

AAMP Simulator Build (Linux) Instructions

Dependencies/Pre-requisites

a) CMake, uuid and SSL:

```
sudo apt-get update
sudo apt-get -y install cmake
sudo apt-get -y install uuid-dev
sudo apt-get -y install libssl-dev
```

b) Gstreamer:

Below gstreamer packages with version = 1.16.2 are required (hard dependency on gst version is due to the prebuilt qt demux library provided)

```
sudo apt-get -y install libgstreamer1.0-dev
sudo apt-get -y install libgstreamer-plugins-base1.0-dev
sudo apt-get -y install libgstreamer-plugins-good1.0-dev
sudo apt-get -y install libgstreamer-plugins-bad1.0-dev
sudo apt-get -y install gstreamer1.0-libav
sudo apt-get -y install gstreamer1.0-tools
```

c) Git, XML and Curl

```
sudo apt-get -y install git
sudo apt-get -y install libxml2-dev
sudo apt-get -y install libcurl4-openssl-dev
```

1. Clone aamp and dependent components in a new directory

```
git clone -b dev_sprint https://code.rdkcentral.com/r/rdk/components/generic/aamp
git clone -b dev_sprint https://code.rdkcentral.com/r/rdk/components/generic/aampabr
git clone -b dev_sprint https://code.rdkcentral.com/r/rdk/components/generic/gst-plugins-rdk-aamp
git clone -b dev_sprint https://code.rdkcentral.com/r/rdk/components/generic/aampmetrics
git clone git://github.com/DaveGamble/cJSON.git
```

2. Build the components in the following order -

a) aampabr

```
cd aampabr
mkdir build
cd build
cmake ../ -DCMAKE_BUILD_TYPE=Debug
make
sudo make install
cd ../../
```

b) cJSON

```
cd cJSON
mkdir build
cd build
cmake ../
make
sudo make install
cd ../../
```

c) aampmetrics:

```
cd aampmetrics
mkdir build
cd build
cmake ../ -DCMAKE_BUILD_TYPE=Debug
make
sudo make install
cd ../../
```

d) libdash

For Ubuntu, reference <https://github.com/bitmovin/libdash#ubuntu-1204>

```
git clone git://github.com/bitmovin/libdash.git
cd libdash/libdash
git checkout stable_3_0
```

Apply patches downloaded from (patch -p1 < file.patch): <https://code.rdkcentral.com/r/plugins/gitiles/components/generic/rdk-oe/meta-rdk-ext/+/-rdk-next/recipes-multimedia/libdash/libdash/>

```
mkdir build
cd build
cmake ..
make
sudo make install
```

Install:

```
cp bin/libdash.so /usr/local/lib/
sudo mkdir -p /usr/local/include
sudo mkdir -p /usr/local/include/libdash
sudo mkdir -p /usr/local/include/libdash/xml
sudo mkdir -p /usr/local/include/libdash/mpd
sudo mkdir -p /usr/local/include/libdash/helpers
sudo mkdir -p /usr/local/include/libdash/network
sudo mkdir -p /usr/local/include/libdash/portable
sudo mkdir -p /usr/local/include/libdash/metrics
sudo cp -pr ../libdash/include/*.h /usr/local/include/libdash
sudo cp -pr ../libdash/source/xml/*.h /usr/local/include/libdash/xml
sudo cp -pr ../libdash/source/mpd/*.h /usr/local/include/libdash/mpd
sudo cp -pr ../libdash/source/network/*.h /usr/local/include/libdash/network
sudo cp -pr ../libdash/source/portable/*.h /usr/local/include/libdash/portable
sudo cp -pr ../libdash/source/helpers/*.h /usr/local/include/libdash/helpers
sudo cp -pr ../libdash/source/metrics/*.h /usr/local/include/libdash/metrics

echo -e 'prefix=/usr/local \nexec_prefix=${prefix} \nlibdir=${exec_prefix}/lib \nincludedir=${prefix}/include
/libdash \n \nName: libdash \nDescription: ISO/IEC MPEG-DASH library \nVersion: 3.0 \nRequires: libxml-2.0
\nLibs: -L${libdir} -ldash \nLibs.private: -lxml2 \nCflags: -I${includedir}' > libdash.pc
sudo mv libdash.pc /usr/lib/pkgconfig/.

cd ../../../../
(OR)

cd aamp
./install_libdash.sh
```

e) aamp:

```
cd aamp
mkdir build
cd build
cmake ../ -DCMAKE_BUILD_TYPE=Debug -DCMAKE_PLATFORM_UBUNTU=1
make
sudo make install
cd ../../
```

f) gst-plugins-rdk-aamp:

```
cd gst-plugins-rdk-aamp
mkdir build
cd build
cmake ../ -DCMAKE_BUILD_TYPE=Debug
make
sudo make install
cd ../../
```

3. Copy the prebuilt qtdemux library with AAMP patches. Prebuilt qtdemux library is available in tar file attached to this doc - [override_libs.tgz](#)

Run the following commands in a new folder

```
tar -xzf override_libs.tgz
```

Update the environment variable with the new location so that custom qtdemux plugin will be picked

```
export GST_PLUGIN_PATH=`pwd`/override-libs:$GST_PLUGIN_PATH
```

To verify, run `gst-inspect-1.0 qtdemux` and confirm the Filename value under Plugin Details is same as the override library location

4. optionally create `/opt/aamp.cfg` – supports local configuration overrides

5. Create `/opt/aampcli.cfg` with below content for a virtual channel list of assets that could be loaded in `aamp-cli`

```
*1 HOSTED_FRAGMP4 https://bitmovin-a.akamaihd.net/content/MI201109210084\_1/m3u8s-fmp4/f08e80da-bf1d-4e3d-8899-f0f6155f6efa.m3u8
*2 HOSTED_HLS http://devimages.apple.com/iphone/samples/bipbop/bipbopall.m3u8
*3 HOSTED_DASH https://bitmovin-a.akamaihd.net/content/MI201109210084\_1/mpds/f08e80da-bf1d-4e3d-8899-f0f6155f6efa.mpd
```

6. Execute binaries

```
./aamp-cli // for full featured command line interface to aamp
or
./playbintest <url> // to exercise aamp as a plugin
```