

Andmore SDK Manager User Guide

Andrew Bowley
January 2018

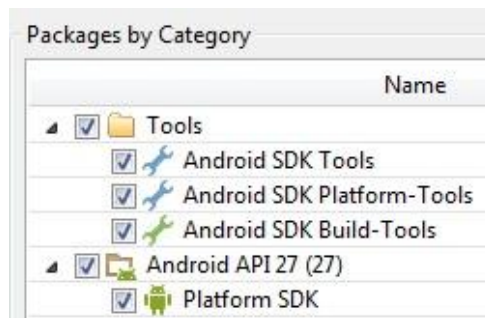
Attachment to bugzilla report with summary %Andmore SDK
Management needs update+

This is a user guide for the SDK Manager component of And-
more being revised for version 0.5.2.

Contents

Introduction

New SDK Installation	3
Accepting Licences.....	5
Package Download and Install	6
Configuring an Existing Installation.....	6
SDK Preferences.....	8
Android Virtual Device (AVD) Management.....	9
Android Virtual Devices Tab	9
Device Definitions Tab.....	12



Detail from Android SDK Manager showing a minimal SDK packages installation.

Introduction

Android continues to evolve over time and Andmore is undergoing some significant development to keep pace. This user guide covers a revision to Software Development Kit (SDK) and Android Virtual Device (AVD) Managers which are integrated into the Andmore IDE. These managers were originally made available in an application distributed with the Android SDK tools and launched using the command `%android+`.

The SDK Manager allows you to view the packages that you have installed for Android development, organizing them into platform categories. It shows updates for the packages already installed and the latest version of new packages available for installation. One new feature you can take advantage of is the ability to filter packages according to category. This allows you to focus on relevant packages.

The AVD Manager allows you to create and run AVDs. An AVD configuration requires a device definition, a system image and a target platform. AVD Manager features broken AVD configuration repair. This can be caused by changing to a different SDK installation. A new feature of the AVD Manager is the ability to find system images with Android TV and Android Wear tags.

Andmore has also been updated in how it deals with SDK installation and configuration. The driver for this is the ability now to configure multiple SDK installations shared by multiple workspaces. Each SDK installation can be assigned a name in order to make it easy to recognize when displayed on a list with other installations. A new `%SDK Selection Wizard+` allows you to either select an existing SDK or install a new one. When an existing SDK is selected, it is profiled so you can get an idea of its status. When installing a new SDK, you are initially presented with a minimum set of packages for the latest platform to which you can make any desired changes using package filtering.

New SDK Installation

When Andmore starts up without an Android SDK configuration it automatically shows the "Select Android SDK" wizard. You need to configure or a new or existing SDK (figure1).

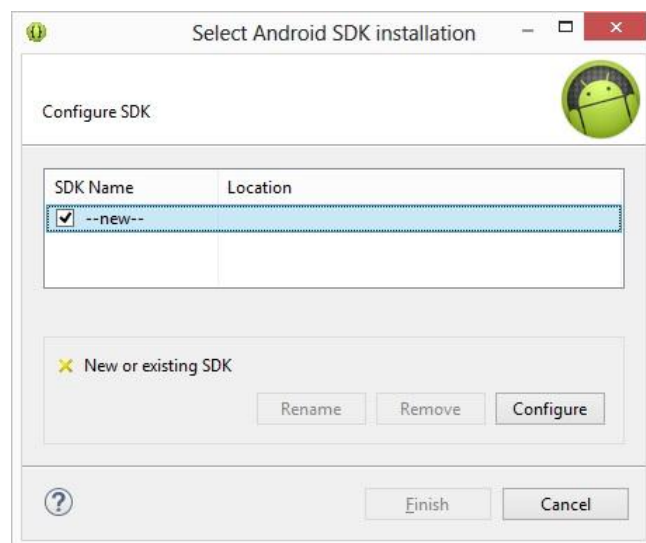
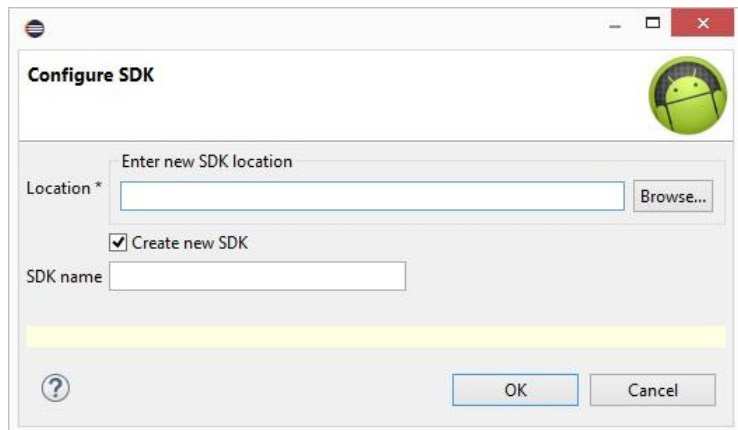


Figure 1. Select Android SDK Wizard on new Andmore installation.

