

WEB DESIGN PROFESSIONAL WORKFLOW BIBLE

Webデザイン プロフェッショナルワークフロー・バイブル

Andy Clarke [著]

今里了次 [訳] 木達一仁 (株式会社ミツエーリンクス) [監訳]

WD
Web Designing
BOOKS

クリエイティビティを刺激し、最大化する、
新しいWebデザインへのアプローチ。
自分の「スキル」と「デザイン」をバージョンアップしたい
Webクリエイターのためのバイブル!

XHTML+CSSデザインの新スタンダード!



Webデザイン プロフェッショナルワークフロー・バイブル

Andy Clarke [著]

今里了次 [訳] 木達一仁 (株式会社ミツエーリンクス) [監訳]

Transcending CSS: The Fine Art of Web Design

Andy Clarke

Authorized translation from the English edition, entitled TRANSCENDING CSS: THE FINE ART OF WEB DESIGN, 1st Edition, ISBN: 0321410971 by CLARKE, ANDY; HOLZSCHLAG, MOLLY, E., published by Pearson Education, Inc, publishing as New Riders, Copyright © 2007

All rights reserved. No part of this book may reproduced, or transmitted in any form or by any means, electronic, mechanical, introducing photocopying, recording or any information storage retrieval system, without permission from Pearson Education, Inc.

Electronic JAPANESE language edition published by MYNAVI CORPORATION, Copyright © 2014
Japanese translation rights arranged with Pearson Education, Inc. through Owl's Agency Inc.

*To my nana, born the year that the Titanic sailed.
And to my son, Alex; you make me very proud.*

- 本書の解説や画面写真については、基本的に原著刊行時（2006年11月）の情報に基づいています。ただし、ブラウザの対応状況や紹介したURLなどについては、日本語版編集時（2008年12月）段階のものに差し替えています。
- 本書に記載された内容は、情報の提供のみを目的としております。日本語版の制作にあたっては正確を期するようつとめました。株式会社マイナビが本書の内容に関して何らかの保証をするものではなく、内容に関するいかなる運用結果についてもいっさいの責任を負いません。あらかじめご了承ください。
- 日本語版に関してのご質問、ご意見などは、株式会社マイナビまでお願いいたします。
- 本書中に登場する会社名および商品名は、該当する各社の商標または登録商標です。本書では®および™マークは省略させていただいております。

序文

Webデザイナーや開発者たちが直面している最大の問題、その1つが、極端なビジュアル思考派と極端な論理的思考派とのコミュニケーションギャップを埋めることです。知的で革新的なデザインセンスと漸進的で複雑なテクニカルスキルをバランスよく持ち合わせている人など、滅多にいません。ただ、Jeffrey Zeldman、Douglas Bowman、Dave Shea……彼らは類い希な存在で、両サイドの知識とスキルを持ち合わせているだけでなく、その知識とスキルを他の人々に的確に伝えることができ、Web業界の発展に大きく貢献してきました。

数年前、私もNew Ridersから「Integrated Web Design」という本を出版し、その中でこの問題について取り組もうとしました。どうしたら、デザイナーが「セマンティクス」や「文書構造」というものを考えてくれるようになるでしょうか？ どうしたら、プログラムの・分析的思考の人がもっと美について価値を見出したり理解してくれるようになるでしょうか？ デザイナーや開発者間のコミュニケーションだけでなく、マネージメントやマーケティングなどの人たちとのコミュニケーションも容易ではありません。効率的なワークフロー、高度なテクノロジー、魅力的なデザインを実現するための強固なプラットフォームを作り出し、Webを革新的で卓越した職業の場とするためには、私たちはまだまだすべきことがたくさんあります。

この状況が、私をAndy Clarkeへと導いてくれたのでしょうか。Andyの仕事を知って見たのは、数年前、彼のブログを通してでした。「マークアップとCSSの名前付けルール」に関する記事があり、私は長い間その記事が気になっていました。彼の記事やデザイン作品を深く知り始めたとき、「この人だ！」と思いました。Andyこそ、実用的で情熱的なアイデアを持って、デザイナーと開発者の間にある溝を埋めてくれる人だと感じました。私たちは、2005年のあの有名な「Britpack Invasion」ムーブメントの頃に、SXSWカンファレンスで出会い、この問題について話しました。それから数ヶ月の内には、私たちのスキルや経験を通して私たちの仲間にこの問題を伝えて刺激しようと、「CSS for Designers」や「CSS for Developers」などを含む一連のワークショップを始めていました。

Andyと会話する中で、必然的に「本にしよう」という話になり

ました。私はそのことに夢になって、AndyをPeachpit/New Ridersの発行人であるNancy Aldrich-Ruenzelに紹介しました。Andyが何をしたいのか、それを少しでも知れば、New Ridersの「Voices That Matter」シリーズに入れ込んでくれるだろうと分かっていたからです。私たち3人はサンフランシスコで会いました。私をよく知る人なら、その場の光景を信じられないかもしれません。3人でエキサイティングな会話が進む中、私が話に割り込むことをしなかったんですから。話はまとまりました。Dave Sheaと私が手掛けた「The Zen of CSS Design」が成功したこともあって、本にしても上手くいくだろうと判断したのです。Andyと一緒に私たちがのビジョンとコンテンツが形となるようにし、私が編集をして、Daveが前書きを書いて、この本が「よりよいデザイナーや開発者になること」を目指している人たちのリソースとなるように作り上げました。

この本を作る上で、いろいろと困難がありました。ご覧のように、Andyはビジュアル思考の人です。私はと言うと、論理的思考派です。Andyは、多くのビジュアル思考の人がそうであるように、イマジネーションやクリエイティビティの世界で生きています。しかし、私の世界を占めているのは、秩序、情報伝達、プロセスです。皮肉なことに、この本を作る上で私たちが直面し乗り越えなければならなかった困難の多くは、こうした異なる性格の人同士の間に見える壁と似ていました。

しかし、私たちは、信じられないほど美しく、そして確固としたテクノロジーと本物のビジョンを示す本を作ることができました。最初に思い描いていた「このような本にしたい」という形を、すべて思い通りに作ることができたのです。この本こそ、「たとえどんな困難な壁でも、みんながそれを乗り越えようと努力していけば、革命は起こりうる」と証明してくれていると思います。私たちの業界に存在する壁を乗り越えようと一生懸命努力しているみなさんと、私のこの体験を共有できることが、誇りであり、尊敬であり、自信であり、名誉でもあります。この本が、そうした同じ望みを持つみなさんの役に立ち、私たちが望むWebの世界へとつながることを期待しています。

すべての愛を込めて

Molly E. Holzschlag
2006年10月、ロンドンにて

前書き

この本ですが、いきなり読むのではなく、まず“見て”ください。ページをめくってみると、この本のデザインスタイルのセンスを感じ取れるのではないのでしょうか。写真とスクリーンショットを豊富に使った、とても素晴らしいアートディレクションです。また、ソースコードだらけのつまらない見せ方ではなく、写真や画像を使って、うまくポイントが示されています。Webの話題をWebの世界だけでなく、実世界の例を用いて分かりやすく解説しています。他のコード解説本の中には、このように素晴らしい見せ方をしている本はありません。

本書は単なるコード解説本ではありません。デザインとコードの素晴らしい共演、デザイン案を実現するためのコードの在り方、現実には縛られている考えを解き放つ方法について解説しています。

この本の中でも触れていますが、Yahoo!のような巨大オンラインスペースでは、古いWebブラウザとの後方互換を維持するためのテクニックを使い始めていますが、企業というものは新しいWebブラウザの特徴を活かして新しい機能をユーザーに提供の方が大切ではないでしょうか。私たち自身が手掛けるプロジェクトでそうした新しい機能を提供するためのテクニック、それをどう使えばいいのかをAndy Clarkeが紹介してくれます。

この本を見ると2005年の始めに出版した「The Zen of CSS Design」のことを思い出します。本書では編集を担当している

Molly Holzschlagが、共著者として参加してくれました。「The Zen of CSS Design」では、CSSについて、その構造的なコーディングの観点と、そしてデザイン理論に基づいた素晴らしいデザインサンプルを通して解説しました。本書は、「The Zen of CSS Design」を読み終えた人の次のステップとして最適だと思います。その次とは何か？ Clarke先生が登場して、答えを教えてください。

Zen Gardenを通して、CSSベースのデザインでも美しいデザインが可能であることを証明できたと思います。また、その可能性はビジュアルデザイナーの手によって大きく広がることも。そして、Transcendent CSSは、今のWebはまだスタート地点に立ったばかりで、未来へと目を向けなければならないことを教えてください。

この本は、他の技術マニュアルで語られた内容をまとめた本ではありません。この本があなたに求めているのは、本のコンセプトを理解し、今のWebを次の高いステージへと導くことです。

Transcendent CSSのコンセプトは、マニフェストである「Transcend the Web of today (今のWebを超越しよう)」です。過去に執着するのではなく、前を向き未来を見つめるのです。そのためにはどうすればいいのか、Clarke先生が教えてください。

Dave Shea
2006年10月

謝辞

1997年に私がこのWeb業界で仕事を始めた頃、Web制作に関する知識は余りありませんでした。2000年にWeb標準の存在を知ってからは、たくさんの人々の作品や知識を材料に勉強させてもらいました。ここに名前を挙げられないくらいたくさんの人たちから学びました。貴重な知識を共有してくれたみなさんに、私の愛と尊敬の念、そして深い感謝の気持ちを贈りたいと思います。

この本の執筆は非常に大変な作業ではありませんでしたが、人生の中で最もやりがいのあるチャレンジでもありました。出版社であるPeachpit/New Ridersのみなさんの忍耐力と理解力には、いくら感謝してもしきれません。Peachpit/New Ridersで私の担当してくれたKaryn JohnsonやコピーエディターのKim Wimpsettは、一言一句読んで編集してくれただけでなく、たくさんアドバイスを与えてくれて、私が行き詰まった時には励ましてくれました。Kim ScottやCharlene Charles-Will、Kate Reberのおかげで、私のデザインを実用的なものへと上げることができました。私を信頼し続けてくれたNancy Aldrich-Ruenzelに、永遠の尊敬の念を示したいと思います。おそらく、この本が完成して、私と同じくらい喜んでくれていることでしょう。

