

FLASH® LITE™ 1.x アプリケーションの開発

© 2007 Adobe Systems Incorporated. All rights reserved.

Flash® Lite™ 1.x アプリケーションの開発

本マニュアルがエンドユーザー使用許諾契約を含むソフトウェアと共に提供される場合、本マニュアルおよびその中に記載されているソフトウェアは、エンドユーザー使用許諾契約にもとづいて提供されるものであり、当該エンドユーザー使用許諾契約の契約条件に従ってのみ使用または複製することが可能となるものです。当該エンドユーザー使用許諾契約により許可されている場合を除き、本マニュアルのいかなる部分といえども、Adobe Systems Incorporated (アドビ システムズ社) の書面による事前の許可なしに、電子的、機械的、録音、その他いかなる形式・手段であれ、複製、検索システムへの保存、または伝送を行うことはできません。本マニュアルの内容は、エンドユーザー使用許諾契約を含むソフトウェアと共に提供されていない場合であっても、著作権法により保護されていることにご留意ください。

本マニュアルに記載される内容は、あくまでも参照用としてのみ使用されること、また、なんら予告なしに変更されることを条件として、提供されるものであり、従って、当該情報が、アドビ システムズ社による確約として解釈されることはありません。アドビ システムズ社は、本マニュアルにおけるいかなる誤りまたは不正確な記述に対しても、いかなる義務や責任を負うものではありません。

新しいアートを創作するためにテンプレートとして取り込もうとする既存のアートワークまたは画像は、著作権法により保護されている可能性のあるものであることをご留意ください。保護されているアートワークまたは画像を新しいアートワークに許可なく取り込んだ場合、著作権者の権利を侵害することがあります。従って、著作権者から必要なすべての許可を必ず取得してください。

例として使用されている会社名は、実在の会社・組織を示すものではありません。

Adobe、Adobe ロゴ、Flash Lite および Flash は、アドビ システムズ社の米国ならびに他の国における商標または登録商標です。

サードパーティ情報

本マニュアルには、アドビ システムズ社が管理していない、サードパーティの Web サイトへのリンクが掲載されていますが、アドビ システムズ社はいかなるリンク先サイトの内容についても責任を持ちません。本マニュアルに記載されているサードパーティの Web サイトには、自己責任においてアクセスしてください。アドビ システムズ社はこれらのリンクを便宜上の目的においてのみ掲載しています。リンクを掲載することにより、アドビ システムズ社がこれらのサードパーティのサイトの内容について何らかの責任を持つことを示すものではありません。



Sorenson™ Spark™ ビデオ圧縮および圧縮解除テクノロジーは、Sorenson Media, Inc. のライセンス供与によって提供されます。

Fraunhofer-IIS/Thomson Multimedia: MPEG レイヤー 3 音声圧縮テクノロジーは、Fraunhofer IIS および Thomson Multimedia (<http://www.iis.fhg.de/amm/>) によりライセンス供与されています。

Independent JPEG Group: 本ソフトウェアの一部は、Independent JPEG Group による著作物に基づきます。

Nellymoser, Inc.: 音声圧縮および圧縮解除テクノロジーは、Nellymoser, Inc. (<http://www.nellymoser.com>) のライセンス供与によって提供されます。

Opera® browser Copyright © 1995-2002 Opera Software ASA and its suppliers. All rights reserved.

Macromedia Flash 8 ビデオは、On2 TrueMotion ビデオテクノロジーを利用しています。© 1992-2005 On2 Technologies, Inc. All Rights Reserved. <http://www.on2.com>.

Visual SourceSafe は、米国およびその他の国における Microsoft Corporation の登録商標または商標です。

更新情報およびその他のサードパーティのコード情報は、http://www.adobe.com/go/thirdparty_jp/ で入手できます。

Adobe Systems Incorporated, 345 Park Avenue, San Jose, California 95110, USA.

Notice to U.S. government end users. The Software and Documentation are "Commercial Items," as that term is defined at 48 C.F.R. §2.101, consisting of "Commercial Computer Software" and "Commercial Computer Software Documentation," as such terms are used in 48 C.F.R. §12.212 or 48 C.F.R. §227.7202, as applicable. Consistent with 48 C.F.R. §12.212 or 48 C.F.R. §§227.7202-1 through 227.7202-4, as applicable, the Commercial Computer Software and Commercial Computer Software Documentation are being licensed to U.S. Government end users (a) only as Commercial Items and (b) with only those rights as are granted to all other end users pursuant to the terms and conditions herein. Unpublished-rights reserved under the copyright laws of the United States. Adobe Systems Incorporated, 345 Park Avenue, San Jose, CA 95110-2704, USA. For U.S. Government End Users, Adobe agrees to comply with all applicable equal opportunity laws including, if appropriate, the provisions of Executive Order 11246, as amended, Section 402 of the Vietnam Era Veterans Readjustment Assistance Act of 1974 (38 USC 4212), and Section 503 of the Rehabilitation Act of 1973, as amended, and the regulations at 41 CFR Parts 60-1 through 60-60, 60-250 ,and 60-741. The affirmative action clause and regulations contained in the preceding sentence shall be incorporated by reference.

目次

はじめに.....	7
Flash Lite オーサリングの新しい機能.....	7
ヘルプメディアについて.....	7
他の情報源.....	8
表記規則.....	8
第1章：インタラクティブ機能とナビゲーション機能の作成.....	9
Flash Lite でサポートされているキー.....	10
Flash Lite でのタブナビゲーションの使用.....	11
キーイベントの処理.....	15
ボタンイベントの処理.....	22
ソフトキーの使用.....	28
第2章：テキストとフォントの操作.....	33
Flash Lite のテキストについて.....	33
テキスト入力フィールドの使用.....	36
Flash Lite でのフォントのレンダリング方法.....	40
Flash Lite のレンダリング品質とアンチエイリアス処理されたテキスト.....	41
SWF ファイルへのフォントアウトラインの埋め込み.....	42
テキストフィールドのサンプルアプリケーション.....	43
スクロールテキストの作成.....	46
第3章：サウンドの操作.....	49
Flash Lite のサウンドについて.....	49
デバイスサウンドの使用.....	50
Flash ネイティブサウンドの使用.....	57
第4章：コンテンツのパフォーマンスとファイルサイズの最適化.....	59

第5章：Flash Lite コンテンツのレビュー	61
Flash Lite のレビュー機能の概要	61
エミュレータでサポートされていないテスト機能.....	62
エミュレータの使用	62
Flash Lite コンテンツタイプ	65
エラー	69
プラットフォームの機能の確認.....	71
索引	75

はじめに

本マニュアルでは、アドビ システムズ社の Macromedia® Flash® Lite™ 1.0 および Macromedia® Flash® Lite™ 1.1(まとめて Flash Lite 1.x と呼びます)を使用して、モバイルデバイス向けのアプリケーションを開発する方法について説明します。デバイスの機種およびテキストやフォントの操作方法に応じて、さまざまなナビゲーションモードを使用できます。本マニュアルでは、さまざまな機種の携帯電話で動作する Flash Lite アプリケーションのランタイムメモリを管理する方法についても説明します。Adobe® Flash® CS3 Professional に搭載されている Adobe® Device Central CS3 を使用すると、アプリケーションをオーサリングツールでプレビューおよびデバッグしてから実機でテストすることができます。

Flash Lite オーサリングの新しい機能

Flash には、Flash Lite アプリケーションの開発を支援するための次のような機能が含まれています。

Adobe Device Central Adobe Device Central には、コンテンツが実際のデバイスでどのように機能するかをプレビューできる携帯端末エミュレータがあります。エミュレータは、対応機種の機能を反映して自動的に設定されます。エミュレータによるデバッグ情報に基づいて、対応機種の非互換性などの問題を事前に知ることもできます。

デバイスドキュメントテンプレート Flash に付属しているドキュメントテンプレートを使用すると、デバイスおよびコンテンツタイプに応じたコンテンツの作成を簡単に開始できます。

ヘルプメディアについて

Flash Lite マニュアルパッケージには、Flash Lite アプリケーションの作成に役立つ次のメディアが含まれています。

- 『Flash Lite 1.x ファーストステップガイド』では、Flash Lite テクノロジーおよびモバイルデバイス用の Flash Lite コンテンツの開発の概要について説明します。また、Flash Lite 1.x アプリケーションの作成手順を説明するチュートリアルも示します。
- 『Flash Lite 1.x アプリケーションの開発』は Flash Lite 1.x コンテンツの作成に関する総合的なガイドで、Adobe Device Central エミュレータでアプリケーションをテストする手順についても説明しています。

- 『Flash Lite 1.x ActionScript リファレンスガイド』には、Flash Lite デベロッパー用の ActionScript 言語機能に関するリファレンスとサンプルコードが含まれています。
- 『Flash Lite 1.x ActionScript の学習』は、リファレンスガイドを補完し、追加のサンプルコードを提供します。また、Flash Lite 1.x ActionScript の基礎となっている Flash 4 ActionScript の記述方法についての概説があります。
- http://www.adobe.com/go/learn_flt_samples_and_tutorials_jpにある Flash Lite サンプルアプリケーションは、マニュアルで説明されている主な概念およびベストプラクティスを示しています。

他の情報源

Flash Lite アプリケーション開発の最新情報、アドバンスドユーザーからのアドバイス、高度なテクニック、サンプル、ヒント、その他の更新情報については、モバイル & デバイスデベロッパーセンター (www.adobe.com/go/developer_flashlite_jp) を参照してください。

テクニカルノート、マニュアルの更新情報、および Flash コミュニティ内の追加情報へのリンクについては、Adobe Flash Lite サポートセンター (www.adobe.com/go/support_flashlite_jp) を参照してください。

表記規則

本マニュアルでは、次の表記規則を使用しています。

- `Code font` (コードフォント) は ActionScript コードを示します。
- *Code font italic* (イタリック体のコードフォント) は ActionScript パラメータを示します。
- **Bold font** (ボールドフォント) はリテラルの入力を示します。
- コード例の二重引用符 (" ") は区切られた文字列を示します。ただし、プログラマは単一引用符を使用することもできます。

インタラクティブ機能とナビゲーション機能の作成

アドビシステムズ社の Macromedia Flash Lite 1.0 および Flash Lite 1.1 は、デバイスのキーを使用したユーザー操作、または機種によってはペンやタッチスクリーンインターフェイスを通じてのユーザー操作をサポートしています。

キーによる操作機能を Flash Lite アプリケーションに追加するには 2 つの方法があります。Flash Lite のデフォルトのタブナビゲーションを使用するか、独自のキーを使用したナビゲーションシステムを作成することができます。

Flash Lite でのタブナビゲーションの動作は Flash デスクトップアプリケーションと同じであり、Tab キーおよび Shift+Tab を使用して画面のオブジェクト間を移動できます。Flash Lite では、デバイスの 4 方向矢印キーが Tab キーおよび Shift+Tab と同じ役割を果たします。Flash Lite のタブナビゲーションは、ボタンとテキスト入力フィールドに対してのみ使用できます。通常は、メニューなどの単純なユーザー操作に使用します。詳細については、[11 ページの「Flash Lite でのタブナビゲーションの使用」](#)を参照してください。

タブナビゲーションを使用する代わりに、独自のキーナビゲーションを使用することができます。この場合、アプリケーションはユーザーがデバイスでキーを押したときに Flash Lite で生成される keypress イベントに応じて適切な処理を行います。このタイプのナビゲーションは、インタラクションモデルが単純なメニューよりも複雑な Flash Lite ゲームやその他のアプリケーションを作成するような場合に使用します。

この章では、次のトピックについて説明します。

Flash Lite でサポートされているキー	10
Flash Lite でのタブナビゲーションの使用	11
キーイベントの処理	15
ボタンイベントの処理	22
ソフトキーの使用	28

Flash Lite でサポートされているキー

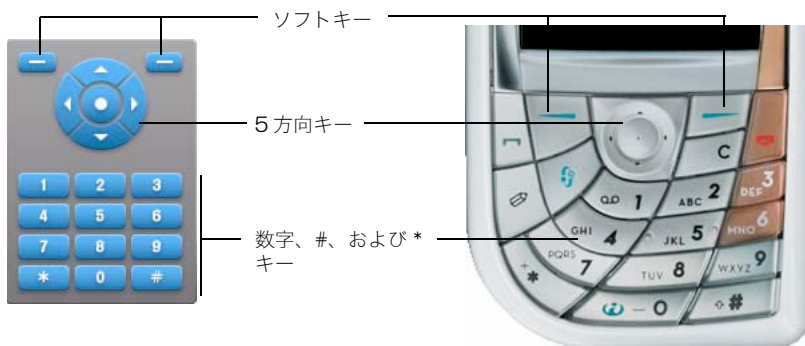
標準的な電話機の英数字キーに加えて、多くのモバイルデバイスには、ユーザーがデバイスのディスプレイ上のアイテムをナビゲートして選択するための5方向キーと、左ソフトキーおよび右ソフトキーがあります。デバイスのソフトキーは、デバイスのディスプレイを使用してその時点での用途を示すさまざまな機能を持つキーです。

5方向キーには、4つの矢印キー(上、下、左、右)と選択キー(通常、4つの矢印キーの中央に配置される)があります。アプリケーションごとに各キーの用途は異なる場合があります。たとえば、Flash Lite ゲームでは矢印キーを使用して画面上のキャラクタを移動し、次に選択キーを押してキャラクタをジャンプさせるなどの別の操作を実行できます。

Flash Lite は、モバイルデバイスの次のキーをサポートしています。

- 5方向キー(上下左右および選択キー)
- 左右のソフトキー
- 0～9、*、および#キー

次の図は、汎用キーパッドおよび実機の各キーを示しています。



すべてのデバイスおよび Flash Lite コンテンツタイプで、これらのすべてのキーがサポートされているわけではありません。たとえば、デバイスによっては Flash Lite アプリケーションで上下の方向キーは使用できるが、左右の方向キーは使用できないという場合があります。また、デバイスおよびコンテンツタイプによっては左右のソフトキーを使用できない場合もあります。Adobe Device Central でアプリケーションをプレビューするとき、対象のデバイスおよびコンテンツタイプで使用できないキーを押すと警告メッセージが表示されます。

Flash Lite でのタブナビゲーションの使用

デスクトップの Flash アプリケーションでは、Tab キーおよび Shift+Tab を使用して画面のオブジェクト間でキーボードフォーカスを切り替えることができます。フォーカスがあるオブジェクトに、キーを押したときの操作が適用されます。Flash Lite では、デバイスの 5 方向キーの矢印キーを使用して、Tab キーおよび Shift+Tab と同じ操作を行うことができます。

Flash Lite は、タブナビゲーションのモードとして 2 方向、4 方向、および折り返し付き 4 方向の 3 種類をサポートしています。デバイスおよび Flash Lite コンテンツタイプに応じて、サポートされるナビゲーションモードは異なります。詳細については、[11 ページの「タブナビゲーションのモード」](#)を参照してください。

Flash Lite のタブナビゲーションは、ボタンとテキストフィールドに適用されます。テキスト入力フィールドにフォーカスがあるときに選択キーを押すと、デバイスの汎用テキスト入力ダイアログボックスが開いて、テキストを入力することができます。入力テキストフィールドを使用してユーザー入力を取得する例については、[43 ページの「テキストフィールドのサンプルアプリケーション」](#)を参照してください。

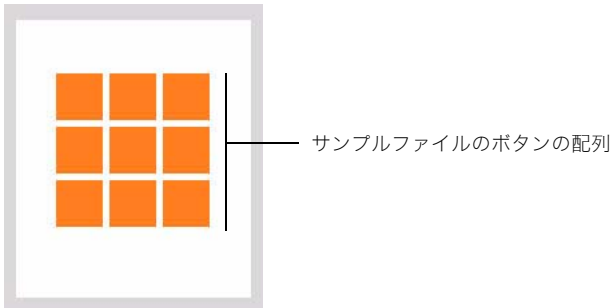
この項では、次のトピックについて説明します。

タブナビゲーションのモード	11
フォーカス矩形について	13
タブナビゲーションの使用のガイドライン	14

タブナビゲーションのモード

Flash Lite は、タブナビゲーションのモードとして 2 方向、4 方向、および折り返し付き 4 方向の 3 種類をサポートしています。デバイスおよび Flash Lite コンテンツタイプに応じて、サポートされるナビゲーションモードは異なります。具体的なデバイスおよびコンテンツタイプで使用するナビゲーションモードについては、[71 ページの「プラットフォームの機能の確認」](#)を参照してください。

以下の項で説明する各ナビゲーションモードでは、Adobe Device Central で表示できるサンプルファイルを参照します。各サンプルファイルは、次に示すように同じ 3x3 グリッドのボタンで構成されています。サンプルファイル間の唯一の相違点は、各ナビゲーションモードをサポートするデバイスおよび Flash Lite コンテンツタイプに応じて設定されていることです。



