

「Icon Design Guidelines, Android 2.0」の日本語訳

本ドキュメントは、developer.android.com サイトで公開されている「Icon Design Guidelines, Android 2.0」をヒム・カンパニー 永井勝則が自主的に日本語に訳したものです。

<http://www.himco.jp/>

knagai@himco.jp

(2010/10)

本ドキュメントの原文は

http://developer.android.com/intl/ja/guide/practices/ui_guidelines/icon_design.html

です。

注意:

アイコンにエフェクトをかける各種数値については、原文の数値と Android Icon Templates Pack で適用されている数値が違っているものがあるので、これらのエフェクトを使用されるときには、原文と Android Icon Templates Pack の数値を確認してください。

Android 2.0 でのアイコンデザインのガイドライン

みなさんが作成されるアプリケーションは、ユーザーインターフェイス (UI) 全体を通した統一感のある外観と操作感を作成することで、その価値が増します。またグラフィックスタイルを無駄のないデザインにすることで、UI はさらにユーザーに訴求力を持ちます。

このドキュメントでは、Android 2.x フレームワークの一般的なスタイルに調和する、アプリケーション UI のさまざまな部分のアイコンを作成するときの情報を提供します。以下に示すガイドラインにしたがうと、ユーザーに統一感のある洗練された印象を与えることができます。

アイコンの作成を手早く開始するには、Android Icon Templates Pack が役立ちます。詳細は「Android Icon Templates Pack」節を参照してください。

密度専用のアイコンセットの提供

Android は、広範な画面サイズや解像度を持つさまざまなデバイスで実行できるよう設計されています。アプリケーションのアイコンをデザインするときには、アプリケーションはさまざまある中でのどのデバイスにインストールされるのか分からない、ということ念頭に置くことが大切です。「[さまざまな画面のサポート](#)」で述べているように、Android プラットフォームでは、どのような画面サイズや解像度であってもすべてのデバイス上で適切に表示される方法で、アイコンが簡単に提供できます。

訳者注：上記「[さまざまな画面のサポート](#)」ページには次の表 1 が記載されています。

表 1：Android SDK に含まれるエミュレータスキンの画面サイズと密度

	低密度 (120)、ldpi	中密度 (160)、mdpi	高密度 (240)、hdpi
小画面	QVGA (240x320)		
標準画面	WQVGA400 (240x400) WQVGA432 (240x432)	HVGA (320x480)	WVGA800 (480x800) WVGA854 (480x854)
大画面		WVGA800* (480x800) WVGA854* (480x854)	

HTC Desire、HTC Desire HD、Samsung Galaxy S はいずれも WVGA(480x800)で、Samsung Galaxy Tab は WSVGA(1024x600)とされています。

一般的に推奨されるのは、下の表1に挙げた、3つの汎用的な画面密度用のアイコンセットを別個に作成し、それらをアプリケーションのその密度専用のリソースディレクトリに保持する方法です。アプリケーションが実行されると、Androidプラットフォームはデバイス画面の特徴を調べ、適切な密度用のリソースからアイコンをロードします。アプリケーションに密度専用のリソースを保持する詳細については、「[画面のサイズと密度に関するリソースディレクトリの修飾子](#)」を参照してください。

Android デバイスの基本となる画面密度は中 (mdpi) です。このため、複数の画面密度用のアイコンの作成に推奨される方法は以下になります。

1. まず基本となる密度用のアイコンをデザインします (デザインするアイコンの実際のピクセル密度は表1を参照)。
2. そのアイコンをアプリケーションのデフォルトの drawable リソースに置き、Android 仮想デバイス (AVD) か、T-Mobile G1 などの HVGA デバイスでアプリケーションを実行します。
3. 結果を観察し、必要に応じて基本アイコンを修正します。
4. 基本となる密度として作成したアイコンに満足できたら、ほかの密度用のアイコンを拡大縮小して作成します。
 - ・基本アイコンを 150% 拡大して、高密度用のアセットを作成します。
 - ・基本アイコンを 75% 縮小して、低密度用のアセットを作成します。
5. アイコンを、アプリケーションの密度専用のリソースディレクトリに置きます。たとえば、
 - ・中密度のアセットは res/drawable-mdpi ディレクトリ (またはデフォルトの res/drawable/ ディレクトリ) に置きます。
 - ・高密度のアセットは res/drawable-hdpi ディレクトリに置きます。
 - ・低密度のアセットは res/drawable-ldpi ディレクトリに置きます。
6. テストし、必要に応じて高密度と低密度のアイコンを修正します。

複数密度のアイコンセットの作成と管理方法のティップスは「[デザイナー向けティップス](#)」を参照してください。

表1: 3つの一般的な各画面密度に対するアイコンサイズを、アイコンの種類別にまとめた表

アイコンの種類	一般的な画面密度に対する標準的なアセットサイズ (ピクセル単位)		
	低密度画面 (ldpi)	中密度画面 (mdpi)	高密度画面 (hdpi)
ランチャー	36 x 36 px	48 x 48 px	72 x 72 px
メニュー	36 x 36 px	48 x 48 px	72 x 72 px
ステータスバー	24 x 24 px	32 x 32 px	48 x 48 px
タブ	24 x 24 px	32 x 32 px	48 x 48 px
ダイアログ	24 x 24 px	32 x 32 px	48 x 48 px

リストビュー	24 x 24 px	32 x 32 px	48 x 48 px
--------	------------	------------	------------

ランチャーアイコン(起動アイコン)

ランチャーアイコンは、デバイスの Home 画面とランチャーウィンドウでアプリケーションを表すグラフィックです。



ランチャーアイコン(Nexus One のデフォルト)

ユーザーは、Home 画面下部にあるアイコンにタッチすることで、ランチャーを開きます。するとランチャーが開き、インストールされているすべてのアプリケーションのアイコンが現れます。アイコンは格子状に並んでいます。ユーザーは、ランチャーアイコンにタッチするか、トラックボールや D パッドといったハードウェアナビゲーションコントロールを使うことでアプリケーションを選択して開きます。

ユーザーはまた、ランチャーウィンドウから Home 画面までアイコンをドラッグできます。これはアプリケーションにアクセスするためのさらに便利な方法です。この場合システムはアプリケーションのランチャーアイコンを Home 画面の壁紙を背景にして表示し、ランチャー内部と同じサイズでレンダリングします。

システムは、すべてのランチャーアイコンが一樣な高さと幅でレンダリングされるように、ランチャーアイコンの拡大縮小を管理します。任意のデバイスでレンダリングされるランチャーアイコンの実際のピクセルサイズは、そのデバイス画面のサイズとピクセル密度の特性によって変わります。みなさんの作成するアイコンが可能な限り最良にレンダリングされるようにするには、低、中、高密度画面用にデザインしたアイコンを提供するようにします。詳細は前述した「密度専用のアイコンセットの提供」か、後の「デザイナー向けティップス」を参

照してください。

スタイル

ランチャーアイコンは以下の一般的なスタイルの指針に沿って作成します。このガイドラインの目的は、アイコンに対する工夫を制限するものではなく、アイコンをほかのユーザーと共有できるように一般化しようとすることにあります。図1はその例です。