Andmore SDK Manager User Guide

Click on the "Configure" button to get the "Configure SDK" dialog (figure2).

You can see that there are just 2 fields and only the first is required, which is the location of where to install the new SDK. This field can be filled by typing or pasting into it, or hit the browse button to navigate to it.



You will notice below the location field is a checkbox ticked to select 'Create new SDK'. Leave it checked to apply new install rules:

Figure 2. The 'Configure SDK' screen for a new installation

1. If the location does not exist, Andmore will try to create it
2. Otherwise, the location must be empty

The "SDK Name" field is optional but is useful if you intend to work with more than one SDK installation.

To complete the configuration, click the OK button. Following validation of the location, the "Android SDK Manager" appears, ready to install the required packages for the latest Android SDK platform (Figure 3.).

You can Adjust the selection as desired. To see more package categories, click on the 'Filter' button. An additional 5 categories are available. as shown below.

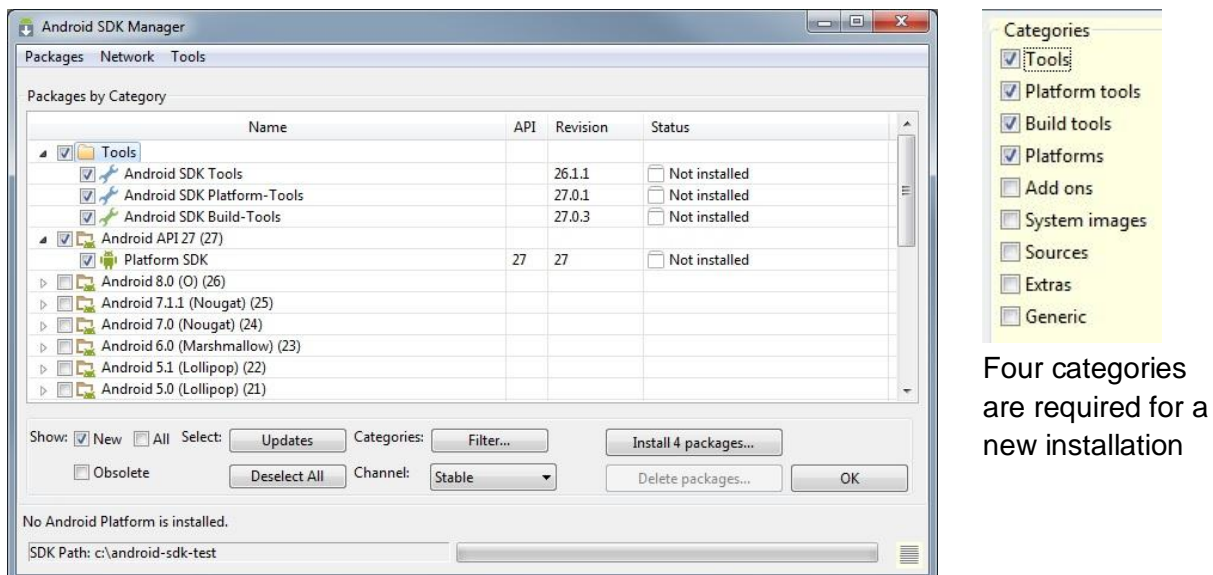


Figure 3. Android SDK Manager selection for new installation

To see even more packages, you can tick the Show: 'All' checkbox to see all available versions of new packages,. You can also change the update channel from 'Stable' to 'Beta', 'Dev' or 'Canary' to see more recent releases and previews.

Andmore SDK Manager User Guide

An example of a change to the initial package selection is shown in figure 4. The highest API platform is changed in favour of the next level down, which provides a greater choice for system images., of which one is selected. The extras package type is now included allowing the HAXM installer to be selected.

Note that platforms below API level 21 are given a separate %Early API+category so they can be ranked below all the other categories in order of appearance. This category recognizes a significant stage in Android evolution- the introduction of the Android runtime (ART). Early APIs are likely not to be of interest for developers doing a new SDK installation.

