

監訳者まえがき

アプリケーションにとっていちばん大事なのは、「使いやすさ」です。この「使いやすさ」とは何でしょうか？

僕は「使いやすさ」とは、目的を妨げないことと考えます。目的に向かって最短経路で、間違えず、つかえず、集中力を乱さない……そういう手段を提供することこそが、使いやすさのエッセンスではないでしょうか。

どんなに革新的であっても、どんなに高性能であっても、使いにくいものは見向きもされない。業務用であったり市場が制限されいたりする場合は、使いにくくても普及しているモノはあります。でも、それはみんな仕方がないから嫌々使っているだけなのです。なので、いったん同機能でより使いやすいモノがリリースされれば、みなそちらに移ってしまいます。

使いやすいインタフェースを、ゼロから生み出すのはとても難しい。それはインタフェースというものが、単なる技術だけでは作れないからです。使いやすいインタフェースを作るためには、心理学、人間工学、美術、プログラミング、そしてその分野のオペレーションや作業フローへの深い造詣……と本当にさまざまな分野の知識を要求します。机上のスケッチや、思いつき一発で使いやすいものが生まれることは滅多にありません。結果、既存のモノをベースとしたスクラップ&ビルドの地道な積み重ねが必須となるのです。

しかし、スクラップ&ビルドによるブラッシュアップなどは、予算やスケジュールには計上しにくい。結果的に、インタフェースは後回しとなり、盛り込みやすい機能ばかりが過多になってしまうわけです。

本書『モバイルデザインパターン』は、そのような問題に対する解のひとつです。プログラミングのように、インタフェース上の課題と解をパターン化することで、初期段階でのインタフェースの模索コストを大幅に軽減することができます。

この本は空手でいえば型、将棋でいえば定跡のようなものです。物事を勉強するには先人の積み上げて来た知識を学ぶのがいちばん早い。愚直に型を学び、そうやって型をスムーズにこなせるようになってはじめて、それぞれの型の中にある本質的なエッ

センスを発見できます。そして定跡の通用しないシチュエーションで、型を破った新しいものを作り出すことができるのです。

本書がそんなふうインタフェースを考えるための、とっかかりになることを願います。

2012年8月13日

深津 貴之

賞賛の声

とってもハンディなカタログです。ページをばらばらめくりながらアイデアを探しています。

— Bill Scott (PayPal社 Web 開発部門シニアディレクター)

すばらしい!

— Erin Malone (Tangible UX社共同経営者)

稼働中のアプリケーションを再設計しなければならなかったのですが、本書が作業を導いてくれました。いくら感謝しても足りません。

— Agatha June (UXデザイナー)

序文

名前を付けることが、対象を理解するための第一歩になります。

5歳になる私の息子も、他の子供たちと同じように雲を眺めるのが大好きです。数週間前、彼はさまざまな種類の雲には異なる名前が与えられているということを知りました。そしてギーク精神を発揮し、雲の名前を覚え始めました。巻雲、積雲、巻層雲、積乱雲、高層雲、レンズ雲など、筆者が知っている名前の雲をすべて覚え、他にもいろいろ覚えるようになりました。雄大積雲などといった名前は、私は聞いたことがありませんでした。

息子は今や、空を見上げるとどの雲が何という名前か教えてくれるようになりました。同時に、今まで以上に細かい点にも気がつくようにもなりました。模様や色、高さ、動きなどに基づいて雲を区別し始めました。見分けるのが難しい種類の雲もありますが、彼にとっては問題にならないようでした。雲の種類によって、これから雨が降りそうかどうかをある程度の正確さで予想もしています。

幼児なりのやり方で、彼は全体的な分析も始めました。「巻層雲は暖かいところに発生する」とか、「雄大積雲は積乱雲へと変化し、嵐をもたらす」といった指摘を述べるようになりました。

このようなこともあって、息子は雲の名前を知り楽しんでいるようです。小さい子供というのは雲や恐竜、虫、人形、映画の登場人物など、好きなものに名前を付けたがるものです。子供たちの創造力は左脳の知識に限定されることはなく、大人の固定観念に左右されることもありません。息子は雲を積雲などと呼んでいますが、今も雲の中にお城やアヒルあるいはカリフラワーなどの形を見いだしているのです。

このような思考様式は我々大人にも当てはまります。「インタフェースに見られるパターンを識別してそれに名前を付けることによって、パターンをよりよく理解できる」というアイデアもここから生まれました。着目すべき対象を指示されれば、脳はより多くの細かな点にも気づくようになります。あるパターンに基づいたインタフェースの振る舞いを知っていれば、そのパターンを利用しているソフトウェアの働きを予測できるようにもなるで

しょう。また、パターンという新たな語彙を使って他者に考えを伝えることも可能になります。

さて、パターンについて学ぶにはどうすればよいでしょうか。

私の息子が雲について学ぶときに、最も役立ったのは多数の絵でした。本やWebサイトに掲載された雲の「カタログ」を見ながら、彼は言葉では表現が難しいような微妙な違いについても知識を広げてゆきました。

インタフェースのパターンについても同様に、最善の学習方法は視覚的な例に触れることであると考えます。私は物書きであり、言葉が大好きです。望まれるなら、パターンとは何か、どのパターンを選ぶべきか、それぞれのパターンの違いは何かといった点についていくらかでも言葉を書き連ねることでしょう。しかし、単にインタフェースをデザインしようとする人（言い換えるなら、自らの知識の一部としてパターンを学ぼうとする人）にとって、言葉は明らかに不必要です。パターンを「感じ取る」ことができる程度の説明と、厳選されたさまざまな実例さえあれば、そのパターンを身につけ知識を強固なものにできるはずです。

本書の著者であるTheresa Neilは、パターンの実例となる画像のすばらしいコレクションを完成させました。この作業には多くの苦勞が伴ったはずですが、私も試みたことがあるのですが、大量のモバイルアプリをレビューしてどれが最も優れているか判断し、スクリーンショットを注意深く収集するというのは簡単なことではありません。

モバイルインタフェースのデザイナーにとって、本書は宝物になることでしょう。初めから順に読むのもよいですが、さまざまな例を活用して自分が行っているデザインを改善してみましよう。

- 例の中で、どれが最も優れているかという点について自分で判断してみましょう。そして、自身が行っているデザインにとって最も適しているものを探しましょう。
- デザインでのインスピレーションの源として本書を活用しましょう。アイコンや色の使い方など、パターン自体とは関係のない点でも本書のスクリーンショットは非常に役立ちます。

- 既存のアプリケーションの働きを知るきっかけにしましょう。有料のアプリケーション（やデバイス）をいくつも購入しなくても、これらについて学ぶことができます。

さらに、読者が普段利用しているモバイルアプリの中から新しいパターンを見いだすのもよいでしょう。パターンには名前があるということを覚えると、他にもどんどん名前を付けたくなるものです。息子が何度か「これが雄大積雲だよ」と教えてくれるのを聞いているうちに、私も雄大積雲について学び、この知識なしでは生きていけないとすら思えてきました。

心ゆくまで本書を味わってください。

Jenifer Tidwell

まえがき

イントロダクション

先日、筆者の周囲でとあるモバイルアプリのプロジェクトが立ち上げられたのですが、ベテランのデザイナーはみな他のプロジェクトで忙しいという状態でした。そこで、1週間足らずの間に新人デザイナーに筆者の経験を伝えなければならなくなりました。筆者はおびただしい量のスクリーンショットを用意して簡潔なチュートリアルを作成し、どれが良いデザインでどれが良くないのかを明らかにしようと試みました。この作業を通じて、モバイルアプリのデザインにおける一連のパターンが明らかになってきました。

これらのパターンのカタログを作成している間に、「真の価値はパターンの識別にあるのではなく、筆者が大量にキャプチャした実例にあるのではないか」ということがわかりました。そこで、抽象的なパターンを延々と書き連ねて自己満足に陥るのではなく、本書ではモバイルアプリのデザインのためのショーケースあるいはギャラリーとなることを目指しました。本書にはiOSやAndroid、BlackBerry、WebOSやWindowsでの400以上のスクリーンショットが収録され、パターンごとに分類されています。本書のWebサイト Mobile Design Pattern Gallery (<http://www.mobiledesignpatterngallery.com>) や Flickr のフォトストリームではさらに多くの実例が紹介されています。

対象読者

本書はモバイルアプリにかかわるプロダクトマネージャー、デザイナーそして開発者のために執筆されました。多くの企業がモバイル分野の戦略を定義し洗練させてゆく状況の中で、デザインのベストプラクティスを見いだすのは（複数のオペレーティングシステムが関与する場合は特に）容易ではありません。シンプルなiPhoneアプリを作成する場合でも、主要なオペレーティングシステムのすべてに対応したアプリケーションが求められる場合でも、本書で紹介するパターンがよくあるデザイン上の課題へのソリューションを提供してくれるはずです。

