

# Containerization using META DAC SDK in RDK-B RPI

## On this Page:

- [Scope](#)
- [Enable Dobby support in Rpi RDK-B build](#)
- [Include DAC SDK in Rpi RDK-B dunfell build](#)
- [Create platform specific lib json to generate container configuration for respective platform](#)
- [Generate OCI bundle from the OCI image using Bundlegen tool](#)
- [Debugging Guide](#)

## Scope

Objective is to create containerized application based on OCI standard and managing the application using Dobby tool in Rpi for RDK-B build.

Things to do:

- Enable Dobby support in Rpi RDK-B build.
- Include DAC SDK in Rpi RDK-B Code.
- Create an OCI image for a sample application using meta-dac-sdk
- Create platform specific lib json to generate container configuration for respective platform
- Install Bundlegen in a VM or any host machine
- Generate OCI bundle from the OCI image using Bundlegen tool
- Start/Stop OCI bundle using Dobby

## Enable Dobby support in Rpi RDK-B build

- Rpi Dunfell repo init and build instruction - [RDK-B \(Raspberry Pi\) Build and Setup Manual - Router Profile](#)
- Add Dobby package in RDK-B build - **meta-cmf-raspberrypi/recipes-core/packagegroups/packagegroup-rdk-ccsp-broadband.bbappend**

Note : From [RDKB\\_2023q2\\_dunfell](#) branch below changes are not needed.

```
+++ b/recipes-core/packagegroups/packagegroup-rdk-ccsp-broadband.bbappend
@@ -6,6 +6,9 @@ RDEPENDS_packagegroup-rdk-ccsp-broadband_append = "\
    libseshat \
    notify-comp \
    start-parodus \
+   doobby \
+   crun \
    \
"
```

- The kernel Image and root filesystem will be created under the `./tmp/deploy/images/raspberrypi*` folder.
- Flash the image : [RDK-B \(Raspberry Pi\) Build and Setup Manual - Router Profile](#)
- Verify doobby service is up and running

```

root@RaspberryPi-Gateway:~# systemctl status dooby
dooby.service - RDK Dobby (Container) daemon
   Loaded: loaded (/lib/systemd/system/dooby.service; enabled; vendor preset: enabled)
   Active: active (running) since Wed 2023-08-23 05:47:12 UTC; 1 months 17 days ago
     Main PID: 233 (DobbyDaemon)
       Tasks: 10 (limit: 3092)
      Memory: 2.8M
     CGroup: /system.slice/dooby.service
            233 /usr/sbin/DobbyDaemon --nofork --noconsole --journal Aug 23 05:47:11 RaspberryPi-Gateway
DobbyDaemon[233]: NFO: Logging socket created at '/tmp/dobbyPty.sock'
Aug 23 05:47:11 RaspberryPi-Gateway DobbyDaemon[233]: NFO: entered Ipc service change thread
Aug 23 05:47:11 RaspberryPi-Gateway DobbyDaemon[233]: NFO: Created log relay from /tmp/dobby-syslog to /dev
/log
Aug 23 05:47:11 RaspberryPi-Gateway DobbyDaemon[233]: NFO: Created log relay from /tmp/dobby-journald to /run
/systemd/journal/socket
Aug 23 05:47:11 RaspberryPi-Gateway DobbyDaemon[233]: NFO: Dobby Logger socket monitoring thread started
Aug 23 05:47:11 RaspberryPi-Gateway DobbyDaemon[233]: NFO: entered Ipc service change thread
Aug 23 05:47:11 RaspberryPi-Gateway DobbyDaemon[233]: NFO: started SIGCHLD monitor thread
Aug 23 05:47:11 RaspberryPi-Gateway DobbyDaemon[233]: NFO: starting watchdog timer with period 7500000
Aug 23 05:47:11 RaspberryPi-Gateway DobbyDaemon[233]: MIL: started Dobby daemon
Aug 23 05:47:12 RaspberryPi-Gateway systemd[1]: Started RDK Dobby (Container) daemon.
root@RaspberryPi-Gateway:~#

root@RaspberryPi-Gateway:~# DobbyTool -V
Version: 3.6.3-e747921419aca764d3f758be2a3b2f39f3a024a0
root@RaspberryPi-Gateway:~#