各サンプルファイルを使用するには、ファイルを Adobe Flash CS3 Professional で開いて Adobe Device Central でプレビューします ([制御]-[ムービープレビュー])。シミュレーション対象デバイスの方向キーをクリックするか、キーボードの方向キーを押すことで、ナビゲーションモード別にユーザーナビゲーションの動作を確認できます。

2方向ナビゲーション 2方向ナビゲーションでは、デバイスの上下の方向キーを押すと、オブジェクト (ボタンまたはテキスト入力フィールド) 間でフォーカスが移動します。左右の方向キーを押しても変化は起きません。下の方向キーを押すと、フォーカスは右側のオブジェクトに移動します。現在フォーカスがあるオブジェクトの右側にオブジェクトがない場合は、下の左端にあるオブジェクトにフォーカスが移動します。右端のオブジェクトにフォーカスがあって、その下にオブジェクトがない場合は、一番上で左端にあるオブジェクトにフォーカスが移動します。上の方向キーを押すと、フォーカスは左側のオブジェクトに移動します。現在フォーカスがあるオブジェクトの左側にオブジェクトがない場合は、そのオブジェクトの上の右端にあるオブジェクトにフォーカスが移動します。

2方向ナビゲーションの例については、2-way fla というサンプルファイルを参照してください。このファイルは、www.adobe.com/go/learn_ft_samples_and_tutorials_jp にあります。サンプルとチュートリアル ページで、Flash Lite バージョンの .zip ファイルを探してダウンロードおよび解凍し、"Samples" フォルダを選択してサンプルにアクセスします。

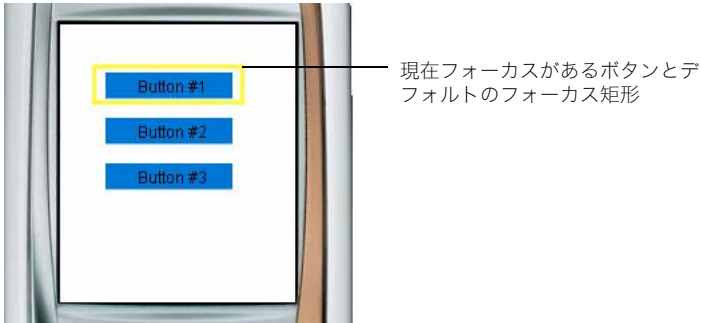
4方向ナビゲーション 4方向ナビゲーションでは、デバイスの左右上下のキーを使用してオブジェクト間でフォーカスを移動できます。左の方向キーを押すと、現在フォーカスがあるオブジェクトの左側にあるボタンにフォーカスが移動します。右の方向キーを押すと、現在フォーカスがあるボタンの右側にあるボタンにフォーカスが移動します。上および下方向キーも同様に、現在フォーカスがあるボタンの上または下にあるボタンにフォーカスが移動します。

4方向のナビゲーションの例については、4-way fla というサンプルファイルを参照してください。このファイルは、www.adobe.com/go/learn_ft_samples_and_tutorials_jp にあります。サンプルとチュートリアル ページで、Flash Lite バージョンの .zip ファイルを探してダウンロードおよび解凍し、"Samples" フォルダを選択してサンプルにアクセスします。

折り返し付きの4方向ナビゲーション このモードは4方向ナビゲーションと同じですが、フォーカスがある右端のボタンの下にボタンがない場合は、一番上の左端のボタンにフォーカスが移動します。

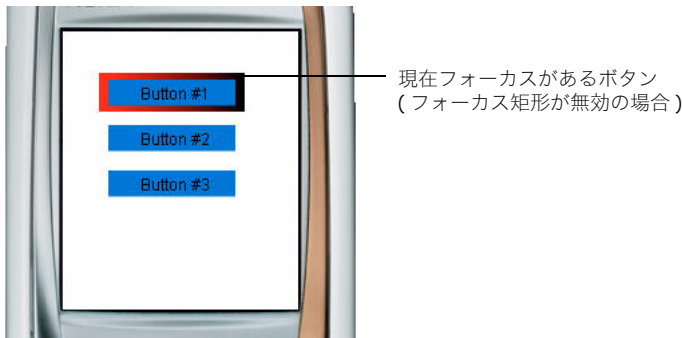
フォーカス矩形について

デフォルトでは、Flash Lite でフォーカスがあるボタンまたはテキスト入力フィールドの回りに黄色の矩形が表示されます。フォーカス矩形が表示されていると、デバイスの選択キーを押したときに画面上で反応するオブジェクトがわかります。たとえば、次の図では、現在フォーカスがあるボタンの周囲にフォーカス矩形が表示されています。

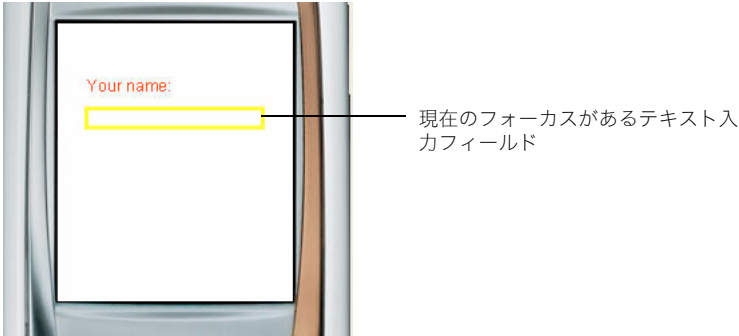


ボタンの場合、フォーカス矩形の境界ボックスはボタンのヒット領域によって決まります。ヒット領域は、(Flash デスクトップアプリケーションで) ボタンのマウスのクリックに反応する部分を定義する非表示の領域です。テキスト入力フィールドの場合、フォーカス矩形の境界ボックスはテキストフィールドのサイズに対応します。

フォーカス矩形のデフォルトの動作を無効にするには、`_focusRect` プロパティを `false` に設定します。オーバー状態を定義するボタンを使用している場合、ボタンにフォーカスが移動したときにその状態が表示されます。このため、ボタンの使用時はフォーカス矩形が必要ない場合があります。たとえば、次の図は前の図と同じアプリケーションを示していますが、フォーカス矩形は無効になっています。フォーカスがあるボタンはオーバー状態を表示しています。



アプリケーションにテキスト入力フィールドが含まれている場合は、フォーカス矩形が表示されないとフォーカスの有無がわからないので、フォーカス矩形を無効にしないでください。たとえば、次の図は現在のフォーカスがあるテキスト入力フィールドを示しています。



オーバー状態を定義したボタンとテキスト入力フィールドが同じ画面にある場合は、次のサンプルコードに示すように、`_focusRect` プロパティを各ボタンの `on(rollOver)` イベントハンドラでは `false` に設定し、各ボタンの `on(rollOut)` ハンドラでは `true` に設定することができます。これにより、フォーカス矩形はテキスト入力フィールドにフォーカスがあるときに表示され、ボタンにフォーカスがあるときに表示されなくなります。

```
// このコードをステージ上の各ボタンにアタッチします。
on(rollOver) {
    _focusRect = false;
}
on(rollOut) {
    _focusRect = true;
}
```

テキスト入力フィールドの使用の詳細については、[36 ページの「テキスト入力フィールドの使用」](#)を参照してください。

タブナビゲーションの使用のガイドライン

タブナビゲーションを使用してインタラクティブ機能を作成する場合、スクリーンに少なくとも 2 つのオブジェクト (テキスト入力フィールド、ボタン、またはその組み合わせ) を同時に配置する必要があります。スクリーンにボタンやテキスト入力フィールドが 1 つしかない場合、ユーザーはフォーカスを変更することができず、ユーザーインターフェイスが動かなくなってしまうと感ずる可能性があります。

アプリケーションの画面にユーザーが操作できるオブジェクトとしてボタンが 1 つしかない場合は、ボタンイベントを使用しないで `keypress` イベントを検知するようにします。詳細については、[15 ページの「キーイベントの処理」](#)を参照してください。

キーイベントの処理

ボタンやテキスト入力フィールドの間でタブナビゲーションを使用する以外に、Flash Lite アプリケーションでは任意の `keypress` イベントに応答することができます。

デバイスおよびコンテンツタイプによっては、一部のデバイスキーをサポートしていない場合もあります。たとえば、2方向のナビゲーション(11ページの「タブナビゲーションのモード」を参照)をサポートするデバイスでは、Flash Lite は左および右矢印キーについて `keypress` イベントを生成しません。

すべてのデバイスで、Flash Lite は次のキーをサポートしています。

- 0～9、*、および # キー
- 選択キー

2方向のナビゲーションをサポートするデバイスでは、Flash Lite は5方向キーの上および下矢印キーもサポートします。4方向のナビゲーションをサポートするデバイスでは、Flash Lite は上、下、左、および右矢印キーをサポートします。

`SetSoftKeys` コマンドをサポートするデバイスでは、Flash Lite は左および右ソフトキーもサポートします。

この項では、次のトピックについて説明します。

キーハンドラスクリプトの作成	15
キーキャッチャーボタンの作成	16
ムービークリップを使用する単純なメニューの作成	18

キーハンドラスクリプトの作成

`keypress` イベントを処理するには、`on(keyPress "key")` ハンドラをボタンインスタンスにアタッチします。ここで、`key` はサポートされるキーイベント名です。たとえば、ステージ上のボタンインスタンスにアタッチされた次のコードは、ユーザーがデバイスの右矢印キーを押したときに実行されます。

```
on(keyPress "<Right>") {  
    trace("You pressed the right arrow key");  
}
```

次の表は、ユーザーがデバイスでキーを押したときに Flash Lite で生成される ActionScript `keypress` イベントの一覧です。

デバイスキー	ActionScript キーイベント	使用可
0-9, *, #	0, 1, 2, 3, 4, 5, 6, 7, 8, -9, *, #	すべてのデバイス
選択キー	<Enter>	すべてのデバイス

デバイスキー	ActionScript キーイベント	使用可
左の方向キー	<Left>	4方向のナビゲーションをサポートするデバイスのみ
右の方向キー	<Right>	4方向のナビゲーションをサポートするデバイスのみ
上の方向キー	<Up>	4方向のナビゲーションをサポートするデバイスのみ
下の方向キー	<Down>	4方向のナビゲーションをサポートするデバイスのみ
左ソフトキー	<PageUp>	SetSoftKeys コマンドをサポートするデバイスのみ
右ソフトキー	<PageDown>	SetSoftKeys コマンドをサポートするデバイスのみ

キーキャッチャーボタンの作成

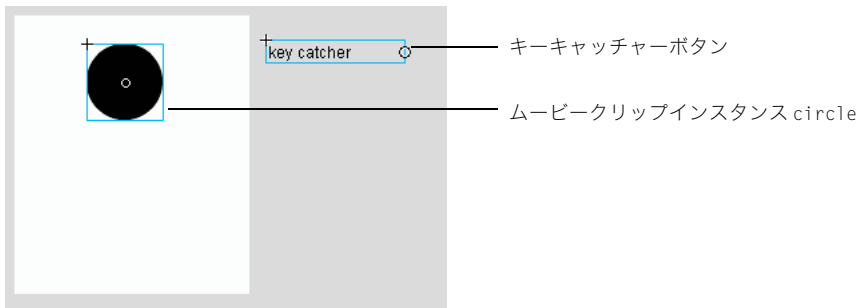
アプリケーションで複数の keypress イベントを処理する必要がある場合は、keypress イベントごとに1つのボタンを作成するか、1つのボタンを使用してすべての keypress イベントを処理することができます。このタイプのボタンは、一般的にキーキャッチャー(またはキーリスナー)ボタンと呼ばれます。キーキャッチャーボタンは、ユーザーに対して表示することを想定していないので、通常はステージの外側(ワークエリア内)に配置されます。

次の手順では、キーキャッチャーボタンを使用して、簡単なアプリケーションの keypress イベントを処理する方法を説明します。このアプリケーションでは、ユーザーはデバイスの4つの矢印キーを押すことによって、ステージの周囲で円を移動できます。

キーキャッチャーボタンを作成および使用するには：

- Flash Lite 1.1 Series 60 テンプレートから新しいドキュメントを作成し、keycatcher.fla という名前で保存します。
Flash Lite テンプレートからのドキュメントの作成の詳細については、『Flash Lite 1.x ファーストステップガイド』の「Flash Lite ドキュメントテンプレートの使用」を参照してください。
- Content という名前のタイムラインでレイヤーを選択します。
- 楕円ツールを使用して、ステージ上に楕円または円を作成し、ムービークリップに変換します。
- 新しいムービークリップが選択されている状態で、プロパティインスペクタの [インスタンス名] テキストボックスに「circle」と入力します。
- テキストツールを使用して、「key catcher」というテキストを含むテキストフィールドを作成し、ボタンシンボルに変換します。

- 新しいボタンシンボルをステージの周囲のワークエリアに配置します。ステージの周囲のワークエリアを表示するには、[表示]-[ワークエリア]を選択します。



- キーキャッチャーボタンインスタンスを選択し、[アクション]パネルを開きます ([ウィンドウ]-[アクション])。
- [アクション]パネルで、次のコードを入力します。

```
on(keyPress "<Left>") {
    circle._x -= 10;
}
on(keyPress "<Right>") {
    circle._x += 10;
}
on(keyPress "<Up>") {
    circle._y -= 10;
}
on(keyPress "<Down>") {
    circle._y += 10;
}
```

- [制御]-[ムービープレビュー]を選択して、アプリケーションをテストします。
Adobe Device Central エミュレータの4つの矢印キーを押して、ステージの周囲に円を移動します。

キーキャッチャーボタンを使用するもう1つの例については、[18 ページの「ムービークリップを使用する単純なメニューの作成」](#)を参照してください。

ムービークリップを使用する単純なメニューの作成

この項では、ムービークリップを使用する単純なメニューを作成する方法について説明します。ここでは、ボタン間のデフォルトのタブナビゲーションおよび各ボタンにコードを追加する方法を使用せずに、キーキャッチャーボタンを使用して `keypress` イベントに応答し、必要に応じてユーザーインターフェイスを更新します。この手法では、ボタンによるメニューのアプローチ (15 ページの「キーイベントの処理」を参照) よりも開発作業は多くなりますが、次のような利点があります。

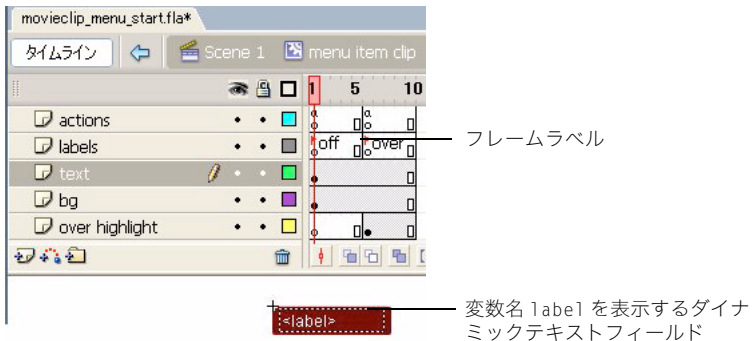
- タブ順序を制御できます。タブ順序を (ボタンの場合のように) Flash Lite プレーヤーが決定するのではなく、開発者がフォーカスを与えるオブジェクトとその `keypress` イベントへの反応を決定できます。
- アプリケーションの状態間でメニューの選択を維持できます。たとえば、ユーザーが最後に選択したメニュー項目を「記憶」して後で同じ項目にフォーカスを戻したい場合があります。これは、ボタンメニューを使用している場合は無理です。ActionScript を使用してボタンフォーカスを割り当てることはできないためです。

次の手順では、一部完成した Flash ドキュメントから始めます。完成版アプリケーション (movieclip_menu_complete fla) のサンプルについては、Flash Lite のサンプルとチュートリアル のページ (www.adobe.com/go/learn_flt_samples_and_tutorials_jp) を参照してください。使用している ActionScript バージョンの .zip ファイルを探してダウンロードおよび解凍し、"Samples" フォルダを選択してサンプルにアクセスします。

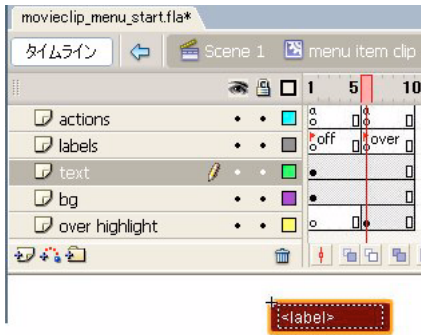
ムービークリップを使用する単純なメニューを作成するには：

1. www.adobe.com/go/learn_flt_samples_and_tutorials_jp にある "movieclip_menu_start fla" という名前のファイルをダウンロードして開きます。サンプルとチュートリアル のページで、Flash Lite バージョンの .zip ファイルを探してダウンロードおよび解凍し、"Samples" フォルダを選択してサンプルにアクセスします。
2. タイムラインで、Menu Items レイヤーを選択します。
3. [ライブラリ] パネルを開いて ([ウィンドウ]-[ライブラリ]), Menu Item Clip というムービークリップシンボルのインスタンスをステージにドラッグします。