Name	API	Revision	Status
<input checked="" type="checkbox"/> Android SDK Tools		26.1.1	<input type="checkbox"/> Not installed
<input checked="" type="checkbox"/> Android SDK Platform-Tools		27.0.0	<input type="checkbox"/> Not installed
<input checked="" type="checkbox"/> Android SDK Build-Tools		27.0.2	<input type="checkbox"/> Not installed
▶ <input type="checkbox"/> Android Android API 27 (27)			
▲ <input type="checkbox"/> Android Android 8.0 (O) (26)			
<input checked="" type="checkbox"/> Platform SDK	26	26	<input type="checkbox"/> Not installed
<input checked="" type="checkbox"/> Google Play Intel x86 Atom System Image	26	7	<input type="checkbox"/> Not installed
<input type="checkbox"/> Google APIs Intel x86 Atom System Image	26	7	<input type="checkbox"/> Not installed
<input type="checkbox"/> Android Wear Intel x86 Atom System Image	26	2	<input type="checkbox"/> Not installed
<input type="checkbox"/> China version of Android Wear Intel x86 Atom System Image	26	3	<input type="checkbox"/> Not installed
<input type="checkbox"/> Android TV Intel x86 Atom System Image	26	5	<input type="checkbox"/> Not installed
▶ <input type="checkbox"/> Android Android 7.1.1 (Nougat) (25)			
▶ <input type="checkbox"/> Android Android 7.0 (Nougat) (24)			
▶ <input type="checkbox"/> Android Android 6.0 (Marshmallow) (23)			
▶ <input type="checkbox"/> Android Android 5.1 (Lollipop) (22)			
▶ <input type="checkbox"/> Android Android 5.0 (Lollipop) (21)			
▲ <input type="checkbox"/> Extras			
<input type="checkbox"/> ConstraintLayout for Android		1.0.2	<input type="checkbox"/> Not installed
<input type="checkbox"/> Solver for ConstraintLayout		1.0.2	<input type="checkbox"/> Not installed
<input checked="" type="checkbox"/> Intel x86 Emulator Accelerator (HAXM installer)		6.2.1	<input type="checkbox"/> Not installed
<input type="checkbox"/> Google Web Driver		?	<input type="checkbox"/> Not installed

Figure 4. Example of a custom package selection

When satisfied with your package selection, hit the %Install packages+button to proceed to the next stage.

Accepting Licences

A "Choose Packages to Install" dialog shows which packages are to be installed and allows you to review the license terms and conditions. You may notice additional packages to the ones you selected and these are dependencies. You can select any package to get more information.

The packages are arranged into license type categories so if you accept the license for one package in a group, then all are accepted. Figure 5. shows the result of clicking on the top package and accepting it's license.

In figure 6. next page, notice in the details "Archive Size" which indicates how large the download will be for this package. It is good to review archive sizes as the total download size may be over a gigabyte.



Figure 5. Packages grouped by license

Andmore SDK Manager User Guide

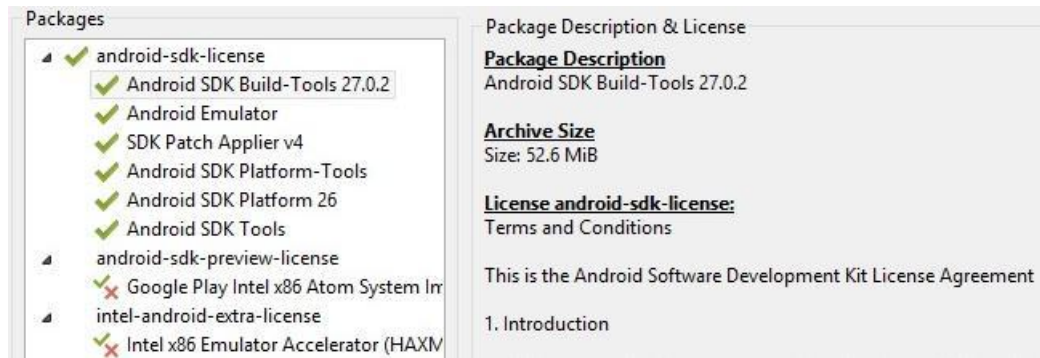


Figure 6. Package description showing Archive Size of 52.6 Mebibytes

Package Download and Install

Complete the license acceptance steps and then click the "Install" button to commence the package download.