表記上のルール

本書では、次に示す表記上のルールに従います。

太字 (Bold)

新しい用語、強調やキーワードフレーズを表します。

等幅 (Constant Width)

プログラムのコードを表します。その断片(変数、関数、キーワードなど)を本文中から参照する場合にも使われます。

等幅イタリック (Constant Width Italic)

ユーザーの環境などに応じて置き換えなければならない文字列を表します。

TIP ヒントや示唆、興味深い事柄に関する補足を表します。

意見と質問

本書(日本語翻訳版)の内容については、最大限の努力をもって検証および確認していますが、誤りや不正確な点、誤解や混乱を招くような表現、単純な誤植に気づかれることもあるかもしれません。本書を読んで気づいたことは、今後の版で改善できるようにお知らせください。将来の改訂に関する提案なども歓迎します。連絡先を以下に示します。

株式会社オライリー・ジャパン
〒160-0002 東京都新宿区坂町26番地27
インテリジェントプラザビル1F
電話 03-3356-5227
FAX 03-3356-5261
電子メール japan@oreilly.co.jp

本書についての正誤表や追加情報などは、次のWebサイトを参照してください。

<http://www.oreilly.co.jp/books/9784873115689/>

<http://www.oreilly.com/catalog/0636920026419/> (英語)

本書に関する技術的な質問や意見は、次のメールアドレスあてにお送りください。

japan@oreilly.co.jp

bookquestions@oreilly.com (英語)

オライリーに関するその他の情報（文献、会議、リソースセンター、O'Reilly Networkに関する情報）は、次のオライリーのWebサイトを参照してください。

<http://www.oreilly.co.jp/>

<http://www.oreilly.com/> (英語)

謝辞

すべてのパターンでイラストを作成してくれたAaron Jasinskiと、Mobile Design Pattern GalleryのWebサイトやブログをデザインしたSmith & RobotのChad、本書を実現させてくれたO'Reilly MediaのMaryとDanに感謝します。この6か月間の作業を共にした才能ある同僚Jessica、James、Kirsten、Marie、Ben、Lulu、Ivan、Enricoにも感謝します。また、週末のたびに執筆にかりっきりになっていた筆者を許してくれた家族にも感謝します。Jenifer TidwellとMatthew君の親子にはすばらしい推薦の言葉を書いていただきました。読者にとって本書が、昼下がりに眺める雲の半分くらいは楽しいものになることを願います。

目次

監訳者まえがき	v
賞賛の声	vii
序文	viii
まえがき	xi
1章 ナビゲーション	1
1.1 主要なナビゲーション	3
1.1.1 Springboard (スプリングボード)	4
1.1.2 List Menu (リストメニュー)	7
1.1.3 Tabs (タブ)	10
1.1.4 Gallery (ギャラリー)	14
1.1.5 Dashboard (ダッシュボード)	16
1.1.6 Metaphor (メタファー)	17
1.1.7 Mega Menu (大型メニュー)	19
1.2 補助的ナビゲーション	20
1.2.1 Page Carousel (ページのカルーセル)	22
1.2.2 Image Carousel (イメージのカルーセル)	24
1.2.3 Expanding List (伸縮するリスト)	26
2章 フォーム	29
2.1 Sign In (サインイン)	31
2.2 Registration (ユーザー登録)	33
2.3 Checkout (購入手続き)	36
2.4 Calculate (計算)	38
2.5 Search Form (検索フォーム)	40
2.6 Multi-Step (複数のステップ)	42
2.7 Long Form (長いフォーム)	45
3章 テーブルとリスト	49
3.1 Basic Table (基本的なテーブル)	52
3.2 Headerless Table (ヘッダーのないテーブル)	52

3.3	Fixed Column (固定された列)	54
3.4	Overview plus Data (概要と個々のデータ)	55
3.5	Grouped Rows (グループ化された行)	57
3.6	Cascading Lists (カスケードイングリスト)	58
3.7	Table with Visual Indicators (インジケータ付きのテーブル)	60
3.8	Editable Table (編集可能なテーブル)	61

4章 検索、並べ替え、フィルター 63

4.1	検索	65
4.1.1	Explicit Search (明示的な検索)	66
4.1.2	Search with Auto-Complete (オートコンプリート付きの検索)	67
4.1.3	Dynamic Search (動的な検索)	69
4.1.4	Scoped Search (絞り込み検索)	70
4.1.5	Saved and Recent Searches (検索結果の履歴)	71
4.1.6	Search Form (検索フォーム)	73
4.1.7	Search Results/View Results (検索結果・表示結果)	75
4.2	並べ替え	77
4.2.1	Onscreen Sort (画面上での並べ替え)	77
4.2.2	Sort Order Selector (並べ替え方法のセレクト)	78
4.2.3	Sort Form (並べ替えのフォーム)	80
4.3	フィルター	81
4.3.1	Onscreen Filter (画面上でのフィルター)	82
4.3.2	Filter Drawer (フィルターの引き出し)	84
4.3.3	Filter Dialog (フィルターダイアログ)	85
4.3.4	Filter Form (フィルターフォーム)	86

5章 ツール 91

5.1	Toolbar (ツールバー)	96
5.2	Option Menu (オプションメニュー)	99
5.3	Call to Action Button (アクションボタン)	100
5.4	Contextual Tools (コンテキスト依存のツール)	102
5.5	Inline Actions (インラインの操作)	105
5.6	Multi-State Button (複数の状態を持つボタン)	106
5.7	Bulk Action (操作の一括実行)	110

6章 グラフ	113
6.1 Chart with Filters (フィルター機能を持つグラフ)	117
6.2 Preview Window (プレビューウィンドウ)	120
6.3 Overview plus Data (概要と個々のデータ)	121
6.4 Datapoint Details (特定の点についての詳細)	123
6.5 Drill Down (階層的な表示)	125
6.6 Zoom (拡大表示)	126
6.7 Pivot Table (ピボットテーブル)	128
6.8 Sparklines (スパークライン)	130
7章 誘導	133
7.1 Dialog (ダイアログ)	136
7.2 Tip (ツールチップ)	137
7.3 Tour (ツアー)	139
7.4 Video Demo (動画によるデモ)	140
7.5 Transparency (透過表示)	142
7.6 1st Time Through (初回のみ表示)	143
7.7 Persistent (継続的な表示)	144
7.8 Discoverable (発見型誘導)	145
8章 フィードバックとアフォーダンス	147
8.1 フィードバック	148
8.1.1 Error Messages (エラーメッセージ)	148
8.1.2 Confirmation (確認)	150
8.1.3 System Status (システムの状態)	152
8.2 アフォーダンス	155
8.2.1 Tap (タップ)	155
8.2.2 Flick (フリック)	157
8.2.3 Drag (ドラッグ)	159
9章 ヘルプ	161
9.1 How To (ハウツー)	162
9.2 Cheat Sheet (チートシート)	164
9.3 Tour (ツアー)	165

10章 アンチパターン	167
10.1 Novel Notions (新しいだけのアイデア)	168
10.2 Metaphor Mismatch (メタファーの不一致)	175
10.2.1 コントロールのミスマッチ	175
10.2.2 アイコンのミスマッチ	177
10.2.3 メンタルモデルのミスマッチ	178
10.3 Idiot Box (作業を妨げる愚行)	180
10.4 Chart Junk (無駄なグラフ)	182
10.5 Oceans of Buttons (無数のボタン)	186
付録A パターン一覧	189
A.1 主要なナビゲーション	190
A.2 補助的ナビゲーション	192
A.3 フォーム	193
A.4 テーブルとリスト	195
A.5 検索	197
A.6 並べ替え	199
A.7 フィルター	200
A.8 ツール	201
A.9 グラフ	203
A.10 誘導	206
A.11 フィードバック	208
A.12 アフォーダンス	209
A.13 ヘルプ	210
アプリ索引	213
索引	215

1章

ナビゲーション



主要なナビゲーション

Springboardパターン

List Menuパターン

Tabsパターン

Galleryパターン

Dashboardパターン

Metaphorパターン

Mega Menuパターン

補助的なナビゲーション

Page Carouselパターン

Image Carouselパターン

Expanding Listパターン

筆者はモバイルアプリストアでのレビューを読むのをいつも楽しみにしています。レビューを通じて、ユーザーがアプリケーションをどのように利用しているのか知ることができます。このようなレビューの仕組みは従来のWebアプリやデスクトップアプリにはなかったものであり、極めて有益なフィードバックを開発者に与えてくれます。顧客の嗜好やアプリケーションへの期待についても、ここから多くの情報を得られます。