テクニカルエディターのAaron GustafsonやMark Boultonと一緒に仕事をできたことをうれしく思います。比類なき人物、Molly E. Holzschlagと一緒に仕事をできたという誇りは、言葉で言い表せません。彼女は約10年に渡ってリーダーまた教育者としてWeb標準に携わり、プロフェッショナルのトップとしての地位

を築いています。Mollyは、私をNew Ridersに紹介してくれただけでなく、バラバラであった私のアイデアが本としてまとまるようにサポートしてくれました。

私はパーカーを着て後ろに乗っていただけ、彼らがスクーターを運転してくれたんです

また、César Acebalやスペインのオビデオ大学のみなさんにも大変感謝しています。彼らの革命的なスクリプト「CSS3 Advanced Layout prototype scripts」のおかげで、この本を仕上げることができました。世界の8番目の不思議と言ってもよい「Flickr」の存在がなければ、この本の挿絵として使っている作品の提供者である、才能に溢れたアーティストや写真家のみなさんに出会うことはありませんでした。クリエイティブコモンズによって作品を提供してくれたみなさんに感謝します。特に、ペインティングを表紙に使わせてもらったRon Huxley、思わず唾を飲み込みたくなるほどおいしいそうな食べ物の写真を使わせてくれたLara FerroniとEric Fung、そして素敵な写真集から好きなだけ選ばせてくれたDave SheaとMolly E. Holzschlag、彼らに感謝しています。そして、さらに特別の感謝を贈りたいのが、肖像写真を撮ってくれたPatrick Laukeです。実際の私よりもかなりクールに映っています。

最後に、私の妻、Sueへ。辛い時にも側にいてくれて、我慢強くサポートし続けてくれた妻へ愛を贈ります。

私に教え、靈感を与え、サポートしてくれた以下の人々に感謝します。あなたたちなしにはこの本は生まれませんでした。

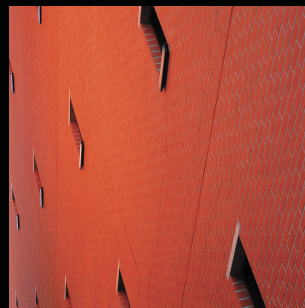
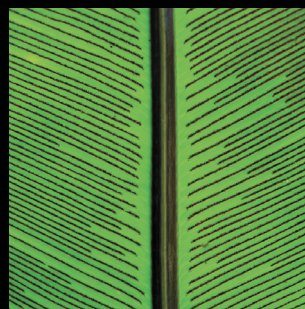
adactio.com、airbagindustries.com、allinthehead.com、andybudd.com、boxofchocolates.ca、brothercake.com、brucelawson.co.uk、cameronmoll.com、clagnut.com、collylogic.com、dean.edwards.name、hicksdesign.co.uk、ian-lloyd.com、jasonsantamaria.com、joelclark.org、joshuaink.com、markboulton.co.uk、meyerweb.com、mezzoblue.com、mikeindustries.com、molly.com、rachelandrew.co.uk、simplebits.com、shauninman.com、sidesh0w.com、simon.incutio.com、splintered.co.uk、stopdesign.com、subtraction.com、tantek.com、themaninblue.com、veen.com/jeff、veerle.duoh.com、vivabit.co.uk、westciv.typepad.com、zeldman.com

1

C O N T E N T S

Discovery 発見 010

1. Transcendent CSSとは?	013
1-1. 何かツールが必要か?	013
1-2. Transcendent CSSが必要となる理由	014
1-3. CSSがクリエイティブの可能性を広げる	018
1-4. アクセシビリティはセールスポイントではなく、デザインの一部	020
1-5. Transcendent CSSへ向けて	024
2. Transcendent CSSの原則	031
2-1. すべてのWebブラウザが 同じデザインを表示できるわけではない	032
2-2. 利用できるCSSセレクタはすべて使用する	033
2-3. 未来に目を向けてCSS3を使用する	039
2-4. CSS未対応部分はJavaScriptやDOMで補う	039
2-5. ハックやフィルタは使用しない	040
2-6. 意味的な命名規則やマイクロフォーマットを使用する	044
2-7. みんなとアイデアを共有し協力し合う	055
3. 整ってきたTranscendent CSSへの環境	059
3-1. 予期せぬCSSの広がり	061
3-2. 段階的なブラウザサポート	063
3-3. 発見、プロセス、インスピレーション、 そして卓越(トランスデンス)	065
4. まずコンテンツの中身からデザインする	067
4-1. コンテンツありきのアプローチ	069
4-2. 最適化されていないCSSレイアウトの典型例	070
4-3. スタイルシートの有無に関わらず 最適化されたコンテンツ	072
5. セマンティクスとは“意味付け”すること	077
5-1. CSS Naked Day	077
5-2. “意味”を“マークアップ”に訳す	078
5-3. コンテンツの伝えること	088
5-4. 意味付けを意識する	088
6. 世界をマークアップしよう	091
6-1. 世界全体がリスト:すべてのモノがリストの構成パーツ	091
6-2. 視野に入るものをリストにする	094
6-3. サンフランシスコから届くカード	098
6-4. 身近にあるものを意味付けしていく	100
6-5. マークアップは“中身”から考えていく	100
7. 学んだことを実践する	109





Process プロセス 110

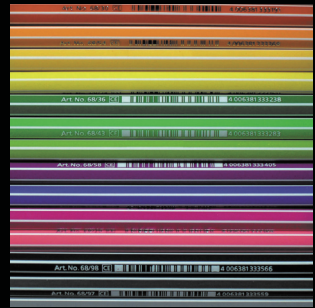
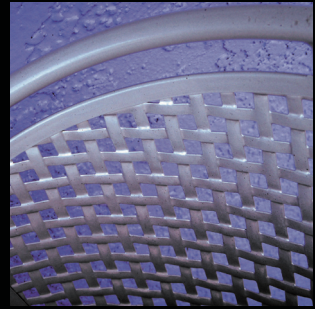
1. 完璧なワークフローを探し求めて	113
1-1. より良い手法を求めて	114
1-2. コンテンツを基盤としたプロセス	116
2. コンテンツの収集	119
3. ワイヤフレームを使った作業	121
3-1. 従来のワイヤフレームの問題点	122
3-2. ワイヤフレームにコストをかける価値はあるか?	124
3-3. 従来のワイヤフレームとインタラクション	125
4. グレーボックス手法を使ってワイヤフレームを改善	129
4-1. 詳細を説明するためのビジュアル素材	131
5. 静的なデザインの作成	133
5-1. ワークフローをより速く進める	133
5-2. 静的なデザインにマークアップガイドを追加	134
6. インタラクティブ・プロトタイプを活用する	137
6-1. インタラクティブ・プロトタイプはリアリティをもたらす	139
6-2. 再利用できるコードを用意しておく	139
6-3. ワイヤフレームとプロトタイプのお手本	140
6-4. WYSIWYG……目先のことだけ見てませんか?	140
7. インタラクティブ・プロトタイプのベストプラクティス	143
7-1. 開発時に利用するWebブラウザ	143
7-2. Webブラウザの拡張機能を活用する	143
7-3. div要素の使用は最小限に	146
7-4. マークアップが妥当かどうかを随時検証	146
7-5. floatではなくpositionプロパティでレイアウトを行う	146
7-6. CSSを整理する	148
8. プロセスを実践する	153
8-1. 材料のチェック	153
8-2. コンテンツありきの考え方によるマークアップ	154
8-3. CSSで静的なデザインを実装する	169
8-4. レイアウトの構築	170
8-5. body要素から作業を開始する	171
8-6. 自分で続けるか、他のスタッフに引き継ぐか	177
8-7. カラーの基本スタイル	177
8-8. ブランディングエリア	178
8-9. ロゴの追加	178
8-10. ナビゲーションのスタイル	180
8-11. フッタのスタイル付け	183
8-12. タイポグラフィのスタイル	186
9. まとめ	189

2

3

Inspiration インスピレーション 190

1. グリッドベースデザイン	193
1-1. デザイナーとグリッド	194
1-2. 黄金比と3分割の法則	197
1-3. 完全にフレキシブルなレイアウト	206
1-4. 合理的なグリッドデザイン	208
2. 最新のWebデザインに見るグリッド	211
2-1. Subtraction	211
2-2. Airbag Industries	214
2-3. Jeff Croft	216
2-4. Veerle's Blog	218
3. Webデザイン以外でのグリッド	221
3-1. 新聞から学ぶ	221
3-2. 伝統的な8カラムデザイン	221
3-3. 現代の6カラムデザイン	222
3-4. その他のカラムデザイン	223
4. 新たなグリッドをWebデザインへ	231
4-1. ニュースサイトのリデザイン	231
4-2. 垂直方向のカラムと水平方向のベースライン	232
4-3. デザインの細部にはマイクログリッドを使う	235
4-4. ベースライングリッドを使った水平方向の配置	235
4-5. 写真をフレキシブルにする	236
5. 予想外の場所で見つかるインスピレーション	239
5-1. ホームページの模様替え	239
5-2. ムードボードのスヌメ	242
5-3. デザインのスクラップブックを作ろう	244
5-4. 雑誌からインターフェイスの インスピレーションを得る	249
5-5. タイポグラフィのインスピレーション	249
5-6. Flashデザインの再検討	251
5-7. 画像や写真からインスピレーションを得る	253
6. より素晴らしいデザインを生み出すための活動	259
6-1. 魅力的なビジュアルを作るだけがデザインではない	259
6-2. 技術だけに目を向けない	260
6-3. ムードを高める	263
6-4. 今までとは異なる視点を持つ	265
6-5. 目的を持ってスクラップブックを作る	269
7. Webデザインを芸術へと高めるために	271
7-1. デザイナーにはまだすべきことがある	272