The "Android SDK Manager" has a progress status area on the bottom of the window. Figure 7a. shows an example of the first package being downloaded. You see a top status line showing **Preparing Packages**. This line displays the name of each package as it is installed. The next line is split into a text window on the left and a progress bar on the right.



Figure 7a Progress area of Android SDK Manager during package download

Figure 7b. is a snapshot of the progress area as the package is being unzipped. Note that to indicate the installation is proceeding, the progress text window flashes the package file contents as it is stored to disk.



Figure 7b. Progress area as package is stored to SDK

You will see a small button on the right of the progress bar. This brings up the SDK Manager message log. This log is also automatically shown if an error occurs.

When the installation is completed:

1. The message **n packages installed** is displayed where n is the number of packages
2. The Android SDK Manager is refreshed and the status of the new packages is updated to "installed"

You can now exit from the Android SDK Manager by hitting the OK button. If the required packages were installed successfully, you will see the new SDK as an checked item inserted at the top of the SDK list

Configuring an Existing Installation

The next case to consider is when you set up a new Andmore workspace on a machine where an Android SDK installation is already available for use. The initial sequence is the same as for a new SDK installation. Andmore shows the "Select Android SDK installation" wizard (figure1, page 2) and you then click on the "Configure" button to get the "Configure SDK" dialog (figure2, page 3).

Now uncheck the "Create new SDK" checkbox below the Location field. You will see the Location prompt change to "Enter existing SDK location". You can type or paste directly into the Location field or hit the "Browse" button to navigate to the desired location. You can also enter a name for the SDK. To complete the configuration, hit OK,. The location will then be validated to ensure is the directory exists and is not empty. The dialog will then close.

On returning to the SDK selection page, you will see the results of entering the existing SDK location:

- A new item appears at the top of the SDK list for the location just entered
- A status icon and message shows the SDK profile.
- You are given the option to update the SDK if the status is good or repair the SDK if it is missing required packages
- The new item will be configured in the Andmore workspace as indicated by being checked

Figure 8. shows the wizard after configuring an SDK installation consisting of a single platform with a good profile. If the "Update" button is clicked then the Android SDK Manager is launched so you can install any pending updates or add / remove packages.

If the SDK profile is bad, as indicated by a red cross in place of the green tick, and the command button text changes to "Repair".

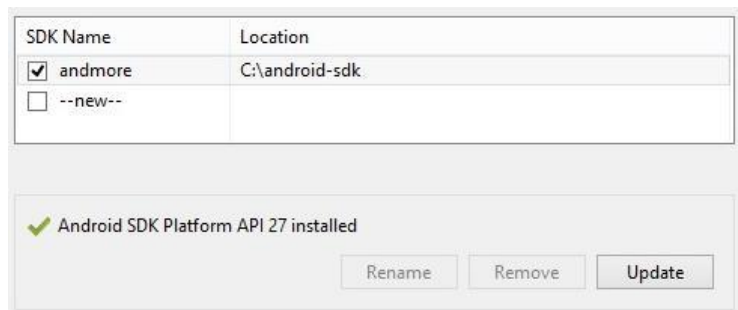


Figure 8. Select Android SDK window after existing SDK configuration

Figure 9. shows other profile indications that are possible. A bad SDK profile is caused by at least one platform package not installed and/or the required packages for a platform being absent. The repair case is treated as a new install, where the initial selection is limited to packages required to install a platform.

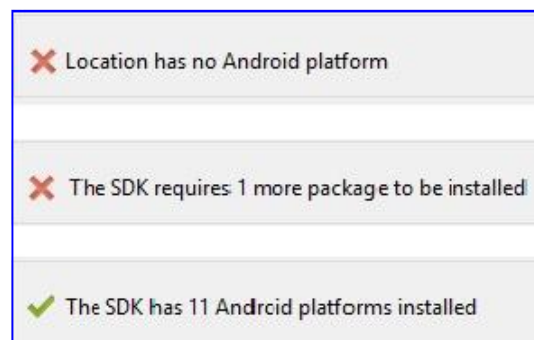


Figure 9. Other profiles

SDK Preferences

There are two different Android SDK preference contexts:

1. Workspace: Configure SDK location and installed packages
2. Project: Configure target platform on which to run the application

The first time you start Andmore with a new workspace, you are automatically requested to configure its SDK location. Afterwards, you are reminded of this configuration when you open the Android Preferences. These preferences provide the opportunity to update the current SDK or change to one in another location.

