

Android 仮想デバイスマネージャーの使い方

1 はじめに

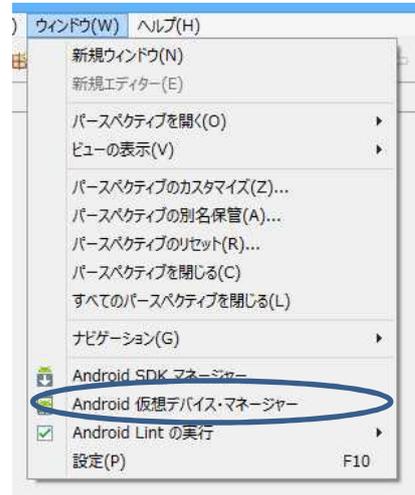
Android 仮想デバイス（以下、AVD : Android Virtual Devices）とは、PC 上で動作する Android 端末のエミュレーターであり、実機が無くともアプリケーションのデバッグを行うためのものです。Android 仮想デバイスマネージャー（以下、AVD マネージャー）とは、この AVD を作成、管理するための Android SDK 付属のアプリケーションです。

2 新しい AVD マネージャーの操作方法

2.1 AVD マネージャーの起動

Android SDK をインストールし、Eclipse に ADT プラグイン（Android Developer Tools）の導入が正しく行われていると、Eclipse の「ウィンドウ」メニューが右図のようになります。

ここから、**Android 仮想デバイス・マネージャー**を選択することで、AVD マネージャーが起動できます。



2.2 AVD マネージャーの画面

起動すると、右図のような画面になります。

Android Virtual Devices タグは、AVD の新規作成、編集、削除や、起動などの操作を行うための画面です。

Device Definitions タグは、AVD の画面サイズや解像度、カメラやセンサーの有無など、AVD の各種機能の定義を管理するためのものです。初期状態で様々な定義が用意されています。



2.3 AVD の作成

Android Virtual Devices タグから「新規」ボタンを押すと、AVD 作成が行えます。

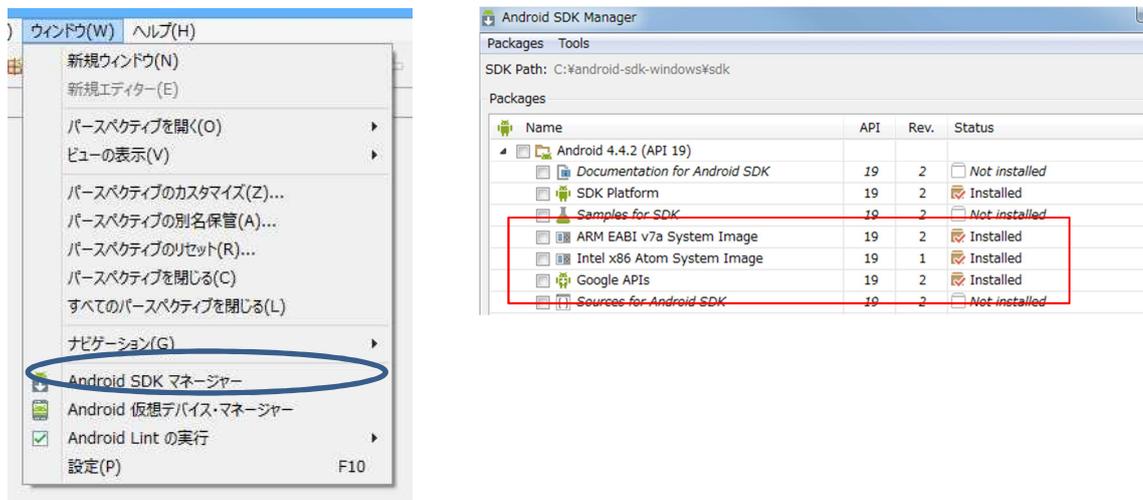
AVD 名は、AVD マネージャー上に表示される AVD の名称です。アルファベット大小文字、数字、ドット、アンダーバー、ハイフン（マイナス記号）しか使用できません。