図1：ランチャーアイコンのスタイルの例

クリーンで今風であること：

- ・ランチャーアイコンは今風であるべきで、時にはひとひねりも必要です。古びてみずばらしいのはだめです。使い古された象徴的なメタファーはつねに避けるべきです。

シンプルで図象的であること：

- ・Android のランチャーアイコンは本質的に風刺的です。作成するアイコンは、小さなサイズでも適切に使用されるように、高度に単純化されデフォルメされるべきで、過度に複雑にしてはいけません。
- ・アプリケーション全体を表現する象徴として、アプリケーションの一部を使用する方法も考慮します（たとえば、Music アイコンにスピーカーを使います）。
- ・リアルな（しかし写真のように決して見えない）レンダリングには、幾何学的な形状でも有機的な形状でも、そのままの外形線を使用します。
- ・アイコンは、それより大きなイメージの切り取り版であってはいけません。

手触り感のある質感：

- ・アイコンには光沢のない、手触り感のあるマテリアルが必要です。詳細は後の「マテリアルとカラー」を参照してください。

正対し、ライトは上から:

・新しい Android2.0 とそれ以降のプラットフォームでは、ランチャーアイコンは正対し、パースはほとんどなく、上からライトが当たっていることが求められます。

またすべてのアイコンには、テキストラベルを別につけます。これによってアイコンのデザイン時、埋め込んだテキストに関わる作業がなくなり、アイコンの視覚的な独自性と覚えやすさに集中して取り組むことができます。

ビルトインの Android アプリケーションが使用するランチャーアイコンの例は、本ドキュメント付録の「標準的なランチャーアイコン」を参照してください。

OKとNG

以下は、アプリケーションのアイコンを作成するときに考慮すべき、“OKとNG”の例です。

Android ランチャーアイコンは...	Android ランチャーアイコンは...
<ul style="list-style-type: none">・モダンで、小さく、マット(光沢なし)で、手触り感があること。・正対し、上からのライトを受け、何も欠けず、カラーパレットのカラーが使われている。	<ul style="list-style-type: none">・時代遅れで、複雑すぎ、光沢があり、平板なベクターであってはいけない。・回転していたり、切り取られ、彩度を上げすぎではいけない。
	
	

図2: Android ランチャーアイコンの“OKとNG”の例、上が OK、下が NG

マテリアルとカラー

ランチャーアイコンには手触り感があり、上からライトが当たったマテリアルを使用すべきです。形状は単純であっても、実際の素材から削り出したような質感を生み出すよう努めるべきです。

Android プラットフォームのデフォルトアプリケーション用ランチャーアイコンでは、図3に示すマテリアルが使用されています。みなさんのアイコンではこれらのマテリアルを使うこともできますし、新たに作成することもできます。

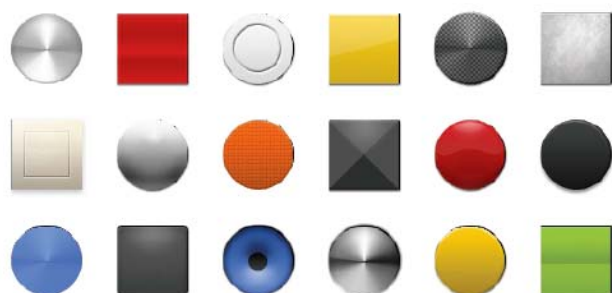


図3: アイコンの作成に使用できるマテリアルの例

Android のランチャーアイコンは通常、大きな基本シェイプとその中に収めた小さなシェイプで構成され、中間色と原色を組み合わせで使います。中間色を組み合わせた場合でも、コントラストは高く維持すべきです。1つのアイコンでは原色を2色以上使わないようにします。

ランチャーアイコンのカラーは、中間色と原色の範囲を限定したカラーパレットのものを使用します。彩度は上げすぎないようにします。

図4では、ランチャーアイコンの使用に推奨されるカラーパレットを示しています。このパレットの要素は基本カラーと強調カラーの両方に使用できます。パレットのカラーは、白から黒へ垂直方向に線状にグラデーションするオーバーレイと合わせて使用できます。これによってアイコンに上からライトが当たった印象を与えつつ、彩度が維持できます。

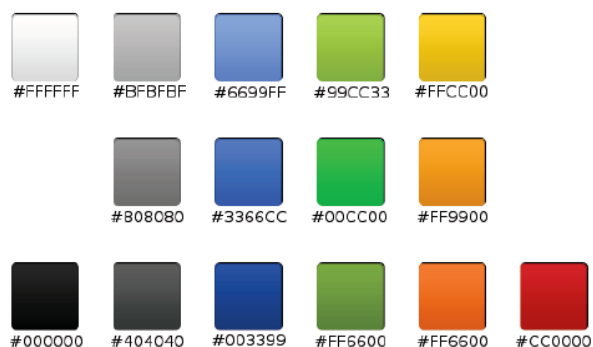


図4: アイコンの作成に推奨されるカラーパレット

図3のマテリアルと、図4のカラーパレットのカラーを強調カラーとして組み合わせると、図5に示すようなマテリアル

アルの組み合わせが作成できます。この作業の開始に当たっては、Photoshop のテンプレートファイル (Launcher-icon-template.psd) を含むアイコンパックが利用できます。このパックには、デフォルトのマテリアルやカラー、グラデーションが用意されています。

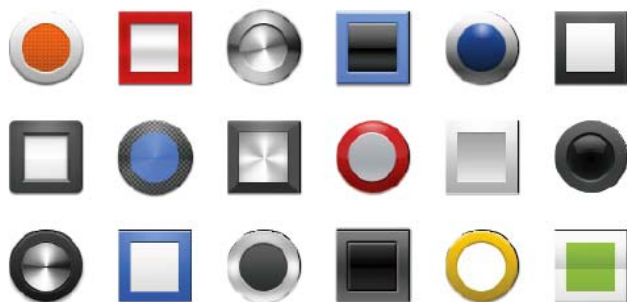


図4：推奨パレットのカラーを基本カラーと強調カラーとして組み合わせたマテリアルの例

サイズと位置

ランチャーアイコンの形状にはさまざまなものが使用できますが(たとえば地球やかばん、吹き出しなど)、アセット内で適切に縮小し配置して、一貫した視覚的な重みをつける必要があります(地球もかばんも吹き出しも、適切に縮小し配置することで、同じような大きさのアイコンに見えるようにする)。

図6では、アセット内でのアイコンのさまざまな配置方法を示しています。図内で詳しく述べているように、アイコンのサイズは、一貫した視覚的な重みづけを行い、シャドウ効果で必要になるスペースを取っておくために、アセットの実際の境界よりも小さくすべきです。アイコンが正方形かそれに近い場合には、縮小する必要があります。

- ・フルアセット(アセットの目一杯のサイズ)の境界ボックスは赤線で示しています。
- ・実際のアイコン自体の推奨される境界ボックスは青線で示しています。このアイコンボックス(青線)は、シャドウや特殊なアイコン処理にスペースが必要になるので、フルアセットボックス(赤線)よりも小さくします。
- ・正方形アイコンの推奨される境界ボックスはオレンジで示しています。正方形アイコンのボックスは、正方形でないアイコンのボックスとの一貫した視覚的な重みづけを行うため、正方形でないアイコンのボックスよりも小さくします(正方形アイコンを目一杯大きくすると、正方形でないアイコンよりも見た目が大きくなる)。

