

Autodesk® Scaleform®

Getting Started with CLIK Buttons

本書では、Scaleform® Common Lightweight Interface Kit (CLIK™) のボタン コンポーネントを設定して使用方法について説明しています。

著者: Matthew Doyle
バージョン: 2.0
最終更新日: 2010 年 8 月 19 日

Copyright Notice

Autodesk® Scaleform® 4.4

© 2014 Autodesk, Inc. All rights reserved. Except as otherwise permitted by Autodesk, Inc., this publication, or parts thereof, may not be reproduced in any form, by any method, for any purpose.

Certain materials included in this publication are reprinted with the permission of the copyright holder.

The following are registered trademarks or trademarks of Autodesk, Inc., and/or its subsidiaries and/or affiliates in the USA and other countries: 123D, 3ds Max, Algor, Alias, AliasStudio, ATC, AutoCAD LT, AutoCAD, Autodesk, the Autodesk logo, Autodesk 123D, Autodesk CAM 360, Autodesk Homestyler, Autodesk Inventor, Autodesk MapGuide, Autodesk Streamline, AutoLISP, AutoSketch, AutoSnap, AutoTrack, Backburner, Backdraft, Beast, BIM 360, Burn, Buzzsaw, CADmep, CAiCE, CAMduct, CFdesign, Civil 3D, Cleaner, Combustion, Communication Specification, Configurator 360™, Constructware, Content Explorer, Creative Bridge, Dancing Baby (image), DesignCenter, DesignKids, DesignStudio, Discreet, DWF, DWG, DWG (design/logo), DWG Extreme, DWG TrueConvert, DWG TrueView, DWGX, DXF, Ecotect, ESTmep, Evolver, FABmep, Face Robot, FBX, Fempro, Fire, Flame, Flare, Flint, FMDesktop, ForceEffect, FormIt, Freewheel, Fusion 360, Glue, Green Building Studio, Heidi, Homestyler, HumanIK, i-drop, ImageModeler, Incinerator, Inferno, InfraWorks, InfraWorks 360, Instructables, Instructables (stylized robot design/logo), Inventor, Inventor HSM, Inventor LT, Kynapse, Kynogon, LandXplorer, Lustre, MatchMover, Maya, Maya LT, Mechanical Desktop, MIMI, Mockup 360, Moldflow Plastics Advisers, Moldflow Plastics Insight, Moldflow, Moondust, MotionBuilder, Movimento, MPA (design/logo), MPA, MPI (design/logo), MPX (design/logo), MPX, Mudbox, Navisworks, ObjectARX, ObjectDBX, Opticore, Pipeplus, Pixlr, Pixlr-o-matic, Productstream, Publisher 360, RasterDWG, RealDWG, ReCap, ReCap 360, Remote, Revit LT, Revit, RiverCAD, Robot, Scaleform, Showcase, Showcase 360 ShowMotion, Sim 360, SketchBook, Smoke, Socialcam, Softimage, Sparks, SteeringWheels, Stitcher, Stone, StormNET, TinkerBox, ToolClip, Topobase, Toxik, TrustedDWG, T-Splines, ViewCube, Visual LISP, Visual, VRED, Wire, Wiretap, WiretapCentral, XSI.

All other brand names, product names or trademarks belong to their respective holders.

Disclaimer

THIS PUBLICATION AND THE INFORMATION CONTAINED HEREIN IS MADE AVAILABLE BY AUTODESK, INC. "AS IS." AUTODESK, INC. DISCLAIMS ALL WARRANTIES, EITHER EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE REGARDING THESE MATERIALS.

Autodesk Scaleform の連絡先:

ドキュメント	Getting Started with CLIK Buttons (CLIK ボタンを使ったのクイック ガイド)
住所	Autodesk Scaleform Corporation 6305 Ivy Lane, Suite 310 Greenbelt, MD 20770, USA
ホームページ	www.scaleform.com
電子メール	info@scaleform.com
電話	(301) 446-3200
Fax	(301) 446-3199

目次

1. はじめに.....	1
2. Button.....	2
2.1 作業を始める.....	2
2.2 ボタンのレイヤー	4
2.3 Button のステート	5
2.3.1 Up	6
2.3.2 Over	6
2.3.3 Down	6
2.3.4 Disabled	7
2.4 Button のパラメータ.....	8
2.4.1 Inspectable	8
3. Toggle Button.....	9
3.1 Toggle Button のステート	10
3.1.1 Selected_Up	10
3.1.2 Selected_Over.....	11
3.1.3 Selected_Down	11
3.1.4 Selected_Disabled	12
4. Animated Button.....	13
4.1 Animated Button のトランジション.....	13
4.1.1 Over	15
4.1.2 Down.....	15
4.1.3 Release	15
4.1.4 Out.....	16
4.2 キーボード トランジション	16
4.2.1 フォーカス インジケータ	16
4.2.2 KB_Down	17
4.2.3 KB_Release.....	17
5. Animated Toggle Button	19
5.1 Animated Toggle Button のトランジション	19
5.1.1 Selected_Selecting.....	21
5.1.2 Selected_Release.....	21

5.1.3 Selected_Out	21
5.1.4 Selecting	22
5.2 キーボード トランジション	22
5.2.1 Selected_KB_Down.....	22
5.2.2 Selected_KB_Release	22
5.2.3 Selected_KB_Selecting	23
5.2.4 KB_Selecting.....	23
6. 最後に.....	24

1. はじめに

Scaleform の Common Lightweight Interface Kit (CLIK) を使うと、デベロッパーはゲームの効果的なフロントエンド メニュー インターフェイスをすばやくプロトタイプ化して、そのプロトタイプを短時間で効率的に完成することができます。また、ユーザーは自身のデザインを簡単に繰り返し利用することもできます。このキットにはボタン、スライダ、プログレス バー、ドロップダウン メニュー、テキスト領域、スクロール リストを含む 15 以上のウィジェットやコンポーネントが含まれています。本書では、CLIK コンポーネントの核の一つであるボタン コンポーネントに焦点を合わせます。主なボタンの種類は Button、ToggleButton、AnimatedButton、AnimatedToggleButton の 4 つです。後の 3 つは標準の Button コンポーネントを特殊加工したバリエーションです。このコンポーネントとそのバリエーションをいつどのように使用するかを学ぶことが、その他の CLIK コンポーネントを理解するための最初のステップとなります。

本書の前に「[Getting Started with CLIK](#)」を読むことをお勧めします。本書で説明している概念の多くがそのマニュアルで紹介されているからです。

ご注意： このチュートリアルでは、私どもが現時点で最善と思われる方法をお薦めしております。しかしながら、Flash でのオーサリングでは同じ結果を得るためにいくつかの手法が存在するのも事実です。そのため、開発者がより良い方法を見出すことも充分あり得ることです。

