelloworld layers/meta-openembedded/meta-oe
```

- **Upload patch for review**

```
% git add . && git commit -sv -m "cmakehelloworld: Add recipe"
```

- **Adapt to your patch workflow system**

```
% git fetch github && git checkout -b kraj/esdk-foo github/master && git cherry-pick master  
% git push github HEAD:kraj/esdk-foo  
% git format-patch -s --subject-prefix='meta-oe'][PATCH' github/master && git send-email -to ...
```

- **Add this recipe to image e.g. IMAGE\_INSTALL**

- **Regenerate eSDK and re-publish**

# Using eSDK

- **Modify existing package**

```
% devtool modify cmakehelloworld  
% devtool build cmakehelloworld
```

- **Edit sources, make changes**

- **Edit->compile->debug**

- **Push changes to component via infra/CI**

```
--- a/HelloWorld.cpp  
+++ b/HelloWorld.cpp  
@@ -4,7 +4,7 @@ using namespace std;  
using namespace Hello;  
int main(int argc, char *argv[]) {  
- Speaker* speaker = new Speaker();  
+ auto speaker = Speaker{};  
- speaker->sayHello();  
+ speaker.sayHello();  
}
```



# Updating

# Update eSDK

- **Configuring SDK Update server**
  - Add to `conf/sdk-extra.conf`

```
SSTATE_MIRRORS = "file://.* http://10.0.0.10:8000/sstate-cache/PATH"  
SDK_UPDATE_URL = "http://10.0.0.10:8000"
```

# Update eSDK

- Publish eSDK

```
% oe-publish-sdk tmp/deploy/sdk/yoe-x86_64-yoe-debug-image-riscv32-  
qemuriscv32-toolchain-ext-3.2.0-rc.1.sh /mnt/b/yoe/sdk_feeds
```

```
INFO: Copying the SDK to destination  
INFO: Unpacking SDK  
Yoe Linux Extensible SDK installer version 3.2.0-rc.1  
=====
```

You are about to install the SDK to "/tmp/xxx". Proceed [Y/n]? Y

```
Extracting  
SDK.....  
.....done  
Setting it up...  
Extracting buildtools...  
done  
SDK has been successfully set up and is ready to be used.  
Each time you wish to use the SDK in a new shell session, you need to source the environment setup  
script e.g.  
$ . /tmp/xxx/environment-setup-riscv32-yoe-linux  
INFO: Successfully unpacked /tmp/xxx/yoe-x86_64-yoe-debug-image-riscv32-qemuriscv32-toolchain-ext-  
3.2.0-rc.1.sh to /tmp/xxx  
Initialized empty Git repository in /tmp/xxx/layers/.git/  
INFO: SDK published successfully
```

# Update eSDK

- **Setup SDK Feed Server**

```
% cd /mnt/b/yoel/sdk_feeds  
% python3 -m http.server 8000
```

# Update eSDK

- Update eSDK

```
% devtool sdk-update http://10.0.0.10:8000
```

```
NOTE: Starting bitbake server...
```

```
NOTE: Started PRServer with DBfile: /mnt/b/yoe/3.2.0-rc.1/cache/prserv.sqlite3, IP: 127.0.0.1, PORT: 36659, PID: 778056
```

```
NOTE: Reconnecting to bitbake server...
```

```
NOTE: Retrying server connection (#1)...
```

```
NOTE: Reconnecting to bitbake server...
```

```
NOTE: Reconnecting to bitbake server...
```

```
NOTE: Retrying server connection (#1)...
```

```
NOTE: Retrying server connection (#1)...
```

```
NOTE: Starting bitbake server...
```

```
NOTE: Started PRServer with DBfile: /mnt/b/yoe/3.2.0-rc.1/cache/prserv.sqlite3, IP: 127.0.0.1, PORT: 46785, PID: 965890
```

```
Cloning into 'layers'...
```

```
Fetching objects: 17199, done.
```

```
INFO: Preparing build system... (This may take some time.)
```

# Update eSDK

- Install a new component

```
% devtool sdk-install clang
```

```
% devtool sdk-install --help
```

```
NOTE: Starting bitbake server...
```

```
NOTE: Started PRServer with DBfile: /mnt/b/yoee/3.2.0-
```

```
rc.1/cache/prserv.sqlite3, IP: 127.0.0.1, PORT: 34097, PID: 968877
```

```
usage: devtool sdk-install [-h] [-s] recipename [recipename ...]
```

Installs additional recipe development files into the SDK. (You can use "devtool search" to find available recipes.)

arguments:

    recipename                  Name of the recipe to install the development artifacts for

options:

    -h, --help                  show this help message and exit

    -s, --allow-build          Allow building requested item(s) from source



A decorative pattern of semi-transparent grey hexagons is located in the upper-left corner of the slide.

# Thanks for your time

yocto  
PROJECT

THE  
LINUX  
FOUNDATION



yocto  
PROJECT

THE  
LINUX  
FOUNDATION