```

- Verify the crun version

```

root@RaspberryPi-Gateway:~# crun -V
crun version 1.7.2
commit: 0356bf4aff9a133d655dc13b1d9ac9424706cac4
rundir: /run/crun
spec: 1.0.0
+SELINUX +APPARMOR +CAP +SECCOMP +EBPF +YAJL
root@RaspberryPi-Gateway:~#

```

- RPI4 32 Bit Binary : dooby , crun enabled : [rdkb-generic-broadband-image\\_rdk-next\\_20230823054516.rootfs.wic.bz2](https://rdkcentral.com/r/rdkcmf/manifests)

## Include DAC SDK in Rpi RDK-B dunfell build

- Rpi Dunfell repo init instruction - [RDK-B \(Raspberry Pi\) Build and Setup Manual - Router Profile](#)
- Download DAC SDK in Rpi Workspace - `git clone https://github.com/stagingrdkm/meta-dac-sdk.git`

```

repo init -u https://code.rdkcentral.com/r/rdkcmf/manifests -m rdkb-extsrc.xml -b dunfell
repo sync -j `nproc` --no-clone-bundle --no-tags
git clone https://github.com/stagingrdkm/meta-dac-sdk.git

```

- Include the meta-dac-sdk to bblayers : `conf/bblayer.conf`

Refer to build app using meta-dac-sdk toolchain - [Quickstart: Building, Installing and Running DAC apps#SetupSDKandbuildyourfirstDACapp](#)

## Create an OCI image for a sample application using meta-dac-sdk



helloworld-test.tgz

Add below files inside meta-dac-sdk for building sample "helloworld-test" application

```
classes/dac-image-helloworld.bbclass
recipes-example/example/helloworld-test.bb
recipes-example/example/helloworld-test/
recipes-example/example/helloworld-test/main.cpp ( main.cpp )
recipes-example/example/helloworld-test/LICENSE
recipes-example/images/dac-image-helloworld-test.bb
recipes-example/images/metadatas/helloworld-test-appmetadata.json
```

To create OCI Image for the sample application, run the respective recipe

```
MACHINE=MACHINE=raspberrypi4-rdk-broadband source meta-cmf-raspberrypi/setup-environment
bitbake dac-image-helloworld-test
```

Image path : build-raspberrypi-rdk-broadband/tmp/deploy/images/raspberrypi-rdk-broadband\*/

Image name : dac-image-\*.oci-image.tar

Example - build/tmp/deploy/images/raspberrypi3/dac-image-helloworld-test-raspberrypi3-20230325195708.rootfs-oci-latest-arm-linux.oci-image.tar

Note : In latest rdk-b releases meta-qt5 layer is not included , while building one might face the below issue

Issue observed in [RDKB\\_2023q2\\_dunfell](#) and rdk-next branch.

```
ERROR: ParseError at /home/dbhat/dac_check_2023q2_64bit/meta-dac-sdk/recipes-example/example/qt-egl-test.bb:
24: Could not inherit file classes/qmake5.bbclass
To resolve this error , follow any of below approach.
```

- remove/comment the line "inherit qmake5" in [qt-egl-test.bb](#) recipe file
- download meta-qt5 and include the layer in bblayer.conf

## Create platform specific lib json to generate container configuration for respective platform

Refer - <https://github.com/rdkcentral/BundleGen/blob/master/docs/TemplateLibs.md>

**\*\*Note** : Already available for Rpi3, support for Rpi4 is in progress

## Install Bundlegen in a VM or any host machine

Reference : [Containerization feature integration in Raspberry-Pi RDK Video stack](#) [ Currently the page is restricted to RDKM ]