2. Button

これは、スライダ、プログレス バー、スクロール バーを含め、ほとんどの CLIK コンポーネントの基礎を形成するコア コンポーネント、またはウィジェットです。これ以外のコンポーネントの多くはこの基本的な Button コンポーネントの機能を継承し、ほとんどのユーザー インターフェイス (UI) の作業は、この Button コンポーネントを使って、または拡張して行われます。このコア Button コンポーネントはアニメーション化されていない、必要最小限のボタンです。このコンポーネントの中心的な動作は、mouseover イベント、マウスのクリック、キーボードの押下、ゲームのコントローラの入力に応答するというものです。

このコア Button コンポーネントの用途として考えられるのは以下のような場合です：

- メニュー ボタン
 - OK / キャンセル / 適用
 - 画面の選択
- ゲーム内のコンソール コントロール ボタン
- ミニゲーム ボタン

この Button コンポーネントとこれ以降の章で説明しているそれ以外のバリエーションは、Button.as ファイルの同じ ActionScript™ 2 (AS) クラスを使用しています。このファイルには Button を動作させるために必要なすべてのコードが含まれています。Button コンポーネントを使用するためにこのファイルを見直す必要はありませんが、そうすることをお勧めします。

2.1 作業を始める

1. まず始めに Adobe® Flash®で以下の CLIK_Components.flc ファイルを開きます：

Windows ライセンス済み SDK は以下：

C:/Program Files/Scaleform/GFx SDK

4.4/Resources/AS2/CLIK/components/CLIK_Components.flc

Mac ライセンス済み SDK は以下：

Scaleform 4.4 SDK がインストールされたディレクトリの

scaleform_gfx_4.4_macos/Resources/AS2/CLIK/components

2. Ctrl キーを押したまま [ライブラリ] パネルで以下のコンポーネントをマウスの左ボタンでクリックします：
 - a. *Button*
 - b. *ToggleButton*
 - c. *AnimatedButton*
 - d. *AnimatedToggleButton*

3. Ctrl キーを離し、*Button* を右クリックします。4 つのボタン バリエーションがすべて強調表示されているはずです。
4. [コピー]を選択します。
5. Ctrl+N キーを押す、または Flash メニューの [ファイル]–[新規] を選択して、新規の Adobe® Flash®ファイル (AS2.0) を作成します。
6. この新規 Flash ファイルの [ライブラリ] パネル内の空白のエリアで右クリックします。
7. [ペースト] を選択して、[ライブラリ] パネルに 4 つのボタン バリエーションをペーストします。
8. [ライブラリ] パネルで *Button* をマウスの左ボタンでクリックして選択します。
9. [ライブラリ] パネルから *Button* コンポーネントをクリックして、ステージにドラッグします。
10. 残りの 3 つの *ToggleButton*、*AnimatedButton*、*AnimatedToggleButton* についてステップ 8–9 を繰り返します。

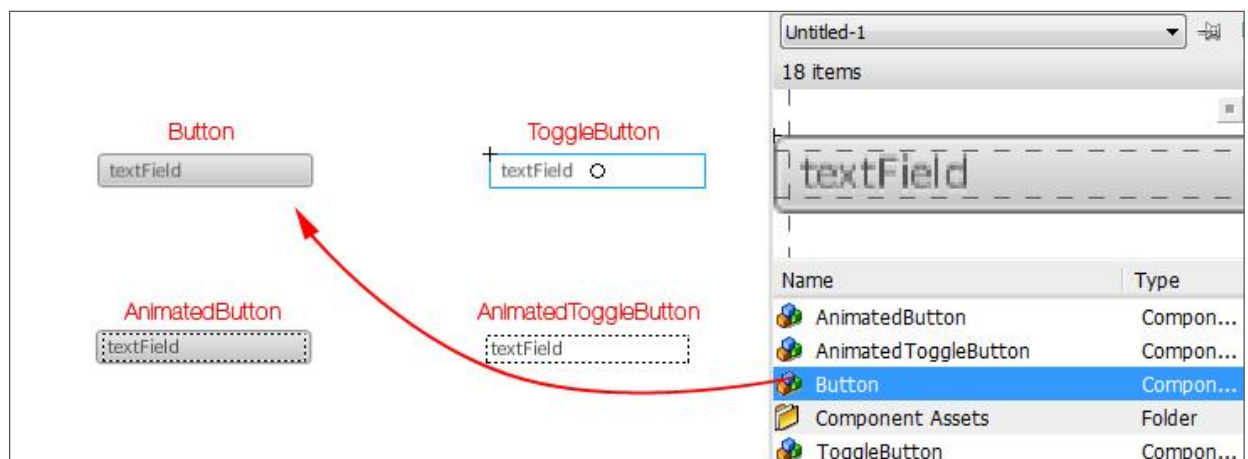


図 1 : 4 つのボタン バリエーションをクリックしてステージにドラッグする

2.2 ボタンのレイヤー

基本の Button コンポーネントと他の 3 つのバリエーション ToggleButton、AnimatedButton、AnimatedToggleButton はそれぞれ 5 つのレイヤーで構成されています。Button コンポーネントのレイヤーを確認するには、ステージ上の Button をダブルクリックします。バリエーションの 1 つではなく、必ず標準のボタンをダブルクリックして、これ以降の章に従ってください。

レイヤーは以下のとおりです：

- *actions* : AS ステートメントの配置に使用されます。
- *labels* : ボタンの各ステートを表す視覚的なヒントとして使用されます (第 2.3 章を参照してください)。
- *textField* : 実行時にボタンが表示するテキストに使用されます。
- *button* : さまざまなステートのボタンのグラフィック イメージの配置に使用されます。

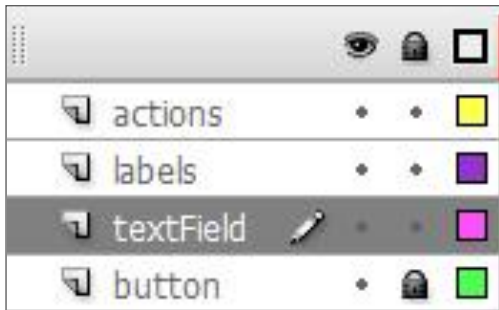


図 2 : Button のレイヤー

このようなレイヤーは現状のまま必要なわけではありません。レイヤーは再配置、名前の変更、結合、または他のレイヤーを選択して完全に削除される場合があります。ボタンは UI デザイナーが選択した任意のレイヤーで構成できますが、構成の起点として、上記のレイヤーを提供しています。ただし、*actions* レイヤーの AS や *labels* レイヤーのラベルが付いたキーフレームが存在しない場合や適切な位置にない場合は、ボタンは正しく動作しないことを覚えておってください。同様に、各ステートのグラフィック表現は、タイムライン上の *labels* のキーフレームに対して適切な位置に含まれていなければなりません。したがって、UI デザイナーにはこの Button コンポーネントの動作方法をよく理解するまで、現状のまま使用することをお勧めします。