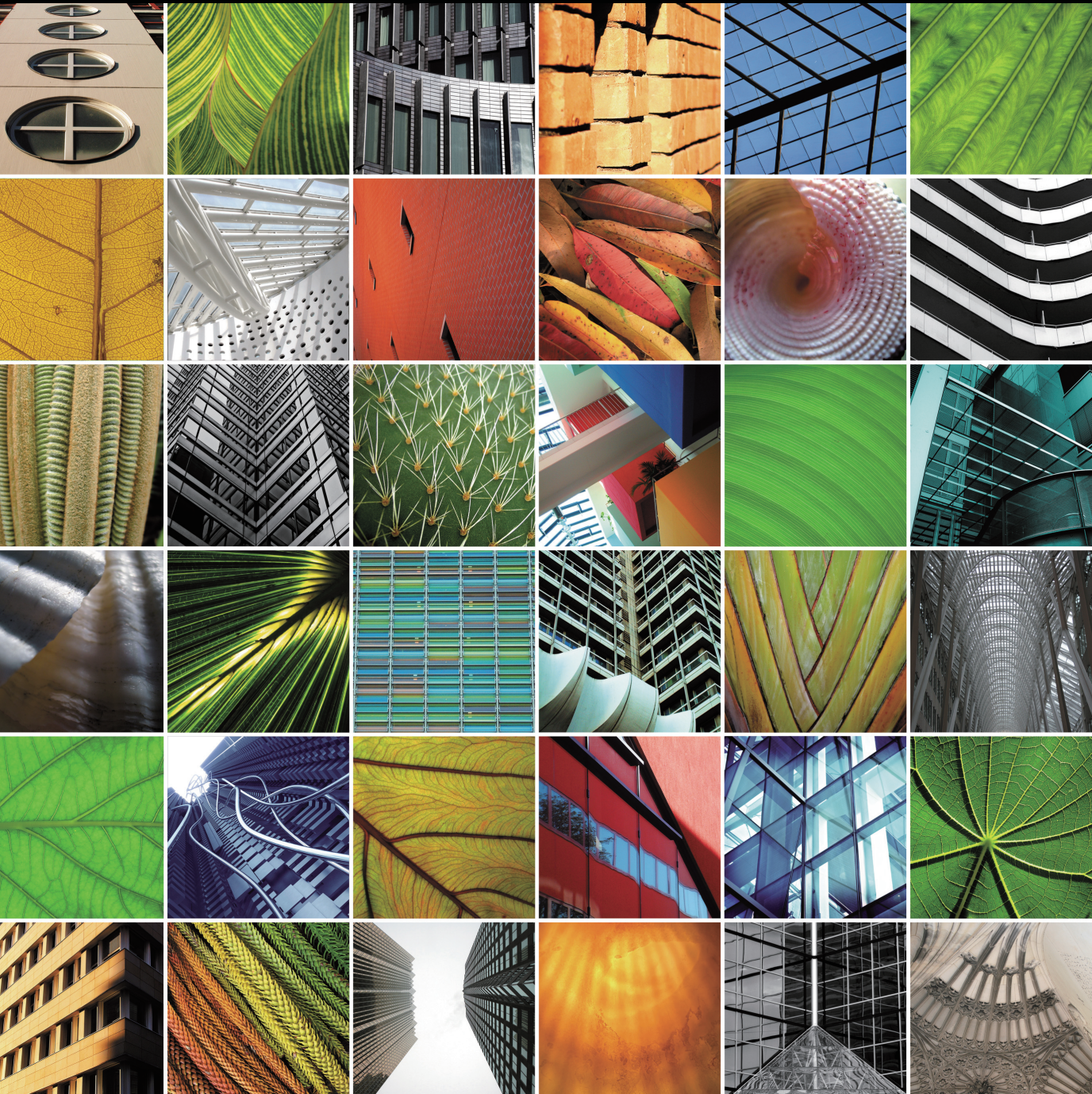


Transcendence

卓越 274

4

1. Transcendent CSS	277
1-1. 絶対配置	278
1-2. 相対配置	290
1-3. floatを使ったクリエイティビティ	296
1-4. サイドバーを作る	302
1-5. テクニックを組み合わせる	304
2. CSS3 (3度目の正直)	317
2-1. CSS3のモジュール	317
2-2. 新しい標準策定に携わる	319
2-3. バック・トゥ・ザ・フューチャー	320
2-4. CSS3のセレクタモジュールを用いたデザイン	320
2-5. 縞模様で読みやすさを改善	321
2-6. 背景&ボーダーモジュールを使う	324
2-7. 複数背景画像でデザインする	328
2-8. マルチカラムのレイアウト	339
2-9. マルチカラムレイアウトモジュールでデザインする	344
3. アドバンストレイアウト	349
3-1. グリッドへの回帰	350
3-2. アドバンストレイアウトモジュールを使って デザインする	354
4. 最後に	367
Credit	368
監修者あとがき	370
Index	371



Chapter1

Discovery

発見

Webデザインのプロセスを違った角度から考える
ワークフローの改善方法を発見する
エキサイティングな可能性に期待する



1. Transcendent CSSとは？

「Transcendent CSS (卓越したCSS)」という言葉をはじめて聞く人も多いでしょう。

Transcendent CSSとは、最先端の素晴らしいCSSテクニックのことを指しているだけでなく、「そうしたCSSを学ぶ中で得た知恵や経験を活かして立派なアーティストやデザイナーになろう」という姿勢でもあります。古いものよりは新しいものに目を向けて新しい視点でインスピレーションを受けるようにする、現状に満足せずより機動的で適切なWebデザインワークフローを生み出せないか考える、たえずデザインとテクノロジーについて学ぶようにする、こうした姿勢を持つことがTranscendent CSSなのです。

1-1. 何かツールが必要か？

このTranscendent CSSや本書で説明している内容を仕事に取り入れたい場合にどんなツールが必要になるかというところ……何も必要ありません。すでにあなたが持ち合わせている知識があればおそらく十分でしょう。スパナやモンキーレンチは要りませんのでご心配なく。特別なソフトウェアや新しいサーバ構成も必要ありません。

とはいえ、本書は初心者の方でも分かるというものではありません。読者がある程度のXHTMLやCSSの知識を持ち、Web標準の基本コンセプトを理解しているという前提で解説しています。もしあなたが未だにテーブルレイアウトを行っているレベルだとしても、本書を通じてCSSのセレクタやプロパティの基本を学ぶことはないでしょう。そうした基本的な内容については、素晴らしい参考書がたくさんあるのでそちらで学んでください。でも、あなたがCSSの初心者であっても、本書で紹介しているコンセプトや事例を見ることでインスパイアされ、「CSSをもっと深く学びたい」と感じてくれるのではないかと期待しています。

また、本書を読み進める上で、みなさんにはオープンマインドでいて欲しいのです。解説している内容のすべてがみなさんの制作環境や仕事内容に当てはまるとはいかないでしょうが、ぜひ、新しいアイデアとして受け入れてください。それをあなた自身に合ったスタイルへとアレンジすることができるはずです。そして何よりも大切なのは、「次世代の手法を習得して斬新でエキサイティングなWebを創りたい」という願望を持つことです。

1-2. Transcendent CSSが必要となる理由

私はデザイナーで、デザインをすることが好きです。できれば、いつかはクラシックカーやAppleのiPodのようにみんなから愛される象徴的なものをデザインして……そうしたらたくさんのお金が入ってきますし、ガレージをクラシックスクーターや60年代のMiniでぎっしりにしたいですね。妄想はさておき、実際は、私はガレージを持っていませんし、今のWebデザインという仕事が大好きで満足しています。

でも、常に楽しいというわけではありません。何日もフラストレーションが続くので、コンピュータをスパッとシャットダウンさせて、何か全く違うことをしに出かけることが結構あります。たとえば、ひとつふたつの問題が解決せずに悩み苦しんだ後、ラジオと大きく成長したネギだけを持って物置小屋の中で何時間も過ごしたことが幾度かありました。そこでネギに話しかけるわけではないですが、自分自身と会話をし続けていると悩んでいた問題の解決方法が見つかり、またWebデザインへの意欲が戻ってくるのです。Webデザインがこんなにチャレンジングな仕事だとは思っていませんでした。「そこに山があるから」といって危険を伴うエベレスト登頂に望む登山家はいますが、そんなチャレンジ精神を持ってWebデザインを始めた人はいないと思います。

しかし、Webデザインという仕事の多くは、私のようにビジュアルで物事を考えるデザイナーにとっては難しいものです。毎日、アドビのPhotoshopやFireworksのようなビジュアルツールを使ってスクリーン上で自由にデザインしているだけに、今のWeb構築技術の多くは、特に文書構造を意識したHTMLマークアップやCSSの記述方法は分かりにくく、直観でやろうにもうまくできません。



CSSはデザイナーフレンドリーではない

テクノロジーというのはデザイン上の問題を解決するために生まれてくるものですが、CSSはあまりデザイナーフレンドリーとはいえません。その一因として、CSSはデザイナーが扱うものでありながら、デザイナーではなく技術者によって開発されたからではないかと思います。CSSの基本原則は以下のように非常にシンプルです。

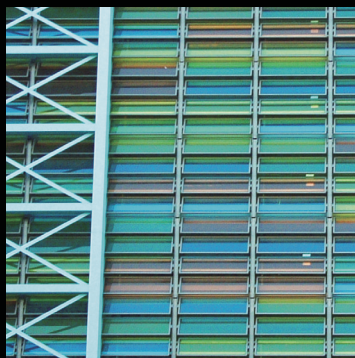
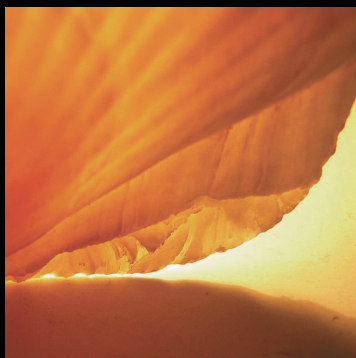
```
p { color : #000; }  
body { background-color : #fff; }
```

しかし、デザイナーにとって、「カスケード」や「継承」という技術用語を理解するのは難しいでしょう。それ以外にも、絶対配置やマージンの相殺、ボックスモデルといったルール、そうしたものがデザイナーには分かりにくく、CSSを習得するまでにかなりの時間がかかっているようです。