Requirement : Ubuntu 22

Reference for Bundlegen : <https://github.com/rdkcentral/BundleGen>

#### # Install dependencies

```
$ sudo apt update
$ sudo apt upgrade
$ sudo apt install -y make git go-md2man
# Install Go 1.13
$ wget https://dl.google.com/go/go1.13.12.linux-amd64.tar.gz
$ tar -xvf go1.13.12.linux-amd64.tar.gz
$ sudo mv go /usr/local
$ mkdir -p $HOME/go
$ echo 'export GOPATH=$HOME/go' >> $HOME/.bashrc
$ echo 'export GOROOT=/usr/local/go' >> $HOME/.bashrc
$ echo 'export PATH=$GOPATH/bin:$GOROOT/bin:$PATH' >> $HOME/.bashrc
$ source $HOME/.bashrc
# Build/install umoci
$ go get -d github.com/opencontainers/umoci
$ cd $GOPATH/src/github.com/opencontainers/umoci/
$ make
$ sudo make install
# Install skopeo
$ . /etc/os-release
$ sudo sh -c "echo 'deb
$ sudo apt update && sudo apt install skopeo
```

#### Install Bundlegen

```
git clone https://github.com/rdkcentral/BundleGen.git
cd BundleGen
$ python3 setup.py build
$ python3 setup.py install
```

### Generate OCI bundle from the OCI image using Bundlegen tool

- Copy the oci image (dac-image-\*.oci-image.tar) to local VM where Bundlegen is installed.
- Run `./test/build_and_test_on_rpi.sh <VM_IP> dac-image-*.oci-image.tar`
- Above step will generate an oci bundle image ( .gz), and uncompressed folder.  
example : `rpi3_reference-dac-image-helloworld-test-raspberrypi-rdk-broadband-20230325195708.rootfs-oci-latest-arm-linux.oci-image.tar.gz`

Note : Once the OCI bundle is created , verify the **config.json** for extra mount options and extra libraries, remove them from config.json.

### Start/Stop OCI bundle using Dobby

To run the OCI bundle using doobby, copy the above generated bundle to Rpi board

```

cd /
mkdir dac
copy bundle image to /dac
tar -xvzf rpi3_reference-dac-image-helloworld-test-raspberrypi-rdk-broadband-20230325195708.rootfs-oci-latest-arm-linux.oci-image.tar.gz
chmod -R 744 rpi3_reference-dac-image-helloworld-test-raspberrypi-rdk-broadband-20230325195708.rootfs-oci-latest-arm-linux.oci-image
DobbyTool start helloworld-test rpi3_reference-dac-image-helloworld-test-raspberrypi-rdk-broadband-20230325195708.rootfs-oci-latest-arm-linux.oci-image/
DobbyTool list

```

```

root@RaspberryPi-Gateway:/# cd dac/

```

```

root@RaspberryPi-Gateway:/dac# ls

```

```

rpi3_reference-dac-image-helloworld-test-raspberrypi-rdk-broadband-20230325195708.rootfs-oci-latest-arm-linux.oci-image

```

```

root@RaspberryPi-Gateway:/dac# chmod -R 777 rpi3_reference-dac-image-helloworld-test-raspberrypi-rdk-broadband-20230325195708.rootfs-oci-latest-arm-linux.oci-image/
root@RaspberryPi-Gateway:/dac#