2.3 Button のステート

各 CLIK コンポーネントはステートと呼ばれるもので構成されています。アニメーション ボタンの場合はトランジションで構成されています (トランジションについては、アニメーション ボタンの章で説明します)。ステートはユーザーの操作やコードを使って設定できます。このようなステートには視覚的な表現が含まれ、そのコンポーネントの他のステートと異なる場合と同じ場合があります。今回の標準のボタンには 4 つのステートしかありません。各ステートの位置はボタンのタイムライン上の Labels レイヤーのキーフレームで表され、そのステートに対応するグラフィックは、同じ場所の States と Border レイヤーに存在します。

主なボタンのステートは以下のとおりです:

- *up*
- *over*
- *down*
- *disabled*

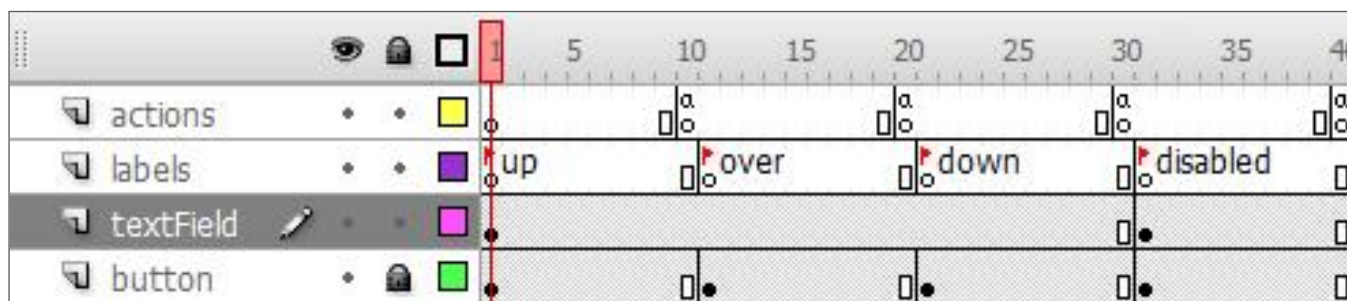


図 3 : Labels レイヤー上にはっきりとラベル付けされたステートを持つ Button のタイムライン



図 4 : 並べて表示した、Button ステートのグラフィック ワイヤーフレーム表現

2.3.1 Up

up ステートはボタンの標準の「静止中」の状態です。このボタンを使ったユーザーの操作がない状態で、ボタンがデフォルトで表示されるステートです。

表示：ボタンのデフォルトのステートで、何の操作も行われなかった場合

用途：標準のステートでボタンを表現します。

2.3.2 Over

over ステートはボタンが強調表示された状態です。ユーザーがマウスのカーソルをこのボタンの上に移動したときに表示されます。ボタンからカーソルを離すと、ボタンは *up* ステートに戻ります。

表示：ユーザーがボタンの上にマウス カーソルを移動したとき、あるいはマウスの押下、キーボードの矢印キー、ゲーム コントローラの方向パッド/アナログ スティックで、ボタンにフォーカス適用されたとき。フォーカスについては、第 2.3.3.1 章を参照してください。

用途：マウスが上にある、またはフォーカスが適用されているボタンを表現します。

2.3.3 Down

down ステートは、マウスの左ボタンのクリック、キーボードの Enter キーの押下、またはゲーム コントローラの該当するボタンの押下によって、ユーザーがこのボタンを押したときに表示されます。ボタン コンポーネントは、マウス ボタン、Enter キー、ゲーム コントローラのボタンが押されている限り、*down* ステートを維持します。解放するとボタンは *over* ステートに戻ります。

デフォルトでは、そのボタン上でマウス ボタンが押されたときに、ボタンにフォーカスが適用されます。フォーカスが適用された場合、マウスがボタン上になかったとしても、ボタンは「静止中」に標準の *up* ステートではなく、*over* ステートを維持します。*over* ステートは、マウスのクリック、キーボードの矢印キー、またはゲーム コントローラで別のコンポーネントを選択することでフォーカスがどこかに移動するまで、ボタンのデフォルトのステートになります。ユーザーがマウス ボタンを押したままボタンからマウス カーソルを離した場合、ボタンは *over* ステートに戻りフォーカスを維持します。フォーカスはマウスを押したときに移動します。

表示：ユーザーがボタン コンポーネント上でマウスの左ボタンを押したとき、Enter キーを押したとき、あるいはゲーム コントローラの該当するボタンを押したとき。

用途：クリックされているボタンを表現します。

2.3.3.1 フォーカス

フォーカスが適用されたコンポーネントは、すべてのキーボード イベントやコントローラ イベントのターゲットになります。つまり、ボタンにフォーカスが適用されているとき、Enter キーを押すとそのボタンのみに影響し、それ以外のコンポーネントは関係ないということです。フォーカスはマウスの左ボタンの押下、あるいはキーボードの矢印キーの使用によって移動して、近隣のコンポーネントに移すことができます。ゲーム コントローラの方向パッドおよびアナログ スティック、またはそのいずれかがキーボードの矢印キーにマップされている場合、それを使ってフォーカスを移動することもできます。また、フォーカスは AS コードで移動することもできます。

2.3.4 Disabled

disabled ステートはオフにされたボタンを表現します。オフにされたボタンは押すこともフォーカスを適用することもできず、*mouseover* イベントにも反応しません。ボタンは [コンポーネントインスペクタ] の [パラメータ] タブを使って、または以下のコードを使って *ActionScript* でオフにすることができます:

```
buttonInstanceName.disabled = true;
```

表示 : *ActionScript* または [パラメータ] タブで設定

用途 : 現在オフになっていて、使用できないボタンの表現に使用されます。

2.4 Button のパラメータ

Button コンポーネントは [パラメータ] タブや [コンポーネント インспекタ] で確認できるパラメータのリストを備えており、これらのパラメータは UI デザイナーがコードを変更せずに、基本的なボタンのプロパティを設定できるように設計されています。このようなパラメータは以下のとおりです：