CSSが公開されてから長く、技術者がCSSを牽引した時代が続きました。彼らは、「specificity（詳細度：各セレクタの優先度を数値で表す方法）」といったCSSの概念を理解することに努力すると同時に、クロスブラウザで同じデザインを実現するために必要な「CSSハック」を無数に考案してきました。しかし、それによってビジュアルデザイナーと（デザインのコーディングを行う）開発者の知識的なギャップが埋まることはなく、未だデザイナーはCSSにフラストレーションを持ったままです。

Webブラウザの対応不足がCSSの進展を妨げた

CSS自体の難しさだけでなく、Webブラウザの問題もあります。マイクロソフトのInternet Explorer 3が主要WebブラウザとしてはじめてCSSを実装して以来、CSSの導入にはイライ



ラが続いています。Webブラウザのバグ、レンダリングエラー、そして特定のWebブラウザのおかしな挙動にはとても苦しめられました。

ただこの状況も時間とともに解消されてきています。Netscape 4.xはInternet Explorer 3よりも良くなりましたし、Mac版のInternet Explorer 5では初めて「CSSが使える」と言えるようになりました。そして、今ではいほとりどりのバグ名が列挙されるほどのInternet Explorer 6も、リリースされた当時は、他のどのWebブラウザよりもCSSをサポートしたWebブラウザでした。

私には14才の息子がいて、Webデザインの道へ進み始めたところです。数年後、息子が「double-margin float（「マージン2倍バグ」。バージョン6以前のIEで、floatプロパティで浮動化したボックスの浮動させた方向のマージン量が倍になってしまうというバグ）」「peekaboo bug（「いないいないばあバグ」。バージョン6以前のIEで、floatプロパティで浮動化されたボックスを内包したボックスの一部が表示されたり、されなかったりするバグ）」といったバグ対応の記事を読むと、そんなことがあったのかと笑うかもしれません。また、今のデザイナーが「3pxのズレ（バージョン6以前のIEで、floatプロパティで浮動化したボックスと隣接するはずのテキストとの間に3ピクセルの隙間ができてしまうバグ）」対応を当たり前のようにやっていたと聞くと不思議に感じるのではないのでしょうか。

個人の力を見くびってはいけない

Webブラウザの改善とともに、Tantek Çelik、Todd Fahrner、Eric Meyerといった開発者の努力により、デザイナーにも道が開けてきました。彼ら開発者が生み出した知識やテクニックを使い、Douglas BowmanやDave Sheaといったデザイナーが手掛けた作品は、CSSの導入が単なる理想ではなく、実践可能であることを示してくれました。最初のCSS仕様が公開されてから数年間、デザイナーが日々直面するCSSの問題に対して、その解決策を見つけようと世界中の人々が取り組んだのです。他にも忘れてはいけないのが、Rachel Andrew、Molly E. Holzschlag、Dori Smithらが取り組んでいる草の根的な「Web Standards Project」の存在です。彼らはWebブラウザベンダーやソフトウェアデベロッパーとコミュニケーションをとり、また知識を広めようと努力することで、Web標準の重要性を認知させることに大きく貢献しました。こうした小さなグループでありながらも努力した個々の存在がなければ、CSSが今日のようにWebデザインのパワフルなツールとなることはなかったでしょう。CSSが登場した初期の頃に比べれば、今のCSS導入はそれほど難しくはありません。これからこの業界に入ってくる新しいWebデザイナーの人たちは、テーブルでレイアウトすることもなく、Netscape 4のような古いWebブラウザでCSSレイアウトを実現しようと苦しむこともないでしょう。

▶▶▶NOTE ビジュアルWebツールの物足りなさ

DreamweaverのようなビジュアルWebオーサリングツールはバージョンアップの度に、ツール自身が生成するHTMLやCSSのソースを改善してきましたが、どのツールもWebのビジュアルデザイナーが直面している大きな問題を解決するには至っていません……自由にデザインしたいけれど、それを文書構造を意識したHTMLマークアップで実現できないのです。今日では、プロのWebデザイナーにとって、文書構造を意識したHTMLマークアップやCSSの基礎を理解することは、必須条件となっています。それだけに、ビジュアルWebオーサリングツールには、もっとデザイナーがデザインからコードを、コードからビジュアルイメージを理解できるようなツールになってほしいものです。

“アクセシビリティを謳うサイトの中には、
見た目がかなりひどいものがあります。
でも、その原因は、アクセシビリティの確保を目指したためではなく、
それらのサイトをデザインしたデザイナーにあります。
そもそもアクセシビリティはビジュアル要素に影響するものではありません。
同じことが「Web標準」についてもいえると思います。
W3Cのサイトのデザインを見て、
XMLやCSS2を必死に習得したいと感じる
デザイナーの人はいないでしょう。”

JEFFREY ZELDMAN

『Designing With Web Standards 第一版』 2003年5月

1-3. CSSがクリエイティブの可能性を広げる

最近では、多くのWebブラウザがCSSなどのWeb標準をかなりのレベルでサポートするようになり、しっかりとした土台の上でデザインをすることができるようになりました。今は、じっと我慢しているのではなく、新しいデザインを目指し前へと動き出す時なのです。

テーブルレイアウトはその構造上、固定のグリッドデザインしかできず、いわゆる2カラム〜3カラムデザインばかりになってしまいます(今でもこのカラムデザインを採用しているサイトは数え切れないくらいありますが)。それがCSSレイアウトだと、**float**を使うことでクリエイティブな位置指定ができますし、**z-index**を使ったレイヤー構造も可能です。さらには、ボックスモデルを使ってどんなHTML要素にもスタイルを付けることができます。このようにCSSを導入することで、新たなクリエイティブの幅が広がり、テーブルレイアウトでは実現できなかったデザインを生み出すことができるのです(図1.1)。

PCモニタにこだわらないデザイン

最初のCSS仕様書を公開する2年前、Netscapeが最初のWebブラウザをリリースするちょうど半年前、W3CのCSSワーキンググループの会長となるChris Lilleyは、その後数年間におけるWebの状況をズバリ予測していました。

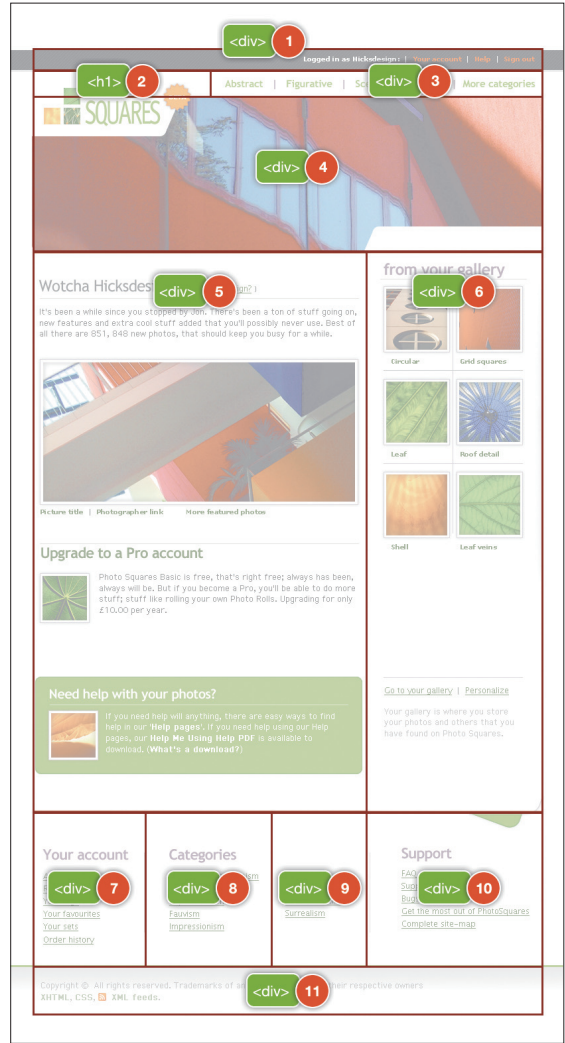
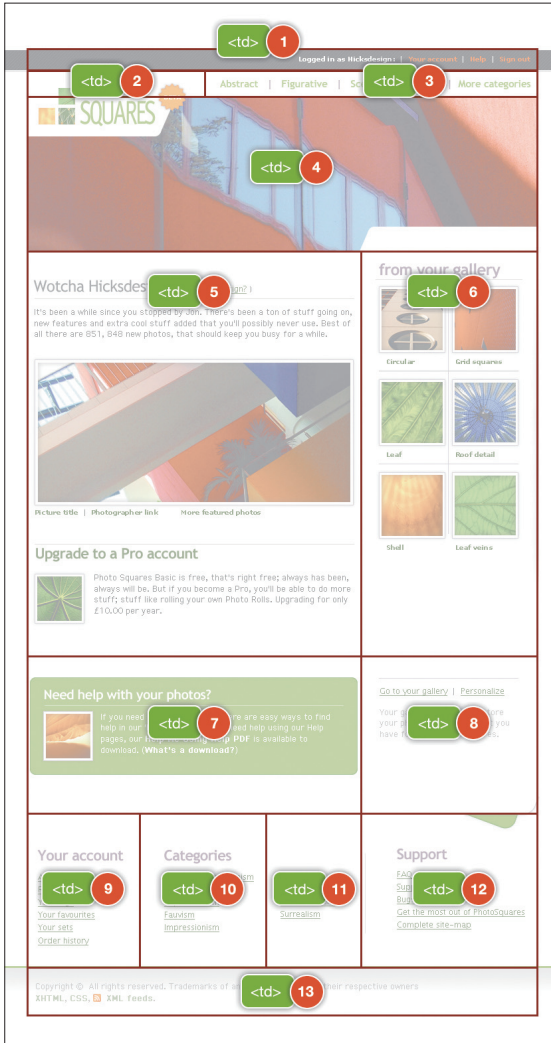
“もし、スタイルシートがそれと同じような技術がHTMLに利用されなければ、HTMLドキュメントは特定のWebブラウザ、特定のサイズ、特定のフォントといった制限された環境でしか上手く表示されなくなってしまう。その代償は避けられないだろう。”

—— Chris Lilley (CSSワーキンググループ前議長、当時は「the Style and Formatting Properties Working Group」と呼ばれていた)、1994年5月