高密度 (hdpi) 画面用 アイコンサイズ						
フルアセット	72 x 72 px		✓	✓	✓	✓
アイコン	60 x 60 px					
正方形アイコン	56 x 56 px					
中密度 (mdpi) 画面用 アイコンサイズ						
フルアセット	48 x 48 px		✓	✓	✓	✓
アイコン	40 x 40 px					
正方形アイコン	38 x 38 px					
低密度 (ldpi) 画面用 アイコンサイズ						
フルアセット	36 x 36 px		✓	✓	✓	✓
アイコン	30 x 30 px					
正方形アイコン	28 x 28 px					

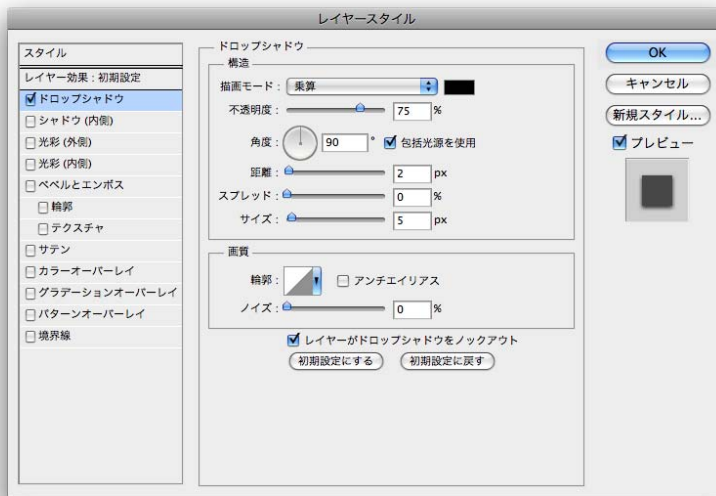
図6: アイコンアセットの境界内でのアイコンのサイズと位置

ランチャーアイコンテンプレートの使用

Android Icon Templates Pack 2.0 には、デフォルトアイコンのマテリアルやカラーのパレットを含むテンプレートが用意されています。このテンプレートは .psd 形式で提供されているので、Adobe Photoshop や同種のラスターイメージエディタ (Gimp など) で開くことができます。

開始するにはまず、[Android Icon Templates Pack 2.0](#) をダウンロードします。パックをダウンロードしたらそれを展開し、Adobe Photoshop や同種のラスターイメージ編集プログラムで Launcher-icon-template.psd を開きます。このテンプレートは、ランチャーアイコンを作成するときの開始点として使用できます。

アイコンを作成したら、次の仕様にしたがってシャドウ効果が追加できます。



WVGA(高密度)画面用シャドウ:

効果: ドロップシャドウ

カラー: #000000

描画モード: 乗算

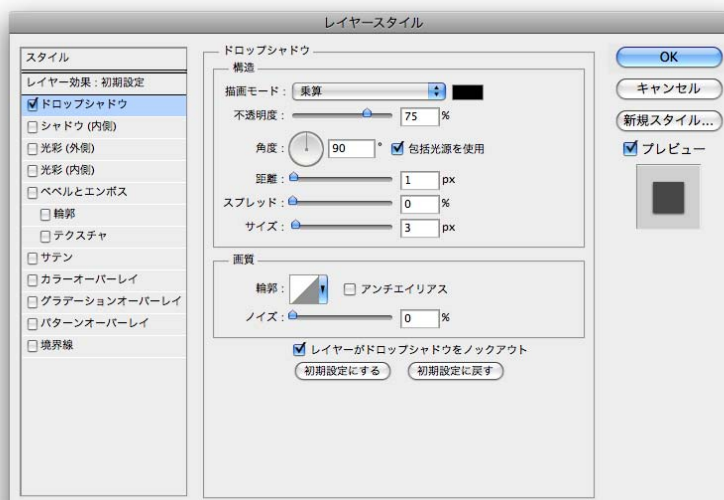
不透明度: 75%

角度: 90 度

距離: 2px

スプレッド: 0%

サイズ: 5px



HVGA(中密度)画面用シャドウ:

効果: ドロップシャドウ

カラー:#000000

描画モード:乗算

不透明度:75%

角度:90度

距離:1px

スプレッド:0%

サイズ:3px

シャドウを追加しアイコンを完成させたら、透明度を有効にしたPNGファイルとして書き出します。その際には必ず、高密度画面用のアイコンのサイズは 72 x 72 ピクセルに、中密度画面用のアイコンのサイズは 48 x 48 ピクセルにします。ランチャーアセットを高、中、低密度画面用に用意しなければならない理由については、「[複数画面のサポート](#)」を参照してください。

メニューアイコン

メニューアイコンは、ユーザーが Menu ボタンを押したときに表示されるポップアップメニュー内のグラフィック要素です。正面を向いたフラットな画像で、少し奥行きを持たせます。メニューアイコン内の要素は 3D 風に視覚化してはいけません。



メニューアイコン(画面下部の[検索]、[アップロード]など)

「密度専用のアイコンセットの提供」節で述べたように、アイコンセットは高、中、低密度画面別に作成すべきです。これによって、アプリケーションがインストールされるすべてのデバイスでアイコンが適切に表示され

ようになります。各密度について推奨されるアイコンの完成サイズは表1に挙げています。また複数のアイコンセットの使用方法については「デザイナー向けティップス」が参考になります。

構造

- ・一貫性を保つため、すべてのメニューアイコンでは同じメインパレットと同じ効果を使用します。詳細は「カラーパレット」節を参照してください。
- ・メニューアイコンの角は丸めるべきですが、これは論理的に妥当な場合だけです。たとえば図7に示す角丸の論理的な位置は屋根で、ほかの部分は丸めません。
- ・ここで指定しているサイズはすべて、6ピクセルのセーフフレームを取った 48 x 48 ピクセルのアートボードサイズにもとづいています（外枠が 48 x 48 ピクセルで、安全のため上下左右に 6ピクセルのセーフゾーンを設けた中にアイコンをデザインする）。
- ・次の「ライト、エフェクト、シャドウ」節で述べているメニューアイコンのエフェクト（外側グロー）は 6 ピクセルのセーフフレームに重ねられますが、これは必要なときに限ります。基本となる形状はつねにセーフフレームの中に収める必要があります。
- ・最終的なアートは透明度を有効にした PNG ファイルとして書き出す必要があります。
- ・Adobe Photoshop でメニューアイコンを作成するときのテンプレートは Icon Templates Pack にあります。