一般的に、4つ星や5つ星のレビューのほとんどは具体性を欠いたものです。例えば「何とすばらしいアプリなのでしょう。見栄えもよく、きちんと機能しています」といったレビューがしばしば見られます。一方、1つ星や2つ星のレビューははるかに具体的であり、ユーザーが遭遇した問題が詳細に述べられています。問題点の多くは以下のように分類できます。

- クラッシュ
- 重要な機能（同期、フィルター、アカウントの連携など）の欠如
- 低機能なナビゲーション（元の画面に戻れない、探しているものが見つからない、など）
- 混乱を招くインタフェースデザイン

これらのうち上の2つはデザインパターンによって解決することはできませんが、残りの2つについてはデザインパターンが確実に役立ちます。この章で紹介するナビゲーションのためのパターンに従えば、ユーザーはアプリケーションに用意されている便利な機能を発見し活用できるようになるでしょう。

1.1 主要なナビゲーション

優れたデザインと同様に、優れたナビゲーションも前面には出てこないものです。優れたナビゲーションを備えたアプリは直感的であり、（それが友達探しであれ自動車ローンの申し込みであれ）どんな作業でも容易に行えるようにしてくれます。アプリの中でコンテンツの間をナビゲートするための手法には多くの選択肢がありますが、ここでは主要なナビゲーション（メインメニューで利用するものなど）のための7つのパターンに着目します（図1-1）。

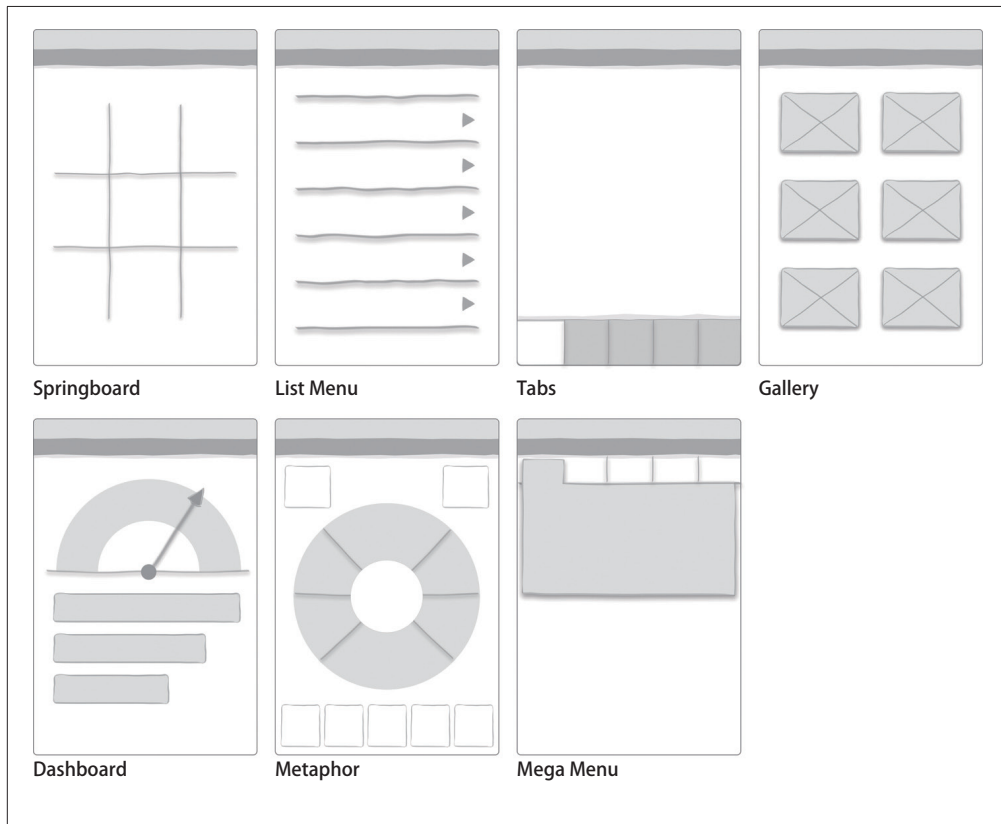


図 1-1 主要なナビゲーションのパターン

1.1.1 Springboard (スプリングボード)

Springboardパターンはオペレーティングシステムに依存しないので、すべてのデバイス上で同じように機能します。Launchpadパターンと呼ばれることもあります。このパターンの特徴は、起動直後のページに表示されるメニュー項目がアプリを利用するきっかけになるという点です。FacebookはiOSのホーム画面と同様にSpringboardパターンを採用しており、他のアプリもこれに追随しました(図1-2、図1-3、図1-4)。



図1-2 Facebookに見られるSpringboardパターンとOvi Maps

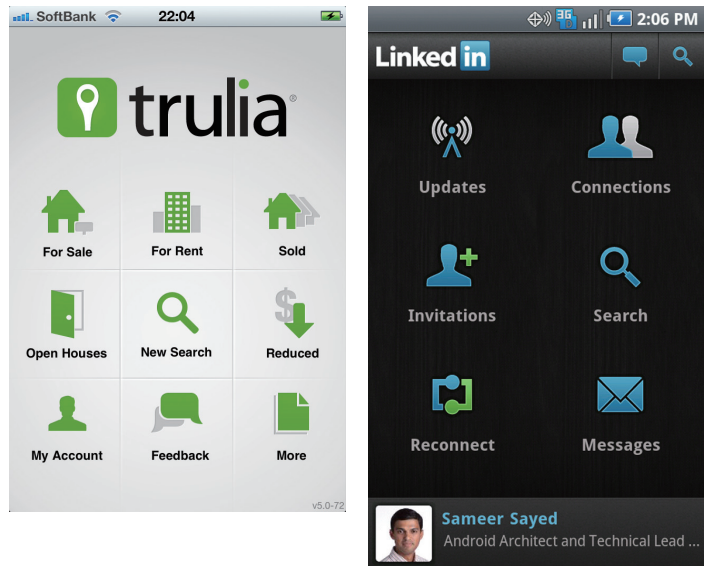


図 1-3 TruliaとLinkedIn

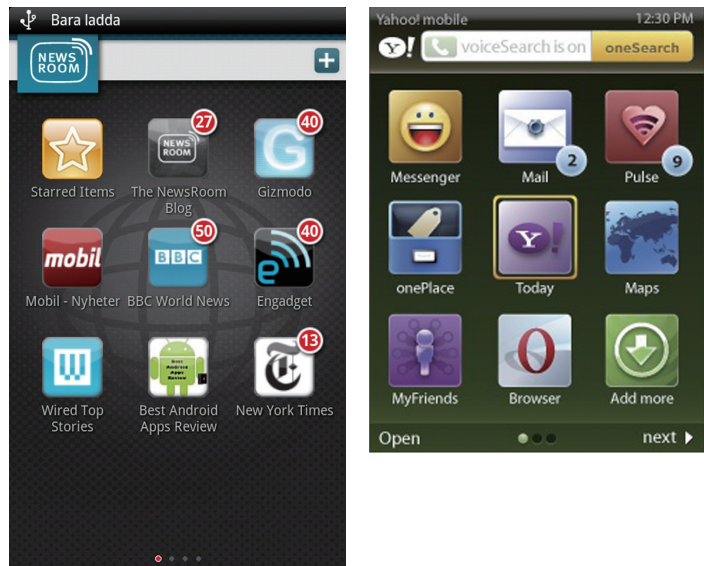


図 1-4 PalmでのNewsRoomとSymbianでのYahoo!