しかし、何年にも渡る努力の結果、今やWebは携帯電話、ゲーム機、テレビでも見られるようになりました。将来的には、これから発明されるものも含め、すべてのポータブルデバイスでWebを見ることができるようになるでしょう。

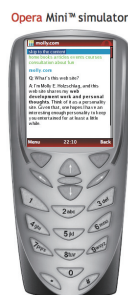
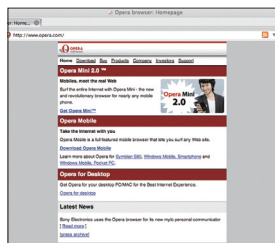
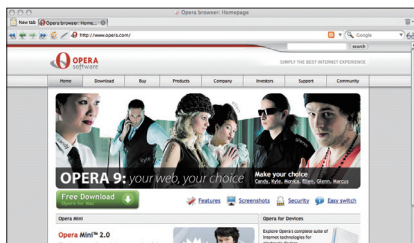
実際のところ、次に何がWebを表示できるようになるのかは誰にも分かりません。近い将来、私の息子は、Webはデスクトップマシンから接続するものだったということに笑ってしまうかもしれません。ちょうど私が、レンガのようなサイズと重さのバッテリーパックのついた初期の携帯電話のことを思い出して笑ってしまうように。

このようにメディアが激しく変化していく状況を考えると、Chris Lilleyの言葉はまさに的を射ましたし、当時よりも今のほうが一層強くそう感じます。今日のデザイナーは、あらゆる異なる環境に適応できるデザインを心掛けなければなりません。そうした中で、軽くて文書構造を



1.1

テーブルレイアウトと同じことが、CSSを使うことで実現できる。左図：テーブルレイアウトの例、右図：コンテンツの順序を変えずにテーブル要素をdiv要素で置き換えた結果。



1.2

デバイスに合わせて異なるCSSを適用。

意識したHTMLやCSSというのは、デスクトップマシン向けに制作したデザインを、プリンタ、小さなスクリーンのハンドヘルドコンピュータ、パーソナルメディアプレーヤ、携帯電話といった他のデバイスにも上手く適用させるための大切な要素と言えるでしょう (図 1.2)。

1-4. アクセシビリティはセールスポイントではなく、デザインの一部

当然ながら、プロのデザイナーや開発者に対して、優れたインフォメーションアーキテクチャやユーザビリティ、そしてアクセシビリティの能力が求められるようになりました。特に、幅広い人々に対応できるアクセシビリティの確保は、道徳上の観点からだけではなく、商業的な観点からも重視されています。しかし、多くのデザイナーや開発者、さらにはアクセシビリティ分野の専門家の中にも、アクセシビリティは障害のある人のニーズに応えるためだけのものだと、間違った解釈をしている人もいます。

アクセシビリティはユーザビリティでもある

これまでアクセシビリティといえば、視覚障害がある人々にもサイトが使えるようにすべきだ、アクセシビリティのガイドラインや関連法規に従うべきだ、という視点で語られることがほとんどでした。しかし、それらだけでは、アクセシビリティの重要なポイントをとらえているとは言えません。

もちろん、障害を持つ人々も利用できるようにすることも大切なことですが、もっと大きな視点を持つべきです。「アクセシビリティはユーザビリティでもある」。この視点でアクセシビリティが語られることはほとんどありませんでした。プロのデザイナーとして、コンテンツをデザインする際は、使う人の能力や利用するデバイスにかかわらず、世界中の人々がアクセスできて、できるだけ多くの人々のニーズに答えることができるようにすべきです。

たとえば、出張中のビジネスマンが出先からハンドヘルドコンピュータを使って社内の彼のマシンにログオンして販売数量をチェックできるようにするには、ユーザビリティとアクセシビリ

ティの両方がかかかってきます。おかしな話ですが、多くの映画サイトで携帯電話用のおまけコンテンツを提供しているにもかかわらず、そのサイトに携帯電話ではアクセスできません。みなさんにぜひ理解してほしいのは、良いデザインはできるだけ多くの人々がアクセスできるように、できるだけ多くの障害を取り除くということです。

「守っています」では不十分

残念ながら、未だに多くのデザイナーは、アクセシビリティのことを「デザイナーの日常業務の1つ」としてではなく、「クリエイティブを制限するもの」「守らなければならないガイドライン」「従わなければならない法律」ととらえているようです。また、アクセシビリティは、サイトの完成後に何らかの基準をクリアしているかどうかを確認する際の外部要素として考えられることが多々ありました。

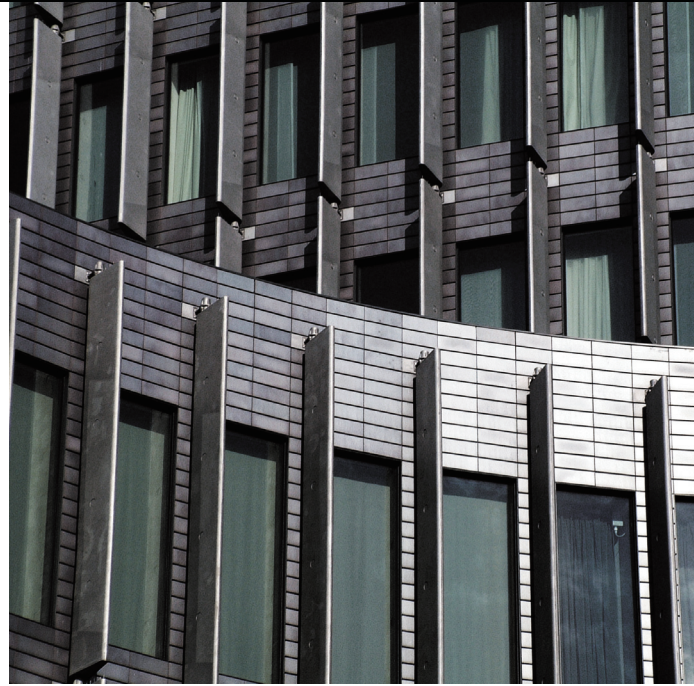
“もしアクセシビリティのことを「セールスポイント」としてとらえているならば、あなたはアクセシビリティを無視してしまうでしょう。いずれにせよ、ほとんどの開発者は無視してしまうでしょうけど。多くの開発者は、アクセシビリティのこと自体をあまり知らないし、その重要性となればなおさらです。”

—— Joe Clark (<http://joelclark.org/ice/iceweb2006-notes.html>)

残念なことに、Webコンテンツやサービスのアクセシビリティを確保するための作業は、良くてもデザインや開発工程の最後の方、つまり完成間近に行われるようで、悪ければ完成後の後付けとして修正追加されるといった状況です。もし最先端のAjaxインターフェイスとアクセシビリティのどちらを選ぶかとしたら、多くの企業はダイナミックなインターフェイスの方を選び、アクセシビリティの確保は将来のリニューアル時の検討材料として見送るのではないのでしょうか。

しかし、コンテンツを作り始める際に、きっちりと整理されしっかりと意味付けされた文書構造にしてCSSで見た目をスタイルするようしておけば、最初からアクセシビリティを取り入れたデザイン工程を実践できるのです。この方法を行えば、すべての人にとってプラスになるでしょう。アクセシビリティにあまり関心のないデザイナーでもアクセシビリティを取り入れることができますし、クライアントにとっても後で修正する必要がないので余計なコストを抑えることができます。そして何よりも大切なこととして、サイトのユーザーが欲しい情報をより簡単に入手できるようになるのです。

***NOTE 2005年6月、私は自分のサイトで「Accessibility and a society of control (アクセシビリティと法規制社会)」と題した記事を公開しました。趣旨としては、障害を持つ人々がもっと快適にWebを利用できるようにするためには、政府による法律や規則でそれを義務化することは逆効果である、というものです。この記事には、「Webアクセシビリティへの関心を持ち行動を起こさせるための最良の方法は何か」ということについて興味深いコメントをたくさんいただきました。



▶▶COLUMN Transcendent CSSをクライアントに上手く説明する方法

多くのデザイナーが仕事でCSSを存分に使いたいと感じつつも、「クライアントが期待していることはすべてのWebブラウザ上で同じように見えること」という思いから、それを我慢せざるを得ませんでした。実際、クライアントや企業は、デザイナーならクロスブラウザにデザインしてくれるものだと期待しているでしょう。そこで、デザイナーがクライアントを教育するということはあまりおすすめできることではありませんが、クライアントにTranscendent CSSの概念を説明するために役立つ例え話を紹介しましょう。

そもそも、デザインやテクノロジーの分野において、「時代や技術の移り変わりに関係なく、同じ機能あるいはユーザーエクスペリエンスを提供し続ける」という考えは当てはまりません。消費者は、テクノロジーは進歩していくものだと期待していますし、最先端の商品を購入していると思いたいものです。そういう視点でクライアントにも説明するのです。あなたのデザインは最先端のコーディング技術を使って最新版のWebブラウザに向けたもので、しかも古いWebブラウザを使っている人々にも素晴らしい体験を提供できる、という風に。

ハイビジョンテレビ

ハイビジョンテレビは、従来のテレビと比べて、高解像度な映像、より高いクオリティの音質を楽しめるというのが特徴です。2006年にイギリスでハイビジョン放送の開始が告知されて以来、電機店やテレビメーカーはこぞって「ハイビジョンブーム」に乗りました。そうしたブームの中であっても、ハイビジョンテレビに買い換えたりハイビジョン放送サービスに申し込まなかったりしたからといって、昼ドラや刑事ドラマ、サッカーなどの番組が見られなくなるわけではありません。ちょっと古いテレビでも、ハイビジョンに比べて少しは画質が劣るでしょうが、ちゃんと番組を見ることができます。

iPod

Appleが最初に発売したiPodは、最新のiPodに比べると機能も少ないですし、違うユーザーエクスペリエンスを提供しています。ポータブルミュージックプレーヤーの市場では競争が非常に激しいにも関わらず、Appleは新しいバージョンを出すたびにアルバムジャケットのアートワーク表示機能やミュージックビデオ・テレビジョン再生機能などの新機能を搭載することで、市場のフラッグシップ商品として君臨しています。古いバージョンのiPodを持っていてもその機能がアップグレードされることはありませんが、消費者もそのことは分かっていて、新しい機能を手に入れるには新しいiPodを購入するしかないのです。