このムービークリップには、2 つのキーフレームまたはビジュアル状態が含まれています。メニュー項目の最初の選択されていない状態と、メニュー項目がフォーカスを受け取ったときに表示される選択された状態です。次の図は、ムービークリップのタイムラインにある最初のキーフレームです。メニュー項目のラベルを表示するダイナミックテキストフィールドと背景の赤いグラフィックを示しています。テキストフィールドと背景のグラフィックは、このムービークリップのタイムライン上にあるすべてのフレームにわたって表示されます。

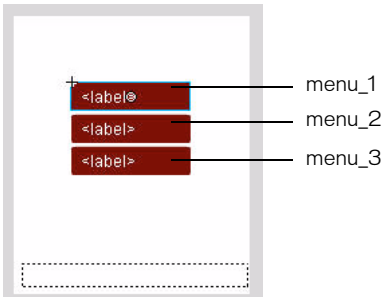


次の図はムービークリップのタイムラインのフレーム 5 を示しています。このフレームと最初のフレームの視覚的な相違点は、メニュー項目の赤い背景を囲む黄色のハイライトされた枠線だけです。



4. メニューアイテムのムービークリップのインスタンスをもう 2 つステージにドラッグし、縦 1 列に整列します。
5. 上のムービークリップを選択し、プロパティインスペクタの [インスタンス名] テキストボックスに「menu_1」と入力します。

- 中央と下のムービークリップに、それぞれ `menu_2` および `menu_3` というインスタンス名を割り当てます。



各インスタンス名の最後の数値は、これから追加するコードで各ムービークリップを動的に参照するためのものです。

- テキストツールを使用して、ステージの下端に沿ってテキストフィールドを作成します。
- プロパティインスペクタの[テキストの種類]ポップアップメニューから[ダイナミックテキスト]を選択し、[変数]テキストボックスに「`status`」と入力します。
単純なメニューの例と同じように、このテキストフィールドには現在選択されているメニュー項目に関するステータスメッセージが表示されます。
- タイムラインで、**Actions** レイヤーの最初のフレームを選択します。
- [アクション]パネルを開いて([ウィンドウ]-[アクション])、次のコードを入力します。

```
// メニュー項目のラベルを初期化します :
menu_1.label = "News";
menu_2.label = "Sports";
menu_3.label = "Weather";

// メニューアイテムの数を指定する変数を初期化します
numItems = 3;

// selectedItem 変数を初期化します。この変数には、
// 現在選択されているメニュー項目のインデックスが入ります
selectedItem = 1;

// ステータステキストフィールドを初期化します
currentLabel = eval("menu_" + selectedItem + ".label");
status = "Press to select " + currentLabel;

// 最初のメニューアイテムを " オーバー " 状態に送ります
tellTarget("menu_1") {
    gotoAndStop("over");
}
```

- タイムラインで、**Key Catcher** という名前のレイヤーを選択します。

12. ライブラリを開いて、key catcher というボタンのインスタンスをステージにドラッグします。
次に、このボタンに、ユーザーの keypress イベントを処理するイベントハンドラコードを追加し、ユーザーインターフェイスを更新します。
13. ステージでボタンを選択状態にし、[アクション]パネルを開きます。
14. [アクション]パネルに次のコードを入力(またはコピー&ペースト)します:

```
on(keyPress "<Down>") {
    if(selectedItem < numItems) {
        // 前に選択したメニューアイテムのハイライトをオフにします :
        tellTarget("menu_" add selectedItem) {
            gotoAndStop("off");
        }
        // selectedItem 変数をインクリメントし、
        // 新しい選択アイテムのハイライトをオンにします
        selectedItem++;
        tellTarget("menu_" add selectedItem) {
            gotoAndStop("over");
        }
        // 選択したアイテムのラベルを使ってステータステキストフィールドを更新します :
        currentLabel = eval("menu_" add selectedItem add ":label");
        status = "Press to select " add currentLabel;
    }
}

on(keyPress "<Up>") {
    if(selectedItem > 1) {
        // 前に選択したアイテムのハイライトをオフにします :
        tellTarget("menu_" add selectedItem) {
            gotoAndStop("off");
        }
        // selectedItem をデクリメントし、新しい選択アイテムのハイライトをオンにします
        selectedItem--;
        tellTarget("menu_" add selectedItem) {
            gotoAndStop("over");
        }
        // 選択したアイテムのラベルを使ってステータステキストフィールドを更新します :
        currentLabel = eval("menu_" add selectedItem add ":label");
        status = "Press to select " add currentLabel;
    }
}

on(keyPress "<Enter>") {
    // 選択したアイテムを使ってステータスフィールドを更新します
    status = "You selected " add currentLabel;
}
```

15. [制御]-[ムービープレビュー]を選択して Adobe Device Central シミュレータでアプリケーションをテストします。

メニューを操作するには、マウスでエミュレータの上下の方向キーをクリックするか、キーボードの対応する方向キーを押します。

ボタンイベントの処理

Flash Lite は、ActionScript ボタンイベントとして、press、release、rollOver、および rollOut をサポートしています。これらのイベントを処理するには、`on(event)` ハンドラをボタンインスタンスにアタッチします。ここで、`event` は次の表に示されているサポートされているボタンイベントのいずれかです。

ボタンイベント	イベントが生成される時点
press	ボタンにフォーカスがある場合にユーザーがデバイスの選択キーを押したとき
release	ボタンにフォーカスがある場合にユーザーがデバイスの選択キーを離れたとき
rollOver	ボタンがフォーカスを受け取ったとき
rollOut	ボタンがフォーカスを失ったとき

次の手順では、ボタンイベントを処理する単純なアプリケーションを作成する方法を説明します。ボタンを使用してメニューを作成する例については、[24 ページの「ボタンとタブナビゲーションを使用する単純なメニューの作成」](#)を参照してください。

ボタンイベントハンドラスクリプトを作成するには：

1. Flash Lite 1.1 Series 60 テンプレートから新しいドキュメントを作成し、`button_handler.fla` という名前で保存します。

Flash Lite テンプレートからのドキュメントの作成の詳細については、『Flash Lite 1.x ファーストステップガイド』の「Flash Lite ドキュメントテンプレートの使用」を参照してください。
2. [ウィンドウ]-[サンプルライブラリ]-[ボタン]を選択して、作成済みのボタンシンボルの外部ライブラリを開きます。
3. [ライブラリ]パネルで、"Circle Buttons" フォルダをダブルクリックして開きます。
4. メニューボタンシンボルのインスタンスをステージにドラッグします。
5. ボタンを選択し、[アクション]パネルを開きます([ウィンドウ]-[アクション])。

6. [アクション]パネルで、次のコードを入力します。

```
on(press) {
    trace("You pressed Button 1");
}
on(release) {
    trace("You released Button 1");
}
on(rollOver) {
    trace("Button 1 has focus");
}
on(rollOut) {
    trace("Button 1 lost focus");
}
```

7. 同じボタンの別のインスタンスをステージにドラッグし、最初のボタンのすぐ下に配置します。

8. ステージ上で 2 番目のボタンが選択されている状態で、[アクション]パネルを開いて、次のコードを入力します。

```
on(press) {
    trace("You pressed Button 2");
}
on(release) {
    trace("You released Button 2");
}
on(rollOver) {
    trace("Button 2 has focus");
}
on(rollOut) {
    trace("Button 2 lost focus");
}
```

9. タイムラインで、ActionScript レイヤーのフレーム 1 を選択します。

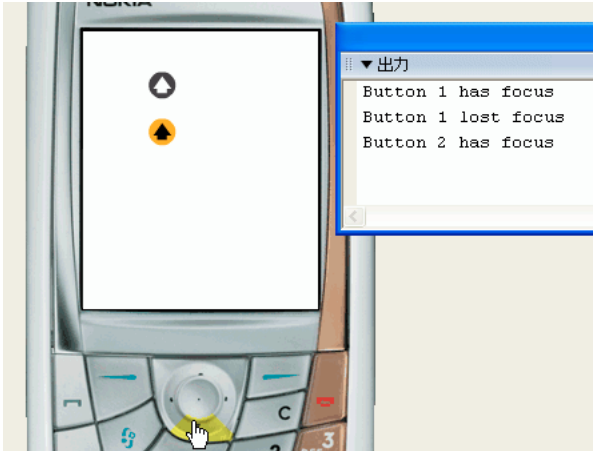
10. [アクション]パネルで、次のコードを入力します。

```
_focusRect = false;
```

これによって、フォーカスのあるボタンの周囲に描画される黄色のフォーカス矩形が無効になります。この例では、ボタンにフォーカスがあるときに表示されるオーバー状態が存在するので、デフォルトのフォーカス矩形は必要ありません。

11. Adobe Device Central エミュレータでアプリケーションをテストします ([制御]-[ムービープレビュー])。

エミュレータの上および下矢印キーを押すと、[出力] パネルにメッセージが表示されます。



ボタンとタブナビゲーションを使用する単純なメニューの作成

この項では、ボタンとタブナビゲーションを使用する単純なメニューの作成方法を示します。メニューを作成するには、3つのボタンシンボルを使用します。各シンボルが各メニューオプションに対応します。ユーザーが各メニュー項目に移動したとき(対応するボタンにフォーカスを与えたとき)とユーザーがデバイスの選択キーを押してメニュー項目を選択したときに、メッセージを表示するイベント処理コードを各ボタンに追加します。Flash Lite でのボタンイベントの処理の詳細については、[22 ページの「ボタンイベントの処理」](#)を参照してください。

Nokia 7610 およびスタンドアロンプレイヤーのコンテンツタイプ向けに設定された一部完成済みの Flash ドキュメントを使用して開始します。これらの設定は、異なるデバイスやコンテンツタイプに応じて変更できます(詳細については、Adobe Device Central ヘルプを参照してください)。

ボタンを使用する単純なメニューを作成するには：

1. "simple_menu_start fla" という名前のファイルを開きます。このファイルは、www.adobe.com/go/learn_ftl_samples_and_tutorials_jp にあります。サンプルとチュートリアル ページで、Flash Lite バージョンの .zip ファイルを探してダウンロードおよび解凍し、「Samples」フォルダを選択してサンプルにアクセスします。
2. [ライブラリ] パネルを開きます ([ウィンドウ]-[ライブラリ])。
ライブラリ内に、News Button、Weather Button、および Sports Button という 3 つのボタンシンボルがあります。
3. タイムライン内で ([ウィンドウ]-[タイムライン])、Menu Buttons レイヤーを選択します。

4. News Button シンボルのインスタンスを [ライブラリ] パネルからステージにドラッグします。
5. Sports Button と Weather Button に対し、手順 4 を繰り返します。
6. 次の図に示すように、3 つのボタンを縦に揃えて並べます。



7. [ツール] パネルで、テキストツールを選択して、ステージの下部にテキストフィールドを作成します。
このテキストフィールドには、ユーザーが各メニュー項目に移動したときに短いメッセージが表示されます。
8. 新しいテキストフィールドを選択状態にし、プロパティインスペクタで次の操作を行います。
 - a. [テキストの種類] ポップアップメニューから [ダイナミックテキスト] を選択します。
 - b. [変数] テキストボックスに 「status」と入力します。
9. ステージ上で、[News] ボタンを選択し、[アクション] パネルを開きます ([ウィンドウ]-[アクション])。

10. [アクション]パネルに次のコードを入力します。

```
on(rollOver) {  
    status = "Press to select News"  
}  
on(press) {  
    status = "You selected news"  
}
```

このコードは、ユーザーが [News] メニューボタン上にマウスポインタを移動したときにダイナミックテキストフィールドにテキストを割り当てます。

11. [Sports] ボタンを選択し、[アクション]パネルに次のコードを入力します。

```
on(rollOver) {  
    status = "Press to select Sports";  
}  
on(press) {  
    status = "You selected Sports";  
}
```

12. [Weather] ボタンを選択し、[アクション]パネルに次のコードを入力します。

```
on(rollOver) {  
    status = "Press to select Weather";  
}  
on(press) {  
    status = "You selected Weather";  
}
```

13. タイムラインで、Actions レイヤーのフレーム 1 を選択します。

14. [アクション]パネルに次のコードを入力します。

```
_focusRect = false;
```

これによって、フォーカスのあるボタンおよびテキストフィールドの周囲にデフォルトで描画される黄色のフォーカス矩形が無効になります (13 ページの「フォーカス矩形について」を参照)。この時点で、ステージは次の図のようになります。



15. [制御]-[ムービープレビュー]を選択して Adobe Device Central エミュレータでアプリケーションをプレビューします。

メニューオプション間を移動するには、下の方向キーをマウスでクリックするか、コンピュータのキーボードで下の方向キーを押します。メニュー項目を選択するには、選択キーをマウスでクリックするか、コンピュータのキーボードで Enter キーを押します。



ソフトキーの使用

デバイスのソフトキーは、デバイスの表示内容に応じて用途が変わる多機能キーです。たとえば、次のアプリケーションでは、ソフトキーの上のラベルは、ユーザーが右ソフトキーを押して次のおすすめディナーを表示したり、左ソフトキーを押してアプリケーションのホーム画面に戻ることができることを示します。



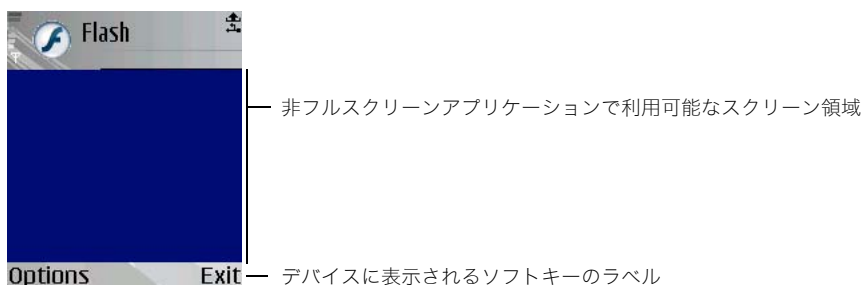
左および右ソフトキーを使用するには、最初に SetSoftKeys コマンドを呼び出す必要があります (『Flash Lite 1.x ActionScript リファレンスガイド』の「SetSoftKeys」を参照)。この結果、ユーザーが右ソフトキーを押すと PageDown イベントが生成され、ユーザーが左ソフトキーを押すと PageUp イベントが生成されます。これらのイベントにตอบสนองして所定の動作を行う ActionScript イベントハンドラコードを書きます。

SetSoftKeys コマンドは 2 つのパラメータを取ります。これらのパラメータは、アプリケーションがフルスクリーンモードで実行されていないときに表示される左ソフトキーと右ソフトキーのラベルを指定します。フルスクリーンモードで実行されているアプリケーションの場合は、指定したラベルは表示されないため、独自のラベルを作成して、ステージ上のソフトキーの場所に配置する必要があります。

たとえば、次の SetSoftKeys コマンド呼び出しについて考えてみます。

```
fsccommand2("SetSoftKeys", "Options", "Exit");
```

次の図は、(フルスクリーンモードではない) 通常モードの実際のデバイスで実行されるアプリケーションで、このコマンドを実行した結果を示しています。



フルスクリーンモードを有効にした場合、つまり `fsccommand("fullscreen", true)` コマンドを呼び出した場合、SetSoftKeys コマンドのパラメータとして指定したラベルは表示されません。このため、フルスクリーンモードのアプリケーションでは、次の図のように独自のソフトキーラベルを作成する必要があります。

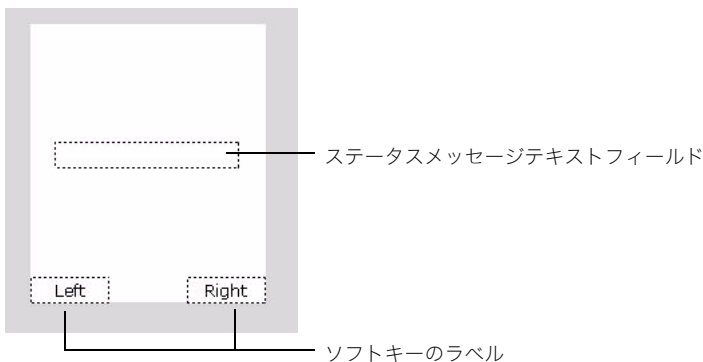



アプリケーションでソフトキーを使用するには：

1. Flash Lite 1.1 Series 60 デバイステンプレートから新しいドキュメントを作成します。
デバイステンプレートの詳細については、[17 ページの「Flash Lite ドキュメントテンプレートの使用」](#)を参照してください。
2. [デバイス設定] ダイアログボックスを開き、スタンドアローンプレーヤーをコンテンツタイプとして選択します。Symbian > Series 60 フォルダ内のデバイスの1つをテストデバイスの一覧に追加します。

3. タイムラインで、Content レイヤーを選択します。
4. テキストツールを使用して、Left (または別の指定したテキスト) という静止テキストフィールドを作成し、ステージの左下隅でデバイスの左のソフトキー上に配置します。
5. Right という別の静止テキストフィールドを作成し、ステージの右下隅でデバイスの右のソフトキー上に配置します。
6. テキストツールを使用して、別のダイナミックテキストフィールドを作成し、ステージの中央に配置します。

