Cocoa+Java~Mac OS X 10.2対応への覚書

2002年11月11日 赤松正行

はじめに

◆本稿について

本稿は、「Mac OS Xプログラミング入門」シリーズの書籍「Cocoa+Java〜Mac OS X 10.1対応版」 (広文社刊、ISBN4-87778-093-9)を元に、Mac OS X 10.2において変更や追加があった事項を説明 している。各事項に対応する同書のページ番号を括弧内に記している。

◆Mac OS X 10.2について

(p.4) コードネーム「Jaguar」としても知られるMac OS X 10.2は、2002年8月24日(米国時間)に 発売され、数多くの新機能の搭載や堅実な改良が行われている。Mac OS X 10.2には、従来に引き続き開 発用ツールが付属しており、OSを購入するだけで開発を行うことができる。本稿で使用したおもな開発 ツールのバージョンは以下の通りである。

Project Builder $\mathcal{N} - \mathcal{V} \exists \mathcal{V} 2.0.1$ (July 2002 Developer Tools) Interface Builder $\mathcal{N} - \mathcal{V} \exists \mathcal{V} 2.3$ (v283)

第2章 アプリケーション開発の実際

◆2.2 Developer Toolsのインストール

(p.19) 「Developer Tools」CD-ROMの内容は10.1とほぼ同じだが、一部のファイル名が異なっている。

	💿 Developer Tools	C
(·· ··) []]	📼 📙 🏦 💚	A
戻る 進む 表示	コンピュータ ホーム よく使う項目 アブ	リケーション
×	6 項目、4 MB 空き	
About AppleScript St	rudio About Developer Tools What's	New
Carbon Lib CDK		
CarbonLib SDK	Developer.mpkg Packa	ges

図2-001 「Developer Tools」CD-ROMの内容

(p.20) インストーラのファイル名は「Developer.mpkg」となり、拡張子が変わった。

(p.21) インストール時には管理者パスワードの入力が求められなくなった。

◆2.3 Project Builderの設定

(p.28) Project Builderの設定アシスタントで、エディタウィンドウの設定に変わって、ウィンドウ環境の設定を行うようになった。ウィンドウの取り扱いについて、3種類の方法からいずれかを選ぶ。



図2-002 ウィンドウ環境の設定

(p.28) Project Builderの設定アシスタントで、ファイル管理の設定が加わった。プロジェクトやプロ ジェクトウィンドウを閉じる時の動作を設定する。

ファイル管理	里
☑ 閉じるときにプロジェク	トウインドウの状態を保存する
プロジェクトを閉じるときに、 記憶させるには、このチェック	開いているすべてのウインドウとその内容をプロジェクトに ボックスをチェックします。
☑ プロジェクトウインドウ	を閉じるときにプロジェクト全体を閉じる
プロジェクトウインドウを閉じ じるには、このチェックボック	るときに、プロジェクトに関連するすべてのウインドウを閉 スをチェックします。
(キャンセル)	前へ 完了

図2-003 ファイル管理の設定

◆2.4 新規プロジェクトの作成

(p.29) 新規プロジェクトのアシスタントで選択できるプロジェクトが増えた。しかし、ここでは同じく 「Cocoa-Java Application」を選べば良い。



図2-004 作成するプロジェクトの種類を選択する。

(p.31) プロジェクトウィンドウのツールバーに表示されるアイコンの並びが変わった。また、「Project Builder Release Notes」は表示されない。

🦳 🧉 Project Builder ファイル 編	長 形式 表示	検索 プロジェクト ビルド デバッグ CVS ウインドウ ヘルプ 🎫
000		HTMLColor
🔨 🐔 🚊 隊 💿 нтм	Color	
◎ グループとファイル		Q.検索 V. ビルド I. I. 実実行 V. T. デバッグ
▼ HTMLColor	Z X OO	
Chasses	2	
► 📁 Resources		
Frameworks		
Products	77	
	2	
	O	
	1	
	1	
	2	
	in the second se	
	\prec	
	2	エディタがちりませ/
	7	エノイラがのりょせん
	4	
	1	
	2	
	Ϋ́,	
	Ì	
	J.	
	•	