永久に続くソフトウェアのアップグレード

ソフトウェア業界では、バグが残ったり、不安定なままでのソフトウェア発売、ほぼ毎日フリーズやクラッシュするOSというのは日常茶飯事です。こうした欠点があるにも関わらず、消費者は新しいバージョンが発売されるとなると、アップグレードしようと考えています。

クライアントは、デザイナーにWeb制作を依頼する立場の人間であると同時に、日々のテクノロジーの進歩に接している消費者でもあります。ならば、クライアントの人々にも、最新版のWebブラウザ(モダンブラウザ)ではより素晴らしいユーザーエクスペリエンスやデザインを提供できることを分かってもらえるのではないのでしょうか。

1-5. Transcendent CSSへ向けて

印刷やマルチメディアのデザイナーに比べて、HTMLやCSSを使うWebデザイナーにはデザイン上の制約が多く、なかなかその制約を克服することができません。最初のCSS仕様が公開されたものの、Webブラウザの性能が限られていたため、デザイナーがクリエイティビティを発揮するには、見た目のデザインを文書構造用のHTMLタグで指定するなど本来の目的と違った使い方をしたり、あるいはFlashで制作するといった手段しかありませんでした。そうした中、WebブラウザのベンダーはWeb標準、特にCSSとW3C DOM (Document Object Model) のサポートを強化し続けてきました。ところが、高いブラウザシェアを誇るMicrosoftのInternet Explorerの開発がバージョン6を最後にしばらくのあいだストップし、(今ではありがたいことですが)古いバージョンの利用率が低下しているということもありましたが、デザイナーはより高度なCSSテクニックを使うことをためらっていました。

※※※ 訳注 その後、Internet Explorerはバージョン7が2006年にリリースされ、また2009年中にはバージョン8がリリースされる見込みです。

その後、Mozilla Firefox、Mozilla Camino、Safari、Operaといった競合WebブラウザがInternet Explorerのシェアを奪い始めました(表1.1)。先進的なデザイナーたちは、その状況を敏感に察して、他のモダンブラウザでは実現可能なデザイン要素を付加することで、モダンブラウザのユーザーを優遇する手段を探し始めました。そうやって生まれたテクニックが「プログレッシブ・エンハンスメント」あるいは「MOSe」と呼ばれるものです。

表1.1 Webブラウザの利用率 2004-2006年7月時点

	Internet Explorer 7 (beta)	Internet Explorer 6	Internet Explorer 5	Firefox	Mozilla	Netscape	Opera
2006	1.9%	57.8%	4.2%	25.0%	2.2%	0.4%	1.4%
2005	-	67.9%	5.9%	19.8%	2.6%	0.5%	1.2%
2004	-	67.2%	13.2%	-	12.6%	1.8%	2.0%

MOSe

2003年、カナダのデザイナーで「CSS Zen Garden」の制作者であり著者でもあるDave Sheaが、クロスブラウザを意識したデザインで生じる問題を解決する独自のアプローチ方法を発表しました。彼はそのアプローチを「MOSe (Mozilla Opera Safari enhancement)」と呼んでいます。MOSeというのは、Internet Explorer 6 (古いバージョンも含め) が特定のCSSセレクタ (子セレクタ、隣接セレクタ、属性セレクタ) をサポートしていないことを逆手にとって、モダンブラウザのユーザーにはより素晴らしいデザインを提供しようというものです。たとえるなら、すべてのWebブラウザで表示できるデザインの上に、モダンブラウザだけが表示できるレイヤーをもう一枚被せるような感じです。そのアプローチの内容は、@import規則テクニックほど単純なものでした。@import規則テクニックは、Netscape 4が@import規則を解釈できないことを逆手にとって、サポートしていないCSSを適用させないようにするというものでしたが、Dave Sheaのアプローチとは大きな違いがあります。@規則テクニックは古いブラウザを対象にしていますが、Dave Sheaのアプローチでは、たくさんの欠陥があったにも関わらずCSS仕様の多くをサポートしていたInternet Explorer 6を対象にしていることです。Internet Explorer 6は当時もっとも普及していたWebブラウザであっただけに、Internet Explorer 6で表示できないデザイン要素は最小限に止めなければなりませんでした。

>>>COLUMN Netscape 4.xへの対応

Netscape 4.xがサポートしているCSSは限られています。そこでNetscape 4.xがサポートしていないCSSスタイルを適用させないために、多くのデザイナーは@import規則を使って基本スタイル以外のCSSスタイルを隠しています。基本のCSSスタイルは以下のように記述して、Netscape 4.xを含むすべてのWebブラウザに適用されるようにします。

```
| <link rel="stylesheet" type="text/css" href="simple.css" />
```

より高度なCSSスタイルは以下のように記述して、サポートしているWebブラウザだけに適用されるようにします。

```
| @import "modern.css";
```

すべてのWebブラウザがsimple.cssファイルを読み込むことができますが、modern.cssファイルは@importをサポートしているWebブラウザしか読み込むことができません。modern.cssファイルはsimple.cssファイルの後に読み込まれるため、simple.cssでmodern.cssよりも個別指定をしていない限り、そのスタイルルールが上書きされます。

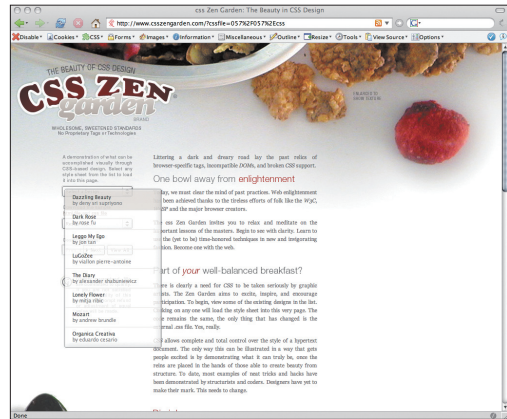
“MOSe手法のキーポイントは、
CSSが普及し始めた頃の
Netscape Navigator 4.xへ
対応させるためのデザイン手法と似ているところがあります。
MOSeでは、まずInternet Explorerで機能する基本ページを作り、
そこにInternet Explorerが
サポートしていないセクタを使って機能を追加していくのです。”

DAVE SHEA

www.mezzoblue.com/archives/2003/06/25/mose/

●CSS ZEN GARDENより

「This Is Cereal」(右上図)は、Shaun InmanによるCSS Zen Gardenの作品で、Internet Explorerがサポートしていないセレクトタを使い、モダンブラウザのユーザーにはよりリッチな体験を提供しています。たとえば、モダンブラウザでは、順不同リスト部分が透過PNGを使ったドロップダウンメニューとして表示されます。このテクニックはInmanが考案したものではありませんが、彼の使い方には心を動かされます。古いWebブラウザを使っているユーザーには、このリスト部分はスタイルが施されているにしてもシンプルな順不同リストとして表示されます。おそらく、そのユーザーは別のデザインバージョンがあるとは気づかないでしょう。以来、このようなちょっとした「MOSe」が、CSS Zen Gardenの作品の中でよく見られるようになりました(図1.3)。



1.3

CSS Zen GardenでMOSeを使った作品。

●AND ALL THAT MALARKEY

Inmanに影響を受けて、私も個人サイトをデザインし直し、2つのデザインパターンを作りました。1つは1960年代にイギリスで起きたモッズミュージックのムーブメントをモチーフとしたフルカラーデザイン。もう1つは、ブリティッシュ・スカのレコードレーベル「2Tone」で有名なツートーンイメージをモチーフにした白黒のデザインです。モッズよりスカの方が先に流行になったので、スカのデザインはInternet Explorer用にして、モッズのデザインはCSS 2.1のセレクタを使ってモダンブラウザだけが表示できるようにしました。こうやってWebブラウザの性能ごとに、デザインを変えるという試みはなかなか楽しいものでした。

▶▶NOTE Internet Explorerのユーザーだけモッズデザインを見ることができないというちょっとしたお仕置きをしてしまいましたが、それを快く思わなかった人がいるのも事実です。そのことについては、www.stuffandnonsense.co.uk/archives/and_all_that_design_malarkey.htmlで見ることができます。

▶▶訳注 「AND ALL THAT MALARKEY」は原著刊行当時の著者サイトのタイトルです。現在は「Stuff and Nonsense」というタイトルに変更されています。また、著者のサイトは現在リデザインされていますが、相変わらずIE6以下にはローファイ（白黒）バージョンを提供しています。

プログレッシブ・エンハンスメント

Dave Sheaが「MOSe」の記事を発表した数ヶ月後に、Triangle TechJournalサイトにて別な手法「プログレッシブ・エンハンスメント」についての記事が掲載されました。

“プログレッシブ・エンハンスメントは、ユーザーの要求に応じて情報を配信するというアプ



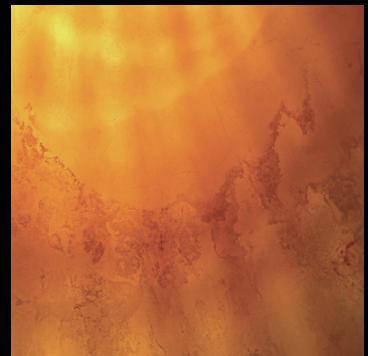
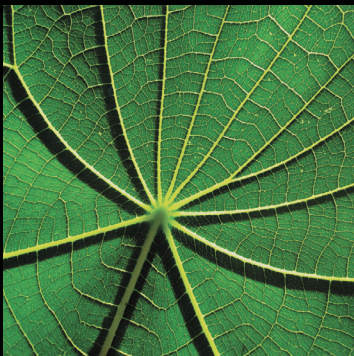
ローチです。しかも、アクセシビリティや将来的な互換性を持ち合わせ、新たなデバイスでもその性能に応じたユーザーエクスペリエンスを提供できます。”

—— Debra Chandra and Steve Champeon (http://hesketh.com/publications/progressive_enhancement_paving_way_for_future.html)