Any time an Android project is open you can navigate to the project Android project properties to view the available target platforms and change which one is the target for the project.

Details on both workspace and project preferences follow.

Workspace Preferences

It is possible you will have more than one Android SDK installation set up on your machine. Workspace Preferences allows you to assign any of these installations to the current workspace at any time. Note that all SDKs previously configured in Workspace Preferences are listed automatically, otherwise you will need to enter the directory containing the desired SDK yourself.

To configure the Workspace Preferences, from the Andmore top menu select Window -> Preferences -> Android.. This brings up a preference page with description %Android SDK %
This page shows the details of the SDK currently assigned to the workspace (Figure 10.)

This page shows the installed target platforms and the location where the SDK is installed. You will find that the details cannot be edited. To make changes, you need to hit either the %Change+or %Update+button.

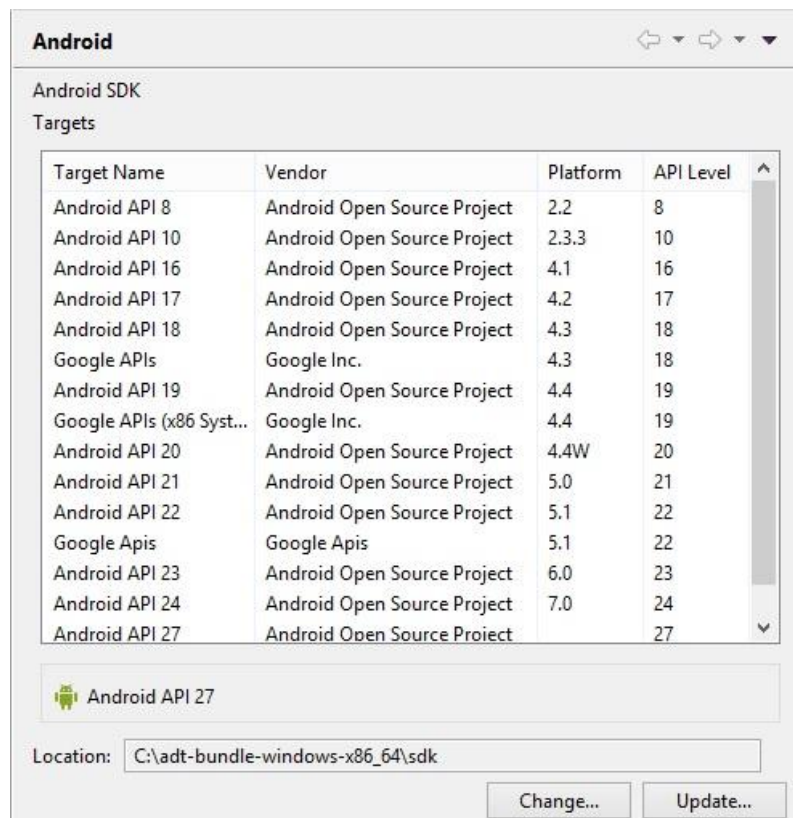


Figure 10. Android SDK Preference Page

Andmore SDK Manager User Guide

Hitting %Change+brings up the %Select SDK Wizard+. The wizard allows you to select a previously configured SDK, or install a new one. (figure 11). The selected SDK is indicated by a ticked checkbox.