図2-005 プロジェクトウィンドウの初期状態

◆2.5 アプリケーションのビルドと実行

(p.33) デフォルトでは、ビルド中の進行状況は「ビルド」タブに表示されないようなった。なお、「ビルド」パネルの下部にあるパネルを区切るバーを上にドラッグすれば、進行状況を見ることができる。また、プロジェクトウィンドウ下部のステイタスバーにビルドの進行状況が表示され、ビルドが正常に行われると「ビルドは問題なく完了しました」と表示される。

(p.34) ソースコードに間違いがあった場合に、Javaのコンパイルが完了しないというバグは、10.2の Developer Toolsでは解消されている。従って、「注意!」事項は無視して良い。ここで記載した対処方法 を設定する必要もない。

◆2.6 ビューの配置

(p.35) 「グループとファイル」リストに表示されるファイルの左側にチェックボックスが表示されるようになった。このチェックボックスのチェックを外せば、そのファイルはビルドの対象から外される。例えば、ソースコードを修正した場合でも、チェックを外せば、ソースコードはコンパイルされない。



図2-006 「グループとファイル」リストの表示

(p.36)「Cocoa-Views」パレットの内容が一部変化し、以下の図のようになった。テキストフィールドは右上にある白地の長方形なので、これをドラッグしてウィンドウにドロップする。



図2-007 「Cocoa-Views」パレット

(p.37) カラーウェルの初期状態が白っぽい青に変更されたので、これに合わせてテキストフィールドに 最初に表示するテキストとして「#OeOeff」を設定する。

(p.38) 「Cocoa-Views」パレットのメッセージテキストは、フォントサイズ違いで3種類表示されるようになった。ここでは左下にある「System Font Text」をドラッグし、ウィンドウにドロップする。

◆2.7 ウィンドウの設定

(p.40) QuickTime Playerなどで見られるメタル調のウィンドウに設定することもできる。ウィンドウを メタル調にするには、InfoウィンドのAttributesの「Options」にある「Texture Window」をチェックす る。



図2-008 メタル調のウィンドウ

◆2.8 インターフェースのテスト

(p.41) Colorsパレットのデザインが変わり、クレヨン表示など一部機能が増えている。使用方法として は以前とほぼ同じである。



図2-009 Colorsパレット

◆2.10 アウトレットの作成

(p.44)InfoウィンドウにおけるクラスのAttributes表示が変更されている。アウトレットとアクションは タブによって表示を切り替える。また、アウトレットやアクションの追加や削除を行う「+」ボタンと「- 」ボタンは、「add」ボタンと「remove」ボタンに変更された。重要な変更点として、アウトレットのタ イプをポップアップ・リストから指定できるようになった。ここではcodeTextFieldアウトレットのタイプ として「NSTextField」を指定する。

000	ColorEncod	er Class Info
000	colorencou	
F	Attributes	
Language:	O Objectiv	ve-C
Lunguage.	 Java 	View in Editor
ClassName:	ColorEnco	der
	1 Outlet	0 Actions
Outlet Name		Туре
codeTextFiel	d	NSTextField
<u>codeTextFiel</u>	d	NSTextField
<u>codeTextFiel</u>	d	NSTextField
codeTextFiel	d	NSTextField
codeTextFiel	d	NSTextField

図2-010 InfoウィンドウによるクラスのAttributes表示

◆2.16 メソッドのコーディング

(p.54) Interface Builderでアウトレットのタイプ指定ができるようになったことに伴い、以下のように 生成されるソースコードでは、アウトレットに対応するインスタンスフィールドのクラスが自動的に記述さ れるようになった。従って、インスタンスフィールドをObjectクラスからNSTextFieldクラスに書き換え る必要はない。また、インスタンスフィールドは自動的にpublic指定される。本文で記述している「クラス フィールド」は「インスタンスフィールド」に訂正されたい。

```
/* ColorEncoder */
```

```
import com.apple.cocoa.foundation.*;
import com.apple.cocoa.application.*;
public class ColorEncoder {
    public NSTextField codeTextField; /* IBOutlet */
    public void encode(Object sender) { /* IBAction */
    }
```