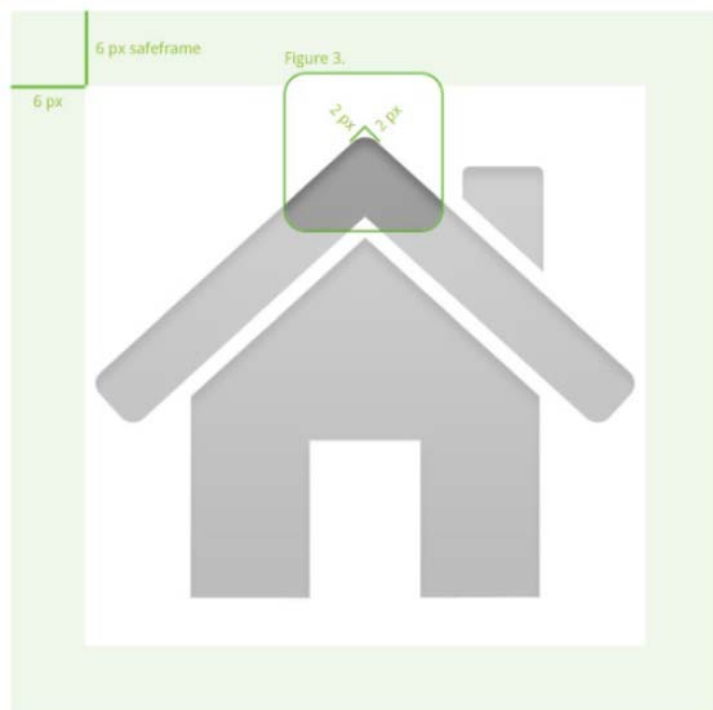


図7：メニューアイコンで使用するセーフフレームと丸めた角。アイコンサイズは 48 x 48

ライト、エフェクト、シャドウ

メニューアイコンはフラットで正面を向いています。図8に示すように、奥行き感を出すために少しのへこみとほかのエフェクトも加えられます。

	図8:メニューアイコンのライトとエフェクト、シャドウ	
	①正面	メインのカラーパレットを使ってグラデーションで塗る
	②シャドウ(内側)	黒 不透明度 21% 角度 120 度 距離 2px サイズ 2px
	③光彩(外側)	白 不透明 20% スプレッド 10% サイズ 3px
④ベベル(内側)	深さ 1% 方向 下へ サイズ 0px 角度 90 度 高度 11 度 ハイライト 白 不透明度 68% シャドウ 黒 不透明度 25%	

カラーパレット

	白 R 255 G 255 B 255 光彩(外側)とベベルのハイライトに使用
	グラデーション塗り カラーの塗りに使用 1: R 163 G 163 B 163 2: R 120 G 120 B 120
	黒 R 0 G 0 B 0 シャドウ(内側)とベベルのシャドウに使用

手順

1. Adobe Illustrator などのツールを使って基本となるシェイプを作成します。
2. シェイプを Adobe Photoshop などのツールに読み込み、透明な背景の上の 48 x 48 ピクセルのイメージにぴったり合うよう縮小します。
3. 図8で述べたエフェクトを追加します。

4. アイコンを、透明度を有効にした PNG ファイルとして 48 x 48 のサイズで書き出します。

OKとNG

以下は、メニューアイコンを作成するときに考慮すべき、“OKとNG”の例です。



ステータスバーアイコン

ステータスバーアイコンはアプリケーションからの通知をステータスバーに表すために使用されます。グラフィックとしてはメニューアイコンによく似ていますが、メニューアイコンより小さく、コントラストが高くなっています。



ステータスバーアイコン(画面の最上部、時刻の左)

「密度専用のアイコンセットの提供」節で述べたように、アイコンセットは高、中、低密度画面別に作成すべきです。これによって、アプリケーションがインストールされるすべてのデバイスでアイコンが適切に表示されるようになります。各密度について推奨されるアイコンの完成サイズは表1に挙げています。また複数のアイコンセットの使用方法については「デザイナー向けティップス」が参考になります。

構造

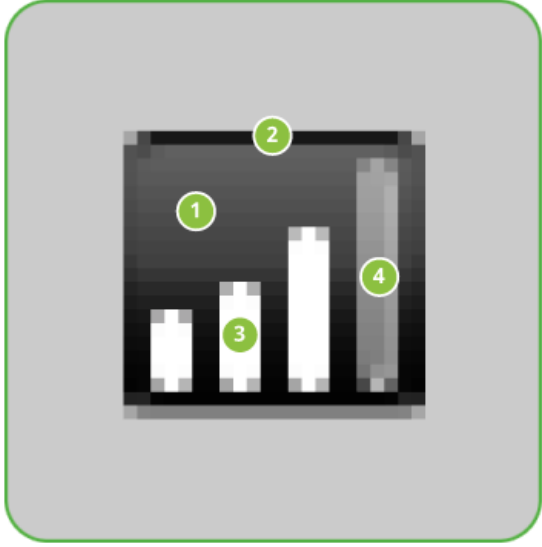
- ・ステータスバーの基本シェイプと詳細部分には、図9に示すようにつねに丸めた角が適用されなければなりません。
- ・ここで指定しているサイズはすべて、2ピクセルのセーフフレームを取った 25 x 25 ピクセルのアートボードサイズにもとづいています(外枠が 25 x 25 ピクセルで、安全のため上下左右に 2ピクセルのセーフゾーンを設けた中にアイコンをデザインする)。
- ・ステータスバーアイコンの左右は、必要に応じてセーフフレームに重ねることができますが、上下を重ねてはいけません。
- ・最終的なアートは透明度を有効にした PNG ファイルとして書き出す必要があります。
- ・Adobe Photoshop でステータスバーアイコンを作成するときのテンプレートは Icon Templates Pack にあります。



図9: ステータスバーアイコンで使用するセーフフレームと丸めた角。アイコンサイズは 25 x 25





ライト、エフェクト、シャドウ

ステータスバーアイコンは、小さなサイズでもはっきり見えるように、少しへこみ、ハイコントラストで正面を向いています。

	図 10:ステータスバーアイコンのライト、エフェクト、シャドウ	
	① 正面	メインのカラーパレットを使ってグラデーションで塗る
	② ベベル(内側)	深さ 100% 方向 下へ サイズ 0px 角度 90度 高度 30度 ハイライト 白 不透明度 75% シャドウ 黒 不透明度 75%
	③ 詳細	白
④ 動作しない詳細	パレットのグレーのグラデーション + ベベル(内側): 滑らかに 深さ 1% 方向 下へ サイズ 0px 角度 117度 高度 42度 ハイライト 白 不透明度 70% シャドウ なし	

カラーパレット

電話機能に関するステータスバーアイコンのみフルカラーを使用し、それ以外のステータスバーアイコンはすべて単彩すべきです。

	白 R 255 G 255 B 255 アイコン内の詳細とベベルのハイライトに使用
	グラデーション アイコン内の動作しない詳細に使用 1: R 169 G 169 B 169 2: R 126 G 126 B 126
	グラデーション塗り カラーの塗りに使用 1: R 105 G 105 B 105 2: R 10 G 10 B 10
	黒 R 0 G 0 B 0 ベベルのシャドウに使用

手順

1. Adobe Photoshop などのツールで、透明な背景の上の 25 x 25 ピクセルのイメージ内に基本となるシェイプを作成します。セーフフレームに注意し、上部と下部 2 ピクセルは空けるようにします。
2. 図8で示したように、角を丸めます。
3. 図 10 に示したように、ライト、エフェクト、シャドウを加えます。
4. アイコンを、透明度を有効にした PNG ファイルとして 25 x 25 ピクセルで書き出します。

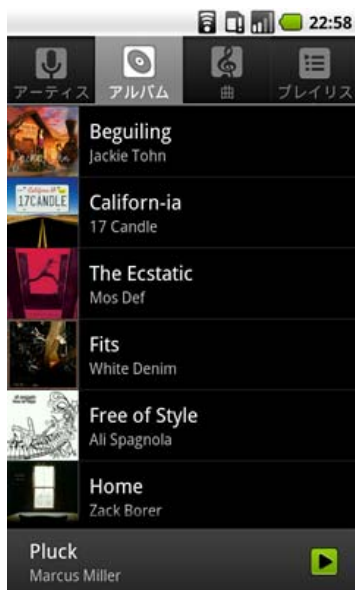
OKとNG

以下は、ステータスバーアイコンを作成するときに考慮すべき、“OKとNG”の例です。



タブアイコン

タブアイコンは、マルチタブインターフェイスの個々のタブを表すために使用されるグラフィック要素です。タブアイコンはそれぞれ、非選択と選択の2つの状態を持っています。