- *disabled* : ボタンをオフにする場合は true に設定します。
- *disableFocus* : ボタンにフォーカスが適用されないようにする場合は true に設定します。
- *disableConstraints* : ラベルのカウンタ スケーリングをオフにする場合は true に設定します。
- *label* : 実行時にボタンが表示するテキスト ラベルをここに入力します。
- *toggle* : ボタンのオン / オフの切り替えを許可する場合は true に設定します。
- *visible* : 実行時にボタンを非表示にする場合は false に設定します。



図 5 : CS3 ([パラメータ] タブ) の Button コンポーネントのパラメータ

2.4.1 Inspectable

パラメータは Button.as ファイルを変更することでリストに追加できます。これはメンバ変数、またはブール値 (true または false) など変更可能なプロパティを備えた getter/setter 関数の上に inspectable ステートメントを追加することで行います。たとえば、以下の inspectable ステートメントをボタンのラベルの設定を担当するコード ブロックの上に直接追加して、UI デザイナーがコードではなく Flash 内の [パラメータ] タブでボタン ラベルを設定できるようにしました：

```
[Inspectable(name="label", defaultValue="")] ]
```

Button.as で toggle 変数の真上にある以下の inspectable ステートメントは、UI デザイナーが [パラメータ] タブで true/false を切り替えられるようにします。

```
[Inspectable(defaultValue="false")] ]
```

3. Toggle Button

Toggle Button は標準ボタンにチェックボックス スタイルの機能を追加します。これにより、選択された状態と選択が解除された状態を持つボタンを作成することができます。Toggle Button は、ユーザーがボタンをクリックするたびに、選択された状態から選択解除された状態—オンからオフ (またはその逆でオフからオン) へ「切り替え」られます。AS で切り替えることもできます。

Toggle Button コンポーネントの用途として考えられるのは以下のような場合です：

- メニュー ボタン
 - ブルーム設定の視覚効果をオン / オフする
 - サウンドをミュート / ミュート解除する
 - 高品質ライティングをオン / オフする
 - 自動保存をオン / オフする
- オン / オフ ステートを必要とするゲーム内のコンソールのコントロール
- オン / オフ ステートを必要とするミニゲーム ボタン

Toggle Button は標準ボタンと同じ AS クラスを使用します。唯一の違いは *toggle* プロパティが “true” に設定されていることです。AS クラスは Toggle Button を本質的にサポートするので、ユーザーが提供された Toggle Button コンポーネントの上で標準の Button コンポーネントを使用することを選択した場合、タイムライン上にステート フレームを追加するかどうかはユーザー次第です。ただし、ユーザーにはあらかじめ構成された Toggle Button を使って時間を節約することをお勧めします。第 4 章と第 5 章で説明する Animated Button と Animated Toggle Button の場合も同じです。

Toggle ボタンを適切に動作させるために、CS4 の [コンポーネントインスペクタ] の [パラメータ] タブ (CS3 の場合は [パラメータ] タブ) の toggle フィールドが “true” に設定されていることを確認します。任意の CLIK Button コンポーネントの toggle プロパティを設定することで、そのボタンを Toggle Button のように強制的に動作させることができます。

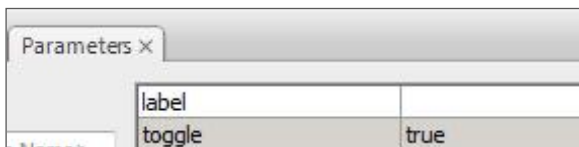


図 6 : 'true' に設定された toggle パラメータ

3.1 Toggle Button のステート

標準の Button ステート (*up*、*over*、*down*、*disabled*) に加えて、Toggle Button には以下の特殊なステートも含まれています：

- *selected_up*
- *selected_over*
- *selected_down*
- *selected_disabled*

この 4 つの特殊なステートを使って、「オン」または選択された状態に切り替えられた Toggle Button を表現し、一方でオリジナルの Button ステート、*up*、*over*、*down*、*disabled* を Toggle Button で使用して、「オフ」または選択解除の状態に切り替えられたボタンを表現します。

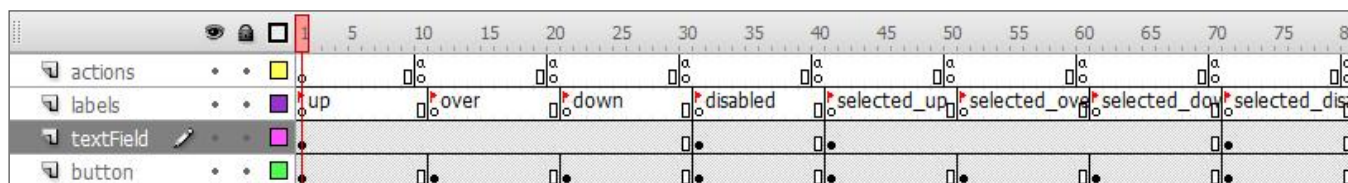


図 7：はっきりとラベル付けされたステートを持つ Toggle Button のタイムライン



図 8：並べて表示した新規の Toggle Button ステートのグラフィック ワイヤフレーム表現

3.1.1 Selected_Up

selected_up は、ユーザーが「オン」または「選択された」状態に切り替えた、フォーカスが適用されていない Toggle Button の標準の「静止中」のステートです。このステートは通常、縁が太くなる、あるいは色が変わるなど、ボタンが「オン」であることを示すある種のグラフィック インジケータを含んでいます。

表示：ユーザーが現在「オフ」または「選択解除された」Toggle Button をクリックしたとき、ボタンは「選択された」状態、または「オン」になり、*selected_over* ステートに進みます。また、別のコンポーネントをクリックして、あるいは矢印キーやゲーム コントローラを使ってユーザーがこのボタンからフォーカスを移動すると、ボタンは *selected_up* ステートに変わります。

用途：「選択された」または「オン」に設定された、フォーカスが適用されていない Toggle Button を表現します。

3.1.2 Selected_Over

selected_over はユーザーがマウス カーソルを切り替えたボタンに移動したとき、または矢印キーやゲーム コントローラを使って、「オン」または「選択された」状態のフォーカスが適用されていない Toggle Button に移動したときに表示されます。マウスを移動する、または矢印キーやゲーム コントローラを使ってボタンから離れることで、ボタンは *selected_up* ステートに戻ります。

表示：「オン」または「選択された」状態に設定された Toggle Button 上にユーザーがマウス カーソルを移動したとき、または矢印キーやゲーム コントローラを使ってボタンにフォーカスを適用したとき