}

(p.55) encodeアクションに対応するメソッドのアーギュメントは、本文通りObjectクラスから NSColorWellクラスに書き換えなければならない。従って、クラスを訂正したソースコードは以下のよう になる。

/* ColorEncoder */

```
import com.apple.cocoa.foundation.*;
import com.apple.cocoa.application.*;
public class ColorEncoder {
    public NSTextField codeTextField; /* IBOutlet */
    public void encode(NSColorWell sender) { /* IBAction */
    }
}
```

◆2.18 デバッガの利用

(p.62) デフォルトでは、使用するデバッガが「Gdb」に設定されているので、ソースコードエディタで ブレイクポイントを指定しても、実行時にブレークポイントで一時停止しない。このような場合には、プロ ジェクトメニューから「アクティブな実行可能ファイルを編集」を選び、デバッガ欄で「Javaデバッガ」 を使用するように指定すれば良い。



図2-011 使用するデバッガの指定

(p.66)使用するデバッガは、「ターゲット」タブを開き、「Executables」の「HTMLColor」を選択し て表示される実行可能ファイルの編集項目のうち、デバッガ欄で行う。上記のように、プロジェクトメ ニューから「アクティブな実行可能ファイルを編集」を選んでも良い。ここでは「Gdb」と「Javaデバッ ガ」に加えて、「OSA Debugger」も指定できるようになった。

◆2.20 メニューの処理

(p.68) ColorPanelControllerクラスに追加したsourceColorWellのアウトレットのタイプは、「NSColorWell」を指定する。

(p.73) 生成されたソースコードでは、ColorPanelControllerクラスのインスタンスフィールド sourceColorWellは自動的にNSColorWellクラスになっているので、修正する必要はない。なお、open() メソッドのパラメータであるsenderは、本文通りNSMenuItemクラスに修正する。

◆2.24 ヘルプの修正

(p.82) HTML形式のヘルプファイルを指定するには、「ターゲット」タブを開き、「Targets」リストの「HTMLColor」を選択する。そして「Info.plistのエントリ」の「詳細設定ビュー」を選ぶ。そして、「同 位置に新規作成」ボタンをクリックすれば、新しいプロパティが追加されるので、その名前を 「NSHelpFile」、クラスを「文字列」、値を「help.html」とする。



図2-012 ヘルプとして用いるHTMLファイルの指定

◆2.25 アイコンの設定

(p.89) アプリケーションのアイコンを指定するには、「ターゲット」タブを開き、「Targets」リストの「HTMLColor」を選択する。そして「Info.plistのエントリ」を選び、アイコンファイル欄に「HTMLColorAppIcon.icns」と入力する。

🧉 🛛 Project Builder	ファイル 編集 形式 表示 検索 プロジェクト ビルド デバッグ CVS ウインドウ ヘルプ 🏼
00	HTMLColor - ターゲット ∶ HTMLColor
5 🐔 🚊 🌂	
♥Targets ● ● ● HTMLColor ▶Build Styles ♥Executables ● ◆ HTMLColor	
	- C 💿 🧐 ターゲット:HTMLColor 🗢 🕜 🖸 🔞 😒
	♥ プロジェクト"HTMLColor"のターゲット"HTMLColor"
	 要約 認別子: タイブ: APPL シグネチャ: ???? タイブ: APPL シグネチャ: ???? アパージョン: 0.1 CCC コンパイラ設定 リンカ設定 ブレフィックスヘッダ 検索パス Java コンパイラ設定 Java コンパイラ設定 詳細設定ビュー ▼ 情報を表示 ▼ 情報を表示 ● アイコン
	アイコン Cocoa 固有の設定 Pure Java 固有の設定 Pure Java 固有の設定 中田 の設定
	► UPL のタイプ COCOA 固有の設定 ▼

