



PRÉSENTATION ANDROID

Présentation

Android est un OS (Operating System ou système d'exploitation en français) open source pensé pour les téléphones mobiles et développé par l'**Open Handset Alliance** (OHA) sous l'autorité de Google. L'OHA est composé d'environ 80 sociétés telles que Samsung, HTC, SFR, Orange, Asus, Qualcomm...

Android est basé sur un kernel Linux et possède une licence **Apache License 2.0**. Il sépare la couche matérielle de la couche logicielle, c'est-à-dire que n'importe quel téléphone Android peut faire fonctionner la même application et ainsi créer un écosystème riche pour les constructeurs, les utilisateurs et les développeurs.

Android est aussi un framework, et en tant que développeur vous avez accès au SDK (Software Development Kit - kit de développement) et à tout le code source de la plateforme, ce qui vous permet de comprendre son fonctionnement, de créer des versions personnalisées de l'OS et bien sûr, de développer des applications.

Par conséquent, plusieurs appareils exotiques tournant sous Android ont vu le jour : autoradios, montres, casques de ski, etc.

3

Historique

L'histoire d'Android commence en octobre 2003, lorsqu'Andy Rubin, Rich Miner, Nick Sears et Chris White créent la société Android à Palo Alto (Californie).

Google a racheté la société en août 2005. Deux ans plus tard, l'Open Handset Alliance est annoncée et Android devient officiellement open source.

La première version du SDK Android 1.0 sort en 2008 avec le premier téléphone sous Android (HTC Dream).

En avril 2009, la version 1.5 (API 3) d'Android sort. Cette version baptisée **Cupcake** (Petit gâteau) inaugure les nouveaux noms des versions d'Android.

Ce qui donnera pour les futures versions :

Donut (Beignet) 1.6 (API 4) : septembre 2009.

Eclair (Éclair) 2.0/2.1 (API 7) : octobre 2009.

Froyo (Yaourt glacé) 2.2.x (API 8) : mai 2010.

4

Gingerbread (Pain d'épice) 2.3.x (API 10) : décembre 2010.

Honeycomb (Ruche d'abeille) 3.x (API 11 - 12 - 13) : février 2011.

Ice Cream Sandwich (Sandwich à la crème glacée) 4.0.x (API 14 - 15) : octobre 2011. Cette version unifie la version tablette (Honeycomb) et la version smartphone (Gingerbread).

Jelly Bean (Dragibus) 4.1.x jusqu'à 4.3.1 (API 16, 17 et 18) : juin 2012.

KitKat 4.4.x (API 19) : octobre 2013.

Lollipop (Sucette) 5.x (API 21) : octobre 2014. Cette version est la dernière en date et apporte énormément de nouveautés. Ce sera la version utilisée et présentée tout au long de l'ouvrage.

Le personnage représentant Android est nommé **Bugdroid**. Ce petit robot vert a pour origine, selon la rumeur, un personnage d'un jeu vidéo des années 1990 sur Atari : **Gauntlet : The Third Encounter**.

5

Google Play

Afin de permettre aux utilisateurs d'accéder à un large panel d'applications, Android utilise un market nommé **Google Play**.

Il permet aux utilisateurs de :

Rechercher et télécharger des applications, livres, musiques, magazines et films.

Noter, commenter, désinstaller et mettre à jour les applications déjà installées sur un appareil.

En tant que développeur, publier une application sur Google Play est très simple et se fait en quelques étapes :

Exportation de l'application.

Certification de l'application.

Envoi sur Google Play.

6

SDK Android

Présentation

Ce dernier contient tous les outils nécessaires pour créer une application Android. Il est disponible pour Windows, Mac OS et Linux.

Chaque version du SDK contient :

aapt - Android Asset Packaging Tool : cet outil sert à créer et à analyser des fichiers *.apk .Ces fichiers contiennent le programme de votre application.

adb - Android Debug Bridge : le but de cet outil est d'établir des connexions avec un téléphone Android ou un émulateur en facilitant l'accès à leur shell. Ceci afin de parcourir son contenu ou d'exécuter des commandes. Cet outil sert aussi à transférer une application ou des fichiers sur un téléphone Android ou un émulateur.

7

ddms - Dalvik Debug Monitor Service : Cet outil est utilisé pour le débogage de votre application et permet de :
Faire des captures d'écrans.