タブアイコン: 上部の[アーティスト]、[アルバム]など

「密度専用のアイコンセットの提供」節で述べたように、アイコンセットは高、中、低密度画面別に作成すべきです。これによって、アプリケーションがインストールされるすべてのデバイスでアイコンが適切に表示されるようになります。各密度について推奨されるアイコンの完成サイズは表1に挙げています。また複数のアイコンセットの使用方法については「デザイナー向けティップス」が参考になります。

構造

- ・選択状態のタブアイコンには、メニューアイコンと同じグラデーション塗りとエフェクトがありますが、光彩(外側)はありません。
- ・選択状態のタブアイコンは非選択状態とタブアイコンと同じように見えますが、かすかなシャドウ(内側)があり、正面部にダイアログアイコンと同じグラデーションがかかっています。
- ・タブアイコンのセーフフレームは 1px で、ここにはアンチエイリアス処理された丸いシェイプのエッジだけを重ねさせます。
- ・ここで指定しているサイズはすべて、32 x 32 ピクセルのアートボードサイズにもとづいています。Photoshop のテンプレートの境界ボックスの周囲には 1 ピクセルの余白を維持してください。

- ・最終的なアートは透明度を有効にした 32 x 32 の PNG ファイルとして書き出す必要があります。
- ・Adobe Photoshop でタブアイコンを作成するときのテンプレートは Icon Templates Pack にあります。



図 11: 非選択タブアイコンのセーフフレームとグラデーション塗り。アイコンサイズは 32 x 32



図 12: 選択状態にあるタブアイコンのセーフフレームとグラデーション塗り。アイコンサイズは 32 x 32

非選択状態のタブアイコン

ライト、エフェクト、シャドウ

非選択状態のタブアイコンは選択状態のタブアイコンと同じように見えますが、かすかなシャドウ(内側)があり、正面部にダイアログアイコンと同じグラデーションがかかっています。

	図 13: 非選択タブアイコンのライト、エフェクト、シャドウ	
	① 正面	グラデーションオーバーレイ: 角度 90 度 下のカラー R 223 G 223 B 223 位置 0% 上のカラー R 249 G 249 B 249 位置 75%
	② シャドウ(内側)	黒 不透明度 70% 角度 90 度 距離 1px サイズ 2px
	③ ベベル(内側)	深さ 1% 方向 下へ サイズ 0px 角度 90 度 高度 11 度 ハイライト 白 不透明度 0% シャドウ 黒 不透明度 25%

手順


1. Adobe Illustrator などのツールを使って、基本となるシェイプを作成します。
2. シェイプを Adobe Photoshop などのツールに読み込み、透明の背景上の 32 x 32 ピクセルのイメージにぴったり合うよう縮小します。
3. 図 13 に示す非選択状態のエフェクトを追加します。
4. 透明度を有効にした PNG ファイルとして、32 x 32 のサイズでアイコンを書き出します。

選択状態のタブアイコン

選択状態のタブアイコンには、メニューアイコンと同じグラデーション塗りとエフェクトがありますが、光彩(外側)はありません。

	図 14: 選択タブアイコンのライト、エフェクト、シャドウ	
	① 正面	カラーパレットのカラーからのグラデーション塗りを使う
	② シャドウ(内側)	黒 不透明度 70% 角度 90 度 距離 1px サイズ 2px
	③ ベベル(内側)	深さ 1% 方向 下へ サイズ 1px 角度 90 度 高度 11 度 ハイライト 白 不透明度 75% シャドウ 黒 不透明度 0%

カラーパレット

	グラデーション塗り 1: R 165 G 165 B 165 2: R 123 G 123 B 123 非選択のタブアイコン上で塗り色として使用
---	--

手順

1. Adobe Illustrator などのツールを使って、基本となるシェイプを作成します。
2. シェイプを Adobe Photoshop などのツールに読み込み、透明の背景上の 32 x 32 ピクセルのイメージにぴったり合うよう縮小します。
3. 図 14 に示す選択状態のエフェクトを追加します。
4. 透明度を有効にした PNG ファイルとして、32 x 32 のサイズでアイコンを書き出します。

ダイアログアイコン

ダイアログアイコンは、ユーザーに操作を求めるポップアップダイアログボックスで表示されます。ダイアログアイコンには、暗い背景に映えるように明るいグラデーションとシャドウ(内側)が使用されます。



ダイアログアイコン: ダイアログボックスの左上にあるスピーカーのアイコン

「密度専用のアイコンセットの提供」節で述べたように、アイコンセットは高、中、低密度画面別に作成すべきです。これによって、アプリケーションがインストールされるすべてのデバイスでアイコンが適切に表示されるようになります。各密度について推奨されるアイコンの完成サイズは表 1 に挙げています。また複数のアイコンセットの使用方法については「デザイナー向けティップス」が参考になります。

構造

- ・ダイアログアイコンは 1 ピクセルのセーフフレームを持っています。基本シェイプはこのセーフフレーム内に収めなくては行けませんが、丸いシェイプのアンチエイリアス処理された部分はセーフフレームに重ねることができません。
- ・ここで指定しているサイズはすべて、Adobe Photoshop の 32 x 32 ピクセルのアートボードサイズにもとづいています。Photoshop のテンプレートの境界ボックスの周囲には 1 ピクセルの余白を維持してください。
- ・最終的なアートは透明度を持った PNG ファイルとして書き出す必要があります。
- ・Adobe Photoshop でダイアログアイコンを作成するときのテンプレートは Icon Templates Pack にあります。



図 15: ダイアログアイコンのセーフフレームとグラデーション塗り。アイコンサイズは 32 x 32

ライト、エフェクト、シャドウ

ダイアログアイコンはフラットで正面を向いています。暗い背景に際立たせるため、明るいグラデーションとシャドウ(内側)を使って作成します。

	<p>図 16: ダイアログアイコンのライト、エフェクト、シャドウ</p>	
	<p>① 正面</p>	<p>グラデーションオーバーレイ 角度 90 度 下のカラー R 223 G 223 B 223 位置 0%、上のカラー R 249 G 249 B 249 位置 75%</p>
	<p>② シャドウ(内側)</p>	<p>黒 不透明度 25% 角度 -90 度 距離 1px サイズ 0px</p>

手順

1. Adobe Illustrator などのツールを使って、基本となるシェイプを作成します。
2. シェイプを Adobe Photoshop などのツールに読み込み、透明の背景上の 32 x 32 ピクセルのイメージにぴったり合うよう縮小します。
3. 図 16 に示すエフェクトを追加します。
4. 透明度を有効にした PNG ファイルとして、32 x 32 のサイズでアイコンを書き出します。

リストビューアイコン

リストビューアイコンはダイアログアイコンによく似ていますが、シャドウ(内側)エフェクトが使用されます(オブジェクトの上部にある光源によって影ができる)。リストビューアイコンは [ListView](#) 内でのみ使用されます。このアイコンの例は、Android マーケットアプリケーションのホーム画面やマップアプリケーションの経路画面で見られます。