パーソナライズによって、メニュー項目と共に個人のプロフィール情報を表示可能なスプリングボードも存在します。カスタマイズの機能を通じ、スプリングボードのレイアウトを変更できることも少なくありません (図 1-5)。

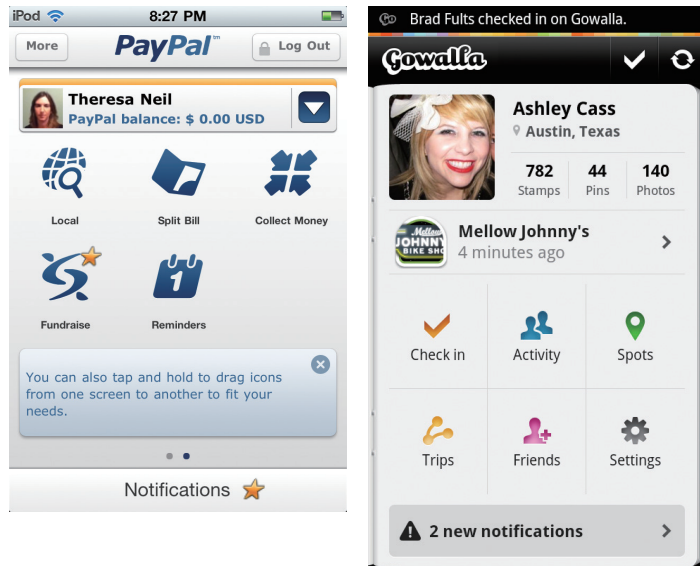


図 1-5 PayPalのパーソナライズされた
スプリングボードと初期のGowalla

グリッド状のレイアウトとしては 3×3 、 2×3 、 2×2 、 1×2 がよく使われます(図 1-6)。しかし、Springboardパターンにとってグリッド状のレイアウトは必須ではなく、重要な意味を持つものについては他より大きく表示されることもあります。例えばiPhone版のMastersでは、動画再生のボタンが2倍以上大きく表示されています(図 1-7)。

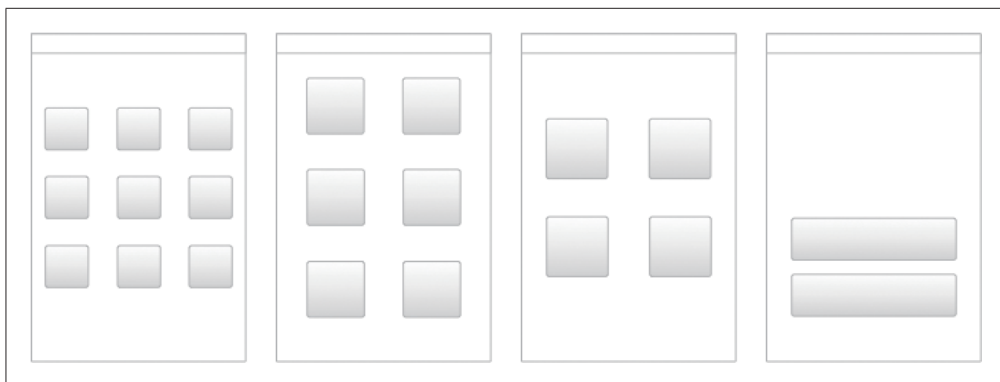


図 1-6 スプリングボードのグリッド状レイアウト



図 1-7 Norton Mobileのグリッド状レイアウト (2×2) とMastersの変則的レイアウト

TIP

それぞれのメニュー項目の重要さが同程度の場合はグリッド状のレイアウトを使い、特定の項目を強調したい場合には変則的なレイアウトを使いましょう。パーソナライズやカスタマイズについても可能かどうか検討しましょう。

1.1.2 List Menu (リストメニュー)

アプリケーションへの入り口の役割を果たしているという点で、List MenuパターンはSpringboardに似ています。例えばパーソナライズされたリスト、グループ化されたリスト、拡張されたリストなど、List Menuパターンには多くのバリエーションがあります。拡張されたリストとは、シンプルなリストに対して検索やブラウズあるいはフィルターなどの機能が追加されたものです(図 1-8、図 1-9、図 1-10、図 1-11)。

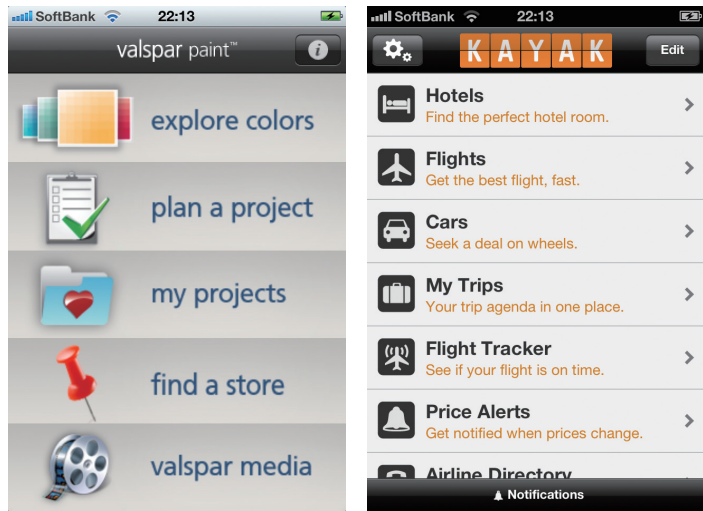


図 1-8 Valspar PaintとKayakのメニューリスト



図 1-9 radiotime PalmとCoziのメニューリスト

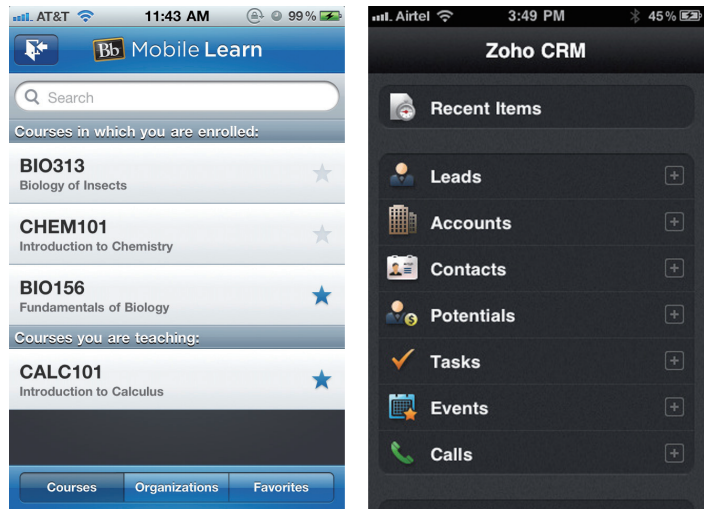


図 1-10 パーソナライズされたリスト (BlackboardとZoho CRM)

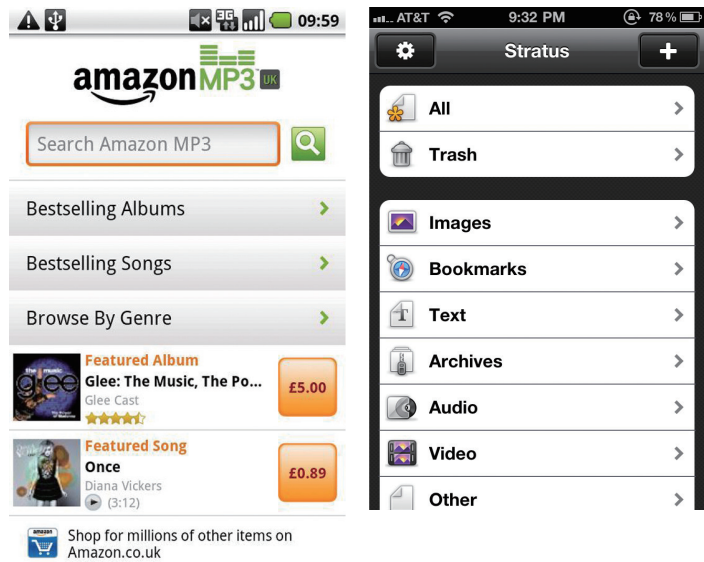


図 1-11 拡張されたリスト (Amazon MP3) とグループ化されたリスト (Stratus)

TIP

項目のタイトルが長い場合や、サブタイトルが必要な場合に List Menu パターンが役立ちます。このパターンに基づいたアプリでは、すべての画面からリストにアクセスできるようにするための仕組みが求められます。一般的にはタイトルバー上に、「メニュー」という文字列が簡条書きのアイコンを含むボタンが表示されます。

1.1.3 Tabs (タブ)

タブの表示位置やデザインについては各オペレーティングシステムが独自のガイドラインを用意しているため、タブによるナビゲーションはオペレーティングシステムごとに異なります (図1-12)。このTabsパターンを採用する際には、それぞれのオペレーティングシステム向けに表示をカスタマイズする必要があります。