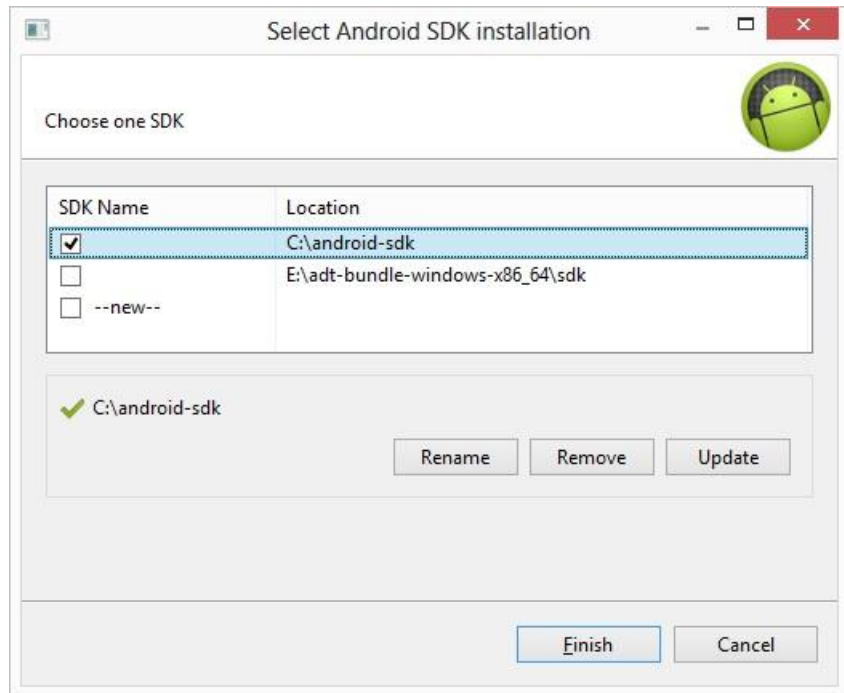


Figure 11. Select Android SDK dialog

To complete the selection, hit the %Finish+button and you will be returned to the Android SDK page, Note that it takes a moment for the Targets to be refreshed after the return to the Android SDK page.

Hitting the %Update+button brings up the Android SDK Manager which will show you what packages are installed and available updates and new packages.

Project Preferences

To view Android target platforms available to a project, right click on the project in the Project Explorer window and select Properties -> Android. You will see targets listed similar to what you see in Workspace Preferences, but with the addition of checkboxes so one target can be set for the project.

Android Virtual Device (AVD) Management

The AVD Manager supports all the possible AVD operations: create, edit, start, delete and repair. The AVD Manager is launched from either the menu or toolbar and presents an "Android Virtual Devices" tab and a "Device Definitions" tab.

Android Virtual Devices Tab

This displays a list of configured devices with status indicated by an icon. Buttons for performing AVD operations on the left are enabled according to which AVD is selected from the list and it's status. For more information on a particular AVD, select it and click on the "Display" button. Figure 12, next page, gives an example of an AVD details display.

Details on the %Details+and %Repair+AVD options follow . The other options require no further explanation.

Details

There are 5 header AVD attributes:

- Name

The name is determined when the device is created and must be unique amongst the AVDs

- CPU/ABI

This identifies the CPU model and architecture of the system image used by the emulator

- Path

The location of the AVD configuration, which will be under the Android home directory

- Target

The Android platform selected for emulation

- Skin

The screen dimensions which reveal the device format and orientation

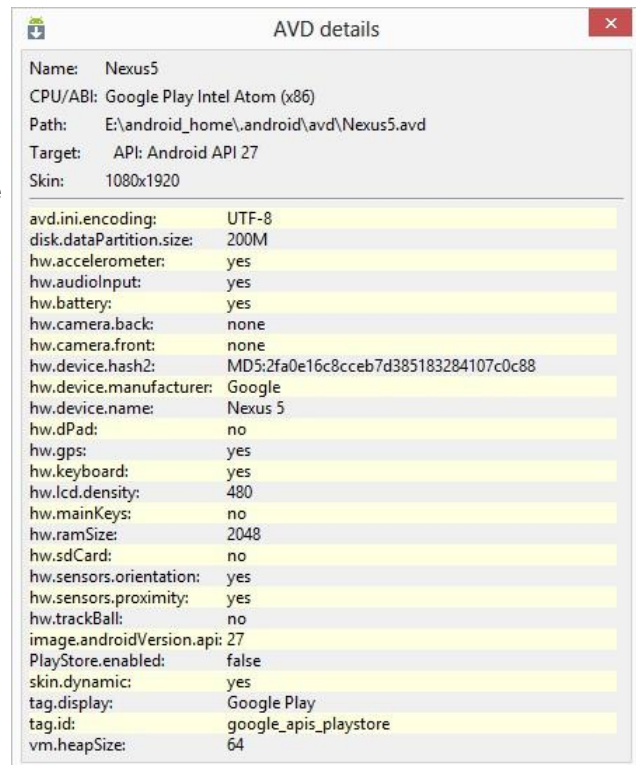


Figure 12. AVD Details window

Other AVD properties are displayed in alphabetical order below the header,. Of these, the most important are **tag.id** and **hw.device.manufacturer**.

Repair

An AVD configuration combines different elements, including system image and Android platform packages. The repair option supports two different cases:

1. Package is missing
2. Device is missing or configuration needs updating

Package is missing

Figure 13 shows details of an AVD which requires a package to be installed. The images are taken from the AVD Manager list and the Details display.