リストビューアイコン: [注目] の下のリスト左に見えるかばんのアイコン

「密度専用のアイコンセットの提供」節で述べたように、アイコンセットは高、中、低密度画面別に作成すべきです。これによって、アプリケーションがインストールされるすべてのデバイスでアイコンが適切に表示されるようになります。各密度について推奨されるアイコンの完成サイズは表 1 に挙げています。また複数のアイコンセットの使用方法については「デザイナー向けティップス」が参考になります。

構造

・リストビューアイコンには通常 1 ピクセルのセーフフレームがありますが、丸いシェイプのアンチエイリアス処理

されたエッジと重なってもかまいません。


- ・ここで指定しているサイズはすべて、Photoshop の 32 x 32 ピクセルのアートボードサイズにもとづいています。テンプレートの境界ボックスの周囲には 1 ピクセルの余白を維持してください。
- ・最終的なアートは透明度を持った PNG ファイルとして書き出す必要があります。
- ・Adobe Photoshop でリストビューアイコンを作成するときのテンプレートは Icon Templates Pack にあります。



図 17: リストビューアイコンのセーフフレームとグラデーション塗り。アイコンサイズは 32 x 32

ライト、エフェクト、シャドウ

リストビューアイコンはフラットで正面を向き、シャドウ(内側)が適用されます。明るいグラデーションとシャドウ(内側)が使用されるので、暗い背景で引き立ちます。

	図 18:リストビューアイコンのライト、エフェクト、シャドウ	
	① シャドウ(内側)	黒 不透明度 57% 角度 120 度 描画モード 通常 距離 1px サイズ 1px
	② 背景	黒 標準的なシステムカラー アイコンはリストビューでのみ表示される
Note:リストビューアイコンは、Photoshop の 32 x 32 ピクセルのアートボードに、セーフフレームなしで置かれる		

手順

1. Adobe Illustrator などのツールを使って、基本となるシェイプを作成します。
2. シェイプを Adobe Photoshop などのツールに読み込み、透明の背景上の 32 x 32 ピクセルのイメージにぴったり合うよう縮小します。
3. 図 18 に示すエフェクトを追加します。
4. 透明度を有効にした PNG ファイルとして、32 x 32 のサイズでアイコンを書き出します。

デザイナー向けティップス

以下は、アプリケーションで使用するアイコンやそのほかの drawable のアセットの開発で役立つと思われるティップスです。このティップスでは、Photoshop や同種のラスターイメージ編集プログラムの使用を想定しています。

アイコンアセットには共通の命名規則を使用する

ファイルに名前をつけるときには、アルファベット順でソートするとき、関係するアセットがディレクトリ内でグループ化されるような名前をつけるようにします。具体的には、アイコンの各タイプに共通する接頭辞をつけま

す。以下はその例です。

アセットの種類	接頭辞	例
アイコン	ic_	ic_star.png
ランチャーアイコン	ic_launcher	ic_launcher_calendar.png
メニューアイコン	ic_menu	ic_menu_archive.png
ステータスバーアイコン	ic_stat_sys か ic_stat_notify	ic_stat_notify_msg.png
タブアイコン	ic_tab	ic_tab_recent.png
ダイアログアイコン	ic_dialog	ic_dialog_info.png

共通する接頭辞の使用は必須ではありません。ただ作業が行いやすくなるだけのことです。

作業スペースは、ファイルを複数の密度に応じて分類するよう設定する

異なる画面密度に対応するように複数のアセットを開発するということは、ファイルのコピーを複数作成することです。ファイルの複数のコピーを安全に保持し見つけやすくするには、作業スペースに解像度ごとにアセットのファイルを分類するディレクトリ構造を作成方法が推奨されます。以下はその例です。

```
assets/...
  ldpi/...
    _pre_production/...
      作業ファイル.psd
    アセットの完成ファイル.psd
  mdpi/...
    _pre_production/...
      作業ファイル.psd
    アセットの完成ファイル.psd
  hdpi/...
    _pre_production/...
      作業ファイル.psd
    アセットの完成ファイル.psd
```

これは密度専用のディレクトリが並列した平行構造で、完成したアセットは最終的にアプリケーションのリソースに保持することになります。作業スペースのこの構造はアプリケーションのディレクトリ構造と似ているので、アプリケーションのリソースディレクトリにどれをコピーすればよいのか、一目瞭然です。アセットを密度で

分けることはまた、ファイル名の不一致の発見にも役立ちます。

参考までに、以下は典型的なアプリケーションのリソースディレクトリの構造です。

```
res/...
  drawable-ldpi/...
    アセットの完成版.png
  drawable-mdpi/...
    アセットの完成版.png
  drawable-hdpi/...
    アセットの完成版.png
```

まず中密度のアセットを作成する

中密度は Android の基準なので、デザインはまず mdpi アセットの作成から開始します。各種アイコンの実際のピクセルサイズは表1を参照してください。可能な場合には、Photoshop のレイヤーでベクターアートやパスを使用します。これによって、アセットの後からの拡大縮小が容易になります。

用心が必要なアセットや、境界ボックスのサイズが同じアセット同等のファイルについては、作業用の Photoshop ファイルを作成し、_pre_production ディレクトリに保存します。たとえば ic_tabs_phone_mdpi.psd というようなファイルです。これによって、変更が必要な場合、個々のアセットが探しやすく編集しやすくなります。また作業ファイルの名前の末尾にその密度を表す接尾辞(たとえば _mdpi.psd)をつけると、ファイルを編集するとき、間違いがなくなります。

mdpi 作業ファイルからは、密度専用のリソースディレクトリ(今の場合は mdpi/)に、平板化した個々のアセット(psd ファイルから書き出したアイコンの PNG ファイル)を保存します。

中密度リソースから高、低密度アセットを作成する

中密度アセットの作業が終わったら、作業ファイルを mdpi/_pre_production ディレクトリから ldpi と hdpi の対応する場所にコピーします。作業ファイルの名前に密度固有の接頭辞をつけた場合には、接頭辞を対応する密度の名前(_ldpi か_hdpi)に変えます。

次に高密度か低密度のディレクトリで作業ファイルを開き、その密度に一致するようイメージを拡大または縮小します。hdpi アセットを作成するには、イメージを 150%拡大します。ldpi アセットなら 75%縮小します。イメージの縮小は次のようにします。

1. Photoshop か同種のプログラムで作業ファイルを開きます。
2. [イメージ]メニューから[画像解像度]を選びます。
3. [画像解像度]パネルで、[幅]のポップアップメニューを[%]に変えます。
4. [幅]の値を hdpi アセットの場合には 150 に、ldpi アセットの場合には 75 に変更します。
5. [スタイルを拡大・縮小]チェックボックスを選択します。
6. [縦横比を固定] チェックボックスを選択します。
7. [画像の再サンプル] チェックボックスを選択し、ポップアップメニューを[バイキュービック法(滑らかなグラデーションに最適)]に設定します。
8. [OK]をクリックします。