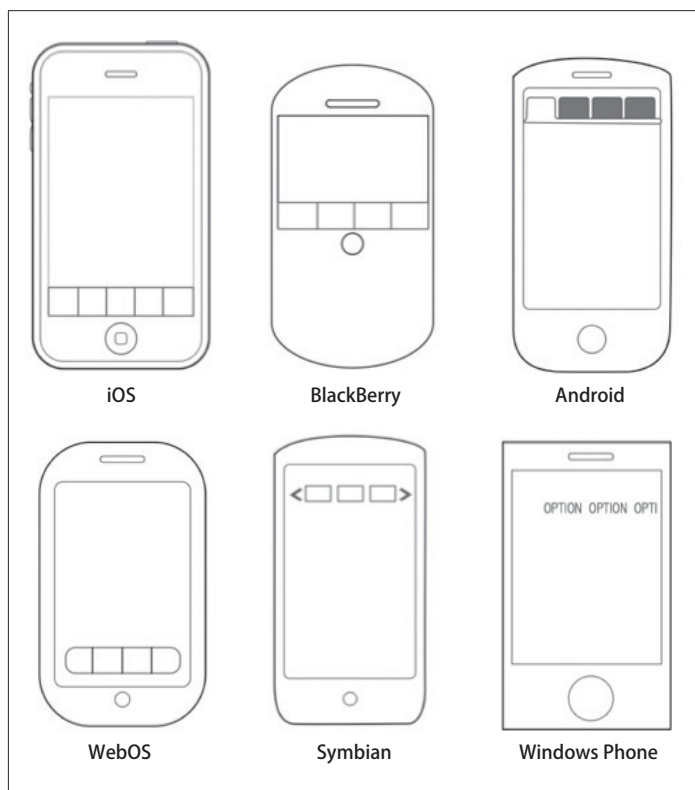


図1-12 オペレーティングシステムごとに異なるタブの表示

iOSとWebOSそしてBlackBerryでは画面下端にタブが表示されます。親指を使った操作にはこの方式が最も適しています (図1-13、図1-14)。

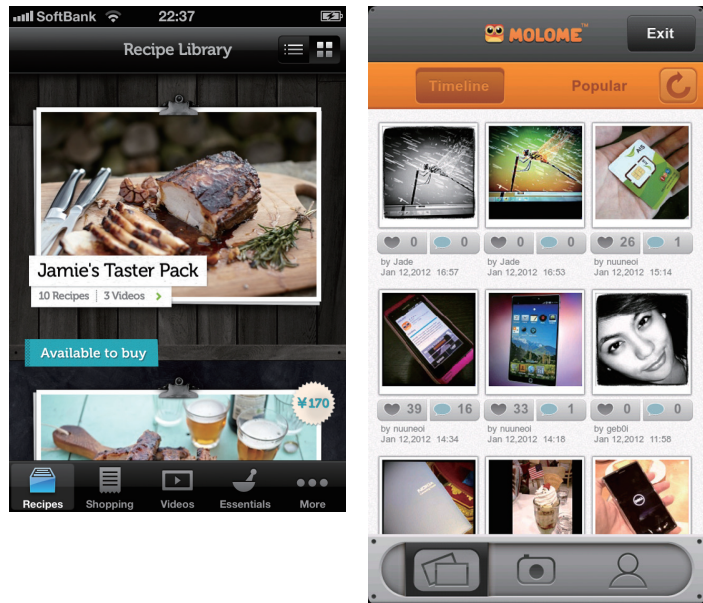


図 1-13 Jamie Oliver Recipes と Molome。画面下端にタブが表示される

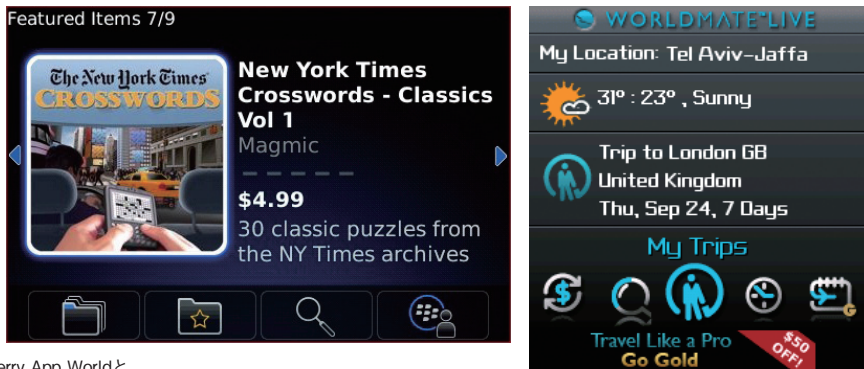


図 1-14 BlackBerry App World と WorldMate。同じく下端の表示

StarbucksやBlue Mobileのアプリのように、タブが左右にスクロール可能な場合もあります(図1-15)。収まりきれない項目のために別に画面を用意する必要がなく便利です。

AndroidやSymbianそしてWindowsでは画面の上部にタブが表示されます。この方式は標準的なWebサイトでのナビゲーションのパターンに基づいており、ユーザーにとって馴染みが深いものです。SymbianとWindowsではタブのスクロールも可能であり、フリックによってより多くの項目を表示できます(図1-16、図1-17)。



図 1-15 StarbucksとBlue Mobile。タブが左右にスクロールする



図 1-16 TimeTractorとSymbianのFring。上部のタブがスクロールする

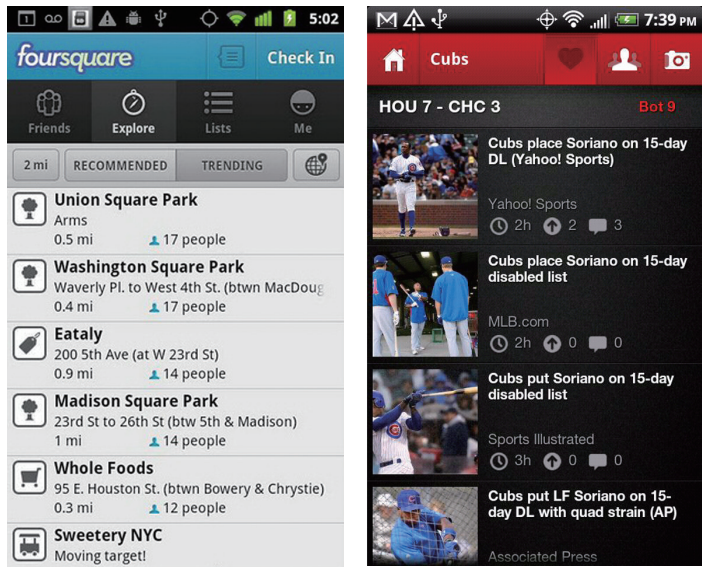


図 1-17 上部に表示されたタブの例。
Foursquare と HitPost

TIP

選択されたタブは視覚的に他と区別して明確に表示しましょう。また、タブには容易に理解できるようなアイコンを表示するか、ラベル付きアイコンを表示する必要があります。

1.1.4 Gallery (ギャラリー)

Galleryパターンでは個々のコンテンツが表示され、これらを通じてナビゲーションが行われます。ここでのコンテンツには記事、レシピ、写真、製品などが含まれます。それぞれのコンテンツはカールセル（「1.2.1 Page Carousel」「1.2.2 Image Carousel」参照）やグリッドあるいはスライドショーとして表示されます（図 1-18、図 1-19）。

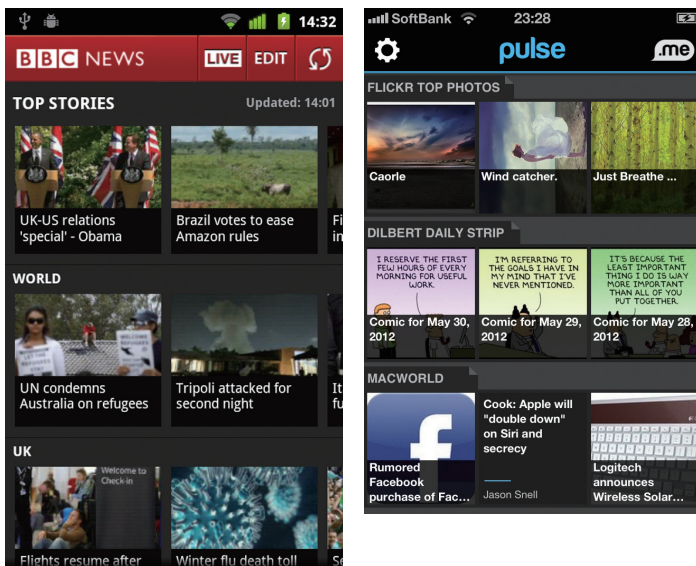


図 1-18 BBCとPulse News

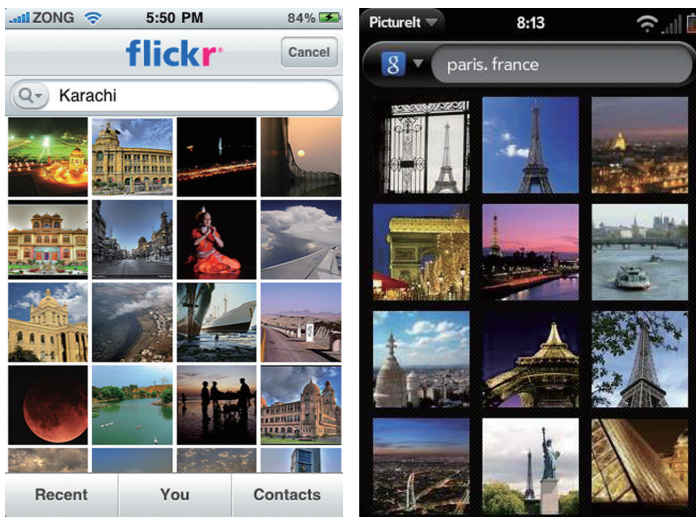


図 1-19 FlickrとPitcorelt

場合によっては、コンテンツをグループ化するとブラウズが容易になります。Dwellでは画面左側にタブが表示され、コンテンツが適度な規模のグループへと分類されています(図1-20)。