しかし、デザイナーにとっては難しく、未だに商業プロジェクトでこのプログレッシブ・エンハンスメントを完全に導入するには至っていません。それを困難にしている原因として、Internet Explorer 6が圧倒的なシェアを占めている状況に加え、「最新のWeb標準をサポートしているかどうかよりも、“最も普及している” Webブラウザを基準にデザインをすればいい」という考えがあるからではないでしょうか。Webのように若くダイナミックに変化するメディアにおいて、「マーケットリーダーとなるWebブラウザに合わせてデザインするだけでいい」という発想は、メディアの進歩を妨げるだけです。

MOSeとプログレッシブ・エンハンスメントの両方が言わんとしていることは、CSS 2.1をサポートしているWebブラウザがあるのであれば、デザイナーはそのWebブラウザのためにCSS 2.1仕様を使ってデザインしよう、ということです。CSS 2.1は商用Webが誕生した数年後には公開されていたにも関わらず、今もなお「高度な技術」「最先端の技術」として語られている状況です。

MOSeやプログレッシブ・エンハンスメントは、個人サイトやブログで使われてはいるものの、商業サイトで使われていることはめったにありません。その結果、斬新なWebデザインは停滞したままで、このままではいずれ腐ってしまうでしょう。プログレッシブ・エンハンスメントは3年後もまだ“プログレッシブ(漸進的)”な手法だと呼ばれているのでしょうか？ その答えはNOでなければなりません。前へと動き出していくべきなのです。



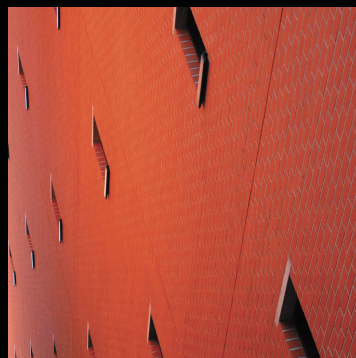
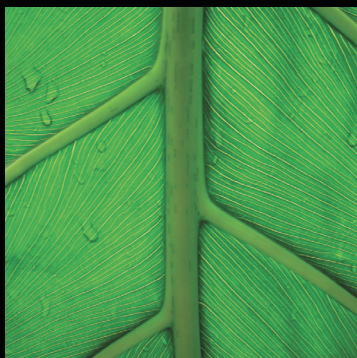


2. Transcendent CSSの原則

この本の主題である「Transcendent CSSの原則」、それを解説していきたいと思います。Transcendent CSSの原則に従えば、Webデザイナーは技術的な制約に悩まされることなくクリエイティブ作業に集中できるようになり、また古いブラウザの制限を気にすることなく未来に目を向けてデザインすることができるようになるのです。

その原則とは、以下の7つです。

1. すべてのWebブラウザが同じデザインを表示できるわけではない
2. 利用できるCSSセレクタはすべて使用する
3. 未来に目を向けてCSS3を使用する
4. CSS未対応部分はJavaScriptやDOMで補う
5. ハックやフィルタは使用しない
6. 意味的な命名規則やマイクロフォーマットを使用する
7. みんなとアイデアを共有し協力し合う



2-1. すべてのWebブラウザが同じデザインを表示できるわけではない

プログレッシブ・エンハンスメントでは、Internet Explorer 6のようなCSSへの対応が不十分なWebブラウザを基準にして、十分に対応しているWebブラウザにはCSSセレクタを利用して機能を追加するという方法でしたが、Transcendent CSSでは、この「CSSへの準拠度が低いWebブラウザを基準にする」という考えを捨てます。プログレッシブ・エンハンスメントが、すべてのWebブラウザで共通している点を使って、すべてのWebブラウザで同じように表示されるデザインを作るというアプローチをとっているのに対して、Transcendent CSSのアプローチではその逆を行います。Transcendent CSSでは堂々と、CSS 2.1そしてCSS 2.1へ対応しているWebブラウザを基準にしてデザインを行います。CSS 2.1の機能をすべて利用し、単に見目に華やかさを追加するのではなく、最も標準準拠度が高いWebブラウザに対して、今実現できるベストなデザインを仕上げるのです(図1.4)。実際にTranscendent CSSを導入すれば、訪問者の中には見劣りするデザインを提供することになるでしょう。どのくらい見劣りするデザインになるかは、あなたの自身の作り方にもよりますし、訪問者の特定の設定・環境にも左右されます。



1.4

著者のサイト(All That Malarkey)では、Webブラウザの種類によって異なるデザインが表示されるようになっていきます。

***訳注 このサイトデザイン「AND ALL THAT MALARKEY」は原著刊行当時の著者サイトのタイトルです。現在は「Stuff and Nonsense」というタイトルに変更されています。また、著者のサイトは現在リデザインされていますが、相変わらずIE6以下にはローファイ(白黒)バージョンを提供しています。

2-2. 利用できるCSSセレクタはすべて使用する

Transcendent CSSでは、すべてのCSS 2.1セレクタを使用しますし、疑似要素やダイナミック疑似クラスなどの他のCSS機能も使います。それでは、各セレクタについて解説しましょう。

属性セレクタ

属性セレクタは驚くほどパワフルで、特定の属性の有無、特定の属性値の有無をもとに要素を識別してスタイルを指定できるのです。たとえば、`href`属性がある要素だけ、属性値に「`http://www.stuffandnonsense.co.uk`」を持つ要素だけ、といったように識別します。以下のソースでは、`alt`属性を含むすべての`image`要素に「グレーのボーダー」というスタイルが適用されます(図1.5)。

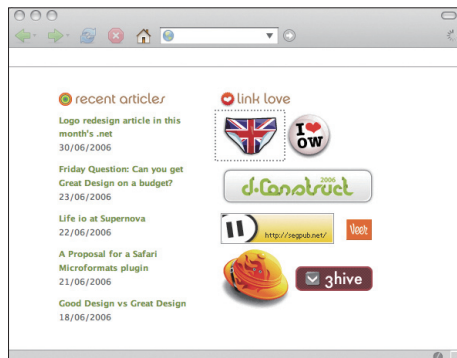
```
img[alt] {
border : 1px dotted #999;
}


```

次のように書くと、`title`属性を含む`a`要素は赤色になるわけです(図1.6)。

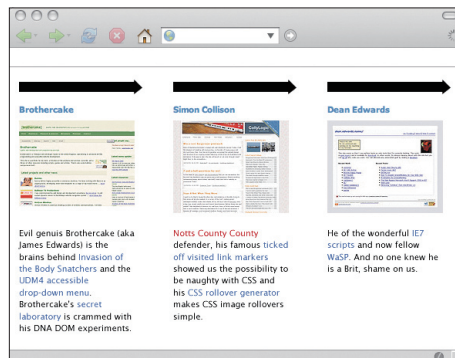
```
a[title] {
color : #c00;
}

<a href="http://www.collylogic.com/" title="Simon Collison">
Former Notts County Defender
</a>
```



1.5

`alt`属性を含む`image`要素だけにアウトラインを表示。



1.6

`title`属性を含む`a`要素をハイライト表示。

また、属性の値をもとに要素を区別してスタイルを適用できます (図 1.7)。

```
p[class="rant"] {
  font-weight : bold;
  text-transform : uppercase;
}

<p class="rant">
  But I am done apologizing. I am realizing more and more that the real
  world produces shocking HTML and CSS is inherently very difficult to
  organize at the best of times, let alone to organize well within a
  project that is jumping all over the place at an extraordinary pace.
</p>
```

(図 1.8)

```
div[id="content_main"] {
  float : left;
}

<div id="content_main">
  <blockquote>
  <p>I hope that things will change. I hope that some young guns will
  take up the challenge, stop following the crowd, and really push CSS
  to its fullest potential.</p>
  <p>Jeremy Keith</p>
  </blockquote>
</div>
```

また、部分一致にも対応しており、属性値の一部をもとに要素を区別することもできます。以下の例では、**cite**属性の値で「<http://www.andybudd.com/>」を含むものだけにスタイルを適用させています (図 1.9)。

```
q[cite*="http://www.andybudd.com/"] {
  padding-left : 100px;
  background : url(images/budd.jpg) no-repeat left top;
}

<q cite=" http://www.andybudd.com/archives/2006/07/layout_grid_bookmarklet/">
  Inspired by Khoi Vinh's post about using a background image of a grid
  for layout, I decided to knock up a quick Photoshop style Layout Grid
  Bookmarklet
</q>
```

属性セレクタの詳細や実用例については、Chapter 4にて解説します。



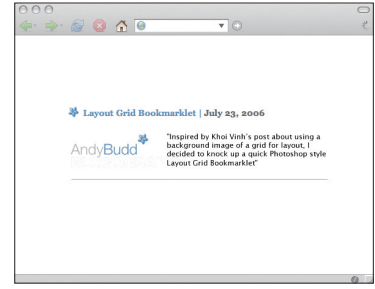
1.7

class属性の値がrantのp要素だけ大文字に。



1.8

id属性の値がcontent_mainのdiv要素だけ左寄せに。



1.9

cite属性の値で「http://www.andybudd.com/」を含む引用文に背景画像を追加。

子供セレクト

子供セレクトは、要素の直下に位置する子要素をターゲットとします。たとえば、同一ページ内にあるa要素でも、リスト要素の直下にあるa要素とそれ以外のa要素を区別してスタイルを適用することができます。子供セレクトは、2個以上のセレクトタからなり、「>」で区切ります。

以下のスタイルは、div要素の直下に位置するa要素すべてに適用されます。

```
div > a {  
  text-indent : -9999px;  
}
```

>>>NOTE 複数のセレクトタを指定する場合には、結合子(コンビネータ)を使用します。結合子の種類としては、「半角スペース」「>」「+」「.(コンマ)」「,(コロン)」があります。

このスタイルルールは、div要素の直接の子要素となるa要素だけに適用され、子孫要素を含むそれ以外のa要素には適用されません。たとえば、以下の例では、ul要素内のa要素にはテキストインデントは適用されません(p.37の 図1.10)。

```
<div>  
<a href="#content">Skip to content</a>  
<ul>  
<li><a href="http://www.andybudd.com">Andy Budd</a>  
<li><a href="http://www.adactio.com">Jeremy Keith</a>  
<li><a href="http://www.clagnut.com">Richard Rutter</a>  
</ul>  
</div>
```

3.1
2017

1

2

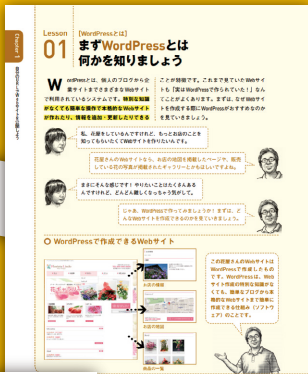
「Webクリエイティブ・マーケティング」

デザイン性・機能性に優れた
Webサイトが作れるようになる!

