

## Web標準時代に求められるサイト構築法

木達一仁  
株式会社ミツエーリンクス WEB開発チーム  
フロントエンドエンジニア  
Web Standards Projectメンバー  
k-kidachi@mitsue.co.jp / kidachi@kazuhi.to

### ●自己紹介

株式会社ミツエーリンクス WEB開発チーム フロントエンドエンジニア(2004年2月~)  
Web標準準拠サービスの立ち上げ/運用  
Web標準Blogの運営  
W3C Advisory Committee Representative  
海外のWeb標準関連書籍和訳版の監修  
Web Standards Project (WaSP)メンバー  
mixiにてWeb標準コミュニティを主催(2006年5月25日現在3260人が参加)  
7月15日「Web標準の日」開催

### ●本日のメインテーマ

Web標準時代に求められるサイト構築法  
既存サイトをWeb標準準拠する想定でワークフローを概説  
「Web Designing」2006年3月号に書かせていただいた内容がベース

### ●ワークフロー(1)全体の流れ

1. 要求分析
2. 現状把握
3. 要件定義
4. 基本設計
5. 詳細設計
6. 実装
7. 品質保証
8. 納品

### ●ワークフロー(2)「段取り八分」を意識すべし

サイト全体に適用される仕様をしっかりと事前に策定するプロジェクトに関わる全員が十分認識する必要性  
さもなくば想定外の場面で手戻りが発生することも  
5 Planes by JJG

### ●要求分析(1)全般的事項

クライアントへのヒアリングを通じ、全般的事項の確認を行う

- ・クライアント側のニーズ/ゴール
- ・ユーザ側のニーズ/ゴール
- ・クライアント視点でWeb標準準拠に最も強く期待するメリット
- ・各種ガイドラインの有無と継承の要/不要
- ・更新体制/更新状況
  - ・アクティブ/非アクティブコンテンツの切り分け(更新頻度、更新内容)
  - ・更新担当者の知識/技術レベルとポジション
  - ・Web標準準拠後の運用方針
- ・納品時期

### ●要求分析(2)技術的事項

クライアントへのヒアリングを通じ、技術的事項の確認を行う

- ・URI維持の要/不要
  - ・(X)HTML文書
  - ・画像等のその他のリソース
- ・メタ情報(特にmeta要素のkeywords属性値、description属性値)の要/不要
- ・印刷時対応(printメディア向けスタイルシート)の要/不要

### ●要求分析(3)ヒアリングの重要性

まずはクライアント側の主要な動機(SEOなのか、あるいはアクセシビリティ向上の一環なのか、等)を確認

どれだけ実装部分に手間やコストをかけようと、それ「だけ」ではクライアントニーズとユーザニーズの双方を満たすことは困難

Web標準それ自体はツールであり、決して万能薬などではない  
コンテンツこそが最も重要

### ●要求分析(4)運用状況の正確な把握

アクティブコンテンツと非アクティブコンテンツの切り分け(Pace Layering)

スケジュールを検討するうえで非常に重要

サイトが巨大であればあるほどプロジェクトは長期化

公開中のコンテンツの更新差分を適宜統合しなければならぬ

更新内容の記録やその授受をルール化

### ●要求分析(5)

単に準拠するのか、リデザインの環で準拠するのか?

現状どのようなマークアップがなされているのか?

スタイルシートの導入具合は?

日々の運用を担当している人々の知識/経験レベルは?

利用されているオーサーリングツールは何か?

ヒアリング次第では、0から新規に構築したほうが効果的/効率的な場合もある

### ●よくあるお話(1)

見た目を提供しない=ユーザーの切り捨て?

Webをあたかも紙媒体かその類似品のように誤解

古いブラウザにスタイルシートを提供しなかったり、最近のブラウザであっても細かな部分で見た目が異なることを極端に嫌悪

Webはユニバーサルな空間であり、さまざまなユーザがさまざまな閲覧環境からアクセス

全てに「1ピクセルたりとも違わず」同じ見た目を提供するのは所詮無理

見た目よりも重要であるはずの情報伝達やコミュニケーションをより確実に実現するツールとしてのWeb標準

特定のブラウザに他と同じ見た目を提供しないからといって、それはユーザーの切り捨てにはならない

Web標準に準拠することで情報にアクセスできるユーザは増える

### ●よくあるお話(2)

新しい仕様だからとりえずXHTMLで構築すべき?