イメージを拡大または縮小したら、その密度専用のリソースディレクトリに保存します。

9-patch イメージを拡大縮小する場合には、イメージのエッジのチェックマークを処理する方法を述べた2つ後の節を参照してください。

訳者注：

9-patchは[Draw 9-patch](#)ページで解説されています。以下はその翻訳です。

Draw 9-patch

Draw 9-patch ツール(Android SDK の tools/draw9patch)を使うと、グラフィックが WYSIWYG エディターで簡単に作成できます。

Nine-patchグラフィックとその使い方については、[Nine-patchイメージ](#)トピックのNine-patch節を参照してください。

以下は Draw 9-patch ツールを使って Nine-patch グラフィックを作成する概要です。ここでは NinePatch にしたい PNG 画像が必要になります。

1. ターミナルから、SDK の/tools ディレクトリにある draw9patch を起動します。
2. PNG 画像を Draw 9-patch ウィンドウにドラッグします(または[File]→[Open 9-patch]を選んで PNG 画像を開きます)。ワークスペースが開きます。

左ペインは描画領域で、伸縮可能なパッチ(つぎ、当て布のこと)と内容領域の線が編集できます。右ペインはプレビュー領域で、伸縮後のグラフィックがプレビューできます。

3. 1ピクセルの周囲領域内(画像の周りには1ピクセルの余白が作成される)をクリックし、線を描画します。この線は伸縮を可能にするパッチと(オプションで)内容領域を定義します(伸縮させたくない角などはクリックしません)。前に描いた線は右クリック(MacはShift-クリック)で消去できます。
4. 終わったら[File]→[Save 9-patch]を選択します。画像が.9.pngというファイル名で保存されます。

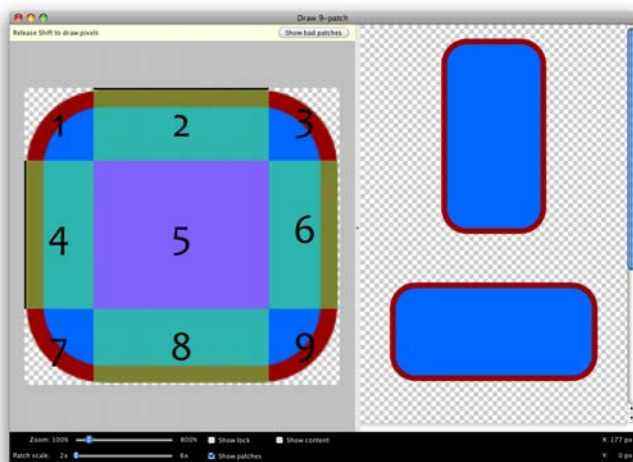
Note: 通常のPNGファイル(*.png)は、画像の周囲に追加される空の1ピクセルの境界とともにロードされます。伸縮可能にするパッチと内容領域はこの1ピクセルの境界に描画します。1度保存した9-patchファイル(*.9.png)は、パッチがすでに存在しているので新たに追加されず、保存したときそのままロードされます。

オプションで以下の制御が行えます。

- ・[Zoom]: 描画領域内でグラフィックのズームレベルを調整します。
- ・[Patch scale]: プレビュー領域で画像の伸縮を調整します。
- ・[Show lock]: マウスオーバー時にグラフィックの描画されない領域を視覚化します。
- ・[Show patches]: 描画領域で伸縮可能なパッチをプレビューします(伸縮可能なパッチがピンクで表示されます)。
- ・[Show content]: プレビュー画像で内容領域を強調表示します(内容は紫で表示されます)。
- ・[Show bad patches]: 伸縮時、グラフィック内にアーティファクト(ノイズなどよくない結果のこと)を発生させる可能性のある赤い境界をパッチ領域の周りに追加します。この“悪いパッチ“をすべて取り除くと、画像は伸縮されても視覚的な一貫性が保たれます。

訳者注:

9パッチは簡単にいうと、次のような仕組みで実現されます。



この図では1ピクセルの黒線を2の上と4の左で定義しています。すると、この画像を1から9までの部分

に分けることができます。1、3、7、9 は伸縮しない部分、2と8 は横方向にのみ伸縮する部分、4と6 は縦方向にのみ伸縮する部分、5 は縦にも横にも伸縮する部分です。角丸矩形をこのように9つの部分に分けることで、右のプレビュー領域のようにきれいに拡大されます。

拡大縮小後、必要に応じてビットマップレイヤーを再描画する

イメージを、ベクターレイヤーでなくビットマップレイヤーから拡大した場合には、高い密度に適応するよう手動で再描画する必要があるかもしれません。たとえば mdpi 用に 60 x 60 の円をビットマップとして描いた場合には、hdpi 用に 90 x 90 の円として描き直す必要が生じるかもしれません。

nine-patch イメージを拡大縮小するときには、チェックマークを削除してから拡大縮小し、その後チェックマークを再配置する

nine-patch イメージには、イメージのエッジにチェックマーク(黒線のこと)が含まれています。nine-patch イメージを拡大縮小するときには、このチェックマークも拡大縮小されるので、結果は正確でなくなります。nine-patch イメージの拡大縮小で推奨される方法は、拡大縮小する前にソースイメージからチェックマークを削除し、拡大縮小した後チェックマークを適切なサイズで手動で再配置する方法です。

作業ファイルを新しい解像度に拡大縮小した後、チェックマークを簡単に決めるには、拡大縮小する前にまず、チェックマークを含む平板化したイメージの一時的な複製を作成します。

1. [選択範囲]→[すべてを選択]を選びます。
2. [編集]→[結合部分をコピー]を選びます。
3. [ファイル]→[新規]を選び、[OK]をクリックします。
4. [編集]→[ペースト]を選びます。

一時コピーを作成したら作業ファイルに戻り、イメージを拡大縮小する前に、作業ファイルからチェックマークを削除します。

1. [イメージ]→[キャンバスサイズ]コマンドを選びます。
2. [キャンバスサイズ]パネルで、[幅]と[高さ]値から2ピクセル引きます。
3. アンカーをセンターに設定します。
4. [OK]をクリックします。

作業ファイルをターゲット密度に拡大縮小します。拡大縮小した作業ファイルで、チェックマークを再描画できるようにキャンバスを拡大します。

1. [イメージ]→[キャンバスサイズ]コマンドを選びます。
2. [キャンバスサイズ]パネルで、[幅]と[高さ]値に 2 ピクセル足します。
3. アンカーをセンターに設定します。
4. [OK]をクリックします。

チェックマークを決めるには、平板化した複製イメージに戻り、それをターゲット解像度に拡大縮小します。

参照用として使用するため、拡大縮小した複製イメージを作業ファイルの新規レイヤーにコピーします。イメージのエッジの 1 ピクセル外側で新しいチェックマークを描画するための新しいレイヤーを作成します。チェックマークは不透明度 100%の黒で、透明度があってははいけません。またチェックマーク以外の領域は 100%透明です。このようにしないとシステムがこのイメージを nine-patch イメージと正しく解釈しません。

拡大縮小した複製イメージを参考に、新しいチェックマークを、参照レイヤーに整列させた新しいレイヤーに描きます。チェックマークのピクセルは全部拾い上げるよう注意します。参照レイヤーで部分的に不透明なピクセルも、新しいレイヤーでは完全な不透明になります。