図2-013 アプリケーションのアイコンファイルの指定

◆2.26 アプリケーション情報の設定

(p.90) アプリケーション情報を設定するには、「ターゲット」タブを開き、「Targets」リストの「HTMLColor」を選択する。そして「Info.plistのエントリ」を選び、「基本情報」や「情報を表示」の各項目に適切なテキストを入力する。



図2-014 アプリケーション情報の設定

(p.91) アプリケーションのビルドに関する設定は、「ターゲット」タブを開き、「Targets」リストの「HTMLColor」を選択する。そして、「設定」を選び、各種項目を設定する。



図2-015 ビルドに関する設定

◆2.27 ローカライズ

(p.97) 各種のローカライズ作業を行った後、アプリケーションをビルドして実行しても、ローカライズ 作業が反映されないことがある。例えば、HTML形式のヘルプファイルをローカライズしても、アプリケー ション実行時には英語で表示されるだろう。これは、ローカライズしていないグローバルであるHTMLファ イルがビルドファイルの中に残っているためである。このような場合は、ツールバーの帚(ほうき)アイコ ンのクリーンボタンをクリックして(またはビルドメニューから「クリーニング」を選んで)、ターゲット をクリーニングし、再度ビルドし、実行すれば良い。

第3章 さまざまなアプリケーション例

◆3.1 ドキュメント型アプリケーション

■ドキュメントのアウトレット設定

(p.106) MyDocumentクラスに追加したtextViewアウトレットのタイプは、「NSTextView」を指定する。

000	MyDocumer Attributes	nt Class Info	•	
Language:	 Objectiv Java 	ve-C View i	n Editor)
ClassName:	MyDocume	ent		1
ſ	2 Outlets	0 Actions)	
Outlet Name		Type		
textView		NSTextVie	w	

図3-001 「textView」アウトレットのタイプ指定

■ドキュメントのタイプ設定

(p.108) ドキュメントのタイプ設定を行うには、「ターゲット」タブをクリックし、「Targets」リストの「FirstText」を選択する。そして「Info.plistのエントリ」にある「書類のタイプ」で行う。

	$L_{\mu\nu}$ $F_{\mu\nu}$ F
$\bigcirc \bigcirc \bigcirc \bigcirc$ FirstText – $\vartheta - \varphi \neg \psi \land$: FirstText	
🔨 🐪 🖳 🦻 FirstText ;	
Targate	◎、検索 人 ヘビルド 人 夏実行 人 1 デバッグ 」
● ▶ ◎ FirstText ► O O ◎ ターゲット : FirstText ◆	S 8 8 S
▶Build Styles ◎ プロジェクト"FirstText	"のターゲット"FirstText"
▼Executables ● ▲ FirstTaxt 要約	57 777 11000 000
▼ 書類のタイフ	
● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ●	役割 拡張子 OSタイプ
● 「板成定 」 インストール設定 NSRTFPboardTy	e Editor rtf rtf
GCC コンパイラ設定	
リンカ設定 プレフィックスヘッダ	
検索パス	
Java コンパイラ設定	
Java アーカイブ設定 詳細設定ビュー	
▼Info.plistのエントリ	
★ ▼シンブルピュー	
日報を表示	
アイコン 書類タイプの情報	
Cocoa 固有の設定 名前	: NSRTFPboardType 役割: Editor 🛟
Cocoa Java 固有の設定 L	: rtf
と 書類のタイプ OS タイプ	: rtf
い URLのタイプ 詳細設定ビュー アイコンファイル	・ ファイルパッケージ
マビルドフェーズ	
▲ ヘッダ 書類のクラス	: MyDocument
	(追加) 取り除く 変更
Java リソースファイル	
■ フレームワークとライ: ▼ IIRI のタイプ	Ŧ
	1.