どんなサイトでもXHTMLを採用すべきかといえば、必ずしもそうとはいえない

HTML 4.01を解釈するブラウザが姿を消してしまわない限り、ニーズが無くなることもない

現状、XHTMLには1.0と1.1の2種類があるが、後者については後方互換性の観点から注意が必要

(X)HTML文書をXMLデータとして配信したいのか、それともHTMLデータとして配信したいのか、あるいは情報の相互運用性といった観点から、ニーズを明確にしたうえで文書型は検討

### ●よくあるお話(3)

Webにおける品質=見栄えのリッチさ?

従来、Webコンテンツは納品時の見た目でその是非が論じられることが一般的

アクセシビリティやSEOへのニーズが声高に叫ばれる昨今、そのようなチェックでは納品物の品質を確認することなど到底不可能

ブラウザによって可視化されない部分の品質にも留意すべき

クライアント側でもHTMLについて一定の知識を持ち、納品された(X)HTML文書が文法的かつ意味的に妥当かある程度確認できたほうが良い

たとえばCSSレイアウトによりリッチな見た目のWebサイトが構築できたとしても、肝心のマークアップが妥当でなければ、Web標準に準拠したとは言えない

### ●現状把握(1)確認事項

新規構築であれば要件定義、そうでなければ現状把握プロセスへ

ディレクターおよび設計担当者が共同で実施

現状把握プロセスで確認すべき事項は、要求分析プロセスの結果に基づく

・サイト利用者の使用するユーザーエージェント

(要アクセスログ解析)

・デザインガイドラインの内容(ガイドラインが存在し、

かつ継承する必要のある場合)

・コーディングガイドラインの内容(ガイドラインが存在し、

かつ継承する必要のある場合)

・サーバ環境(必要に応じ)

・HTTPヘッダの内容(メディアタイプ)

・利用中のサーバサイド技術(SSI, PHP, etc.)

・コンテンツ分析

・リソース命名/配置規則

・マークアップ指針

・CSS記述指針

### ●現状把握(2)アクセスログ解析の必要性

サイトの現状把握の一環として、アクセスログ解析を通じ、ターゲットブラウザ(詳しくは後述)を策定するための情報を収集

実際のユーザがどのブラウザのどのバージョンを使っているのかは、サイトのコンテンツに強く紐づけられた貴重なデータ

可能であれば直近の3ヶ月から1年程度のデータを集計し、その内訳の変動も把握

それらの事実を踏まえ、さらにIE7など、今後登場する予定のより新しくWeb標準に準拠したブラウザへの対応も含め、前方互換性と後方互換性をバランスしながらターゲットブラウザ(後述)を選定

### ●要件定義(1)定義対象

要求分析および現状把握の各プロセスで得られた結果を受け、要件定義をディレクターと設計担当者が共同で実施  
構築対象のスクールの明確化、Web標準準拠後のサポートの必要性の確認や納品物の定義を行うほか、特に下記の3項目につき、要件を定義

- ・マークアップ仕様
- ・スタイルシート仕様
- ・ターゲットブラウザ

## Web標準時代に求められるサイト構築法

木達一仁  
株式会社ミツエーリンクス WEB開発チーム  
フロントエンドエンジニア  
Web Standards Projectメンバー  
k-kidachi@mitsue.co.jp / kidachi@kazuhi.to

### ●要件定義(2)マークアップ仕様

多くの場合、W3Cが勧告するマークアップ言語に準拠して制作

SGMLベースのHTMLを選ぶのであれば、HTML 4.01が相応しい

XMLベースのHTMLであれば、XHTML 1.0を採用する機会が多い(XHTML 1.1は後方互換性や利用すべきメディアタイプの観点から、企業サイトでは特に導入が進んでいるとは言い難い)

XHTML文書をXMLデータではなくHTMLデータとして扱うことが一般的な現状においては、HTML 4.01とXHTML 1.0のいずれかを選択するのが現実的

フレームはアクセシビリティやSEOの観点から好ましくなく、ゆえにFramesetの文書型を採用する機会は減少

StrictかTransitionalかは、コンテンツの特性をよく検討したうえで判断

### ●要件定義(3)

#### レンダリングモードとDOCTYPE宣言、XML宣言

近年登場したモダンブラウザの多くは、「標準準拠モード」と「後方互換モード」という異なる描画モードをもっており、DOCTYPE宣言やXML宣言の内容によって切り替わる