このテキストフィールドには、アプリケーションを実行して左右のソフトキーを押したときにメッセージが表示されます。ドキュメントのステージは次の図のようになります。



 実際のアプリケーションでは、ソフトキーのラベルとして、通常のテキストフィールドの代わりにグラフィックやムービークリップシンボルなどを表示することもできます。

7. ステータステキストフィールドが選択されている状態で、プロパティインスペクタの [変数] テキストボックスに「status」と入力します。
8. キーキャッチャーボタンを作成します (16 ページの「キーキャッチャーボタンの作成」を参照)。[アクション] パネルで、次のコードをボタンに追加します。

```
// 左のソフトキーの keypress イベントを処理します
on(keyPress "<PageUp>") {
    status = "You pressed the Left soft key.";
}
// 右のソフトキーの keypress イベントを処理します
on(keyPress "<PageDown>") {
    status = "You pressed the Right soft key.";
}
```

9. タイムラインで、Actions レイヤーのフレーム 1 を選択します。

10. [アクション]パネルに次のコードを入力します。

```
fscommand2("SetSoftKeys", "Left", "Right");  
fscommand2("FullScreen", true);
```

SetSoftKeys コマンドの2つのパラメータ(この例では Left と Right)は、アプリケーションがフルスクリーンモードで表示されていないときにソフトキーの上に表示されるラベルを指定します。この例では、アプリケーションは FullScreen コマンド(『Flash Lite 1.x ActionScript リファレンスガイド』の「FullScreen」を参照)を使用してアプリケーションを強制的にフルスクリーンモードにします。パラメータに指定する値には、任意のテキスト文字列または式を使用できます。

```
fscommand2("SetSoftKeys", foo, bar);
```

11. [制御]-[ムービープレビュー]を選択して Adobe Device Central でアプリケーションをプレビューします。

マウスで左右のソフトキーをクリックするか、キーボードの PageUp キーおよび PageDown キーを押して、アプリケーションの動作を確認します。



この章では、アドビ システムズ社の Macromedia Flash Lite 1.x アプリケーションに静止テキストフィールドやダイナミックテキストを追加する方法、およびテキスト入力フィールドを追加する方法について説明します。

この章では、次のトピックについて説明します。

Flash Lite のテキストについて	33
テキスト入力フィールドの使用	36
Flash Lite でのフォントのレンダリング方法	40
Flash Lite のレンダリング品質とアンチエイリアス処理されたテキスト	41
SWF ファイルへのフォントアウトラインの埋め込み	42
テキストフィールドのサンプルアプリケーション	43
スクロールテキストの作成	46

Flash Lite のテキストについて

Flash Lite では、静止、ダイナミック、入力の 3 種類のテキストフィールドをサポートしています。静止テキストフィールドのコンテンツは再生中に変化しません。たとえば、ページのタイトルやラベルを表示するには静止テキストフィールドを使用します。静止テキストフィールドの作成の詳細については、[43 ページの「テキストフィールドのサンプルアプリケーション」](#)を参照してください。

ダイナミックテキストフィールドでは、実行時にコンテンツを制御できます。ActionScript の変数名をダイナミックテキストフィールドに関連付けてコードで参照することができます。たとえば、Flash Lite で計算機を作成する場合、ダイナミックテキストフィールドを使用して計算結果を表示することができます。ダイナミックテキストフィールドの使用の詳細については、[34 ページの「テキストフィールドへの変数名の割り当て」](#)を参照してください。

テキスト入力フィールドはダイナミックテキストフィールドに似ていますが、テキスト入力フィールドではユーザーがデバイスの汎用入力ダイアログボックスを開いてテキストを入力できます。テキスト入力フィールドの使用の詳細については、[36 ページの「テキスト入力フィールドの使用」](#)を参照してください。

Flash Lite でのフォントのレンダリング方法について

デバイスのディスプレイにテキストを表示する場合、Flash Lite はデバイスで使用可能なフォントを使用するか、SWF ファイルに埋め込まれているフォントデータを使用することができます。デバイスフォントを使用すると SWF ファイルサイズが小さくなるという利点がありますが、フォントの表示を制御しにくいという欠点があります。フォントデータを SWF ファイルに埋め込むと、フォントの表示を制御しやすくなりますが、ファイルサイズが大きくなります。

埋め込みフォントデータに対して、Flash Lite はアンチエイリアス処理されたテキストとエイリアス処理された (ビットマップ) テキストの両方をサポートしています。これにより、小さいポイントサイズのテキストが読みやすくなります。サードパーティのフォントデザイナーによるピクセルフォントを使用して小さいテキストを読みやすくすることもできます。詳細については、[40 ページの「Flash Lite でのフォントのレンダリング方法」](#) および [42 ページの「SWF ファイルへのフォントアウトラインの埋め込み」](#) を参照してください。

Flash Lite 1.x ではサポートされない Flash Player のテキスト機能

Flash Lite 1.x のテキストフィールドは、デスクトップバージョンの Flash Player で使用可能な以下の機能をサポートしていません。

- Flash Player 8 以降で使用可能な拡張されたフォントレンダリングテクノロジー
- HTML、CSS、またはその他のリッチテキストフォーマット機能
- TextField オブジェクトおよび TextFormat オブジェクト

テキストフィールドへの変数名の割り当て

ダイナミックテキストフィールドと入力テキストフィールドに ActionScript の変数名を関連付けて、ActionScript を使用してテキストのコンテンツを取得または設定することができます。

変数名をダイナミックテキストフィールドまたはテキスト入力フィールドに関連付けるには：

1. Flash Lite 1.1 Symbian Series 60 テンプレートから新しいドキュメントを作成し、dynamic_text.fla という名前で作成します。

Flash Lite ドキュメントテンプレートの使用の詳細については、『Flash Lite 1.x ファーストステップガイド』の「Flash Lite ドキュメントテンプレートの使用」を参照してください。

2. ステージ上にダイナミックテキストフィールドまたはテキスト入力フィールドを作成し、それを選択します。

3. プロパティインスペクタで、[変数] テキストボックスに「nameVar」と入力します。

入力する値は有効な変数識別子になっている必要があります。つまり、半角を使用し、先頭文字は英字、アンダースコア (_)、ドル記号 (\$) のいずれかとし、それ以降の文字は、英字、数字、アンダースコア、ドル記号のいずれかである必要があります。

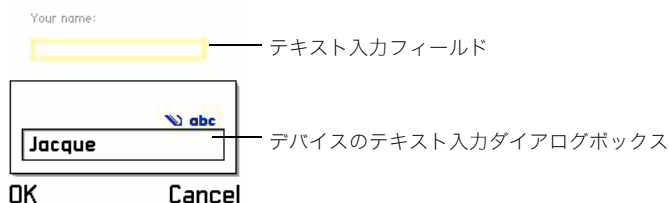


4. タイムラインで、ActionScript レイヤーのフレーム 1 を選択します。
5. [アクション] パネルを開き ([ウィンドウ]-[アクション])、次のコードを入力します。
`nameVar = "George Washington"`
6. Adobe Device Central エミュレータでアプリケーションをテストします ([制御]-[ムービープレビュー])

ダイナミックテキストフィールドのもう1つの使用例については、[43 ページの「テキストフィールドのサンプルアプリケーション」](#)を参照してください。

テキスト入力フィールドの使用

Flash Lite のテキスト入力フィールドでは、ダイナミックテキストフィールドと同じように、ActionScript を使用して実行時にコンテンツを取得または設定することができます。さらに、テキスト入力フィールドでは、Flash Lite アプリケーションでデバイスの汎用テキスト入力ダイアログボックスを使用して、ユーザー入力を取得できます (Flash Lite はインラインテキスト入力をサポートしていません)。次の図は、Symbian Series 60 デバイスの入力ダイアログボックスを示しています。

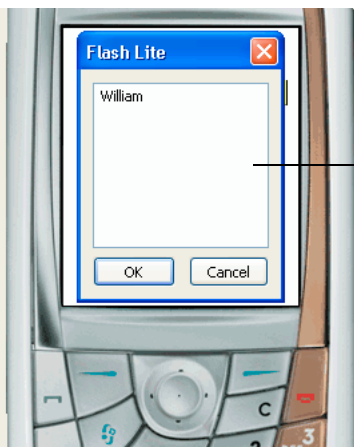


デバイスのテキスト入力ダイアログボックスを開くには、最初にテキスト入力フィールドにフォーカスを与え、次にデバイスの選択キーを押します。デフォルトでは、フォーカスがあるテキスト入力フィールドの回りに黄色の矩形が表示されます。

テキスト入力ダイアログボックスはモーダルなので、ダイアログボックスにフォーカスがある間は、ユーザーは Flash コンテンツを操作できません。ダイアログボックスにフォーカスがある間は、Flash アプリケーションの再生ヘッドも停止します。

ユーザーが [OK] (左のソフトキー) をクリックすると、テキスト入力ダイアログボックスが閉じ、Flash Lite によって自動的にテキスト入力フィールドにテキストが入力されます。ユーザーが [キャンセル] (右のソフトキー) をクリックすると、テキストはテキスト入力フィールドに入力されません。

Adobe Flash CS3 でアプリケーションをテストすると、Adobe Device Central エミュレータによってテキスト入力ダイアログボックスの機能が再現されます。次の図は、エミュレータで実行されているテキスト入力ダイアログボックスを示しています。

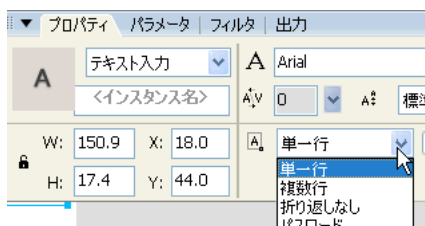


エミュレータで実行されているテキスト入力ダイアログ

アプリケーションでテキスト入力フィールドを使用する例については、[43 ページの「テキストフィールドのサンプルアプリケーション」](#)を参照してください。

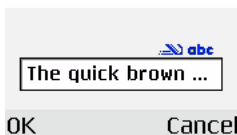
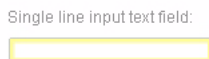
テキスト入力フィールドの種類

Flash Lite では、単一行、複数行、およびパスワードテキスト入力フィールドをサポートしています。次の図のように、プロパティスペクタの [行タイプ] ポップアップメニューを使用してテキスト入力フィールドの種類を指定します。



テキスト入力フィールドについて指定した行のタイプによって、ユーザーがテキストフィールドを編集するときのデバイスのテキスト入力ダイアログボックスの動作が決まります。

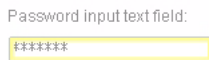
たとえば、ユーザーが単一行のテキスト入力フィールドを編集する場合、デバイスのテキスト入力ダイアログボックスには、単一行のテキスト入力ボックスが表示されます。ユーザーが表示できる文字数よりも多くの文字を入力すると、テキスト入力ボックスは横にスクロールします。次の図は、単一行のテキスト入力フィールドを編集する場合のデバイスのテキスト入力ダイアログボックスを示しています。



ユーザーが複数行のテキスト入力フィールドを編集する場合は、次の図のように、デバイスのテキスト入力ダイアログボックスが必要に応じて拡張され、ユーザーが入力したテキストがすべて表示されます。



ユーザーがパスワードテキスト入力フィールドを編集する場合は、デバイスのテキスト入力ダイアログボックスでは、ユーザーが入力した各文字が (少し遅れて) アスタリスクでマスクされます。



文字入力の制限

SetInputTextType コマンドを使用して、ユーザーがテキスト入力ダイアログボックスに入力できる文字を制限できます。たとえば、アプリケーションに、ユーザーが年齢などの数値を入力するためのテキスト入力フィールドがあるとします。また、このテキスト入力フィールドに ageVar という名前の変数が含まれているとします。

ユーザーがこのテキスト入力フィールドに数値だけを入力できるようにするには、アプリケーションに次のコードを追加します。

```
fsccommand2("SetInputTextType", "ageVar", "Numeric");
```

ユーザーがテキスト入力ダイアログボックスを開いたときに、このテキストフィールドには数値だけを入力できます。

詳細については、『Flash Lite 1.x ActionScript リファレンスガイド』の SetInputTextType を参照してください。

テキスト入力フィールドとフォーカス矩形

デフォルトでは、次の図のように、キーフォーカスがあるテキスト入力フィールドの周囲に黄色の矩形が描画されます。



フォーカス矩形は、グローバルプロパティ `_focusRect` を `false` に設定することによって無効にすることができます。ただし、無効にすると、ユーザーはテキストフィールドにキーフォーカスがあることがわからないので、デバイスで選択キーを押す判断ができなくなります。テキスト入力フィールドを使用する場合は、フォーカス矩形を無効にしないことをお勧めします。

黄色のフォーカス矩形の動作制御の詳細については、[13 ページの「フォーカス矩形について」](#) および『Flash Lite 1.x ActionScript リファレンスガイド』の「`_focusrect`」を参照してください。

Flash Lite でのフォントのレンダリング方法

Flash Lite では、次のいずれかの方法でテキストフィールドのフォントをレンダリングできます。

デバイスで利用可能なフォントを使用 テキストフィールドには、デバイスで利用できることがわかっているフォントを適用するか、[フォント] ポップアップメニューで選択可能な 3 つの汎用デバイスフォント (_ゴシック、_明朝、または_等幅) のいずれかを指定できます。汎用デバイスフォントを選択した場合は、実行時にその汎用フォントとデバイスフォントのマッチングが試行されます。たとえば、_sans が sans serif フォント (使用可能な場合) にマップされます。

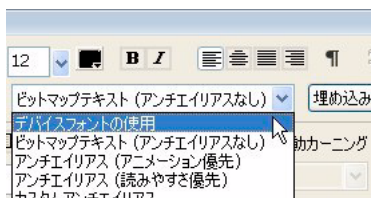
フォントをビットマップとしてレンダリング フォントのアウトラインをピクセルの境界に整列することによってビットマップテキストをレンダリングし、小さいポイントサイズ (10 ポイント以下) でもテキストを読みやすくします。このオプションを使用するには、パブリッシュする SWF ファイルに選択したフォントのアウトラインを含める必要があります (詳細については、[42 ページの「SWF ファイルへのフォントアウトラインの埋め込み」](#)を参照してください)。

フォントをアンチエイリアス処理されたベクターとしてレンダリング フォントアウトラインのベクター形式の表現を使用してアンチエイリアス処理されたテキストをレンダリングし、パブリッシュする SWF ファイルに埋め込みます (詳細については、[42 ページの「SWF ファイルへのフォントアウトラインの埋め込み」](#)を参照してください)。

テキストフィールドのフォントのレンダリング方法を選択するには、プロパティインスペクタの [フォントのレンダリング方法] ポップアップメニューを使用します。[フォントのレンダリング方法] ポップアップメニューには、5 つのレンダリングオプションがありますが、Flash Lite 1.0 および 1.1 で使用できるのはそのうちの 3 つだけです。残りの 2 つの方法 ([アンチエイリアス (読みやすさ優先)] と [カスタムアンチエイリアス]) は、Flash Player 8 以降に対応するアプリケーションでのみ使用できます。

テキストフィールドのフォントのレンダリング方法を選択するには：

1. ステージ上のテキストフィールドを選択します。
2. プロパティインスペクタで、[フォントのレンダリング方法] ポップアップメニューから次のいずれかのオプションを選択します。



- [デバイスフォントの使用] を選択すると、デバイスで使用可能なフォントが使用されます。パブリッシュする SWF ファイルにはフォントデータが埋め込まれません。

- [ビットマップテキスト (アンチエイリアスなし)] を選択すると、フォントアウトラインがピクセルの境界に揃えられ、小さい文字が判読しやすくなります。このオプションでは、Flash でパブリッシュする SWF ファイルにフォントのアウトラインを埋め込む必要があります(詳細については、[42 ページの「SWF ファイルへのフォントアウトラインの埋め込み」](#)を参照してください)。
- [アンチエイリアス (アニメーション優先)] を選択すると、現在のレンダリング品質の設定に従ってテキストフィールドのフォントがアンチエイリアス処理されます (詳細については、[41 ページの「Flash Lite のレンダリング品質とアンチエイリアス処理されたテキスト」](#)を参照してください)。このオプションでは、Flash でパブリッシュする SWF ファイルにフォントのアウトラインを埋め込む必要があります (詳細については、[41 ページの「Flash Lite のレンダリング品質とアンチエイリアス処理されたテキスト」](#)を参照してください)。

Flash Lite 1.0 および 1.1 では、[アンチエイリアス (読みやすさ優先)] および [カスタムアンチエイリアス] フォントレンダリングオプションはサポートされていません。これらのレンダリングオプションは、デスクトップコンピュータの Flash Player 8 以降でのみ使用できます。