人気のWordPress入門書 第2版です。前回の書籍と同様、セミナー経験の豊富な著者陣が、実践的なノウハウを織り交ぜつつ丁寧に解説しています。WordPressの最新バージョンに対応した手順をすべて画面付きで解説しているの、Webサイトをはじめて作る人でも安心して読み進められます。HTML、CSS、PHPなどの知識がなくても、パソコン・スマートフォン両方の表示に対応した本格的なWebサイトを作れることでしょう。

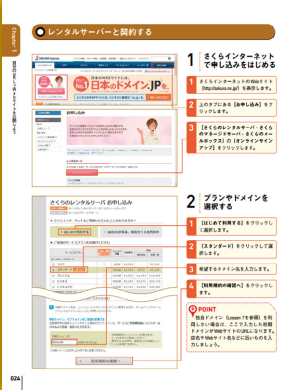
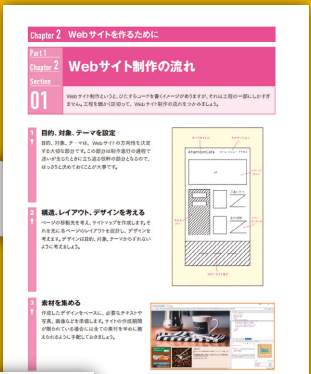
仕事できちんとWebを扱えるように
になりたい人のための新・定番書

そもそもWebサイトって何?というあなたでも、インターネットの基礎からHTMLとCSSの基本や使いこなしを学び、総まとめとしてシンプルなコーポレートサイトを制作できます。環境構築もていねいに説明しました。Web関連の企業に内定が決まった学生や、Web関係の部署に異動になった社会人など、必要に迫られて一通りきちんと学びたい人のための新・定番書です。ダウンロードできるサンプルデータで実際に作りながら学べます。

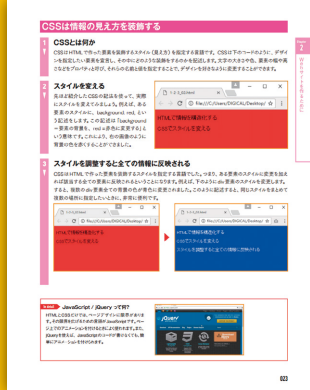


実際のセミナーや講義を元にしたやりとりによって、理解を深められる

ここで学ぶことは何なのか、どのような用途があるのかをイメージしよう



操作の手順は、大きな画面でひとつひとつのステップを丁寧に解説!



コードと実際のWebページを参照しながら学習できる!

いちばんやさしい
WordPressの教本

人気講師が教える
本格Webサイトの作り方
第2版 WordPress 4.x 対応
インプレス
石川栄和・大串肇・星野邦敏 (著者)
256 ページ
価格: 1,663 円 (PDF)



CMS

HTML5&CSS3
マスターブック

マイナビ出版
デジタル/ONTENTS (著)
248 ページ
価格: 2,138 円 (PDF)



HTML
/CSS

問題は設計にあり!?
UX視点のデザインプロセス



3

**デザインング
Web アクセシビリティ**

この書籍に書かれている内容は、良質なユーザー体験を生むためのノウハウです。普通に戦略・要件策定→情報設計→ビジュアルデザイン→実装・コーディングを行えば、達成基準を満たせることが理解できるでしょう。

ポーンデジタル
太田良典・伊原力也(著者)
296 ページ 価格: 2,700 円(PDF・EPUB)

**4コママンガを交えて
Webサイト制作を解説!**



4

**わかばちゃんと学ぶ
Webサイト制作の基本**

4コママンガを交えて、ユーモアたっぷりに Web サイト製作の基本を初心者向けにわかりやすく解説しています。企画の段階から、HTMLやCSSの基本、JavaScriptやPHPの概要、Webサイトの公開・運営に触れています。

シーアンドアール研究所
淡川あい(著者)
価格: 2,138 円(PDF・EPUB)

**UIデザインの原則と
心理学上の根拠がわかる**



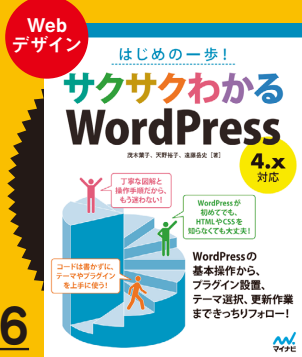
5

**UIデザインの心理学
わかりやすさ・使いやすきの法則**

UIデザインコンサルタントによるデザイン原則の科学。「物事を認識するときどのようなバイアスがかかるか」「物事をどのように見分けてとらえるか」など、UIデザイン原則と心理学上の根拠がわかります。

インプレス
Jeff Johnson(著者)、武倉広幸・武倉るみ(翻訳)
288 ページ 価格: 3,456 円(PDF)

**WordPress4 で迷わず
Webサイトができあがる!**



6

**はじめの一步!
サクサクわかる WordPress 4.x 対応**

WordPress4.x (2016年10月最新版)を使って、ホームページの作り方をていねいに教える本。これまでWordPressを使ったことがない方、HTMLやCSSの知識がない方にとっても分かりやすいよう、図解中心で解説します。

マイナビ出版 茂木葉子・天野裕子・遠藤岳史(著者)
256 ページ 価格: 1,814 円(PDF)

**写真で顧客の心をつかむ!
Instagramの宣伝ノウハウ**



7

**できる100の新法則
Instagram マーケティング**

1枚の写真で顧客の心を引き寄せる手法を現役ソーシャルメディアマーケターが解説。企業がInstagramでマーケティング、ブランディングを行い、成果を出すために必要なすべてのノウハウを収録しています。

インプレス 株式会社オプト・山田智恵・小川由衣・石井リナ(著者)、
できるシリーズ編集部(編集) 240 ページ 価格: 1,944 円(PDF)

**絶対に挫折しない
コンテンツマーケティング**



8

**いちばんやさしい
コンテンツマーケティングの教本**

人気講師が教える宣伝せずに売れる仕組み作り業界の第一人者が「絶対に挫折しない」をコンセプトにコンテンツマーケティングを詳しく解説。商品の売り込みに限界を感じている人、マーケティングの概念だけでなく実践に本気で取り組みたい人などにオススメです。

インプレス 宗像淳・魚山將(著者)
200 ページ 価格: 1,814 円(PDF)

**ビジネス視点から
UXの価値を探る!**



9

**マンガと解説のセットで
Webマーケティングを理解**



10

**種類の多いWeb広告を
上手に運用するための本**



11



UX × Biz Book

顧客志向のビジネス・アプローチとしてのUXデザイン

ビジネスの場で「顧客とのエンゲージメント」の重要性が増している中、UXアプローチは不可欠です。本書はさまざまな現場で活躍する執筆陣が、複合的な視点でUXおよびUXDのビジネス価値を分かりやすく解説します。

マイナビ出版 明海 司・井登 友一・奥谷 孝司・川田 学・坂本 貴史・橋 守・田平 博嗣・塚本 洋・森谷 徹厚・原 裕 (著者)
240 ページ 価格：3,024 円 (PDF)

**マンガでわかる
Webマーケティング 改訂版
Webマーケッター瞳の挑戦!**

マンガとストーリーに沿った丁寧な説明で、Webマーケティングの本質をわかりやすく解説しています。この改訂版では最新のデジタルマーケティングに対応し、現場で役立つノウハウをさらに盛り込みました。

インプレス 村上佳代 (著者)、ソウ (イラスト)、
星井博文 (原作者・原案) 296 ページ 価格：1,944 円 (PDF)

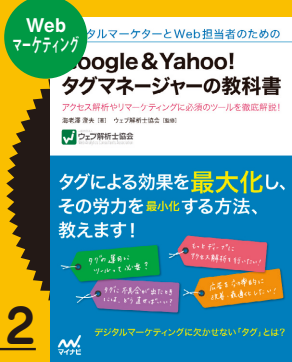
**基礎から学ぶ
Web広告の成功法則**

本書はWeb広告の上手な使い方を解説する本です。特にユーザーをどのようにして動かすのか、という視点を持つことで、Web広告の種類の多さや、新しい技術に惑わされず、長く使える広告運用のスキルが身に着きます。

マイナビ出版 本間和城 (著者)
232 ページ 価格：2,462 円 (PDF)



**タグを効率よく管理して
Webマーケの労力を最小化**



12

**すぐに役立つ実践的な
グロースハックの手法!**



13

**Webマーケティングを
ビジネスに取り入れよう!**



14

**デジタルマーケターと Web担当者のための
Google&Yahoo! タグマネージャー
の教科書**

「Googleタグマネージャー」と「Yahoo! タグマネージャー」について、設定や運用の実際、さらに効果測定や運用のTipsまでを詳しく解説。本書を読んでタグによる効果を最大化し、そのための労力を最小化しましょう。

マイナビ出版 海老澤淳夫 (著者)、ウェブ解析士協会 (監修)
304 ページ 価格：2,894 円 (PDF)

**いちばんやさしいグロースハックの教本
人気講師が教える
急成長マーケティング戦略**

注目の成長戦略「グロースハック」の実践が学べます。業界の第一人者が自社製品を育てた経験をもとに、すぐに役立つ手法やフレームワークを解説。製品やサービスを最短で収益化して成長に導くノウハウが満載です。

インプレス 金山裕樹・梶谷健人 (著者)
176 ページ 価格：1,728 円 (EPUB)

**いまさら聞けない
Webマーケティング
初歩から学べる集客のセオリー**

近ごろはさまざまなマーケティング手法が登場しています。本書はそうしたトレンドを押さえつつ「マーケティング戦略を基本から学びたい」「Webマーケティングを実践に活かしたい」と考えている方に向けた一冊です。

マイナビ出版 佐藤和明 (著者)
272 ページ 価格：2,138 円 (PDF)