用途：「オン」または「選択された」状態に設定された、フォーカスが適用されている Toggle Button を表現します。

3.1.3 Selected_Down

selected_down はユーザーがマウスの左ボタンのクリック、Enter キー、またはゲーム コントローラ上の該当するボタンを使って、「オン」または「選択された」状態に設定された Toggle Button を押したときに表示されます。このボタンは、マウス ボタン、Enter キー、ゲーム コントローラのボタンが押されている限り、*selected_down* ステートを維持します。Toggle Button を押すという行為はその選択された状態を切り替えるので、ボタンは *over* ステートに戻ります。解放されると、このステートは「選択解除」または「オフ」を表現します。*down* ステートの場合は逆になります。この場合、ボタンは *selected_over* ステートに移動します。

標準の Button と同様に、選択された Toggle Button を押すとフォーカスがそのボタンに移動します。ボタンのデフォルト ステートは、Toggle Button が「オン」の場合は *selected_over* ステートに設定され、「オフ」の場合は *over* ステートに設定されます。ユーザーがマウス ボタンを押したままボタン コンポーネントからマウス カーソルを離して、その後マウス ボタンを解放した場合、このボタン コンポーネントは「オン」の場合は *selected_over* に戻り、「オフ」の場合は *over* ステートに戻ります。

表示：マウス カーソルが Toggle Button 上にある間にユーザーがマウスの左ボタンを押したとき、または Toggle Button ボタンにフォーカスが適用されている間に、Enter キー、あるいはゲーム コントローラ ボタンを押したとき。

用途：「オン」に設定された、フォーカスが適用されている Toggle Button がクリックされていることを表現します。

3.1.4 Selected_Disabled

selected_disabled ステートは、標準の Button の *disabled* ステートと同様に、「オン」に設定されていた Toggle Button のステートがオフにされたことを表現します。オフになった Toggle Button は押すこともフォーカスを適用することもできず、mouseover イベント、キーボード イベント、ゲーム コントローラ イベントにも反応しません。Toggle Button は [パラメータ] タブ、または以下のコードを使って AS でオフにすることができます：

```
buttonInstanceName.disabled = true;
```

表示：ActionScript または [パラメータ] タブで設定

用途：選択された、または「オン」に設定された Toggle Button がオフにされ、使用できないことを表現します。

4. Animated Button

Animated Button は基本的に標準の Button コンポーネントと同じですが、ステートからステート (up から over、over から down など) へのアニメーション トランジションが含まれる場合があります。ボタンのさまざまなステートのトランジションとして、タイムライン ベースのアニメーションが欲しい場合は常に Animated Button を使用します。アニメーション効果なしで各ボタン ステートに簡単に切り替わる標準の Button とは対照的です。

Animated Button コンポーネントの用途として考えられるのは以下のような場合です:

- アニメーション トランジションを持つメニュー ボタン
 - OK / キャンセル / 適用
 - 画面の選択
- アニメーション トランジションを持つゲーム内のコンソール コントロール ボタン
- アニメーション トランジションを持つミニゲーム ボタン

コアのボタンとアニメーション ボタンの実際の違いは視覚的なものだけです。ボタンのステートが変わるにつれて、ボタン上のアニメーション効果をデザイン的に求める場合、Animated Button が最適です。そうでない場合は、コアの Button が最善の選択肢です。このボタンにはトランジション キーフレームとアニメーション トゥイーン形式のデータの量が少ないからです。

Animated Button の作成にコードを追加する必要はありません。CLIK Button クラスは、本書で説明しているさまざまなバリエーションにすべての機能を提供しています。

4.1 Animated Button のトランジション

Animated Button は別の用語を使って、関連するステートのキーフレームを記述します。ステート自身を参照するのではなく、各キーフレームは 2 つのステート間のアニメーション トランジションを表現します。*up*、*down*、*over* はステートとしての意味を今でも備えています。アニメーション ボタンの場合、その名前を類似したトランジションのためにリサイクルしました。

トランジションによってボタンが移動するステートとして、*up*、*down*、*over*、*selected_up*、*selected_down*、*selected_over* を考えます。これは、信頼性の高いアニメーション ボタン コンポーネントの、可能性のある多様なトランジションの複雑性の理解に役立ちます。

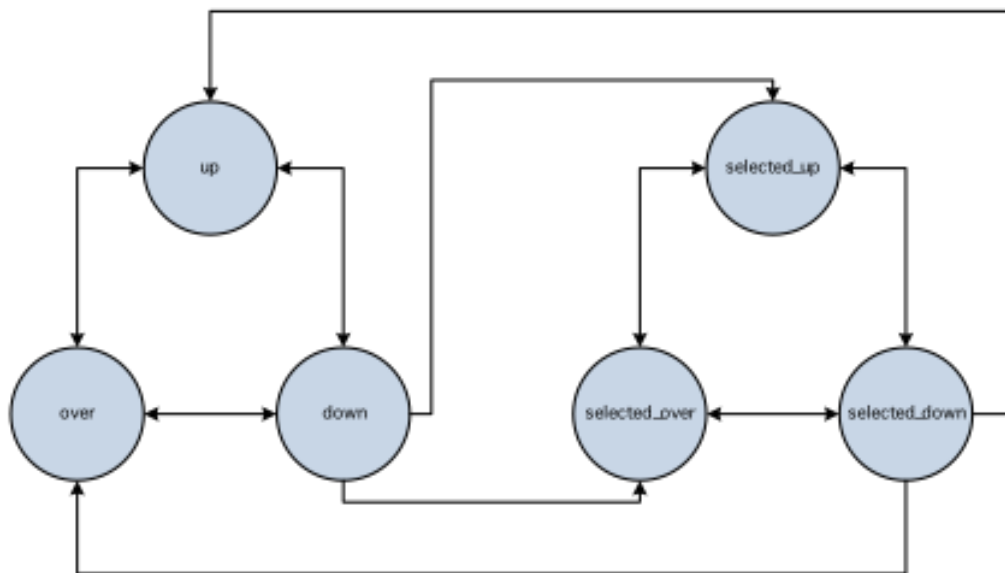


図 9 : Animated Button ステートのトランジション

ステート間でクラシックなアニメーション トゥイーンを持つには、最後のキーフレームを末端に、または最後のステートの表示に設定します。これはアニメーションの最終的な静止地点で、アニメーションが終わったときにボタンが表示するステートを表現する必要があります。次に、最初のフレームから最後のフレームまでをブレンドするトゥイーンを作成します。たとえば、*over* トランジションの場合、アニメーションの最初のキーフレームはボタンの *up* ステートに似たグラフィックになり、最後のキーフレームはボタンの実際の *over* ステートを表すグラフィックになります。

up、*over*、*down*、*disabled* のコアの Button トランジションに加えて、Animated Button には以下の特異なトランジションが含まれています：