Flash Lite のレンダリング品質とアンチエイリアス処理されたテキスト

Flash Lite ではレンダリング品質として低、中、高の 3 種類を設定できます。レンダリング品質を高く設定すると、レンダリングされるベクターのアウトラインがなめらかで正確になります。品質の設定が低いほど、表示されるアウトラインはなめらかでなくなります。デフォルトでは、中の品質を使用してアウトラインがレンダリングされます。SetQuality コマンドを使用してレンダリング品質を制御できます (詳細については、Flash Lite 1.x ActionScript リファレンスガイドの「SetQuality」を参照してください)。

Flash Lite では、フォントアウトラインのベクター表現を使用してアンチエイリアス処理されたテキストをレンダリングします。テキストをできるだけなめらかにアンチエイリアス処理するには、プレーヤーのレンダリング品質を高に設定します。レンダリング品質は、アンチエイリアス処理したテキストだけでなく、画面上のすべてのベクターコンテンツに影響します。高、中、低のレンダリング品質でアンチエイリアス処理したテキストフィールド (Arial、24 ポイント) を以下に示します。

*High
quality
setting*

*Medium
quality
setting*

*Low
quality
setting*

CPU の負荷が高いアニメーションやトゥイーンシーケンスなどで、アニメーションのパフォーマンスとフレームレートを最大限に向上する場合は、レンダリング品質を一時的に低く設定して、アニメーションが完了した後で前の設定に戻すことができます。

SWF ファイルへのフォントアウトラインの埋め込み

テキストフィールドのフォントをレンダリングするために、Flash Lite ではデバイスで利用可能なフォントを使用するか、パブリッシュされた SWF ファイルに埋め込まれたフォントアウトラインを使用することができます (詳細については、[40 ページの「Flash Lite でのフォントのレンダリング方法」](#)を参照してください)。SWF ファイルにフォントアウトラインを埋め込むと、テキストフィールドのフォントはすべてのターゲットプラットフォームで同じように表示されます。ただし、ファイルサイズは大きくなります。Flash Lite では、フォントアウトラインをビットマップ (非アンチエイリアス) またはアンチエイリアス処理したテキストとしてレンダリングする必要があります。

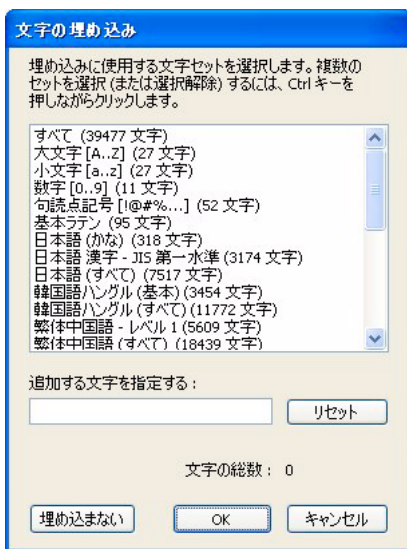
アンチエイリアスまたはビットマップとしてフォントをレンダリングする方法を使用する静止テキストフィールドの場合は、テキストフィールドの内容を表示するのに必要なフォントアウトラインが自動的に埋め込まれます。たとえば、静止テキストフィールドに **Submit** という単語が含まれている場合、この 6 つの文字 (**S**、**u**、**b**、**m**、**i**、および **t**) を表示するのに必要なフォントアウトラインが自動的に埋め込まれます。静止テキストフィールドの内容は変化しないので、SWF ファイルに含める必要があるのは、表示される特定の文字のフォントアウトラインだけです。

アンチエイリアスおよびビットマップのフォントレンダリング方法を使用するダイナミックおよびテキスト入力フィールドの場合は、指定する文字のフォントアウトラインを、パブリッシュする SWF ファイルに埋め込む必要があります。この 2 種類のテキストフィールドは、再生中に内容が変動するので、必要なフォントアウトラインを Flash では判断できません。選択したフォントのすべての文字、文字の範囲、または特定の文字のフォントアウトラインを含めることができます。パブリッシュする SWF ファイルに埋め込む文字を指定するには、[文字の埋め込み] ダイアログボックスを使用します。

ダイナミックテキストフィールドとテキスト入力フィールドのフォントアウトラインを埋め込むには：

1. ステージ上のダイナミックテキストフィールドまたはテキスト入力フィールドを選択します。
2. プロパティインスペクタで、[フォントのレンダリング方法] ポップアップメニューから [ビットマップ (アンチエイリアスなし)] または [アンチエイリアス (アニメーション優先)] を選択します。

3. [フォントのレンダリング方法] ポップアップメニューの横にある [埋め込み] ボタンをクリックし、[文字の埋め込み] ダイアログボックスを開きます。



4. 埋め込む文字をリストから選択するか、埋め込む特定の文字をテキストボックスに入力するか、[自動読み込み] をクリックして選択したテキストフィールドに表示されている文字を含めます。
5. [OK] をクリックします。

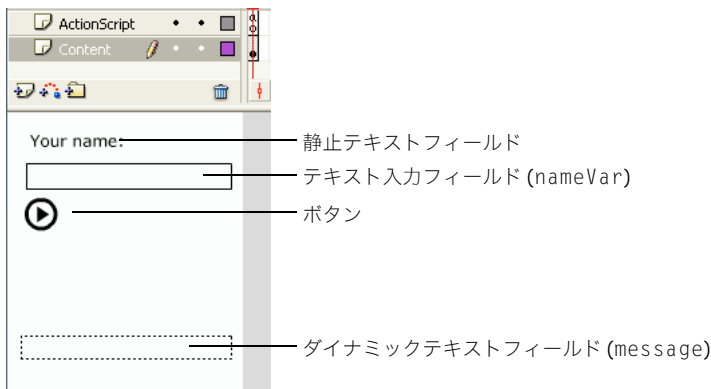
テキストフィールドのサンプルアプリケーション

この項では、静止、ダイナミック、および入力各テキストフィールドを使用する単純なアプリケーションを作成する方法について説明します。アプリケーションは、静止テキストフィールドのテキストを通じて、テキスト入力フィールドに名前を入力することをユーザーに要求します。ユーザーが名前を入力してボタンを選択すると、アプリケーションによってそのユーザー宛てのメッセージが(ダイナミックテキストフィールドを使用して)表示されます。

テキストフィールドのサンプルアプリケーションを作成するには：

1. Flash で、Flash Lite 1-1 - Symbian Series 60 デバイステンプレートから新しいドキュメントを作成します。
デバイステンプレートからのドキュメントの作成の詳細については、『Flash Lite 1.x ファーストステップガイド』の「Flash Lite ドキュメントテンプレートの使用」を参照してください。
2. タイムライン ([ウィンドウ]-[タイムライン]) で、ActionScript レイヤーの最初のフレームを選択します。

3. [アクション] パネルを開きます ([ウィンドウ]-[アクション])
4. [アクション] パネルに「stop();」と入力して、フレームの再生ヘッドを停止します。
5. [ツール] パネルからテキストツールを選択し、「Your name:」というテキストを含むテキストフィールドをステージに作成します。
6. 新しいテキストフィールドを選択した状態で、プロパティインスペクタの[フォントのレンダリング方法]ポップアップメニューから[デバイスフォントの使用]を選択します。
7. 最初のテキストフィールドの下に新しいテキストフィールドを作成し、プロパティインスペクタで、[テキストの種類]ポップアップメニューから[テキスト入力]を選択し、[フォントのレンダリング方法]ポップアップメニューから[デバイスフォントの使用]を選択し、[変数]テキストボックスに「nameVar」と入力して、[テキストの周囲にボーダーを表示]オプションを選択します。
8. [ウィンドウ]-[サンプルライブラリ]-[ボタン]を選択して、作成済みのボタンシンボルのライブラリを開きます。
9. [Buttons] ライブラリで、「Circle Buttons」フォルダをダブルクリックして開きます。
10. "circle button - next" という名前前のシンボルのインスタンスをステージにドラッグします。
11. テキストツールを使用して、ステージの下部に新しいテキストフィールドを作成します。
このテキストフィールドには、ユーザーがテキスト入力フィールドに入力した名前を含むメッセージが表示されます。
12. 新しいテキストフィールドを選択した状態で、プロパティインスペクタで、[変数]テキストボックスに「message」と入力し、[テキストの種類]ポップアップメニューから[ダイナミックテキスト]を選択し、[フォントのレンダリング方法]ポップアップメニューから[デバイスフォントの使用]を選択します。
アプリケーションのステージは、次の図のようになります。



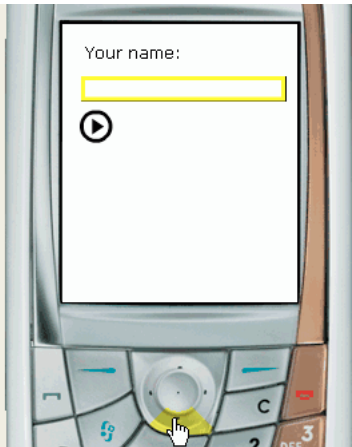
13. ステージ上でボタンを選択し、[アクション] パネルを開きます ([ウィンドウ]-[アクション])

14. [アクション]パネルで、次のコードを入力します。

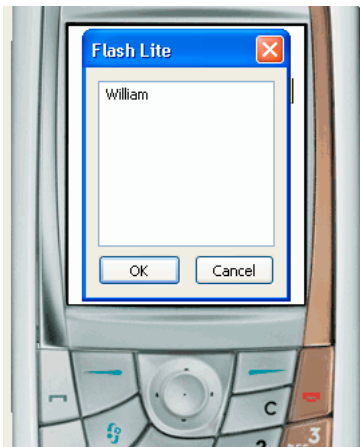
```
on(press) {  
    message = "Hello, " + nameVar;  
}
```

15. [制御]-[ムービープレビュー]を選択して Adobe Device Central シミュレータでアプリケーションをテストします。

- a. エミュレータの下矢印キーを押して、テキスト入力フィールドにフォーカスを移動します。

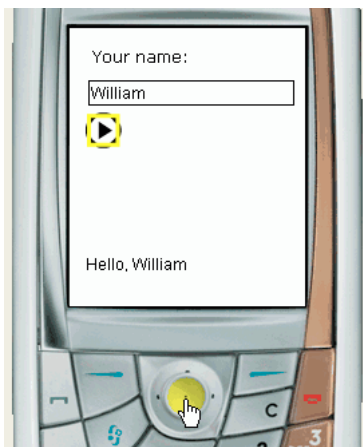


- b. エミュレータの選択キーを押してエミュレータのテキスト入力ダイアログボックスを開き、コンピュータのキーボードを使用して自分の名前を入力します。



- c. [OK] をクリックして、テキスト入力ダイアログボックスを閉じます。

- d. もう一度エミュレータの下矢印キーを押してボタンにフォーカスを移動し、選択キーを押します。



スクロールテキストの作成

Flash Lite 1.1では、`scroll` および `maxscroll` テキストフィールドプロパティをサポートしており、スクロールテキストフィールドを作成することができます。`scroll` プロパティはテキストブロック内で表示可能な最初の行を指定します。この値は取得および設定できます。たとえば、次のコードは、変数名が `story_text` であるテキストフィールドを下へ5行スクロールします。

```
story_text.scroll += 5;
```

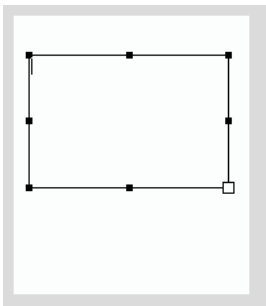
`maxscroll` プロパティは、テキストブロック内でテキストの最終行が表示されている場合に、テキストブロック内で表示可能な最初の行を指定します。このプロパティは読み取り専用です。テキストフィールドの `maxscroll` プロパティを `scroll` プロパティと比較することによって、ユーザーがテキストフィールド内でどれくらいスクロールしたかを特定できます。この機能は、最大スクロール位置に対するユーザーの現在のスクロール位置をフィードバックするようなスクロールバーを作成する場合に便利です。

スクロールテキストフィールドを作成して ActionScript で制御するには：

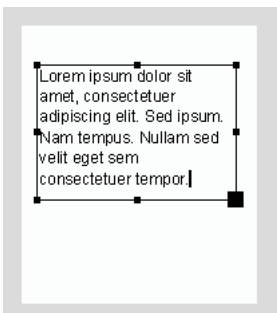
1. Flash で、Flash Lite 1.1 - Symbian Series 60 デバイステンプレートから新しいドキュメントを作成します。

デバイステンプレートからのドキュメントの作成の詳細については、『Flash Lite 1.x ファーストステップガイド』の「Flash Lite ドキュメントテンプレートの使用」を参照してください。

2. テキストツールを使用して、ステージをクリックし、ステージ上で次の図に示されているようなサイズになるまでテキストフィールドをドラッグします。



3. プロパティインスペクタの [行タイプ] ポップアップメニューから [複数行] を選択します。
4. プロパティインスペクタの [テキストの種類] ポップアップメニューから [ダイナミックテキスト] を選択します。
5. プロパティインスペクタの [フォントのレンダリング方法] ポップアップメニューから [デバイスフォントの使用] を選択します。
6. [テキスト] - [スクロール可能] を選択してテキストフィールドをスクロール可能にします。
7. プロパティインスペクタの [変数] テキストボックスに「story」を入力します。これによって、story という名前の ActionScript 変数がテキストフィールドに関連付けられます。
8. テキストフィールド内をダブルクリックし、テキストを入力します。多すぎて一部が下端に表示されないぐらいの量を入力します。



9. 新しいボタンシンボルを作成し、そのインスタンスをステージまたはステージ外の領域に追加します。

このボタンはキーキャッチャーボタンとして機能するので、ユーザーに対して表示する必要はありません。キーキャッチャーボタンの作成の詳細については、[16 ページの「キーキャッチャーボタンの作成」](#)を参照してください。

10. ボタンを選択し、[アクション]パネルを開きます([ウィンドウ]-[アクション])。
11. [アクション]パネルで、次のコードを入力します。

```
on(keyPress "<Down>") {
    story.scroll++;
}
on(keyPress "<Up>") {
    story.scroll--;
}
```

12. [制御]-[ムービープレビュー]を選択して Adobe Device Central シミュレータでアプリケーションをテストします。

キーボードの上下の矢印キーを押すか、エミュレータの上下のキーを押して、テキストを上下にスクロールします。

このサンプルは、例を簡単にするためにテキストフィールドの内容をオーサリングツールで入力するようになっていますが、ActionScript を使用してテキストフィールドの内容を更新するように変更することも簡単にできます。そのためには、複数行テキストフィールドに割り当てた変数名(このサンプルでは story) に目的のテキストを割り当てる ActionScript を作成します。

```
story = "Lorem ipsum dolor sit amet, consectetur adipiscing elit. Sed ipsum. Nam tempus. Nullam sed velit eget sem consectetur tempor. Morbi eleifend venenatis pede. Cras ac lorem eget massa tincidunt iaculis...etc."
```

アドビ システムズ社の Macromedia Flash Lite 1.1 は、デバイスサウンドと標準 (ネイティブ) の Flash サウンドをサポートしています。この章では、モバイルデバイスのアプリケーションにサウンドを追加するための操作について説明します。

この章では、次のトピックについて説明します。

Flash Lite のサウンドについて	49
デバイスサウンドの使用	50
Flash ネイティブサウンドの使用	57

Flash Lite のサウンドについて

Flash Lite 1.1 では、デバイスサウンドと標準 (ネイティブ) の Flash サウンドの 2 種類をサポートしています。デバイスサウンドは、パブリッシュされる SWF ファイルに MIDI や MFi などのデバイス固有のオーディオ形式で保存されます。デバイスサウンドを再生する場合は、Flash Lite からデバイスにサウンドデータが渡され、デバイスでサウンドが復号化されて再生されます。デバイスサウンドをアニメーションと同期することはできません。Flash Lite 1.0 ではデバイスサウンドのみをサポートしています。

Flash Lite 1.0 でサポートされているデバイスオーディオ形式に加えて、Flash Lite 1.1 では、デスクトップバージョンの Adobe Flash Player でサポートされているサウンドと同じ種類の Flash ネイティブオーディオをサポートしています。デバイスサウンドとは異なり、ネイティブサウンドはタイムラインでアニメーションと同期することができます。

Flash Lite 1.0 は MIDI および MFi のデバイスオーディオ形式をサポートしています。これらの形式に加えて、Flash Lite 1.1 では Synthetic music Mobile Application Format (SMAF) デバイスオーディオ形式、および PCM (または WAV)、ADPCM、または MP3 で圧縮された Flash ネイティブサウンドをサポートしています。

イベントおよびストリーム(同期した)サウンド

Flash Lite 1.1は、イベントおよびストリーム(同期した)サウンドをサポートしています。イベントサウンドは、タイムラインに関係なしに再生され、サウンドバッファの終わりに達するか、ActionScriptで停止されるまで再生が続きます。イベントサウンドは、完全にダウンロードされてから再生を開始する必要があります。

ストリームサウンドは、その関連するタイムラインと同期します。通常、オーディオをアニメーションと同期するのに使用します。ストリームサウンドは、その関連するタイムラインの再生ヘッドが停止すると停止します。サウンドの再生をアニメーションと同期するために、必要に応じて、再生中にアニメーションのフレームがとばされ(コマ落ち)ます。