図3-002 ドキュメントタイプの設定

◆3.2 タイマとサウンドの利用

■ユーザインターフェースの作成

(p.117) スライダを動かした時に連続的にテンポ表示の数値が変わるようにするためには、スライダの設定で「options:」欄の「Continuous」にチェックを付ける。

(p.118) 「Start」ボタンのアイコン位置として、Infoウィンドウの「Icon Position:」欄で左上のアイコ ンを選んで、アイコンが無いことを指定する。また、このボタンのテキストは、フォントパネルを開いて 13ポイントに指定すると良い。

000	NSButto	n Info	000	Fonts	
(Attributes	-	Family	Typeface	Sizes
Title: Alt. Title: Icon: Alt. Icon: Sound: Equiv: Type: Behavior: Options Ø Borco Con Ø Enat Sele Sma	Start	ey>	Kino MT Krungthep KufiStandardGK Lucida Blackletter Lucida Calligraphy Lucida Calligraphy Lucida Grande Ex Use Family a System f	Regular Bold tras	13 9 10 11 12 13 14 10

図3-003 「Start」ボタンのアイコン位置指定とフォント指定

■コントローラクラスの作成

(p.119) MetroControllerクラスに追加したアウトレットのタイプとして、beat0からbeat3までと toggleButtonは「NSButton」を指定する。また、tempoFieldは「NSTextField」、tempoSliderは 「NSSlider」を指定する。

Att	ributes 主
Language: 🔇) Objective-C
	Java View in Editor
ClassName:	fetroController
7	Outlets 2 Actions
Outlet Name	Туре
beat0	NSButton 🗘
beat1	NSButton 🗘
beat2	NSButton 🗘
beat3	NSButton 🗘
tempoField	NSTextField +
tempoSlider	NSSlider -
toggleButton	NSButton 🗧
	(Remove) Add

図3-004 MetroControllerクラスに追加したアウトレット

■インスタンスフィールドの定義

(p.125) 生成されたソースコードでは、アウトレットに対応するインスタンスフィールドのクラス型が指 定されているので、修正する必要はない。なお、「NSTimer metroTimer;」以下のインスタンスフィール ドは、本文通り追加する。 ◆3.3 ベジェ曲線でグラフィックを描く

■カスタムビューの作成

(p.138)カスタムビューのクラス指定は、Infoウィンドのポップアップメニューから「Custom Class」 を選んで行う。

Custom Class	•
Class	
DrawingView	
NSBox	
NSBrowser	
NSButton	
NSClipView	
NSColorWell	
NSComboBox	
NSControl	
NSForm	
NSImageView	
NSMatrix	
NSMovieView	
NSOpenGLView	U.
NSOutlineView	
NSPopUpButton	
NSProgressIndicator	
NSQuickDrawView	
NSRulerView	
NSScroller	
NSScrollView	
NSSecureTextField	*

図3-005 カスタムビューのクラス指定

■コントローラクラスの作成

(p.141) DrawingControllerクラスに追加したdrawingViewアウトレットのタイプは「DrawingView」 を指定する。

Attributes				
Language:	O Objecti	ve-C		
	🖲 Java	View in Editor		
ClassName:	DrawingC	ontroller		
	1 Outlet	0 Actions		
Outlet Name		Type		
		DrawingView 🔸		
		DrawingView •		

図3-006 DrawingControllerクラスに追加したアウトレット

■スライダの処理

(p.152)生成されたソースコードでは、アウトレットに対応するインスタンスフィールドのクラス型が指定されているので、修正する必要はない。

第4章 開発ツールの詳細

◆4.1 Project Builder

■Project Builderの構成

(p.161) 図4-2において、プロジェクトウィンドウの下部のバーは「ステイタスバー」である。左下に「ツールバー」と記述されているのは誤り。

■ツールバー

(p.161) ツールバーの左側に表示されるボタンの並びが変更になった。以下の図のように、左から順にビルドボタン(金槌アイコン)、ビルドとデバッグボタン(金槌にスプレー缶が重なったアイコン)、ビルドと実行ボタン(金槌にディスプレイが重なったアイコン)、クリーンボタン(帚アイコン)である。

🧉 Project Builder	ファイル	編集	形式	表示	検索
000			E HT	MLColo	r - ター
 ダループとファイル 	<u>ا</u>	HTMLCo	lor		•
▼ HTMLColor ► Classes		アイル			