装置は、Device Definitions のリストにある定義から装置の設定情報を選択するためのものです。

ターゲットは、AVD 内で動作させる Android のバージョンを選択します。SDK マネージャーでバージョンごとに SDK Platform をインストールすることができますが、そこでインストール済みのバージョンのみ、リストに表示されます。なお、Eclipse から SDK マネージャーを起動し、SDK Platform をインストールして、そのまま続けて AVD マネージャーを起動しても、このリストには加わりません。一度、Eclipse を再起動する必要があります。



CPU/ABI は、Android 4.0 以降で追加されたエミュレーションする CPU の選択です。Android 3.2 以前は ARM プロセッサのエミュレーションのみなので、この項目は操作できません。Android 4.0 以降では、SDK マネージャーで下図のようなところにある各 CPU に対応する System Image をインストールしておく必要があります。なお、System Image は API レベルごとインストールする必要があります。



キーボード は、PC のキーボードから動作中の AVD に直接入力する機能を有効にするかどうかの選択です。

スキン は、AVD の右側に操作パネルを表示させるかどうかの選択です。

フロント・カメラ 及び **バック・カメラ** は、カメラ機能の選択です。「None」は使用しない、「Emulated」はエミュレーションによるダミー映像の入力機能、「Webcam0」は PC に接続した Web カメラの映像の入力機能です（末尾の数字は PC に接続している Web カメラによって異なります）。ただし、Web カメラの映像入力はある程度 PC の能力を必要とするため、安定した利用が難しいこともあります。

メモリー・オプション は AVD に持たせるメモリの量の設定です。最大で 768M です。これは、Windows が一つのタスクに割り当てることのできるメモリの制約に伴うもののようです。**RAM** はメインメモリのサイズ、**VM ヒープ** は Java 仮想マシンが利用できるヒープメモリのサイズを設定します。

内部ストレージ は、AVD の内蔵ストレージのサイズを設定します。

SD カード は、AVD に仮想的な SD カードを認識させるための設定です。SD カード実物を用意する必要はなく、「サイズ」の項目に任意のサイズを設定すれば PC のハードディスク上で仮想 SD カードデバイスを用意し、それを AVD に認識させます。なお、**ファイル** という項目はファイル化された仮想 SD カードのファイルを指定するもので、このファイルは Android SDK に含まれる mksdcard というツールで作成できます。

エミュレーション・オプション は、AVD の特殊機能を設定するもので、「スナップショット」と「ホスト GPU を使用する」の 2 つの設定があります。

スナップショット とは、AVD の状態をホスト PC のディスクに記録することで、AVD の起動を早く行うことができる機能です。

ホスト GPU を使用する とは、エミュレーターの動作にホスト PC の GPU の機能を利用するものです。これにより、AVD 内の描画の高速化が図れます。ただし、この機能は Android 4.0 以降に対応しているため、それ以前のターゲットでこのチェックを有効にすると、AVD が起動しません。

なお、エミュレーション・オプションを両方同時に使用することはできません。両方にチェックを入れるとエラーメッセージが表示され、OK ボタンがグレイアウトして押すことができなくなります。

2.4 AVDの起動

設定を完了すると、AVD マネージャーのリストに AVD が加わります。この行をクリックすると、右側の操作ボタンがいくつか有効になります。

編集は選択した AVD の設定編集、**削除**は AVD の削除、**修復**は AVD が異常終了したときの修復（通常は選択できません）、**詳細**は設定内容の確認、**開始**は AVD の起動です。