図1-20 Dwell

TIP Galleryパターンはコンテンツが頻繁に更新される場合に適しています。

1.1.5 Dashboard (ダッシュボード)

Dashboard (計器盤の意) パターンではKPI (Key Performance Indicator: 重要業績評価指標) が要約して表示されます。それぞれの指標の表示を展開することによって、追加の情報を表示させることもできます。このパターンは財務管理アプリや分析ツール、営業あるいはマーケティングのアプリなどで活用できます (図 1-21)。

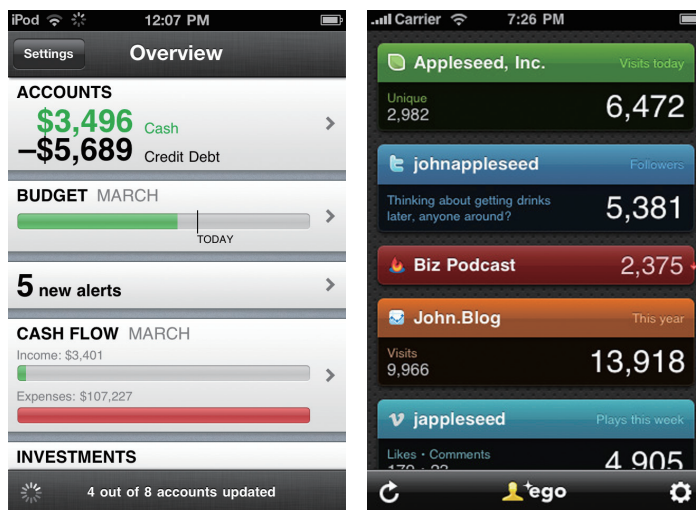


図 1-21 Mint と ego のダッシュボード

TIP ダッシュボードにコンテンツを詰め込みすぎてはいけません。キーとなる指標を厳選して表示させましょう。

1.1.6 Metaphor (メタファー)

このパターンの特徴は、最初に表示されるページが何かのメタファーになっている点です。主にゲームでこのパターンが利用されていますが、ユーザーが何らかの項目（メモ、本、ワインなど）をカタログ化して整理するようなアプリにも適用できます（図1-22、図1-23、図1-24、図1-25）。

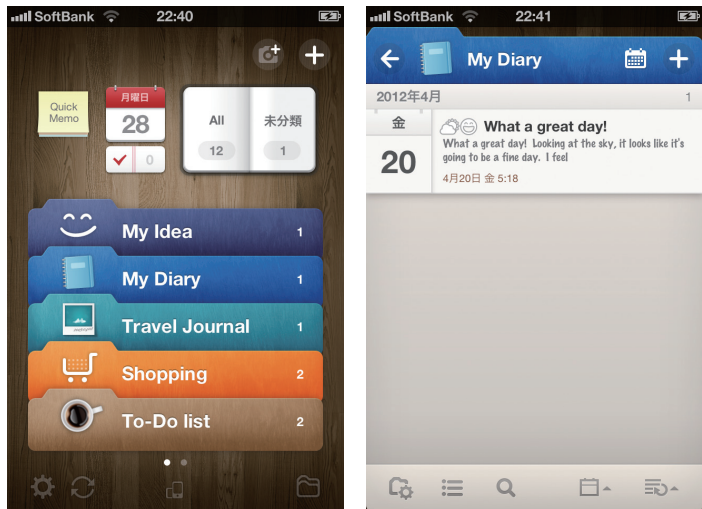


図1-22 Awesome Note



図1-23 Cellar



図 1-24 DoltTomorrow と TripJournal

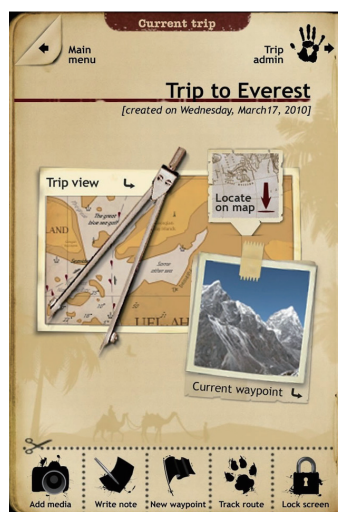


図 1-25 Aldiko Book Reader

Chapter 1

My girlfriend was 15 percent of my age, and I was old-fashioned enough that it bugged me. Her name was Lil, and she was second-generation Disney World, her parents being among the original ad-hocracy that took over the management of Liberty Square and Tom Sawyer Island. She was, quite literally, raised in Walt Disney World and it showed.

It showed. She was neat and efficient in her every little thing, from her shining red hair to her careful accounting of each gear and cog in the animatronics that were in her charge. Her folks were in canopic jars in Kissimmee, deadheading for a few centuries.

TIP 質の低いメタファーは Novel Notion アンチパターン (10.1 節参照) に該当するおそれがあるため、Metaphor パターンの適用は慎重に検討する必要があります。

1.1.7 Mega Menu (大型メニュー)

Mega Menu パターンでは大きなメニューのパネルがオーバーレイ表示されます。ここにはメニュー項目が独自の形式やグループ化に基づいて表示されます。このパターンはWebでも見られます。例えばRipCurlShop.comでは、衣類のサブカテゴリへのナビゲーションにMega Menuパターンが適用されています(図1-26)。

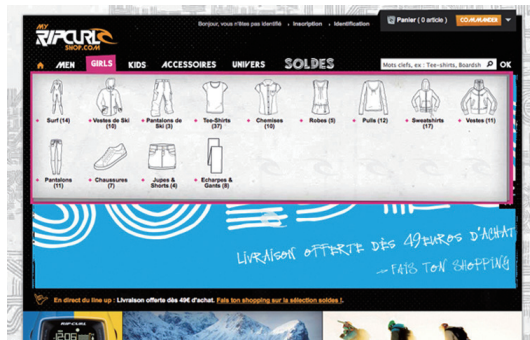


図1-26 RipCurlShop.com

WebOS版のFacebookで使われている大型メニューは、Springboardパターンで見られたような余分なナビゲーションを排し、スムーズなナビゲーションを実現しています。Wal-MartもAndroid版アプリでこのパターンを採用しています(図1-27)。

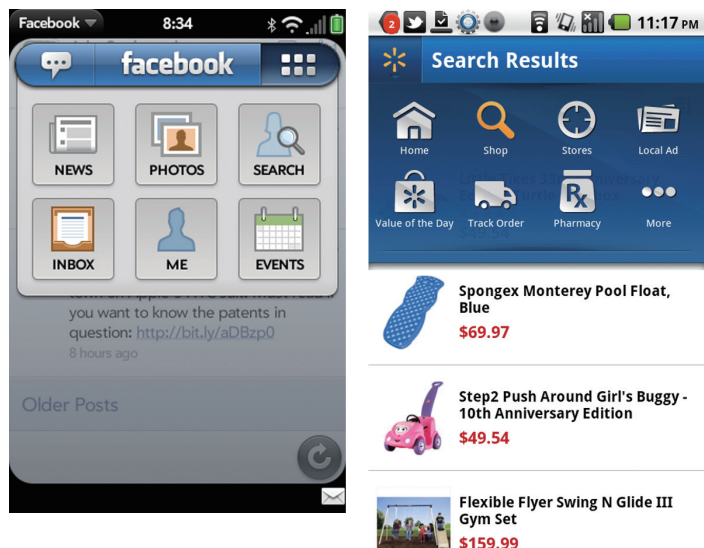


図1-27 WebOS版FacebookとAndroid版Wal-Mart

TIP ナビゲーションのパターンを選択する際には、情報の構造をあらかじめ把握しておく必要があります。アプリの中で主なセクションが数個しかないなら、他のより適切なパターン（例えばTabsパターンなど）を利用すべきです。

