Using Android Emulator Virtual Devices

You can use Android Emulator to create emulations of Android devices that run your own custom Android system images. You can also share your custom Android system images so that other people can run emulations of them. In addition, you can add multi-display support to Android Emulator emulations.

Android Emulator architecture

Android Emulator allows you to run emulations of Android devices on Windows, macOS or Linux machines. The Android Emulator runs the Android operating system in a virtual machine called an Android Virtual Device (AVD). The AVD contains the full Android software stack, and it runs as if it were on a physical device. Figure 1 is a diagram of the Android Emulator's high-level architecture. For more information about the emulator, see Run apps on the Android Emulator{.external}.

Figure 1. Android Emulator architecture

[Building AVD images]

Each AVD includes an Android system image, which runs in that AVD. The AVD Manager includes some system images. And you can build custom AVD system images from your source code and create device emulations to run them.

Note: You need to establish a build environment before building AVD system images.

To build and run an AVD system image:

1. Download the Android source:

   ```
   mkdir aosp-master; cd aosp-master
   repo init -u
   repo sync -j24
   ```

   If you want to build other Android versions, you can find their branch names in the public Android repository. They map to Android Codenames, Tags, and Build Numbers.

2. Build an AVD system image. This is the same process as building an Android device system image. For example, to build a x86 32-bit AVD:

   ```
   mkdir aosp-master; cd aosp-master
   source ./build/envsetup.sh
   lunch sdk_phone_x86
   make -j32
   ```

   If you prefer to build an x86 64-bit AVD, run `lunch` for the 64-bit target:

   ```
   lunch sdk_phone_x86_64
   ```

3. Run the AVD system image in the Android Emulator:

   ```
   emulator
   ```

See the Command-line startup options for more details about running the emulator. Figure 2 shows an example of the Android Emulator running an AVD.

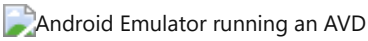

Figure 2. Android Emulator running an AVD

[Sharing AVD system images for others to use with Android Studio] {.devsite-heading role="heading" aria-level="2"} {#sharing_avd_system_images_for_others_to_use_with_android_studio data-text="Sharing AVD system images for others to use with Android Studio" role="presentation"}

Follow these instructions to share your AVD system images with others. They can use your AVD system images with Android Studio to develop and test apps.

1. Make additional `sdk` and `sdk_repo` packages:

   ```
   $ make -j32 sdk sdk_repo
   ```

   This creates two files under `aosp-master/out/host/linux-x86/sdk/sdk_phone_x86` :

   - `sdk-repo-linux-system-images-eng.[username].zip`{translate="no" dir="ltr"}
   - `repo-sys-img.xml`

2. Host the file `sdk-repo-linux-system-images-eng.[username].zip` somewhere accessible to your users, and get its URL to use as the **AVD System Image URL**.

3. Edit `repo-sys-img.xml` accordingly:

   - Update `<sdk:url>` to your **AVD System Image URL**.
   - See sdk-sys-img-03.xsd to learn about other updates to the file.

4. Host `repo-sys-img.xml` somewhere accessible to your users, and get its URL to use as the **Custom Update Site URL**.

To use a custom AVD image, do the following in the SDK Manager:

1. Add the **Custom Update Site URL** as an SDK Update Site.

   This adds your custom AVD system image to the System Images page.

2. Create an AVD by downloading and selecting the custom AVD system image.

[Adding Multi-Display support]{.devsite-heading role="heading" aria-level="2"} {#adding_multi-display_support data-text="Adding Multi-Display support" role="presentation"}

Android 10 enhances Multi-Display (MD) to better support more use cases, such as auto and desktop mode. Android Emulator also supports multi-display emulation. So you can create a specific multi-display environment without setting up the real hardware.

You can add multi-display support to an AVD by making the following changes, or by cherry picking from these CLs.

- Add multi-display provider to the build by adding these lines to file `build/target/product/sdk_phone_x86.mk`:

```
PRODUCT_ARTIFACT_PATH_REQUIREMENT_WHITELIST := \
    system/lib/libemulator_multidisplay_jni.so \
    system/lib64/libemulator_multidisplay_jni.so \
    system/priv-app/MultiDisplayProvider/MultiDisplayProvider.apk \
PRODUCT_PACKAGES += MultiDisplayProvider
```

- Enable the Multi-Display feature flag by adding this line to file `device/generic/goldfish/data/etc/advancedFeatures.ini`:

```
MultiDisplay = on
```

You can find the latest emulator features and release information from the following sources:

- Android Emulator User Guide
- Android Emulator Release Notes