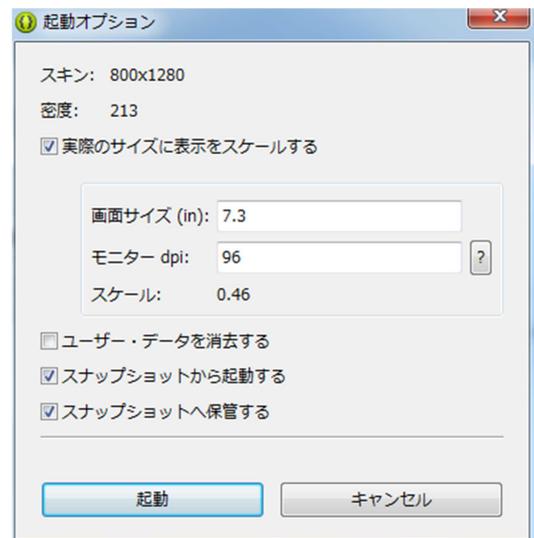
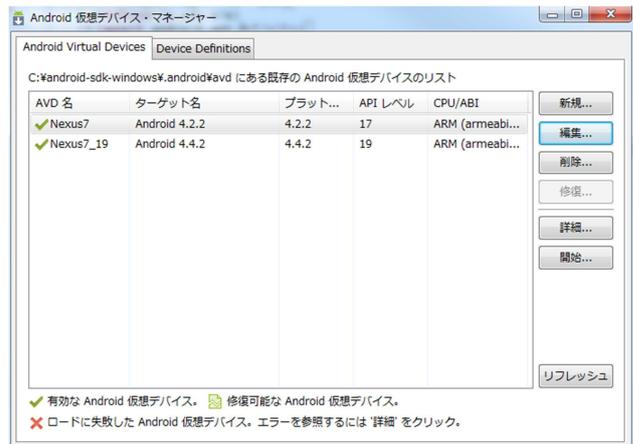
「開始」ボタンを押すと、右図のような画面が表示されます。

実際のサイズに表示をスケールするのチェックを有効にすると、その下のいくつかの項目が入力可能になります。これは、AVD の画面サイズに対して PC のディスプレイ上でどのくらいの比率で表示するかを設定できるものです。

ユーザー・データを消去するは、AVD の内部ストレージのデータを消去してから起動するかどうかの選択です。通常、AVD は一度終了しても、インストールしたアプリケーションは残っています。このチェックを有効にすることでそれらを全て破棄し、初期状態で AVD を起動したいときに使用します。

スナップショットから起動する及び**スナップショットへ保管する**は、AVD 作成画面で「スナップショット」を有効にしているときのみ操作できます。それぞれ、起動するときに保管してあるスナップショットを参照するか、AVD を閉じるときにスナップショットを保管するかを表しています。

起動ボタンを押せば、AVD が起動します。



2.5 AVDの画面

AVD を起動しました。これは、装置に「7.27” (800 x 1280: tvdpi)」を選んだときの構成です。前述した AVD 作成画面で、スキンのチェックを外して AVD を作成したものです。

2.6 スナップショットについて

スナップショットが存在しない状態で「スナップショットから起動する」にチェックを入れて起動しても、新規起動と同じ扱いになるのでエラーになることはありません。

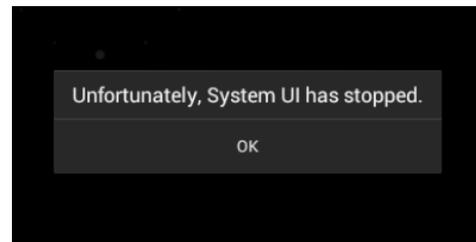
また、常に特定のセットアップをした状態から起動したいということであれば、一度「スナップショットへ保管する」にチェックを入れた状態で AVD を起動してすぐに終了し、以後は「スナップショットから起動する」にはチェックを入れて「スナップショットへ保管する」のチェックを外せば、いつも同じスナップショットから起動できるという活用法もあります。なお、「スナップショットへ保管する」にチェックが入っている場合、AVD を閉じるときにディスクに書き込む処理が入るため、ウィンドウが閉じられるまで少し時間がかかる場合があります。



3 注意点

3.1 アプリケーションが強制終了する

無事に AVD の起動に成功しても、Android 上で右図のようなエラーが表示され、アプリケーションがしばしば強制終了してしまうことがあります。これは、VM ヒープが少なすぎるときに起きやすいようです。そのようなときは、VM ヒープの設定を少し増やしてください。