- *release*
- *out*
- *kd_down*
- *kb_release*

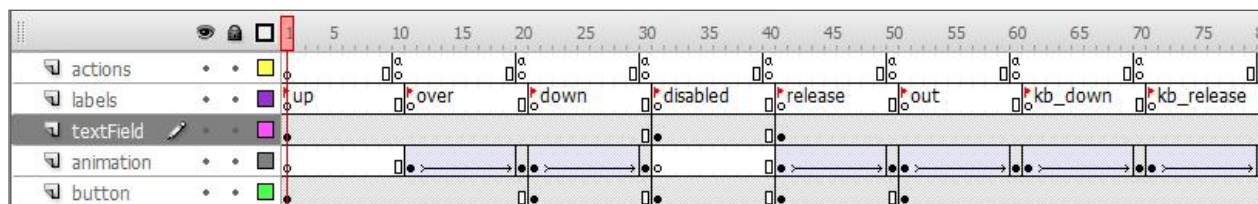


図 10 : はっきりとラベル付けされたトランジションを持つ Animated Button のタイムライン



図 11：並べて表示した新規の Animated Button ステートのグラフィック ワイヤーフレーム表現

Animated Button の *over* と *down* ステートは、コアの Button バージョンの上に新規の機能を備えています。静的なイメージ ステートではなく、これらのステートは最後に静的なステートの表示になるアニメーション トランジションを使用します。

4.1.1 Over

over の動作は Animated Button ではわずかに異なります。これを使って、ボタンの *up* ステートから *over* ステートへのアニメーション トランジションを表示します。

表示：ユーザーがボタンの上にマウス カーソルを移動したとき、または矢印キーやゲーム コントローラを使ってボタンにフォーカスを適用したとき

用途：マウス カーソルが上にある、またはフォーカスが適用されているボタンを表現します。

4.1.2 Down

down は、ボタンの *over* ステートから *down* ステートへのアニメーション トランジションを表示するために使用されます。

表示：ユーザーがボタン上でマウスの左ボタンを押したとき、ボタンにフォーカスが適用されている間に、Enter キーを押したとき、あるいはゲーム コントローラの該当するボタンを押したとき。

用途：クリックされたボタンを表現します。

4.1.3 Release

release は Animated Button と Animated Toggle Button に特有のステートです。これを使って、ボタンの *down* ステート (押下された状態) から *over* ステートへのアニメーション トランジションを表示します。このトランジションの最後のフレーム (*over* ステート) を使って、フォーカスされたステートを表示します。

表示：ユーザーがマウスの左ボタン、Enter キー、またはゲーム コントローラのボタンをクリックして解放したとき。また、ユーザーがボタン上でマウスの左ボタンを押して、マウス カーソルをそのボタンから離れたときも発生します。

用途：マウスの左ボタン、Enter キー、コントローラの同等のボタンが解放された、フォーカスが適用されている Animated Button を表現します。

4.1.4 Out

out はボタンの *over* ステートから *up* ステートへのアニメーション トランジションの表示に使用されます。

表示：ユーザーがマウス カーソルをボタンから離れたとき。

用途：ボタンの境界からマウス カーソルが離れた場合の、フォーカスが適用されていないボタンを表現します。

4.2 キーボード トランジション

キーボード トランジションを使って、マウスではなくキーボードやゲーム コントローラが使用されたとき、さらに別にフォーカス インジケータ ムービー クリップが使用されたとき (フォーカス インジケータの詳細については、第 4.2.1 章を参照してください)、あるステートから他のステートへの特殊なアニメーション トランジションを表示します。別にフォーカス インジケータが使用される場合に、キーボード イベントに必要な追加トランジションには *kb_down* と *kb_release* の 2 つがあり、それぞれ第 4.2.2 章と 4.2.3 章で説明しています。

4.2.1 フォーカス インジケータ

デフォルトでは、Button コンポーネントは *release* トランジションの最後のフレームを使って、フォーカスが適用されたステートを表示します。ただし場合によっては、フォーカスが適用されたステートのインジケータとして、別のグラフィックを持つことが最良の方法となることがあります。特にキーボードとゲーム コントローラの入力、あるいはそのいずれかが使用される場合はそうです。CLIK Button はボタンのムービー クリップの内部に、フォーカス インジケータ ムービー クリップが含まれている場合は、本質的にこの機能をサポートします。このフォーカス インジケータ ムービー クリップ インスタンスは、'focusIndicator' という名前にする必要があり、順に'hide'と'show'というラベルを付けた 2 つのキーフレームで構成しなければなりません。show フレームは縁を発光させるなど、フォーカスを表現するグラフィックを持つ必要があります。hide フレームはフォーカスが適用されていないボタンを表現する必要があり、通常はまったくグラフィックを持たないことになります。このフォーカス インジケータ ムービー クリップを作成するには、以下の手順に従います：

1. ボタンのタイムライン内で新規のレイヤーを作成します。

2. 構成上、そのレイヤーに‘FocusIndicator’という名前を付けます。
3. レイヤーの合計フレーム数が、ボタンのタイムラインの最後まで拡張していることを確認します。
4. ボタンのキーボード フォーカスを表現する新規のムービー クリップを、*FocusIndicator* レイヤーに作成します。
 - a. たとえば、ボタン グラフィックの周囲に、空白の縁が赤い四角形を描画します。
 - b. ステージ上でその四角形を右クリックして [シンボルに変換] を選択します。
 - c. そのムービー クリップに名前を付けて、[OK] をクリックします。
5. [プロパティ] パネルでこのムービー クリップのインスタンス名を‘focusIndicator’ と設定します。
6. *focusIndicator* ムービー クリップをダブルクリックしてそのタイムラインを表示します。
7. タイムラインの内部で、レイヤー1 タイムライン上のフレーム 2 で 2 つ目のキーフレームを追加します。
8. キーフレーム 1 を選択して、[プロパティ] パネルで‘hide’というラベルを付けます。
9. [アクション] パネル (F9 キー) を開き、フレーム 1 に `stop();` というコードを追加します。
10. キーフレーム 2 を選択して、‘show’というラベルを付けます。
11. [アクション] パネルでフレーム 2 に `stop();` というコードを追加します。
12. 赤い四角形が *show* フレームで表示されていて、*hide* フレームでは削除されていることを確認します。

4.2.2 KB_Down

キーボード イベントは、ボタンにフォーカスが適用されている場合に限って、ボタンに渡されます。フォーカス インジケータが使用されない場合、ボタンのフォーカスされた状態は、*over* 状態 (*release* トランジションの最後のフレーム) に戻ります。フォーカス インジケータが使用されている場合、*up* から *down* へのトランジションが必要です。フォーカスが適用されている場合でも、ボタンは *up* 状態にあるからです。

kb_down は、ボタンの *up* 状態から *down* 状態へのアニメーション トランジションの表示に使用されます。

表示：フォーカス インジケータ ムービー クリップを持ったフォーカスが適用されているボタン上で、ユーザーが Enter キー、あるいはゲーム コントローラの該当するボタンを押したとき。