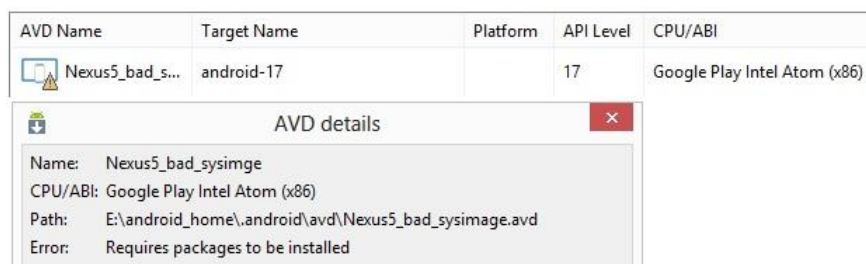


Figure 13. AVD needing repair

Andmore SDK Manager User Guide

To start the repair, make sure the broken AVD device is selected and then click on the Repair button. A mini "Android SDK Manager" window pops up with only a text logging console and progress bar. The SDK Manager loads the package configurations and then goes directly to displaying the package details and requesting you to accept the license terms. To complete the operation, click the "Install" button and the package will then be downloaded and installed (Figure 14).

Once the package is installed, click the OK button to close the SDK Manager. The AVD Manager display automatically updates when the package installation completes.

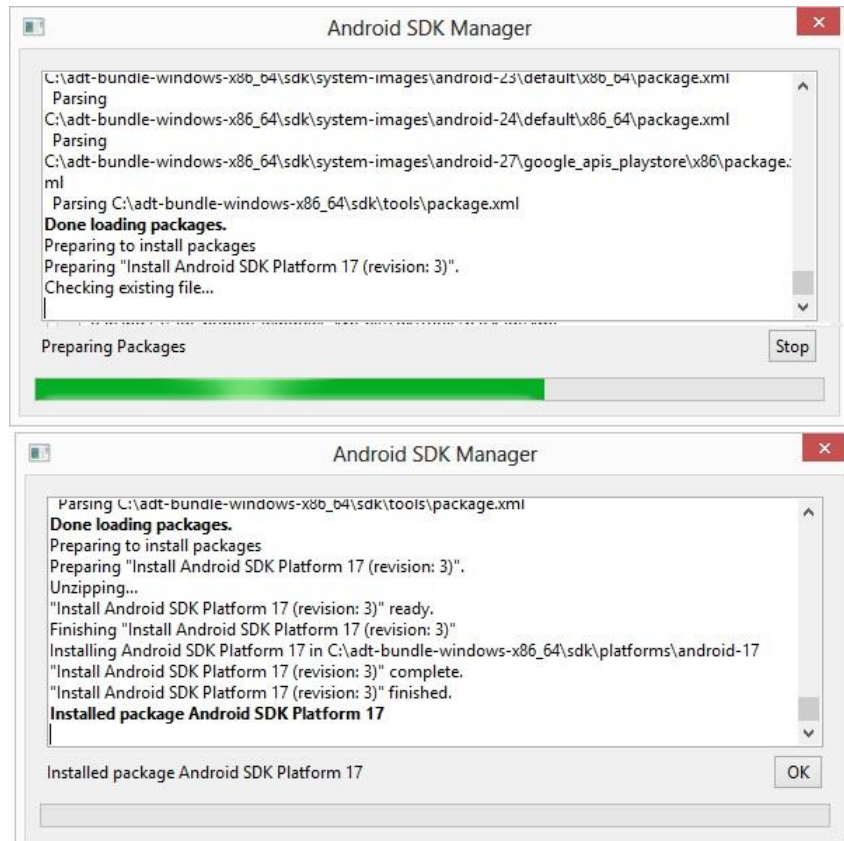


Figure 14. Automatic package installation

Figure 15 shows the how the device appears after the repair. You will see the status icon now looks normal and the "Target Name" and "Platform" details are updated.


AVD Name	Target Name	Platform	API Level	CPU/ABI
 Nexus5_bad_s...	Android API 17	4.2	17	Google Play Intel Atom (x86)

Figure 15. Repaired AVD

Device is missing

The error message for a missing device is

"manufacturer name no longer exists as a device"

Hit the Repair button and the AVD editor window appears showing all configured details except the device, which is left unselected. You can select the desired device and then hit OK.

The AVD will then be ready for use. Note that changing the name of an AVD while in the editor normally results in a duplicate AVD being created, but in case of a repair operation, the original AVD configuration is deleted if the name is changed.