```

```

root@RaspberryPi-Gateway:/dac# DobbyTool -v start helloworld-test rpi3_reference-dac-image-helloworld-test-raspberrypi-rdk-broadband-20230325195708.rootfs-oci-latest-arm-linux.oci-image/
0000000260.953013 NFO: < M:Main.cpp F:main L:1264 > starting dbus service
0000000260.953307 NFO: < M:Main.cpp F:main L:1265 > bus address 'unix:path=/var/run/dbus/system_bus_socket'
0000000260.953533 NFO: < M:Main.cpp F:main L:1266 > service name 'org.rdk.dobby.test.pid14388'
0000000260.958383 NFO: < M:SDBusIpcService.cpp F:eventLoopThread L:1287 > started sd-bus event loop thread
0000000260.958559 NFO: < M:DobbyProxy.cpp F:containerStateChangeThread L:1165 > entered container state change thread
0000000260.959260 NFO: < M:SDBusIpcService.cpp F:eventLoopThread L:1318 > starting sd-bus event loop
started 'helloworld-test' container, descriptor is 734
0000000261.416063 NFO: < M:DobbyProxy.cpp F:containerStateChangeThread L:1224 > exiting container state change thread
0000000261.417027 NFO: < M:SDBusIpcService.cpp F:eventLoopThread L:1323 > stopping sd-bus event loop

```

```

root@RaspberryPi-Gateway:/dac#

```

```

root@RaspberryPi-Gateway:/dac#

```

```

root@RaspberryPi-Gateway:/dac# DobbyTool list

```

```

descriptor | id | state
-----|-----|-----
734 | helloworld-test | running

```

```

root@RaspberryPi-Gateway:/dac#

```

```

root@RaspberryPi-Gateway:/dac#

```

```

root@RaspberryPi-Gateway:/dac#

```

```

root@RaspberryPi-Gateway:/dac# cat rpi3_reference-dac-image-helloworld-test-raspberrypi-rdk-broadband-20230325195708.rootfs-oci-latest-arm-linux.oci-image/rootfs/my_file.txt
*****Hello World *****

```

```

root@RaspberryPi-Gateway:/dac#

```

```

root@RaspberryPi-Gateway:/dac#

```

```

root@RaspberryPi-Gateway:/dac# journalctl -fu doobby

```

```

-- Logs begin at Tue 2023-03-21 04:26:19 UTC. --

```

```

Mar 24 15:22:01 RaspberryPi-Gateway helloworld-test[14410]: Loaded plugin 'thunder' from '/usr/lib/plugins/dobby/libThunderPlugin.so.1'

```

```

Mar 24 15:22:01 RaspberryPi-Gateway helloworld-test[14410]: Plugin logging has nothing to do at postStart

```

```

Mar 24 15:22:01 RaspberryPi-Gateway helloworld-test[14410]: Plugin networking has nothing to do at postStart

```

```

Mar 24 15:22:01 RaspberryPi-Gateway helloworld-test[14410]: Hook poststart completed

```

```

Mar 24 15:22:01 RaspberryPi-Gateway DobbyDaemon[234]: NFO: No file size limit size for container log - setting to unlimited

```

```

Mar 24 15:22:01 RaspberryPi-Gateway DobbyDaemon[234]: NFO: Configuring logging for container 'helloworld-test' (pid: 14395)

```

```

Mar 24 15:22:01 RaspberryPi-Gateway DobbyDaemon[234]: NFO: container 'helloworld-test' started, controller process pid 14395

```

```

Mar 24 15:22:01 RaspberryPi-Gateway DobbyDaemon[234]: MIL: container 'helloworld-test'(734) started

```

```

Mar 24 15:22:30 RaspberryPi-Gateway DobbyDaemon[234]: NFO: List()

```

```

Mar 24 15:22:30 RaspberryPi-Gateway DobbyDaemon[234]: NFO: GetState('734')

```

## Debugging Guide

Log files path

/opt/logs/crun.log

/var/log/<logfile.name> This log file name will be mentioned in config.json

### Observed Error

```
root@RaspberryPi-Gateway:/dac# cat /opt/logs/crun.log
error stat'ing file `/usr/share/X11/xkb`: No such file or directory
error stat'ing file `/usr/lib/libEGL.so`: No such file or directory
root@RaspberryPi-Gateway:/dac#
```

Once the OCI bundle is created , verify the **config.json** for extra mount options and extra libraries, remove them from config.json.

For, helloworld application , these libraries mount is not needed, hence modified the config.json.

PFA,

default config.json : [config\\_default.json](#)

Modified config.json : [config.json](#)