用途：別にフォーカス インジケータを持つ押下されたボタンを表現します。

4.2.3 KB_Release

kb_release はボタンの *down* 状態から *up* 状態へのアニメーション トランジションの表示に使用されます。

表示：フォーカス インジケータ ムービー クリップを持ったフォーカスが適用されているボタン上で、ユーザーが Enter キー、あるいはゲーム コントローラの該当するボタンを解放したとき。また、フォーカスが適用されているボタン上で、ユーザーが Enter キー、あるいはゲーム コントローラの該当するボタンを押したとき。

フォーカス インジケータが使用されていて、ユーザーがマウスをドラッグして離した場合、あるいは矢印キーやゲーム コントローラを使って、Enter キー、ゲーム コントローラ ボタン、マウスの左ボタンを解放する前に、別のボタンに移動したときも表示される場合があります。

用途：別にフォーカス インジケータを持ったフォーカスが適用されたボタンが解放されたときを表現します。

5. Animated Toggle Button

Animated Toggle Button は Toggle Button と Animated Button を組み合わせたハイブリッド コンポーネントです。Toggle Button のさまざまなステートのトランジションとして、タイムライン ベースのアニメーションが欲しい場合は常に、Animated Toggle Button を使用します。アニメーション 効果なしで各ボタンの ステートに簡単に切り替わる標準の Toggle Button とは対照的です。

このボタン タイプは 4 つのタイプの中で最も包括的なものです。このボタンはこれまでの 3 つのタイプ の機能をすべて含み、さらにアニメーション トグル ボタンに特化したトランジションを備えている からです。前章の Animated Button のように、トランジションとして *selected_up*、*selected_down*、*selected_over* をリサイクルしました。これらは今でもステートとしての意味を備えて います。

Animated Toggle Button コンポーネントの用途として考えられるのは以下のような場合です：

- アニメーション トランジション付きの、「オン」と「オフ」を切り替えられるメニュー ボタ ン
 - ブルーム設定の視覚効果をオン / オフする
 - サウンドをミュート / ミュート解除する
 - 高品質ライティングをオン / オフする
 - 自動保存をオン / オフにする
- オン / オフ ステートを必要とするゲーム内のアニメーションを使ったコンソールのコントロー ル
- オン / オフ ステートを必要とするミニゲームのアニメーション ボタン

5.1 Animated Toggle Button のトランジション

Animated Toggle Button は Animated Button と同じ用語を使用して、ボタンに関連するキーフレ ームを記述します。ステートを参照するのではなく、各キーフレームは 2 つのステート間のアニメー ション トランジションを表現します。

Animated Toggle Button には、Button、Toggle Button、Animated Button のこれまでのすべての ステートとトランジションが含まれています。また、以下の 8 種類の専用トランジションも含まれて います：

- *selected_selecting*
- *selected_release*
- *selected_out*
- *selecting*
- *selected_kb_down*
- *selected_kb_release*
- *selected_kb_selecting*
- *kb_selecting*

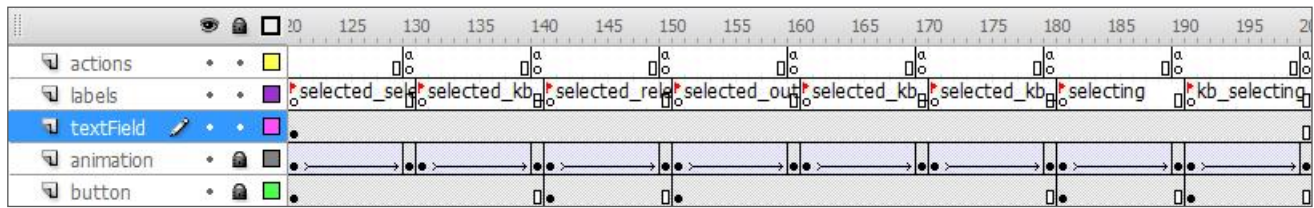


図 12 : はっきりラベル付けされた新規トランジションを持つ Animated Toggle Button のタイムライン

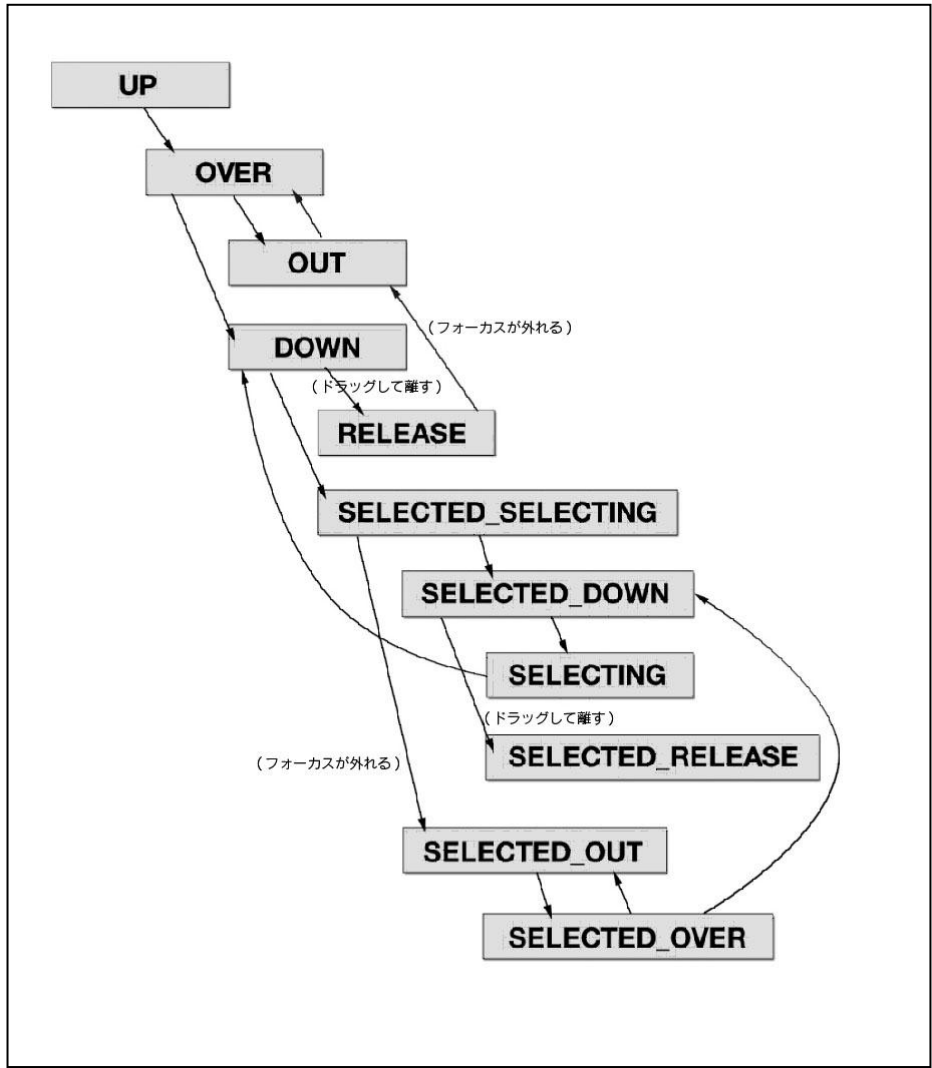


図 13 : Animated Toggle Button のトランジション (キーボード トランジションは含まない)