タイムラインと同期できるのはFlashネイティブサウンドだけです。デバイスサウンドはイベントサウンドとしてのみ使用できます。Flash Lite 1.0はイベントサウンドのみをサポートしています。

デバイスサウンドの使用

デバイスサウンドは、デバイス固有のオーディオ形式(MIDIやMFIなど)で暗号化されたサウンドです。Flash Liteでは、デバイスサウンドファイルをFlashドキュメント内に直接読み込むことはできません。最初に、MP3、WAV、AIFFなどのサポートされている形式でプロキシサウンドを読み込む必要があります。次に、そのプロキシサウンドをMIDIファイルなどの外部モバイルデバイスサウンドにリンクします。ドキュメントのパブリッシュ時に、プロキシサウンドはリンクした外部サウンドに置き換えられます。生成されるSWFファイルには外部サウンドが含まれており、このサウンドがモバイルデバイスでの再生時に使用されます。

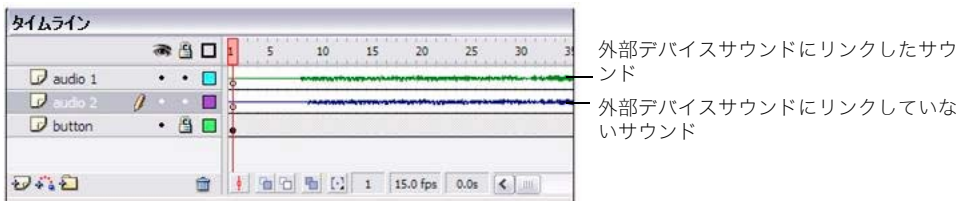
Flash Liteでデバイスサウンドを使用する場合、次の制約に留意してください。

- デバイスサウンドはイベントサウンドとしてのみ使用できます。デバイスサウンドをタイムラインと同期することはできません。
- Flash Liteでは、デバイスサウンドの効果オプション、同期オプション、および編集オプションをサポートしていません。
- ドキュメント内のサウンドごとに、外部デバイスサウンドファイルを指定する必要があります。
- すべての外部ファイルに該当することですが、SWFファイルのパブリッシュ時にデバイスサウンドファイルまたはコンパウンドサウンドファイルが必要です。ただし、これらのファイルはSWFファイルの再生時には必要ありません。

複数のデバイスサウンドを1つのファイルにバンドルすることもできます。この機能は、デバイスサウンド形式が異なる複数のデバイスに対して同じコンテンツを作成する場合に便利です。詳細については、[55 ページの「コンパウンドサウンドの使用」](#)を参照してください。

Flash Lite 1.1では、デバイスサウンドをいつでも再生できます。Flash Lite 1.0では、ユーザーがデバイスでキーを押したときのみデバイスサウンドが再生されます。詳細については、[54 ページの「Flash Lite 1.0でのデバイスサウンドのトリガ」](#)を参照してください。

Flash オーサリングツールのタイムラインには、次の図に示すように、サウンドの波形が表示されます。外部デバイスサウンドにリンクしたサウンドの波形は緑色です。外部デバイスサウンドにリンクしていないサウンドの波形は青色です。



この項では、次のトピックについて説明します。

ボタンへのデバイスサウンドの追加.....	51
Flash Lite 1.0 でのデバイスサウンドのトリガ.....	54
コンパウンドサウンドの使用.....	55

ボタンへのデバイスサウンドの追加

次の手順は、ユーザーがボタンを "クリック" したとき (つまり、ボタンにフォーカスがあるときにユーザーがデバイスの選択キーを押したとき) にサウンドが再生されるように、ボタンシンボルのタイムラインにデバイスサウンドを追加する方法を示しています。そのためには、ボタンシンボルのタイムラインで [ダウン] フレームにプロキシサウンドを割り当てます。次に、再生するデバイスサウンドをプロキシサウンドに関連付けます。

このアプリケーションの完成バージョンである "button_sound_complete fla" は、http://www.adobe.com/go/learn_ftl_samples_and_tutorials_jp にあります。サンプルとチュートリアルで、Flash Lite バージョンの .zip ファイルを探してダウンロードおよび解凍し、"Samples" フォルダを選択してサンプルにアクセスします。

デバイスサウンドを Flash ドキュメントに追加するには：

1. Flash Lite 1.0 Symbian Series 60 ドキュメントテンプレートから新しいドキュメントを作成し、device_sound fla という名前で保存します。

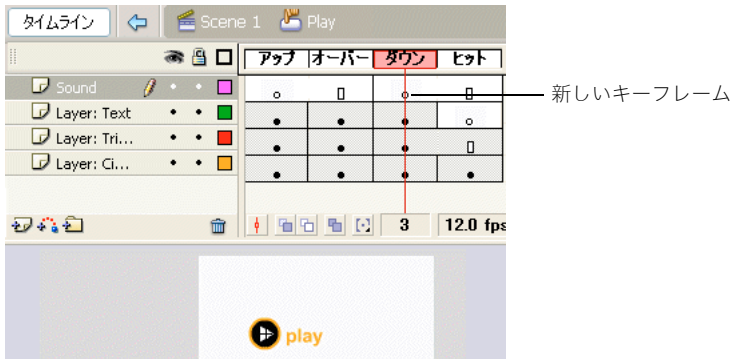
Flash Lite ドキュメントテンプレートの使用の詳細については、『Flash Lite 1.x ファーストステップガイド』の「Flash Lite ドキュメントテンプレートの使用」を参照してください。

2. [ファイル]-[読み込み]-[ライブラリに読み込み] を選択します。

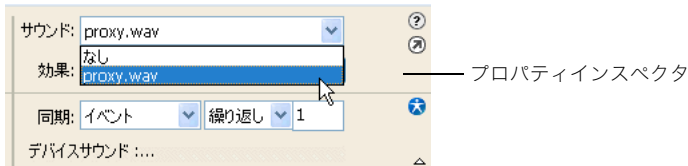
http://www.adobe.com/go/learn_ftl_samples_and_tutorials_jp にあるサンプルとチュートリアルで、Flash Lite バージョンの .zip ファイルをダウンロードおよび解凍し、"Samples" フォルダを選択します。proxy.wav ファイルを選択して [OK] をクリックします。

このサウンドファイルは、含めるデバイスサウンドのプロキシサウンドとして機能します。

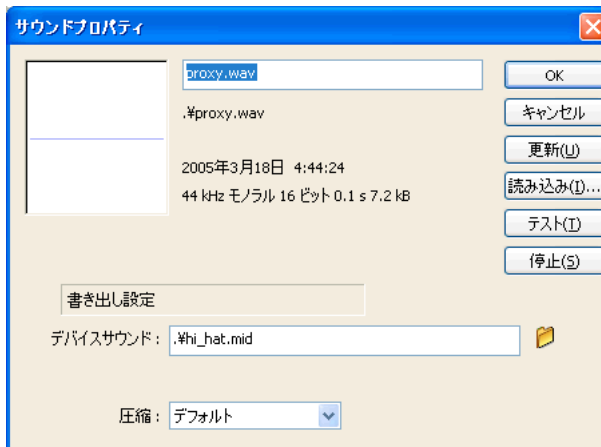
3. [ウィンドウ]-[サンプルライブラリ]-[ボタン] を選択します。
作成済みのボタンシンボルの外部ライブラリが開きます。
4. [Buttons] ライブラリで、"Circle Buttons" フォルダをダブルクリックして開きます。
5. タイムラインで、Content という名前のレイヤーを選択します。
6. ボタンライブラリから Play という名前のボタンシンボルをステージにドラッグします。
7. 新しいボタンをダブルクリックし、編集モードで開きます。
タイムラインの表示が変わり、ボタンの [アップ]、[オーバー]、[ダウン]、[ヒット] という名前のフレームが表示されます。
8. [挿入]-[タイムライン]-[レイヤー] を選択して新しいレイヤーを作成します。
9. [修正]-[タイムライン]-[レイヤープロパティ] を選択し、レイヤーの名前を「サウンド」に変更します。
10. Sound レイヤーの [ダウン] フレームを選択し、F6 キーを押して新しいキーフレームを挿入します。



11. タイムラインで新しいキーフレームを選択した状態で、プロパティインスペクタの [サウンド] ポップアップメニューから [proxy.wav] を選択します。
これによって、プロキシサウンドがキーフレームに割り当てられます。



12. プロキシサウンドを実際のデバイスサウンドファイルにリンクするには、次の操作を行います。
- ライブラリで proxy.wav のシンボルをダブルクリックして、[サウンドプロパティ] ダイアログボックスを開きます。
 - [サウンドプロパティ] ダイアログボックスで、[デバイスサウンド] テキストボックスの右側にあるフォルダアイコンをクリックして、[デバイスサウンドの選択] ダイアログボックスを開きます。
 - http://www.adobe.com/go/learn_flt_samples_and_tutorials_jp にアクセスします。サンプルとチュートリアルで、Flash Lite バージョンの .zip ファイルを探してダウンロードおよび解凍し、"Samples" フォルダを選択して、hi_hat.mid という名前のファイルを選択します。



- [OK] をクリックします。
13. [制御]-[ムービープレビュー] を選択して、Adobe Device Central エミュレータを起動し、SWF ファイルをテストします。
- エミュレータで、下矢印キーを押して [Play] ボタンにフォーカスを移動し、選択キーを押してサウンドを再生します。

Flash Lite 1.0 でのデバイスサウンドのトリガ

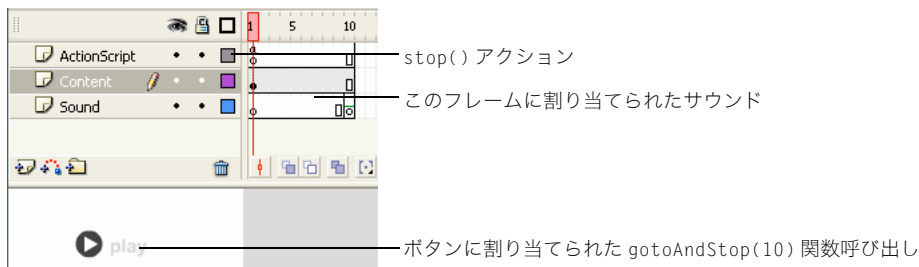
Flash Lite 1.0 では、ユーザーがデバイスでキーを押したときにのみデバイスサウンドが再生されます。この制約を満たすには、2つの方法があります。1つは、ボタンシンボルのタイムラインで [ダウン] フレームにサウンドを割り当てる方法です。ボタンにフォーカスがあるときに、ユーザーがデバイスの選択キーを押すと、ボタンの [ダウン] フレームでサウンドが再生されます。この手法の例については、[51 ページの「ボタンへのデバイスサウンドの追加」](#)を参照してください。

Flash Lite 1.0 でデバイスサウンドをトリガするもう1つの方法は、タイムラインのデバイスサウンドを含んでいるフレームに再生ヘッドを送るキーをユーザーに押させる方法です。再生ヘッドがそのフレーム内に移動したときに、そのフレームにあるすべてのデバイスサウンドが再生されます。

たとえば、フレーム 1 のステージにボタンがあり、フレーム 10 にデバイスサウンドが割り当てられているとします。次の on(press) ハンドラをボタンインスタンスに割り当てます。

```
on(press) {  
    gotoAndStop(10);  
}
```

次の図は、Flash 上にアプリケーションを表示した場合の例を示しています。



ユーザーがボタンを選択すると、フレーム 10 のサウンドが再生されます。この手法では、デバイスサウンドが gotoAndPlay() 関数で指定されたフレームと同じフレームに割り当てられている必要があります。たとえば、前の例で、サウンドがフレーム 10 ではなく、フレーム 11 に割り当てられている場合、再生ヘッドがフレーム 11 に移動してもサウンドは再生されません。

コンパウンドサウンドの使用

Flash Lite 1.1 では、複数形式のデバイス依存サウンドを単一のタグ付きデータブロック内にカプセル化して扱うことができます。これにより、コンテンツ開発者は複数種類のデバイスに対応したコンテンツを単一のパッケージで提供できます。例として、1つの Flash アプリケーションに同じサウンドの MIDI と MFi の両方の形式を含めることができます。この Flash アプリケーションは、MIDI のみをサポートするデバイスでも、MFi のみをサポートするデバイスでも再生可能です。それぞれのデバイスが本来サポートしている特定形式のサウンドが再生されます。

Flash Lite Sound Bundler と呼ばれるユーティリティを使用して、複数のデバイスサウンドを単一のサウンドバンドル (FLS) ファイルにバンドルすることができます。その FLS ファイルを Flash Lite ドキュメントに追加する方法は、標準のデバイスサウンドの場合と同じです。最初に、プロキシサウンドを Flash ドキュメント内に読み込み、次にパブリッシュ時にプロキシサウンドを置き換える FLS ファイルを指定します。Flash Lite アプリケーションへのデバイスサウンドの追加の詳細については、[51 ページの「ポタンへのデバイスサウンドの追加」](#)を参照してください。

X
#

このマニュアルの作成時点では、Sound Bundler ユーティリティは Windows システムでのみサポートされています。

コンパウンドサウンド (FSL) ファイルを作成するには：

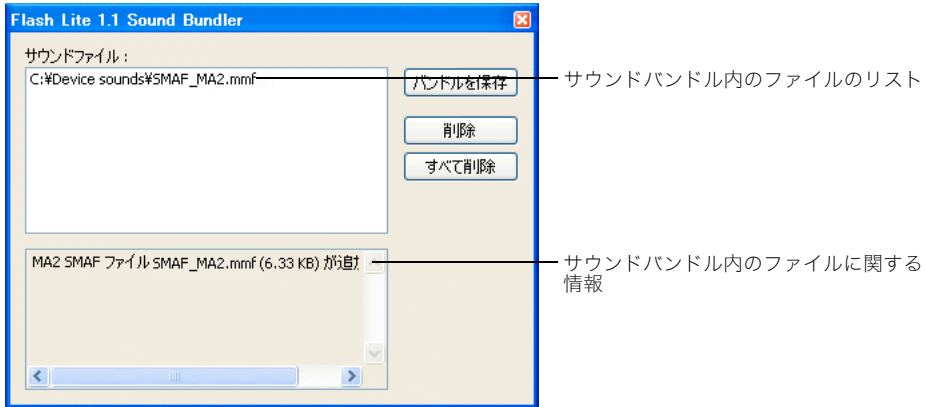
1. www.adobe.com/go/developer_flashlite_jp にある Flash Lite Sound Bundler アプリケーション (FlashLiteBundler.exe) を探してダウンロードします。

Sound Bundler がフローティングウィンドウに表示されます。

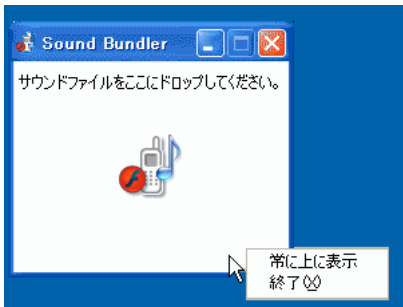


2. デスクトップから、バンドルする最初のサウンドファイルをフローティングウィンドウ内にドラッグします。

[Flash Lite 1.1 Sound Bundler] ウィンドウが表示されます。Sound Bundler ウィンドウの上部は、サウンドバンドルに追加したファイルのリストです。ウィンドウの下部には、サウンド形式、サウンドデータのサイズファイル名など、サウンドバンドル内のサウンドに関する情報が表示されます。



3. バンドルする残りのサウンドファイルをフローティングウィンドウ内にドラッグします。
複数のファイルを同一のオーディオ形式でバンドルすることはできません。たとえば、2つの MIDI ファイルを同じ FLS ファイル内にバンドルすることはできません。
4. サウンドバンドルからファイルを削除するには、サウンドファイルのリストでファイルを選択し、[削除]をクリックします。サウンドバンドル内のすべてのファイルを削除するには、[すべて削除]をクリックします。
5. [バンドルを保存]をクリックして FLS ファイルを保存します。
6. Sound Bundler を終了するには、Sound Bundler ウィンドウを右クリックして [終了] を選択します。



次の手順では、サウンドバンドル (FLS) ファイルを Flash ドキュメントに追加します。操作手順は Flash ドキュメントに標準のデバイスサウンドを追加する手順と同じですが、プロキシサウンドを置き換える単一のデバイスサウンドファイルを指定する代わりに、作成した FLS ファイルを指定する点が異なります。詳細については、[50 ページの「デバイスサウンドの使用」](#)を参照してください。

Flash ネイティブサウンドの使用

Flash Lite 1.1 では、Flash Lite 1.0 でサポートされているデバイスサウンド形式に加えて、Flash の標準、つまりネイティブサウンドをサポートしています。Flash Lite 1.1 のネイティブサウンドは、イベントサウンドまたは同期サウンドのいずれかとして再生できます ([50 ページの「イベントおよびストリーム \(同期した\) サウンド」](#)を参照してください)。