3.2 起動に時間がかかり過ぎる、AVDの動作が重い

起動に時間がかかるのは、スナップショットをうまく活用することで回避できます。しかし、動作中の重さは PC のスペックに直結することなので、なかなか改善は難しいです。Android 4.0 以降であれば、まずホスト GPU の使用を試してみるのも一つの方法です。

また、利用している PC に搭載されている CPU が、**仮想化支援機能**というものに対応していれば、それを利用することで改善できることがあります。ただし、仮想化支援機能を利用すると、スナップショット機能が使えなくなります。設定上、チェックを入れることは可能ですが、実際には機能しません。

3.3 エミュレーション画面の表示がちらつく

Windows 上で AVD を動かすと、画面が乱れたり突然真っ白になったりと、不安定な動作をすることがあります。特に Windows 8 で顕著です。Android 開発者サイトの情報では、Windows 8 は Android SDK のサポート OS に含まれていません。特に、画面サイズが大きな AVD は極めて不安定で、起動すらままならない場合もあります。Windows 7 の方がいくらか安定しているので、開発用途の PC であれば現状では 8 は避けた方が良いでしょう。また、RAM の値を大きく設定しすぎた場合も、動作が不安定になり易いようです。

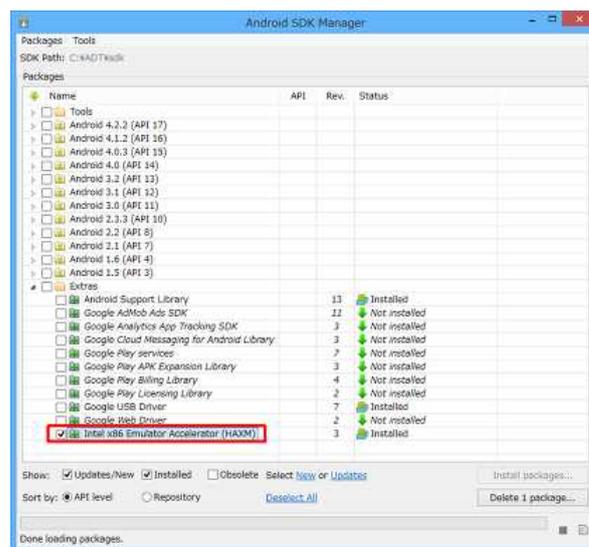
4 仮想化支援機能

Windows では、Intel の提供する HAXM (Hardware Accelerated Execution Manager) をインストールすることで、仮想化支援機能を使って AVD を動作させることができます。ただし、CPU が Intel VT と呼ばれる仮想化支援機能をサポートしている必要があり、必然的に使用できるのは Intel の CPU のみということになります。

HAXM をインストールするには、まず SDK マネージャーでインストーラーをダウンロードします。右図のように、Extras を展開したリストの一番下にあります。

【注意】

Windows 8.1 を使用している場合は上記の箇所からダウンロードした HAXM をインストールして、x86 エミュレーションの AVD を動作させると、CRITICAL_STRUCTURE_CORRUPTION という致命的エラーによって OS が強制再起動されてしまう現象が発生する場合があります。これに対応した HAXM が、インテルの下記サイトで配布されています。「Hotfix for Windows 8.1 only」と記されている方をダウンロードして zip ファイルを解凍してください。なお、もう一方は上記作業で SDK マネージャーによってダウンロードされるものと同一なので、Windows 8.1 以外でもこのサイトから hotfix 版ではない方の HAXM をダウンロードしてインストールすることも有効で



す。

<http://software.intel.com/en-us/articles/intel-hardware-accelerated-execution-manager/>

ここで表示上では「Install」という操作を行っていますが、これはインストーラーをダウンロードするのみであり、OS にインストールしたことにはなりません。操作を完了すると、Android SDK のインストールフォルダーの下の

extras\intel\Hardware_Accelerated_Execution_Manager に HAXM のインストーラーが用意されます。これを実行します。なお、使用している CPU が Intel VT をサポートしていなかったり、サポートされていても BIOS で無効化されていると、インストーラーで警告が表示され、インストールすることはできません。

インストーラーが起動したら、特にパラメーターを変更することなく、デフォルト設定のままインストールしてください。