5.1.1 Selected_Selecting

selected_selecting は、ボタンの *down* ステートから「選択された」または「オン」のフォーカスが適用されたステート (*selected_over*) へのアニメーション トランジションの表示に使用されます。これは「選択解除」のステートから「選択された」ステートへの切り替えトランジションです。

表示：「オン」や「選択された状態」に現在切り替えられていない Animated Toggle Button を、ユーザーがクリックして解放したとき。ボタンは次に「オン」または「選択された状態」に切り替わります。

用途：「選択された状態」に切り替えられ、フォーカスを持っていてもフォーカス インジケータは持たないボタンの表示に使用されます。

5.1.2 Selected_Release

selected_release はボタンの *selected_down* ステート (押下されて現在「選択された」ステートにあるボタン) から、「選択され」、フォーカスが適用された *selected_over* ステートへのアニメーション トランジションの表示に使用されます。

表示：ユーザーが選択されたボタンをクリックして、ボタンからマウス カーソルをドラッグして離れたとき。これは「選択された状態」から「選択解除した状態」への切り替えトランジションを効率的にキャンセルします。

用途：「選択された状態」に切り替えられフォーカスを持っていて、クリック操作がキャンセルされたボタンの表示に使用されます。

5.1.3 Selected_Out

selected_out はボタンの *selected_down* ステート (「選択され」、フォーカスが適用され、さらに押下されたボタン) から、「選択され」かつフォーカスが適用されていない *selected_up* ステートへのアニメーション トランジションの表示に使用されます。

表示：ユーザーがマウスの左ボタンで別のコンポーネントをクリックしたとき、または矢印キーやゲーム コントローラを使って別のコンポーネントを選択したとき。フォーカスが適用され「選択された」ボタンはフォーカスを失うことになります。

用途：フォーカスが適用されていない、ただし、まだフォーカスを失ったばかりの「選択された」状態であるボタンの表現に使用されます。

5.1.4 Selecting

selecting は、ボタンの *selected_down* ステートから *over* ステートへのアニメーション トランジションの表示に使用されます。これは「選択された」ステートから「選択解除」のステートへの切り替え トランジションです。

表示：ユーザーが現在「選択されている」Animated Toggle Button をクリックして解放したとき。次にこのボタンは「選択解除」になります。

用途：「選択解除」されていても、切り替えられたばかりのいまだにフォーカスが適用されているボタンの表現に使用されます。

5.2 キーボード トランジション

Animated Toggle Button は、本書の Animated Button のキーボード トランジションの章で説明した同じセットアップ手順を必要とします。

5.2.1 Selected_KB_Down

selected_kb_down トランジションは、ボタンの *selected_up* ステートから *selected_down* ステートへのアニメーション トランジションの表示に使用されます。

表示：別にフォーカス インジケータ ムービー クリップを持つ、「選択され」、フォーカスが適用された Animated Toggle Button が、キーボードの Enter キー、あるいはゲーム コントローラの該当するボタンで押下されたとき。

用途：Enter キー、またはゲーム コントローラ ボタンで押下された、「選択された」Animated Toggle Button を表現します。

5.2.2 Selected_KB_Release

selected_kb_release は、選択され、フォーカスが適用されたボタンの *selected_down* ステートから *selected_up* ステートへのアニメーション トランジションの表示に使用されます。

表示：別にフォーカス インジケータ ムービー クリップを持つ、「選択され」、フォーカスが適用されている Animated Toggle Button が、Enter キーやゲーム コントローラ ボタンで押された場合、またはマウスの左ボタンでクリックされた場合で、ユーザーが次にマウスをドラッグしてボタンから離れたとき、あるいは矢印キーまたはゲーム コントローラを使って、Enter キー、ゲーム コントローラ ボタン、マウスの左ボタンを解放する前に、別のボタンを選択したとき。

用途：解放済みの「選択され」フォーカスが適用されている Animated Toggle Button を表現します。

5.2.3 Selected_KB_Selecting

selected_kb_selecting はボタンの *down* ステートから *selected_up* ステートへのアニメーション トランジションの表示に使用されます。これは「選択解除」のステートから「選択された」ステートへの切り替えトランジションです。

表示：ユーザーが「選択解除」され、フォーカスが適用された Animated Toggle Button を、Enter キー、またはゲーム コントローラ ボタンの押下と解放で切り替えたとき。

用途：キーボードやゲーム コントローラで切り替えられた、別にフォーカス インジケータ ムービー クリップを持つ「選択され」、フォーカスが適用された Animated Toggle Button の表現に使用されます。

5.2.4 KB_Selecting

kb_selecting トランジションは、ボタンの *selected_down* ステートから *up* ステートへのアニメーション トランジションの表示に使用されます。これは「選択された」ステートから「選択解除」のステートへの切り替えトランジションです。

表示：ユーザーが「選択され」、フォーカスが適用された Animated Toggle Button を、Enter キー、またはゲーム コントローラ ボタンの押下と解放で切り替えたとき。

用途：キーボードやゲーム コントローラで切り替えられた、別にフォーカス インジケータ ムービー クリップを持つ「選択解除」され、フォーカスが適用された Animated Toggle Button の表現に使用されます。

6. 最後に

Scaleform CLIK ボタンのバリエーションは、UI デザイナーにさらに柔軟性を持たせるように設計されています。各ボタン バリエーションのさまざまなトランジションやステートを学習することは、CLIK を使いこなすために不可欠です。各バリエーションをいつ使用するかを知っていれば、時間と労力が節約できます。

- *Button* : 基本のボタン バリエーション
- *Toggle Button* : オン/オフの切り替え機能が追加されたボタン
- *Animated Button* : アニメーション トランジションを持つボタン
- *Animated Toggle Button* : Toggle Button と Animated Button の機能を組み合わせたボタン