図4-001 ツールバー左側のボタンとポップアップメニュー

(p.162) ツールバーの右側に表示されるデバッグ関係のボタンの最も左側に、再起動ボタンが追加された。

ェクト	ビルド	デノ	ドッグ	CVS	ウイ	′ンドウ	ヘルプ	***
HTMLCo	olor						0	
		►			0	()	(1)	
					נעב	/-n)	標準 I/O	
	; 変	数			値		要約	

図4-002 ツールバー右側のデバック関係のボタン

(p.163) ツールバーをカスタマイズするには、表示メニューから「ツールバーをカスタマイズ...」を選ぶ。ツールバーに表示できるボタンの種類が増え、ボタンを小アイコンとして表示できるようになった。



図4-003 ツールバーのカスタマイズ

■コンテンツリスト

(p.164) 「グループとファイル」リストにおいて、ファイルの左側にチェックボックスが表示されるよう になった。このチェックボックスのチェックを外すと、ビルド処理の対象外となる。



図4-004 「グループとファイル」リスト

(p.166) 「ターゲット」リストの「Targets」で選択したターゲットの設定は、新しい形式のエディタで 行うようになった。エディタの左側のリストに設定項目が表示され、リストで選んだ項目を右側のエリアで 設定する。



図4-005 ターゲットの設定

(p.167) 「ターゲット」リストの「Targets」で選択した実行可能ファイルの設定は、新しい形式のエ ディタで行うようになった。

🧉 🛛 Project Builder	ファイル 編集 形式 表示 検索 プロジェクト ビルド デバッグ CVS ウインドウ ヘルプ 📒
000	HTMLColor - 実行可能ファイル:HTMLColor
🔨 🐴 🚊 👌	III II (↓) (↑)
♥Targets ● ● ● HTMLColor ♥Build Styles ●	 ● 検索 くビルド ■実行 ■ デバッグ ● ● 美行可能ファイル:HTMLColor ÷ ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ●
	 ▲ 第時金数 名前 値 使用 ● ● ■ <l< td=""></l<>

図4-006 実行可能ファイルの設定

■エディタ

(p.171) ソースコードエディタの上部にあるナビゲーションバーの右側に表示されるボックスのアイコン が変更された。左から順に、構文チェック、対応するものに移動、エディタを分割、分割しているエディタ を統合、の役割を担う。



図4-007 ソースコードエディタ

(p.171) ソースコードエディタなどに表示されている単語をoptionキーを押しながらダブルクリックすれ ば、その単語が検索パネルの検索文字列に取り込まれ、自動的に検索が開始される。また、その単語がクラ ス名やメソッド名である場合は、そのドキュメントが表示される。

(p.172)従来の「ビルド設定」パネルや「アプリケーション設定」パネル、「詳細設定」パネル、「実行 可能ファイル」パネルは廃止され、ターゲットの設定エディタと実行可能ファイルの設定エディタに統合さ れた。

■操作メニュー

(p.185) 従来の操作メニューは表示メニューに変更された。

🧉 Project Builder	ファイル	編集	形式	表示	検索	プロジェクト	ビルド	デバッグ	CVS	ウインドウ
				次の 前の ヘッ)ファイ)ファイ)ダ/ソ	ル ル ースファイルに	切り替え	~₩→ ~₩← ~%†		
				ブッ	クマー	クに追加		~ ≍ ≋↓		
				独立 グル 現在 現在	てしたエ レープツ Eのエデ Eの分割	ディタで開く リー内に表示 ィタを分割 ウインドウを閉	じる	℃ ₩Ο ℃₩T ₩" %'		
				ツ- ツ-	-ルバー -ルバー	を隠す をカスタマイズ				

図4-008 表示メニュー

(p.192) 従来のSCMメニューはCVSメニューに変更された。

6	Project Builder	ファイル	編集	形式	表示	検索	プロジェクト	ビルド	デバッグ	CVS ウインドウ ヘルプ 💻
										ステータスを更新 最新のリビジョンにアップデート 最新のリビジョンに戻す
										ベースと比較 最新のリビジョンと比較/マージ
										リポジトリに追加 変更をコミット
										CVS 統合を使用する