Andmore SDK Manager User Guide

The error message for a device configuration update required is:

"*manufacturer name* configuration has changed since AVD creation"

Hit the Repair button to update the device configuration.

Device Definitions Tab

Devices are listed ordered by name. The list is displayed in an order that is user friendly with devices using names first sorted alphabetically followed by all devices that use a numeric screen size sorted by actual size. There is a "Tag" column which shows the device tag icon and name. The tag identifies which system images are suitable for use with the device. The majority of devices have a default tag, but those that do not have a distinctive icon to enhance visibility - see Figure 16

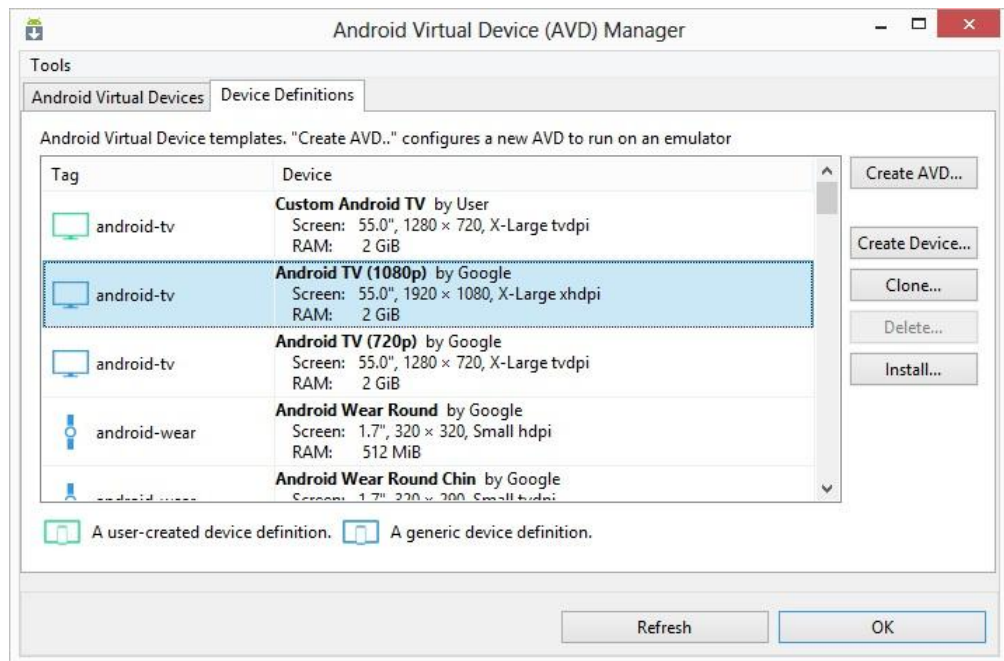


Figure 16 Android TV device

For those devices which have a non-default tag, an install button is provided to assist with selection and installation of system image packages which have a matching tag (figure 17)..






	Name	API	Revision	Status
<input type="checkbox"/>	Android API 27 (27)			
<input type="checkbox"/>	Platform SDK	27	27	 Installed
<input type="checkbox"/>	Android TV Intel x86 Atom System Image	27	2	 Installed
<input type="checkbox"/>	Android 8.0 (O) (26)			
<input type="checkbox"/>	Android 7.1.1 (Nougat) (25)			
<input type="checkbox"/>	Android 7.0 (Nougat) (24)			
<input type="checkbox"/>	Platform SDK	24	24	 Installed
<input type="checkbox"/>	Android TV Intel x86 Atom System Image	24	15	<input type="checkbox"/> Not installed
<input type="checkbox"/>	Android 6.0 (Marshmallow) (23)			
<input type="checkbox"/>	Platform SDK	23	23	 Installed
<input type="checkbox"/>	Android TV Intel x86 Atom System Image	23	14	<input type="checkbox"/> Not installed

Figure 17. Android SDK Manager selection of Android TV system images

Andmore SDK Manager User Guide

The devices defined by the system cannot be modified, so a "Clone" button is provided to create a "User" device based on a system one.

User devices are distinguished by tag icon color (green instead of blue) and the manufacturer is "User". User devices are the only ones that can be deleted.

There is no check on whether a suitable system image package is installed for any of the listed devices. If the system image is missing, you see a "No CPU/ABI system image available for this target" message in the AVD creation window. You can check for system images for devices with non-default tags using the "Install" button. For the rest, you will need to launch the SDK Manager to check for installed packages. You can use the "Categories" button to narrow the selection to just system images.