- Voir les threads en cours d'exécution.
- Voir le Logcat.
- Connaître la liste des processus en cours d'exécution sur l'appareil.
- Simuler l'envoi de messages et d'appels.
- Simuler une localisation, etc.
- Le SDK contient aussi un système permettant de créer et de gérer les émulateurs ainsi que la documentation pour chaque version d'Android et des exemples pour les différentes API.

Voir via Eclipse ou Android Studio.

8

Émulateur

Présentation

L'émulateur est un outil permettant de simuler un téléphone ou une tablette Android. Il vous permet donc de développer sous Android même si vous ne possédez pas de téléphone ou de tablette Android.

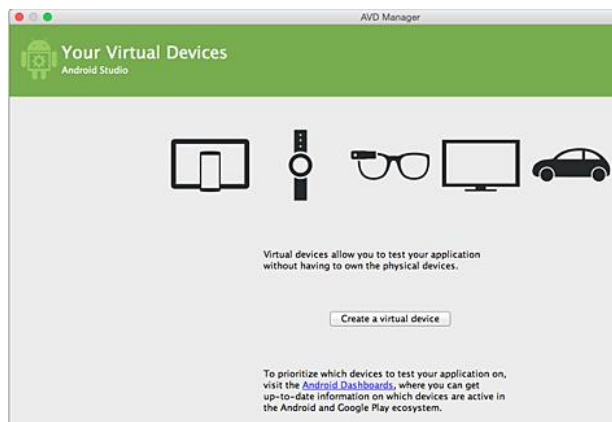
Il permet aussi de créer une large gamme d'émulateurs afin de simuler différents types de téléphones, de tailles d'écrans et de résolutions.

9

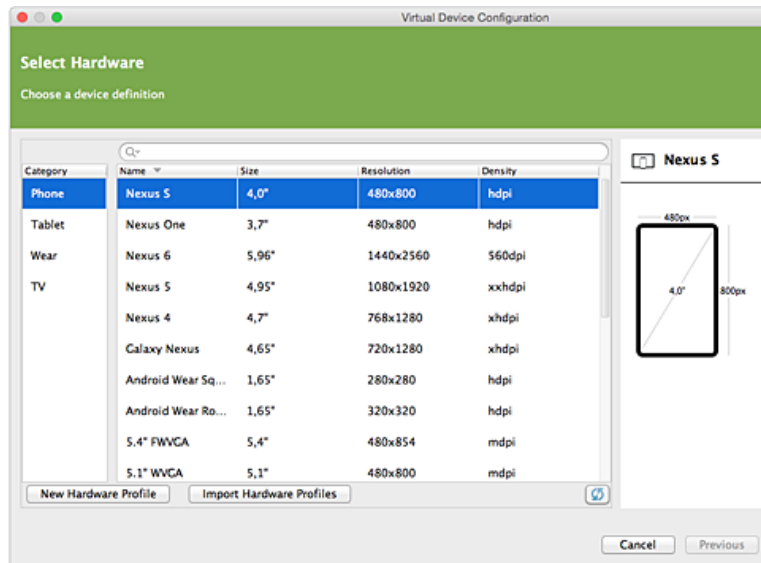
Création

Création d'un émulateur à partir d'Android Studio

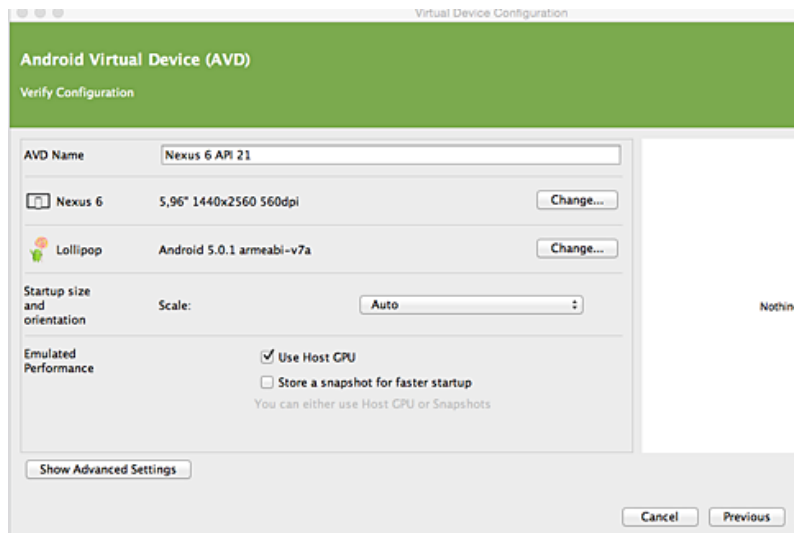
Une icône dédiée aux émulateurs est présente dans Android Studio.