図4-009 CVSメニュー

■プリファレンス

(p.195)プリファレンスウィンドウが拡張され、「タスクテンプレート」、「インデント設定」、「ファ イルタイプ」、「Documentation Viewing」の設定が追加された。また、従来からの設定も項目の追加変 更が行われている。

	127 TANKA 127 TANKA 128 TANKA 129 TANKA 129 TANKA 129 TANKA 120 TANKA	roject Builder"環	境設定 <i>劉</i> 公	{ →if	
全般 5	マスクテンプレート	テキスト編集	構文の色付け	〉 インデント設定	
	基本設定	詳細設定 テン	ノブレートの管理)•	
1 つのウインドウ 1 つのウインドウ たのオプションの選択(*1 つのウインド* *いくつかのウイ? *3数のウインド* *カスタム設定**は	レンジャングレートウイン かくつかのウ は、テンプレートウイン かは、すべての開発タ ンドウ"は、検索、ビル ブ"は、それそれの開発 、"詳細設定"タブで設	マインドウを使用するとき スクを単一のオール・ ド、およびデバッグ・ タスクことれた設定です。	With the second secon	カスタム設定 あえタム設定 を決定します。 します。 ウインドウを開きます。	

図4-010 プリファレンスウィンドウ

◆4.2 Interface Builder

■パレット

(p.212) 「Views」パレットに表示される固定テキストが3種類になった。これらはフォントサイズが異なるだけで、同じビューである。

000	Cocoa-Views 🔘
Text	Te: 📥 제 »
Button	Sept
Switch	Field1: 1.99 Field2:
● Radio ○ Radio	Label Font Text Small System Font Text System Font Text

図4-011 「Views」パレット

(p.212) 「Other」パレットの下部中央に回転式のプログレス・インジケータが加わった。



図4-012 「Other」パレット

(p.214) 「AppleScript」パレットが加わった。青い立方体アイコンで表される「ASKDataSource」オ ブジェクトをインスタンスとして用いることができる。



図4-013 「AppleScript」パレット

■プリファレンス

(p.234)プリファレンスウィンドウに「Editing」パネルが追加された。このパネルでは編集に関する設 定を行う。

00	Interface Builder Preferences					
	General Editing Layout Colors Palettes					
	✓ Inside_out selection					
	Spell check as you type					
	Allow image to be stored inside Nibs					
This is not recommended for modern Nibs						
Unlimited undo levels*						
Number of undo levels: 10						
	* Only takes effect after restarting Interface Builder.					
	li.					

図4-014 「Editing」パネル

第5章 Cocoaフレームワーク

◆5.1 Application Kitフレームワーク

■Application Kitクラス

Application Kitに追加されたクラスは以下の通り。

NSGlyphInfo	文字列のGlyph情報を扱うクラス。
NSTabViewItem	タブによって切り替えられる内容を扱うクラス。
なお、p.249の下から5行目はNSSta	atusItemとなっているが、NSStepperCellが正しい。

NSStepperCell ステッパーの表示やマウス処理を扱うクラス。

■Application Kitインターフェース

Application Kitに追加されたインターフェースは以下の通り。

NSValidatedUserInterfaceItem コントロールを自動的に切り換えるユーザインターフェースのための インターフェース。

◆5.2 Foundationフレームワーク

■ Foundationクラス

Foundationに追加されたクラスは以下の通り。

NSAppleEventDescriptor	AppleEventを記述するためのクラス。
NSAppleScript	AppleScriptのロード、コンパイル、実行を行うクラス。
NSKeyedArchiver	オブジェクトをキーを持ったアーカイブにエンコードするクラス。
NSKeyedUnarchiver	キーを持ったアーカイブからオブジェクトをデコードするクラス。
NSNameSpecifier	集合から名前によってオブジェクトを指定するクラス。
NSUniqueIDSpecifier	集合から重複しないIDによってオブジェクトを指定するクラス。

■Foundationインターフェース

Foundationに追加されたインターフェースは以下の通り。

NSCoding

クラス定義のコーディングのためのインターフェース。

(以上)