ネイティブサウンドを Flash Lite 1.1 のコンテンツに追加する一般的なワークフローは、Flash デスクトップアプリケーションの場合と同じですが、次の例外があります。

- Flash Lite は外部の MP3 ファイルの読み込みをサポートしていません。
- Flash Lite は Sound オブジェクトをサポートしていません。
- Flash Lite はスピーチ圧縮オプションをサポートしていません (『Flash ユーザーガイド』の「書き出し時のサウンドの圧縮」を参照してください)。

Flash でのネイティブサウンドの操作の詳細については、『Flash ユーザーガイド』で次のトピックを参照してください。

- 『Flash ユーザーガイド』の「サウンドの読み込み」
- 『Flash ユーザーガイド』の「ドキュメントへのサウンドの追加」
- 『Flash ユーザーガイド』の「キーフレームでのサウンドの開始および停止」
- 『Flash ユーザーガイド』の「書き出し時のサウンドの圧縮」

コンテンツのパフォーマンスと ファイルサイズの最適化

Flash コンテンツを最適化するには、基本的な原則に忠実である必要があります。たとえば、かつての Flash 開発者は、非常に複雑なアートワーク、必要以上のトゥーン、透明効果の過度の使用を避けなければならない場合がよくありました。

これらデスクトップのパフォーマンス上の問題は、初期のバージョンの Flash でほとんど解決されましたが、Adobe Flash Lite の開発においてはモバイルデバイスの制限があるため、現在もお問題点に直面することがあります。パフォーマンスはデバイスによって異なりますが、ときには劇的に異なる場合もあります。モバイルオーサリングに際しては、多種多様なデバイスにパブリッシュすることが多いため、開発者は対象デバイスの共通項を対象にオーサリングを行う必要があります。

モバイルコンテンツの最適化には、たいいてい交換条件が要求されます。たとえば、ある技術を使うと見栄えがよいが、パフォーマンスは別の方が優れているといった具合です。交換条件をよく計算した上で、エミュレータとターゲットデバイスでのテストを交互に何度も繰り返してください。

Flash の以前のバージョンでは、Flash Lite 1.x のエミュレータは Flash オーサリング環境に属していましたが、Flash CS3 では、エミュレータは Adobe® Device Central CS3 の一機能となっています。Device Central で、Flash Lite プロジェクトのさまざまなデバイスに対するシミュレーションを行い、デバイスの表示、メモリの使用、個々のデバイスでのパフォーマンスを確認することができます。エミュレータを使用して Flash Lite コンテンツをモバイルデバイス用に最適化する方法の詳細については、Device Central マニュアルの「Best practices for content on mobile devices」を参照してください。

Flash Lite コンテンツのプレビュー

Adobe Flash CS3 Professional には Adobe Device Central CS3 で使用可能な Adobe Flash Lite エミュレータが搭載されており、アプリケーションを実機で表示して使用した場合の動作を、オーサリングツールでプレビューすることができます。エミュレータでのアプリケーションの動作に満足したら、アプリケーションを実機で試すことができます。この章では、Flash で使用できる Flash Lite のプレビューおよびデバッグ機能について説明します。

この章では、次のトピックについて説明します。

Flash Lite のプレビュー機能の概要	61
エミュレータでサポートされていないテスト機能	62
エミュレータの使用	62
Flash Lite コンテンツタイプ	65
エラー	69
プラットフォームの機能の確認	71

Flash Lite のプレビュー機能の概要

Flash CS3 Professional の Flash Lite テスト機能は、デバイスプロファイルの膨大なデータベースとデバイスエミュレータを持つ Adobe Device Central の一機能です。Device Central は、Adobe Creative Suite®、Adobe Dreamweaver® など、他のさまざまなアドビ製品でも使用できます。

Adobe Device Central エミュレータでは、Flash オーサリングツール内で Flash Lite コンテンツをプレビューできます。エミュレータは、ターゲットデバイスの動作と外観に合わせて設定されています。Device Central インターフェイスを使用してターゲットデバイスを選択、管理し、アプリケーションのターゲット Flash Lite コンテンツタイプ (着信音、ブラウザ、スタンドアロンアプリケーションなど) を指定します。Device Central では、テストデバイスおよび Flash Lite コンテンツタイプの組み合わせで定義されるデバイス設定により、アプリケーションで使用可能な機能が指定されます。たとえば、サポートされるオーディオフォーマットやネットワーク接続を確立する機能などが指定されます。Flash Lite コンテンツタイプの詳細については、[65 ページの「Flash Lite コンテンツタイプ」](#)を参照してください。

エミュレータの操作方法などの詳細については、Device Central ヘルプを参照してください。

エミュレータでサポートされていないテスト機能

Adobe Device Central エミュレータでは、標準の (デスクトップ) テストウィンドウで使用可能なすべての機能をサポートしているわけではありません。デスクトップ Flash テストウィンドウで使用でき、Adobe Device Central エミュレータでは使用できないテスト機能の一覧を次に示します。

- 変数のリストアップ ([デバッグ]-[変数のリストアップ]) およびオブジェクトのリストアップ ([デバッグ]-[オブジェクトのリストアップ]) 機能
- プロファイラ、ストリーミング、およびフレーム毎データ量の各機能
- [表示]-[ダウンロードのシミュレート] メニューコマンド
- ActionScript デバッガ
- [表示]-[再描画する領域を表示] メニューコマンド
- [制御パネル] ツールバー ([ウィンドウ]-[ツールバー]-[制御パネル])

エミュレータの使用

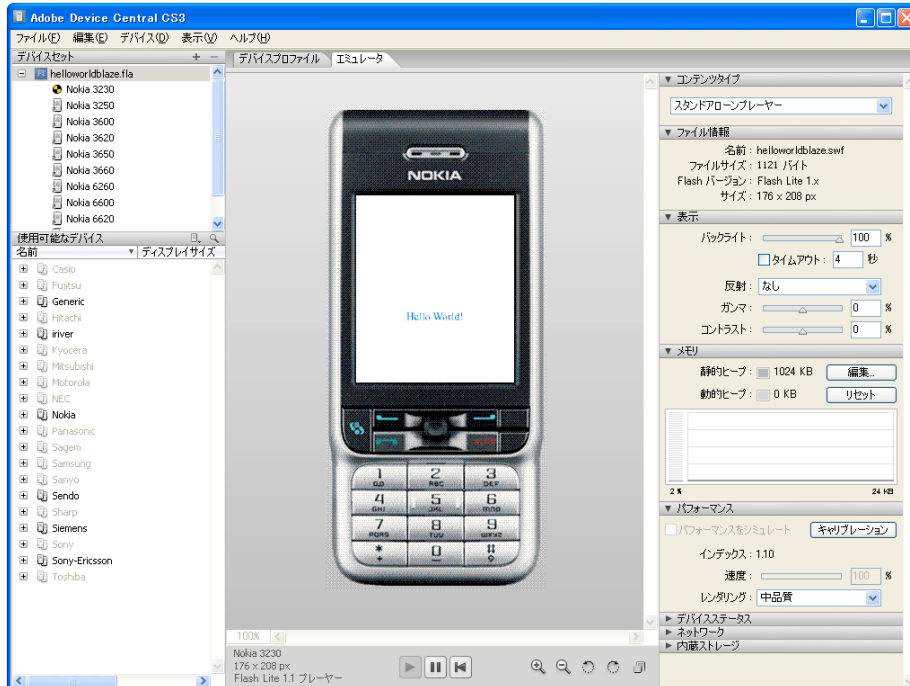
エミュレータを開始するには、Flash デスクトップコンテンツをプレビューするときと同じ方法で、Flash オーサリングツールで [制御]-[ムービープレビュー] を選択します。ただし、Adobe Device Central は、Flash デスクトップコンテンツのテストウィンドウとは外観も機能も異なります。

この項では、次のトピックについて説明します。

- [62 ページの「エミュレータでサポートされていないテスト機能」](#)
- [63 ページの「Adobe Device Central の起動」](#)
- [64 ページの「エミュレータのデバッグオプションの設定」](#)
- [69 ページの「スクリーンサイズと使用可能なステージサイズ」](#)
- [69 ページの「エラー」](#)

Adobe Device Central の起動

[制御]-[ムービープレビュー]を選択するか、Ctrl+Enter を押して Device Central を開始すると、Flash が SWF を Device Central に書き出す進行状況を示すプログレスバーが表示されます。書き出しが完了すると、Device Central が起動してエミュレータがアクティブになり、SWF がロードされます。



Device Central のデバイスセット一覧には、アプリケーションと共に保存されたすべてのターゲットデバイスが表示されます。デフォルトでは、セットの最初のデバイスがシミュレーション対象として選択されます。選択されるコンテンツタイプは、アプリケーションの作成時に一緒に保存されたコンテンツタイプです。Flash Lite コンテンツタイプの詳細については、65 ページの「Flash Lite コンテンツタイプ」を参照してください。

アプリケーションは、さまざまなコンテンツタイプとデバイスでテストできます。コンテンツタイプを変更したり、デバイスセットからデバイスを追加したり削除すると、Flash のデバイス設定も自動的に変更されます。

別のデバイスでコンテンツがどのように表示されるかをテストするには、上部または下部の一覧で別のデバイスをダブルクリックします。テスト中のデバイスの隣に回転アイコンが表示され、選択したデバイスでのデバイス実行シミュレーションに変わります。

エミュレータのデバッグオプションの設定

Adobe Device Central エミュレータでは、コンテンツの実行中にデバッグメッセージを [Flash 出力] パネルに送信できます。また、[出力] パネルのポップアップにも同じメッセージが表示されます。

エミュレータでは次のような種類の情報がレポートされます。

トレースメッセージ。 Flash Lite アプリケーション内で `trace()` 関数呼び出しによって生成されます。`trace()` の使用の詳細については、『Flash Lite 1.x ActionScript リファレンスガイド』の「`trace()`」を参照してください。

情報メッセージ。 選択したテストデバイスや SWF ファイルサイズなどに関する一般的な情報です。これらのメッセージは、エミュレータの [警告] パネルに表示されます。

警告メッセージ。 再生に影響する可能性がある、Flash Lite コンテンツの問題に関する情報です。

エミュレータで生成する情報の種類は、次の方法で絞り込むことができます。

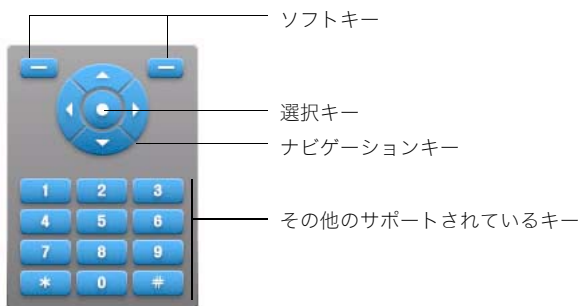
Flash Lite の出力オプションを設定するには：

1. [制御]-[ムービープレビュー]を選択します。アプリケーションが Adobe Device Central に書き出され、エミュレータに表示されます。
2. [表示]-[Flash 出力]-[Show in Device Central]を選択します。
 - [トレース] オプションを選択または選択解除します。
 - [情報] オプションを選択または選択解除します。
 - [警告] オプションを選択または選択解除します。

エミュレータの操作

コンピュータのマウスまたはキーボードショートカットを使用して、エミュレータのキーを操作します。エミュレータでは次のキーを操作できます。

- 数値キー (0～9)、アスタリスク (*) キー、およびシャープ (#) キー
- ナビゲーションキー (上、下、左、右、および選択)
- 左右のソフトキー



マウスを使用してエミュレータのキーを直接クリックするか、次の対応するキーボードショートカットを使用することができます。

- キーボードの方向キー(上、下、左、右)は、エミュレータのナビゲーションキーと対応します。
- Enter キーまたは Return キーは、エミュレータの選択キーに対応します。
- PageUp キーと PageDown キーは、それぞれエミュレータの左と右のソフトキーに対応します。
- キーボードの数値キーは、エミュレータの数値キーと対応します。

アプリケーションをテストする際のエミュレータの操作方法の詳細については、Adobe Device Central のオンラインヘルプを参照してください。

Flash Lite コンテンツタイプ

『Flash Lite 1.x ファーストステップガイド』の「Flash Lite コンテンツタイプについて」で説明されているとおり、Flash Lite はさまざまなデバイスにインストールされます。1つの Flash Lite インストール環境では1つまたは複数のアプリケーションモードがサポートされます。それらのモードを「コンテンツタイプ」といいます。たとえば、Flash Lite を使用してスクリーンセーバーや着信アニメーションを実現するデバイスや、モバイル向け Web ページに埋め込まれた Flash コンテンツをレンダリングするデバイスがあります。

次の表は、このマニュアルの作成時点で利用可能なすべての Flash Lite コンテンツタイプとその説明の一覧です。『Flash Lite 1.x ファーストステップガイド』の「Flash Lite 1.x 対応デバイス」で説明されているように、コンテンツタイプの多くは特定の地域または特定の通信事業者でのみ利用できます。表の右端の欄は、各コンテンツタイプを使用できる地域および通信事業者を示します。利用可能な Flash Lite コンテンツタイプに関する最新情報については、www.adobe.com/go/mobile_supported_devices_jp/にある「Flash Lite サポート対象の携帯デバイス」のページを参照してください。

Flash Lite は、次のコンテンツタイプをサポートしています。

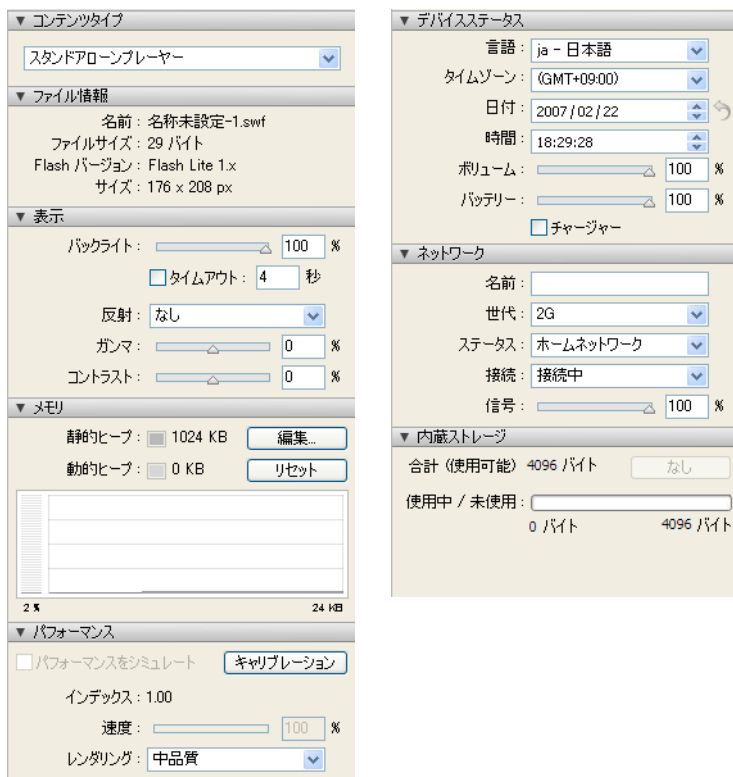
Flash Lite コンテンツタイプ	説明	利用状況
アドレス帳	デバイスのアドレス帳アプリケーションの項目と SWF ファイルを関連付けることができます。	NTT DoCoMo およびボーダフォン(日本のみ)
アラーム	デバイスのアラームとして再生する SWF ファイルを選択できます。	KDDI およびボーダフォン(日本のみ)
ブラウザ	モバイル Web ページに埋め込んでデバイスの Web ブラウザで表示する Flash コンテンツをレンダリングします。	NTT DoCoMo、KDDI、およびボーダフォン(日本のみ)

Flash Lite コンテンツタイプ	説明	利用状況
呼び出し履歴	ユーザーのアドレス帳の各エントリに関連付けられているイメージやアニメーションを、名前や電話番号と共に表示します。	KDDI (カシオ製の機種のみ)
呼び出し画面	着信時にアニメーションを表示します。	NTT DoCoMo および KDDI (日本のみ)
着 Flash	着信音として再生する SWF ファイルを選択できます。	KDDI (日本のみ)
データボックス	デバイスのデータボックスアプリケーションで Flash コンテンツをレンダリングします。これにより、ユーザーはデバイス上のマルチメディアファイルを管理およびプレビューできます。	NTT DoCoMo、KDDI、およびボーダフォン (日本のみ)
データフォルダ	デバイスのデータフォルダアプリケーションで Flash コンテンツをレンダリングします。これにより、ユーザーはデバイス上のマルチメディアファイルを管理およびプレビューできます。	KDDI (日本のみ)
アイコンメニュー	デバイスのランチャーアプリケーションのカスタムアイコンメニューを選択できます。	KDDI (カシオ製の機種のみ)
イメージビューア	イメージビューアアプリケーションを使用して、ユーザーは SWF ファイルを含む、デバイス上のマルチメディアファイルを管理およびプレビューできます。	NTT DoCoMo (日本のみ)
着信コール	着信時にアニメーションを表示します。	NTT DoCoMo、KDDI、およびボーダフォン (日本のみ)
メーラー	電子メールメッセージの送受信時にアニメーションを表示します。	ボーダフォン (日本のみ)
マルチメディア	SWF ファイル (およびその他のマルチメディア形式) をプレビューします。	KDDI (日本のみ)
マイピクチャ	マイピクチャアプリケーションを使用して、ユーザーはデバイス上の SWF ファイルおよびその他のイメージ形式を管理およびプレビューできます。	NTT DoCoMo (日本のみ)
OpenEMIRO	デバイスがスタンバイモードから復帰したときに Flash Lite コンテンツを表示します。これは、他のデバイスのウェイクアップ画面のコンテンツタイプに似ています。	KDDI (カシオ製の機種のみ)