10



11



12

Android4	320 × 480: xxhdpi	14	Android 4.0	arm	15 MB	
Android L	320 × 480: xxhdpi	L	Android L (Preview)	arm	1 GB	
Android Wear Square (E...	280 × 280: hdpi	20	Android 4.4W.2	arm	566 MB	
AVD for Nexus S by Co...	480 × 800: hdpi	15	Google APIs	arm	189 MB	
Nexus10	2560 × 1600: xhdpi	16	Android 4.1.2	arm	81 MB	
Nexus4	768 × 1280: xhdpi	16	Google APIs	arm	203 MB	
Nexus 6 API 21	1440 × 2560: 560dpi	21	Android 5.0.1	arm	650 MB	

+ Create Virtual Device...

Cancel OK

Cliquer «play»

13

Architecture Android

Le choix de l'architecture Android est basé sur l'idée de maîtriser les ressources et la consommation.

Les applications Android s'exécutent sur un système contraint (mémoire disponible, consommation de batterie, différence d'affichage, stockage disponible).

Vous devez en tant que développeur faire attention à plusieurs points :

La création de nouveaux objets.

L'utilisation des ressources (processeur, RAM, stockage, etc.).

La consommation de la batterie.

La diversité des tailles et des résolutions des écrans et des configurations matérielles.

La diversité des versions d'Android disponibles sur le marché.

L'architecture d'Android se compose de cinq parties distinctes :

Application : représente l'ensemble des applications fournies avec Android.

Framework Android : représente le framework permettant aux développeurs de créer des applications en accédant à l'ensemble des API et fonctionnalités disponibles sur le téléphone (fournisseur de contenu, gestionnaire de ressources, gestionnaire de notification, gestionnaire d'activité, etc.).

15

Bibliothèques : Android dispose d'un ensemble de bibliothèques utilisées par les différentes composantes du système.

Android Runtime : contient entre autres la machine virtuelle ART.

Linux Kernel : le noyau Linux (2.6) fournissant une interface avec le matériel, gérant la mémoire, les ressources et les processus Android.

16

ART (Android RunTime)

Android s'appuie sur une nouvelle machine virtuelle (depuis la version 5.0 Lollipop) particulière que l'on appelle **ART** (Android RunTime).

Elle remplace **Dalvik** (disponible dans les anciennes versions d'Android) et possède plusieurs spécificités dont :

1. AOT Compilation (Ahead of Time - compilation anticipée) qui permet de compiler une application à l'installation et non plus à l'exécution (Dalvik). Ce qui améliore grandement les performances des applications, vu qu'elles ne sont plus compilées à chaque lancement.
2. Amélioration du Garbage Collector (ramasse-miettes).
3. Amélioration du développement et du débogage d'applications (les plantages auront des messages plus détaillés).

17

Dalvik, la machine virtuelle utilisée dans les versions précédentes d'Android, se base sur de la compilation à l'exécution (Just In Time), ce qui signifie que votre application était compilée à chaque exécution.

Le passage à ART augmente les performances mais aussi l'espace de stockage nécessaire pour sauvegarder une application (stockage de la version compilée de l'application).

NDK (Native Development Kit)

Le développement natif est possible sous Android à l'aide du NDK (Native Development Kit - kit de développement natif), il s'effectue en C/C++.

Vous pouvez aussi utiliser un mécanisme Java vous permettant d'appeler du code natif depuis une classe JNI (Java Native Interface).

18

APK (Android Package)

Un APK est le binaire représentant une application. Ce format est utilisé pour distribuer et installer les applications.

Pour créer un APK, une application Android est compilée et packagée dans un fichier. Ce fichier contient :

Le code de l'application compilée (.dex).

Les ressources.

Les assets.

Les certificats.

Le fichier manifeste.

19

Pour pouvoir publier votre application sur **Google Play**, vous avez besoin de créer un fichier APK.

Pour générer votre APK depuis **Android Studio**, Android fournit un outil qui s'appelle **aapt** (Android Asset Packaging Tool - outil permettant de créer, visualiser et modifier des archives Android-APK), contenu dans le SDK Android et intégré à l'environnement de développement.

Vous pouvez générer deux types d'APK :

Un **APK signé**.

Un **APK non signé**. Ce format peut être utilisé pour tester votre application mais il ne peut pas être publié sur Google Play.

20

Création d'un keystore (voir dans un autre cours).