1.2 補助的ナビゲーション

メニューについてのパターンだけではこの章は不十分です。ここからは話題を広げ、補助的なナビゲーションについてもいくつかパターンを紹介します。

補助的ナビゲーションとは、ページあるいはモジュールの内部でのナビゲーションを意味します。例えばANZアプリではタブが主となり、スプリングボードが副次的な役割を果たしています（図1-28左）。同様にJamie Oliver Recipesでも、タブによるナビゲーションの補助としてメニューのリストが使われています（図1-28右）。

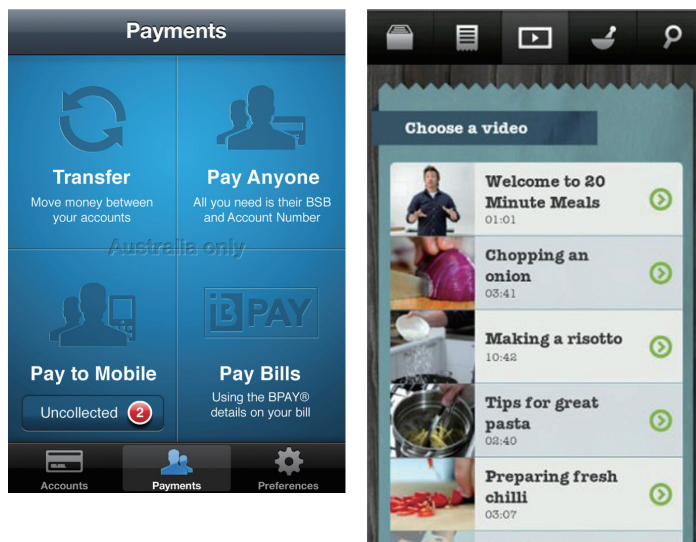


図1-28 タブを補助するスプリングボード(左)とリスト(右)

主要なナビゲーションのためのパターンはすべて、補助的なナビゲーションにも適用できます。タブを別のタブと組み合わせたり、タブとリストあるいはタブとダッシュボード、スプリングボードとギャラリーなどの組み合わせが考えられます (図 1-29)。

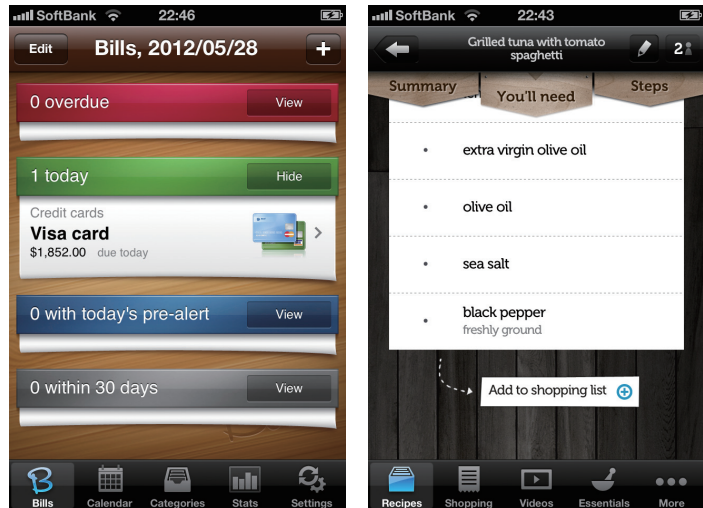


図 1-29 タブを主にダッシュボードを
従にした画面例 (左) とタブの中に別の
タブがある画面例 (右)

主要なナビゲーションには必ずしも適していないけれども、補助的ナビゲーションとしてはよく機能するというパターンもあります (図 1-30)。

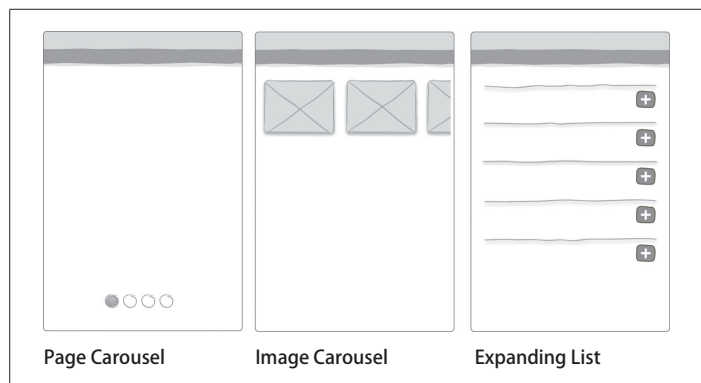


図 1-30 補助的ナビゲーションのため
のパターン

1.2.1 Page Carousel (ページのカルーセル)

Page Carousel (carouselは回転木馬の意) パターンを適用すると、フリックの操作を通じて特定のページに素早く移動できます。ページインジケータ (iOSでの用語。下部に表示される小さな点) によってカルーセル内のページの個数を知ることができ、フリックによって前後のページに移動できます。ここで紹介する4つの例ではいずれも、タブの中にカルーセルが表示されています (図 1-31、図 1-32)。

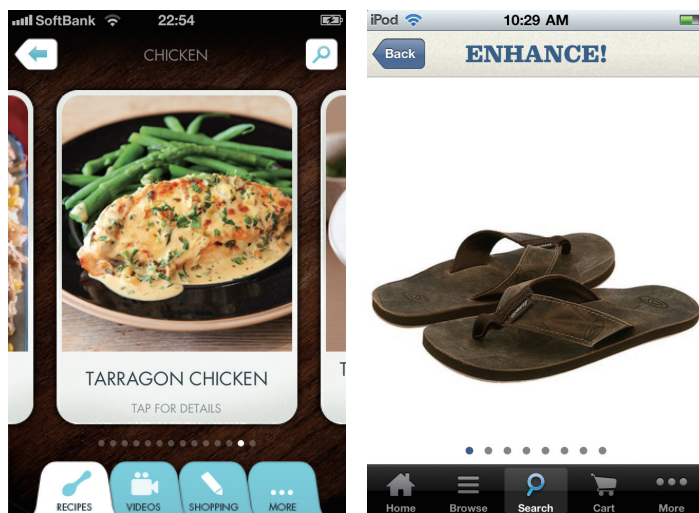


図 1-31 Nigella Quick Collection
と Zappos

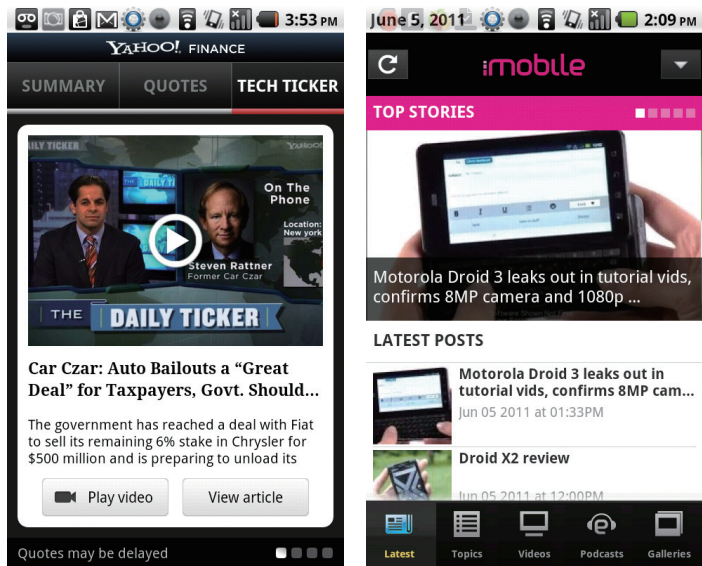


図 1-32 Yahoo! FinanceとiMobile

このパターンには限界もあります。ページが8個以上ある場合は、リストによる表示を検討しましょう(図 1-33)。



図 1-33 ExcellentAnalytics。
ページが多すぎる例

TIP

カルーセルは比較的少数のページでのナビゲーションに適しています。ページの総数や、現在表示されているページの位置を視覚的に表現しましょう。また、カルーセル内を移動するための操作としてはフリックがよく使われます。

1.2.2 Image Carousel (イメージのカルーセル)

画像のためのカルーセルとしては、二次元的な表示だけでなく iTunes での Cover Flow のようなものも考えられます。IMDB では、イメージのカルーセルを使って人気の映画を提示しています。また、筆者は Adobe Flex Showcase 向けに小売サイトのアプリを設計し、その中でイメージのカルーセルを使って注目商品を表示しています (図 1-34)。