Flash Lite コンテンツタイプ	説明	利用状況
スクリーンセーバー	デバイスのスクリーンセーバーを表示します。	KDDI およびボーダフォン (日本のみ)
SMIL プレーヤー	SWF ファイル (およびその他のマルチメディア形式) をプレビューします。	KDDI (日本のみ)
スタンドアローンプレーヤー	Flash Lite をスタンドアローンアプリケーションとして使用して、ユーザーがデバイスに保存されている SWF ファイルや受信ボックスで受信した SWF ファイルを起動して表示できるようにします。	Symbian Series 60 および UIQ の限定デバイスでグローバルに使用可能
待受画面	デバイスの待受画面 (または壁紙画面) を表示します。	NTT DoCoMo および KDDI (日本のみ)
サブ LCD	一部の折りたたみ型携帯端末の外側にあるサブ画面にコンテンツを表示します。	KDDI (日本のみ)
ウェイクアップ画面	電話機の起動時にアニメーションを表示します。	NTT DoCoMo (日本のみ)

エミュレータの Flash Lite に関する情報

エミュレータには、Flash Lite アプリケーション固有の情報を提供するパネルがあります。パネルはウィンドウの右側に表示され、Flash と同じように畳んだり展開することができます。



- [コンテンツタイプ] パネルには、アプリケーションのデフォルトのコンテンツタイプが表示されますが、対応する別のコンテンツタイプを選択することもできます。
- [ファイル情報] パネルには、ファイル名、Flash のファイルバージョン、寸法、サイズ (キロバイト) が表示されます。
- [警告] パネルは問題が発生すると開き、警告メッセージが表示されます。
- [メモリ] と [パフォーマンス] パネルには、スタティックメモリとダイナミックメモリのサイズが表示され、さまざまなパラメータを調整してアプリケーションのパフォーマンスを向上させることができます。
- [デバイスのステータス] パネルには、タイムゾーンやバッテリーレベルなど、さまざまなプラットフォーム固有の (`fscommand()`) 設定が表示されます。
- [ネットワーク] パネルには、ネットワーク接続に関する情報が表示されます。

- [内蔵ストレージ] パネルには、デバイスの記憶域で使用された容量が表示されます。これはデバイスごとの値です。つまり、複数のアプリケーションが固定記憶域に書き込む場合、値はすべてのデータの合計となります。デバイスで実行していたすべてのコンテンツの固定オブジェクトを削除するには、[なし] をクリックしてそのデバイスの記憶域を消去します。

各パネルとその使用方法の詳細については、Adobe Device Central のオンラインヘルプを参照してください。

スクリーンサイズと使用可能なステージサイズ

ターゲットデバイスと Flash Lite コンテンツタイプの組み合わせによって、Flash Lite アプリケーションで使用できるスクリーンの領域が決まります。使用可能なステージ領域は、デバイスのフルスクリーンサイズ以下です。

たとえば、Nokia Series 60 デバイスのスタンドアロンプレーヤーで、SWF ファイルをフルスクリーンモードで実行する場合、使用可能なステージ領域はデバイスのフルスクリーンサイズ (176 x 208 ピクセル) と同じです。その他のデバイス (日本で販売されているデバイスなど) では、特殊なコンテンツタイプ (アドレス帳やスクリーンセーバーなど) で実行される SWF ファイルで使用可能なステージ領域は、デバイスのフルスクリーンサイズよりも小さい場合があります。たとえば、Fujitsu 700i のスクリーンサイズは 240 x 320 ピクセルですが、このデバイスのアドレス帳アプリケーションで実行される SWF ファイルで使用可能なステージ領域は 96 x 72 ピクセルです。

SWF ファイルの実際の寸法が使用可能なステージの寸法と異なる場合は、使用可能なステージ領域に収まるように、Flash Lite プレーヤーによってコンテンツが (縦横比を保って) 伸縮されます。エミュレータでコンテンツをプレビューする場合、アプリケーションのステージサイズが使用可能なステージ領域と異なるときにも警告が表示されます。

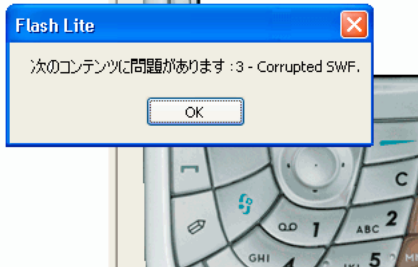
意図しない伸縮の問題を回避するために、Flash ドキュメントのステージサイズを、選択したテストデバイスやコンテンツタイプで使用可能なステージ領域と一致させることをお勧めします。

エラー

Adobe Device Central エミュレータでは、コンテンツのテスト中に警告を生成することができます。エミュレータにのみ表示され、実際エラーまたはエラーの可能性に関する情報提供を目的とした警告と、エミュレータと実際のデバイスに表示される種類の警告があります。

最初の種類の警告は、SWF ファイルに関するデバッグ情報です。たとえば、Flash Lite (または現在選択しているテストデバイスで使用可能な Flash Lite のバージョン) でサポートされていない ActionScript が SWF ファイルに含まれている場合には警告が生成されます。

エミュレータに表示されるもう1つの種類のメッセージは、実際のデバイスにも表示されます。この種類のエラーはエラーダイアログボックスに表示され、アプリケーションを続行するためにユーザーはエラーダイアログボックスを閉じる必要があります。次の図は、エミュレータに表示されたエラーダイアログボックスの例を示しています。



デバイスでは、エラーダイアログボックスには、"次のコンテンツに問題があります" という文字列に続けてエラー番号が表示されます。エミュレータのエラーダイアログボックスには、簡単なエラーの文字列も表示されます。エミュレータの[出力]パネルには、エラーの詳しい説明が表示されます。

次の表は、Flash Lite プレーヤーで発生するすべてのエラーと、エラー番号、エラーダイアログボックスに表示される簡単な説明、および[出力]パネルに表示される詳しい説明の一覧です。

エラー番号	エラー文字列	説明および考えられる原因
1	メモリ不足です。	エミュレータのヒープメモリが不足しています。特に指定しない限り、エミュレータは SWF ファイルで使用するために 1MB のメモリを割り当てます。
2	スタックの限界に達しました。	エミュレータはスタックの限界に到達したか、または限界を超過したことを検出しました。これには、複数レベルのネストされたムービークリップや複雑なベクターの描画など、さまざまな原因が考えられます。
3	SWF が破損しています。	エミュレータは SWF データが壊れていることを検出しました。
4	ActionScript が停止しました。	エミュレータは、SWF ファイル内の特定の ActionScript コードの実行に時間がかかりすぎていることを検出しました。その結果、エミュレータは ActionScript コードの実行を停止しました。
5	ActionScript を処理しています。	エミュレータは、存在しないムービークリップへの参照など、ActionScript エラーを検出しました。
6	ActionScript の無限ループです。	エミュレータは、無限ループまたは深くネストされた ActionScript (深くネストされた if..else ステートメントなど) を検出しました。

エラー番号	エラー文字列	説明および考えられる原因
7	無効なフレームバッファです。	エミュレータは無効なフレームバッファを検出しました。
8	無効な矩形の表示です。	エミュレータは無効な表示矩形を検出しました。
9	無効なフレーム番号です。	エミュレータは、SWF ファイルが無効なフレーム番号に移動または無効なフレーム番号を解決しようとしたことを検出しました。
10	無効なキーです。	エミュレータは無効なキー入力を検出しました。
11	不正な JPEG データです。	エミュレータは、SWF ファイル内の JPEG または PNG データが壊れているか、JPEG データをデコードするのに十分なメモリがないか、JPEG データの形式がサポートされていないことを検出しました。
12	不正なサウンドデータです。	エミュレータは、SWF ファイルにサポートされていないサウンドデータ形式が含まれていることを検出しました。
13	ルートムービーがアンロードされました。	エミュレータは、ルートムービーがアンロードされ、別の SWF ファイルに置き換えられていないことを検出しました。

プラットフォームの機能の確認

ターゲットデバイスおよび Flash Lite コンテンツタイプの各組み合わせにより、ナビゲーションタイプ、サポートされるデバイスサウンドフォーマット、テキスト入力サポートなどの Flash Lite 機能のセットが定義されます。エミュレータの設定ペインで [情報] デバッグオプションを有効にすると、現在選択されているデバイスおよびコンテンツタイプに使用できるプラットフォーム機能が一覧表示されます。デバッグオプションの設定の詳細については、[64 ページの「エミュレータのデバッグオプションの設定」](#)を参照してください。

次の表は、エミュレータでアプリケーションをプレビューするときに [出力] パネルに表示される Flash Lite のプラットフォーム機能の一覧です。

機能名	説明と設定可能な値
DeviceSoundKeyOnly	デバイスが、ユーザーがデバイスのキーを押したときにのみデバイスサウンドを再生するか (Yes)、ユーザーアクションとは独立してデバイスサウンドを再生するか (No) を示します。詳細については、 54 ページの「Flash Lite 1.0 でのデバイスサウンドのトリガ」 を参照してください。
DeviceSoundsOrdered	プラットフォームでサポートされるデバイスサウンドフォーマットのカンマ区切りリストです。サウンドフォーマットの順番は、SWF ファイル内のコンパウンドサウンドファイルが複数のサウンドフォーマットで構成されている場合に、Flash Lite で再生するサウンドの順番を示します。サウンドバンドルの詳細については、 55 ページの「コンパウンドサウンドの使用」 を参照してください。
FSCCommand	Flash Lite で fscommand() または fscommand2() 関数呼び出しを処理する頻度を示します。有効な値は次のとおりです。 OnePerKey: キーを押すたびに fscommand() 呼び出しが1つだけ許可されます。 OnePerKeyPerFrame: イベントハンドラまたはフレームごとに fscommand() 呼び出しが1つだけ許可されます。 All: fscommand() を呼び出す頻度に制限はありません。 None: fscommand() 関数はサポートされていません。
InputText	プラットフォームがテキスト入力をサポートしているか (Yes)、またはサポートしていないか (No) を示します。テキスト入力の詳細については、 36 ページの「テキスト入力フィールドの使用」 を参照してください。
LoadMovie	Flash Lite で loadMovie() 関数呼び出しを処理する頻度を示します。有効な値は次のとおりです。 OnePerKey: キーを押すたびに loadMovie() 呼び出しが1つだけ許可されます。 OnePerKeyPerFrame: イベントハンドラまたはフレームごとに loadMovie() 呼び出しが1つだけ許可されます。 All: loadMovie() を呼び出す頻度に制限はありません。 None: loadMovie() 関数はサポートされていません。
LoadVars	Flash Lite で loadVariables() 関数呼び出しを処理する頻度を示します。有効な値は次のとおりです。 OnePerKey: キーを押すたびに loadVariables() 呼び出しが1つだけ許可されます。 OnePerKeyPerFrame: イベントハンドラまたはフレームごとに loadVariables() 呼び出しが1つだけ許可されます。 All: loadVariables() を呼び出す頻度に制限はありません。 None: loadVariables() 関数はサポートされていません。

機能名	説明と設定可能な値
Loop	SWF コンテンツがタイムラインの最後に到達したときにループする (タイムラインの最初のフレームに戻る) か (Yes)、最後のフレームで停止するか (No) を示します。
MultipleDeviceSound	デバイスが複数のデバイスサウンドのミキシングをサポートしているか (Yes)、またはサポートしていないか (No) を示します。
NativeSounds	Flash Lite プレーヤーでネイティブに再生できるサウンドフォーマット (Flash Lite プレーヤーからデバイスに渡されて再生されるデバイスサウンドと対比) の順不同のリストです。有効な値は、 NativeSound_PCM 、 NativeSound_ADPCM 、および NativeSound_MP3 です。
NavigationType	プラットフォームでサポートされるナビゲーションモードとして、2 方向、4 方向、または折り返し付き 4 方向を示します。ナビゲーションモードの詳細については、 11 ページの「タブナビゲーションのモード」 を参照してください。有効な値は次のとおりです。 2Way : 上下の方向キーのみがサポートされます。 4Way : 4 つすべての方向キー (上下左右) がサポートされます。 4WayWrapAround : フォーカスがディスプレイのトップに戻る点を除いて、4Way と同じです。
SMS	Flash Lite が SMS メッセージの送信をサポートしているか (Yes)、またはサポートしていないか (No) を示します。
getUrl	Flash Lite で <code>getUrl()</code> 関数呼び出しを処理する頻度を示します。有効な値は次のとおりです。 OnePerKey : キーを押すたびに <code>getUrl()</code> 呼び出しが 1 つだけ許可されます。 OnePerKeyPerFrame : イベントハンドラまたはフレームごとに <code>getUrl()</code> 呼び出しが 1 つだけ許可されます。 All : <code>getUrl()</code> を呼び出す頻度に制限はありません。 None : <code>getUrl()</code> 関数はサポートされていません。
keySet	デバイスの Flash Lite でサポートされるキーイベントを示します。キーイベント処理の詳細については、 15 ページの「キーイベントの処理」 を参照してください。有効な値は次のとおりです。 All : すべてのキーイベントが処理されます。 Phone : 0 ~ 9、#、*、選択、および 4 方向ナビゲーションキーと関連付けられたイベントのみが処理されます。

機能名	説明と設定可能な値
mouseType	<p>Flash Lite でサポートされるマウスイベントを示します。有効な値は次のとおりです。</p> <p>None: マウスイベントはサポートされません。</p> <p>Partial: press、release、rollOver、および rollOut イベントはサポートされています。releaseOutside、dragOut、および dragOver イベントはサポートされていません。</p> <p>Mouse: Mouse Up/Mouse Down/Mouse Move メッセージが処理されます。1つの例は仮想カーソル機能を備えた NTT DoCoMo 電話機です。これは、マウスの移動が rollOver/rollOut イベントをトリガすることを意味します。</p>
soundEnabled	<p>デバイスでサウンドが有効になっているか (Yes)、有効になっていないか (No) を示します。</p>

索引

A

Adobe Device Central エミュレータ
デバッグオプション 64

F

Flash Lite エミュレータ
アプリケーションのプレビュー 62
警告メッセージとエラーメッセージ 69
サポートされていない機能 62
操作 64

Flash Lite のコンテンツタイプ、説明 65

Flash Lite のレンダリング品質
テキストフィールド 41
デフォルトのレンダリング品質 41

K

keypress イベント
ActionScript キーコード 15
ActionScript による処理 15
キーキャッチャーボタンの作成 16
キーハンドラスクリプトの作成 15
サポートされるキー 10

S

Sound Bundler ユーティリティ、複合サウンドでの使用
55

あ

アプリケーションモード、Flash Lite 65

い

インタラクティブ機能
keypress イベントの処理 15
キーの押下動作の検出 16
作成、ボタン 22
ソフトキーの使用 28
タブナビゲーション 11
ボタンを使用したメニューの作成 24
ムービークリップによるメニューの作成 18

う

埋め込み、フォントアウトライン
説明 42
方法 42

さ

サウンド
Sound Bundler ユーティリティ 55
イベントおよびストリーム 50
説明 49
デバイス 50
デバイスサウンドとネイティブサウンド 49
複合 55

す

ステージ、スクリーンサイズ 69

そ

ソフトキー
SetSoftKeys コマンド 28
使用 28

た

タブナビゲーション

- 2方向 12
- 4方向 12
- アプリケーション例 24
- 折り返し付きの4方向 12
- ガイドライン 14
- 説明 11
- フォーカス矩形 13
- モード 11

て

テキスト入力フィールド

- サンプルアプリケーション 43
- フォーカス矩形 39
- 文字の制限 39

テキストフィールド

- スクロールテキストの作成 46
- テキスト入力フィールドの文字の制限 39
- テキスト入力フィールド、使用 36
- フォントのレンダリング方法、説明 34

デバイスサウンド

- Flash Lite 1.0でのトリガ 54
- 説明 50
- 追加 51

な

ナビゲーション、「タブナビゲーション」を参照

ね

ネイティブサウンド、概要 57

ふ

フォントのレンダリング方法

- 説明 40
- テキストフィールドへの適用 40
- プラットフォーム機能、決定 71

ほ

ボタンイベント

- 処理 22
- 説明 22

め

メニュー

- ボタンを使用した作成 24
- ムービークリップによる作成 18