ストロークとドロップシャドウはイメージの拡大縮小後調整する

ほとんどの場合、レイヤースタイルの拡大縮小は適切ですが(グラデーションオーバーレイや光彩(内側)など)、拡大縮小するイメージのストロークとドロップシャドウは特に hdpi に拡大するとき、保存する前に 1px にリセットする必要があります。

nine-patch イメージは適切なファイル名の接尾辞を使って保存する

アセットがチェックマーク付きの nine-patch アセットの場合には、必ずファイル名に.9.png 接尾辞をつけた PNG 形式で保存します。この接尾辞がついていないと、システムはそのイメージを nine-patch アセットと認識しないので、意図したサイズ変更が行われません。

イメージアセットを保存するときには、Photoshop ヘッダを削除する

イメージアセットをできるだけ小さく保存するには、ファイルから Photoshop ヘッダを削除するようにします。Photoshop ヘッダは次の手順で削除します。

1. [ファイル]→[Web およびデバイス用に保存]コマンドを選びます。
2. [Web およびデバイス用に保存]パネルで、[プリセット]ポップアップを“PNG-24”に設定します(自動的

にその下のポップアップメニューにも“PNG-24”が選ばれます)。[透明部分]ボックスを選択します(イメージが透明度を使用している場合)。

3. [保存]をクリックします。

密度ごとに対応するアセットでは必ず同じファイル名を使用する

密度ごとのアイコンファイルにはどれも同じファイル名を使用し、それを保持するリソースディレクトリにだけ密度専用の名前をつける必要があります。こうすることでシステムは、デバイスの画面特性にしたがって適切なリソースを探し当てロードできるようになります。このため、各ディレクトリ内のアセットのセットはどれも同じで、密度固有の接尾辞がファイル名に含まれていないことを確認する必要があります。密度固有のリソースと、システムがどのようにして異なるデバイスの必要性を満たすのかについては、“[複数画面のサポート](#)”節を参照してください。

Android Icon Templates Pack の使用

Android Icon Templates Pack は、本ドキュメントで述べてきた仕様に沿ったアイコンの作成を容易にするデザインやフィルタ、設定のテンプレート集です。アイコンのデザインを始めるにはまず、このテンプレートパックをダウンロードすることをおすすめします。

このテンプレートは Adobe Photoshop と Adobe Illustrator のファイル形式で提供されており、Android プラットフォームの標準アイコンが作成されるときに使用されたレイヤーやデザイン処理が保存されています。テンプレートファイルは Photoshop、Illustrator と互換性のある画像編集プログラムにロードできますが、そのレイヤーやデザイン処理が直接操作できるかどうかはお使いのプログラムによって変わります。

Android Icon Templates Pack は以下のリンクからダウンロードできます。

[Android Icon Templates Packのダウンロード](#) (icon_templates-v2.0.zip)

アイコンの付録

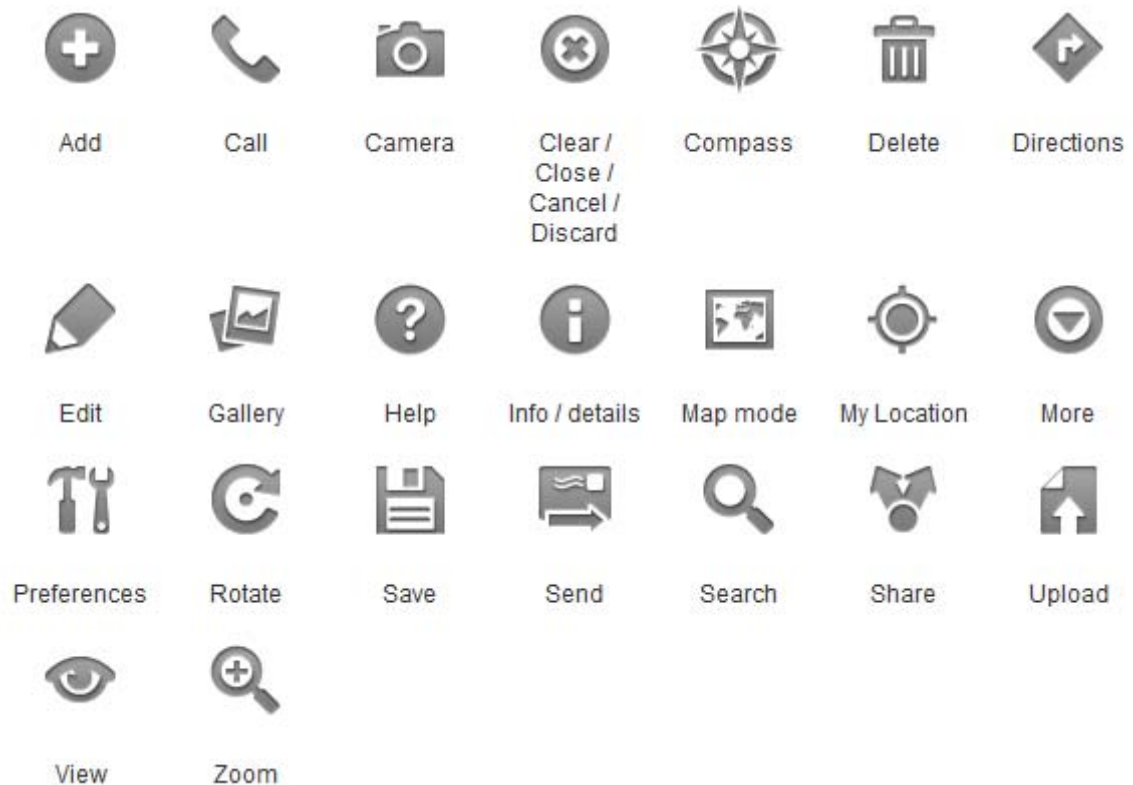
標準的なランチャーアイコン

以下は Android アプリケーションで使用されるランチャーアイコンの例です。これは参照用としてのみ提供されているものなので、みなさんのアプリケーションでは使用しないようにしてください。



標準的なメニューアイコン

以下は Android システムで使用される標準的なメニューアイコンです。これらのリソースはプラットフォームのバージョン間で違うものが表示される可能性があるため、システムのコピーを参照すべきではありません（たとえば SDK の `platforms/android-8/data/res/drawable-mdpi` には多くの画像ファイルがありますが、この場所からは使用しません）。アイコンやほかの内部的な描画リソースを使用したい場合には、それらのアイコンや描画物のローカルコピーをアプリケーションリソースに保持し、そのローカルコピーをアプリケーションコードから参照するようにします。このようにすると、システムに変化があった場合でも、アイコンの外観の制御が維持できます。ただし下記はこれで全部ではありません。



標準的なステータスバーアイコン

以下は Android プラットフォームで使用される標準的なステータスバーアイコンです。これらのリソースはプラットフォームのバージョン間で違うものが表示される可能性があるため、システムのコピーを参照すべきではありません(たとえば SDK の `platforms/android-8/data/res/drawable-mdpi` には多くの画像ファイルがありますが、この場所からは使用しません)。アイコンやほかの内部的な描画リソースを使用したい場合には、それらのアイコンや描画物のローカルコピーをアプリケーションリソースに保持し、そのローカルコピーをアプリケーションコードから参照するようにします。このようにすると、システムに変化があった場合でも、アイコンの外観の制御が維持できます。ただし下記はこれで全部ではありません。



Bluetooth



Email



IM



Voicemail



Warning



Call



Call
forward



Call on
hold



Missed
call