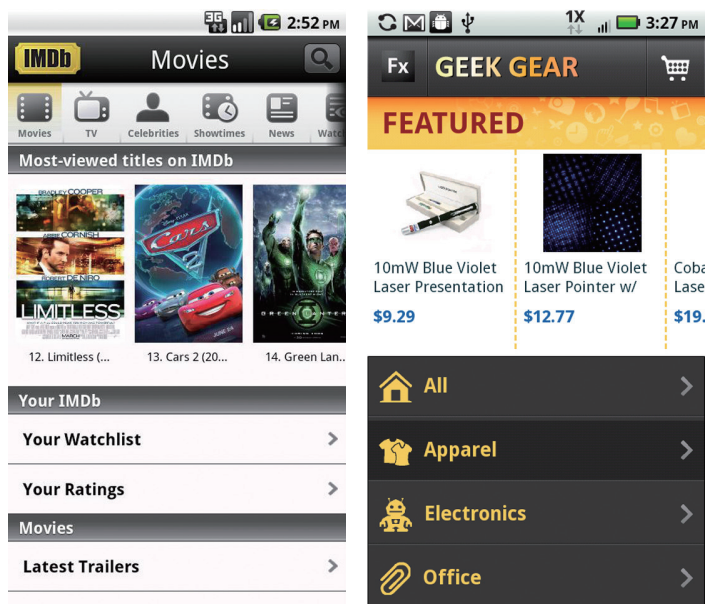


図 1-34 IMDB と Adobe Flex 4.5 Showcase

The Photo Cookbook でもイメージのカルーセルが使われています。ただ、ここではレシピの種類が横方向のカルーセルとして表示され、縦方向の列にレシピがグルーピングされています (図 1-35)。



図 1-35 The Photo Cookbook

Tap'n'Scrapでは、背景や枠の表示を選択する際には二次元のカーセルが使われ、それぞれのスクラップブックをブラウズする際にはCover Flowスタイルの表示が行われます(図1-36)。2つのスタイルがうまく組み合わせられています。

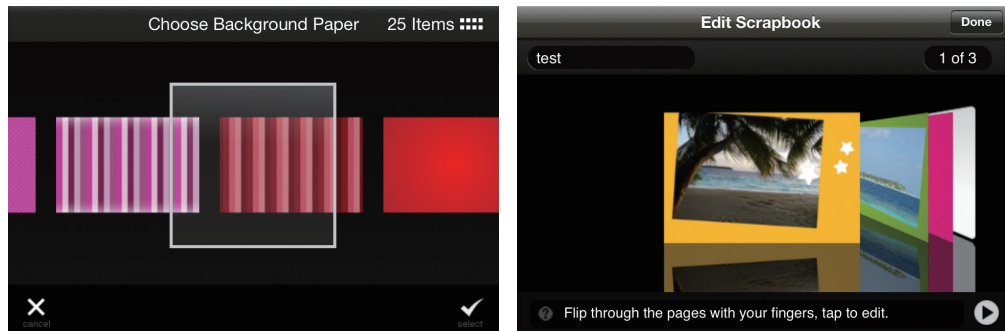


図 1-36 Tap'n'Scrap

ANZのバンキングアプリでは、口座や支払い方法に関する情報がCover Flowとして表示されます。このスタイルは見栄えがよく、デモンストレーションとしては適しているかもしれませんが、複数のカードの間をフリックによって移動しながら支払い方法を選択するというやり方は、初めのうちは物珍しいとしてもすぐに飽きられてしまうでしょう(図1-37)。

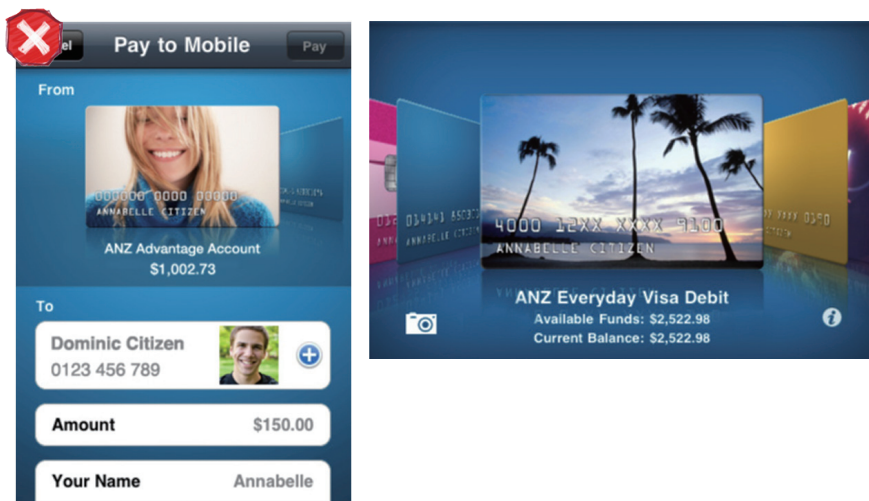


図1-37 ANZで支払い方法を指定するために使われるカルーセル。デモ向きではあるが飽きられやすい

TIP Image Carouselパターンは鮮やかな視覚的コンテンツ（記事、製品、写真など）の表示に向いています。矢印のアイコンや画像の一部あるいはページインジケータ（小さな点の並び）などを通じて、現在表示されているもの以外にもコンテンツが存在することを視覚的なアフォーダンス（8.2節参照）として提示しましょう。

1.2.3 Expanding List (伸縮するリスト)

Expanding Listパターンでは、1つの画面上でメニューのデータ構造の中をたどり、より多くの情報を提示できるようにします。AndroidのGingerbread（バージョン2.3）では、通話の記録の表示にこのパターンが適用されています。同じ番号の相手との通話は1行にまとめて表示され、アイコンをタップすると個々の通話についての情報が表示されます（図1-38）。

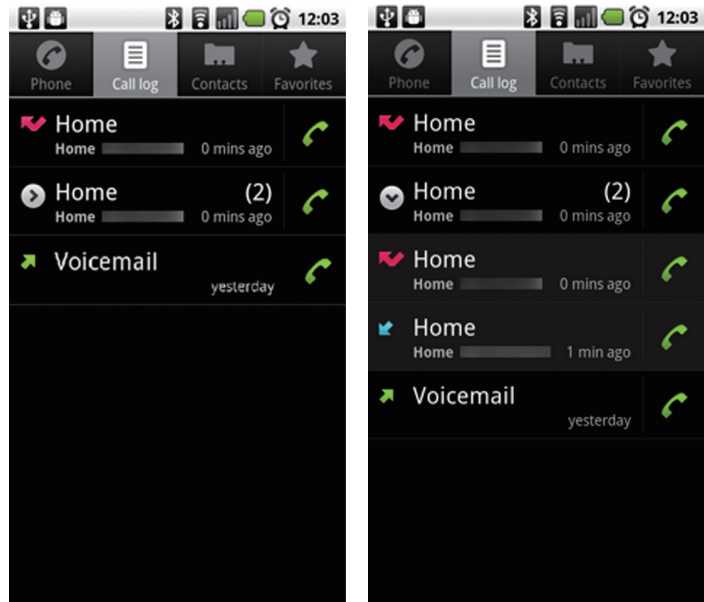


図 1-38 Androidの通話記録

このパターンはモバイルアプリよりもモバイル向けWebサイトでよく見られますが、これらのどちらでもうまく機能します。例えばGap.comのモバイルサイトでは、カスケード（階段）状のリストではなくExpanding Listパターンに基づいて婦人衣料の各カテゴリが表示されています（図1-39）。

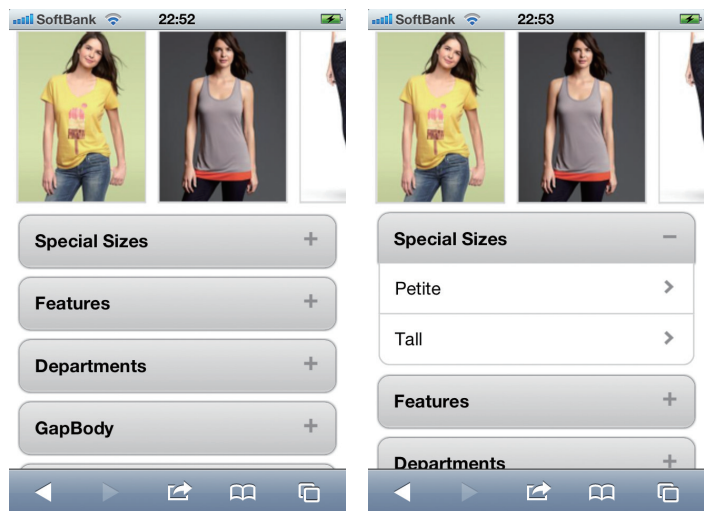


図 1-39 Gap.comのモバイルサイト

TIP

Expanding Listパターンは、対象について徐々に詳細な情報やオプション項目を表示させたい場合に適しています。