XHTML文書はXMLデータであることから、(XHTMLの仕様上必須とはされていないが)XML宣言を記述することが強く推奨されている

XML宣言を記述した場合、圧倒的なシェアを誇るIE6では、標準準拠モードを意図してDOCTYPE宣言を記述したとしても、後方互換モードで描画されてしまう

どのブラウザではどのレンダリングモードで描画されるのかを把握したうえで、DOCTYPE宣言の内容、XML宣言の有無を決定

### ●要件定義(4)文書のメディアタイプ

WinIEは、application/xhtml+xmlやapplication/xml、text/xmlといったメディアタイプへの対応が芳しくない

XHTML文書であってもtext/htmlというメディアタイプで配信しているケースは多い

text/htmlでの配信が禁止されているわけではないが、その場合、WebブラウザはXMLデータではなくHTMLデータとして解釈を行う点に注意

### ●要件定義(5)スタイルシート仕様

まだ草案段階ではあるが、CSS level2 revision1(CSS 2.1)に準拠することが多い

CSS 2.1では、以下の10種類のメディアタイプがあらかじめ用意されている

- ・all
- ・braille
- ・embossed
- ・handheld
- ・print
- ・projection
- ・screen
- ・speech
- ・tty
- ・tv

一般的なPC向けサイト構築でよく使われるのはall、print、screenの3つ

主要なターゲットユーザがスクリーンを持つデバイスの利用者として、判断すべきポイントは印刷時対応の有無

printメディア向けにもそれぞれ専用のスタイルを提供し、レイアウトが崩れる懸念を積極的に解消するほうが良い場合がある

「印刷用ページ」を別途静的に用意したり、あるいはサーバサイドのアプリケーションを用いた動的に生成する必要は無い

### ●要件定義(6)CSSハックの是非

ブラウザの仕様に対する解釈の相違やバグの有無を振り分け条件として見た目上の帳尻を合わせる、いわゆる「CSSハック」と呼ばれるような手法は、極力使わないようにしたい

やむをえない場合には、将来そのハックが使えなくなった場合に十分留意して使用する必要がある

今後登場するIE7では旧バージョンの有していた多くのバグが解消され、またCSS仕様により準拠する見込み

実際にどのハックを使うことになるかは詳細設計を済ませないとわからないはずではあるものの、具体的な記述方式や影響範囲などの情報と共に、適宜Web標準準拠ルールセット(後述)に盛り込む

### ●要件定義(7)ターゲットブラウザその1

現状把握プロセスで実施したアクセスログ解析結果を参考に、サイトの主要なターゲットユーザが利用しているであろうブラウザを中心に、ターゲットブラウザを定義

ターゲットブラウザは以下の2種類に分けて定義

- ・視覚表現保証ブラウザ(設計用ターゲットブラウザ)  
詳細設計プロセスで設計担当者が目視確認に使うブラウザ
- ・検品対象ブラウザ(検品用ターゲットブラウザ)  
検品プロセスで品質保証室が目視確認に使うブラウザ

### ●要件定義(8)ターゲットブラウザその2

多種多様なブラウザに配慮しつつも、検品プロセスにかかる時間やコストを削減することにより、Web標準準拠の総コストを少しでも抑え、かつ一定の品質をクリアするために導入している考え方

設計用ターゲットブラウザにはFirefoxの最新版を常に含めている(FirefoxはWeb標準に高いレベルで準拠しており、制作した文書のレンダリング結果が意図した通りか否かの判断基準として利用するため)

どのブラウザをターゲットブラウザに含めるにせよ、ブラウザ個別に合わせて作るのではなく標準に合わせて作るという大原則をまず理解し、クライアントとも認識を共有しておく

### ●基本設計(1)具体的な設計作業の開始

基本設計と詳細設計は、原則として同じ人間が一貫して作業を行い、方針や品質上の揺らぎを極力避ける

基本設計での主な作業は要件定義のさらに具体的な仕様への落とし込むことであり、仕様書(以後「Web標準準拠ルールセット」と呼称)の策定

ビジュアルデザインを担当するデザイナー、また必要に応じてアクセシビリティ、あるいはSEOを専門とするチームとも連携

### ●基本設計(2)ビジュアルデザイナーとの連携

必要に応じ、下記事項を情報共有

- ・対象解像度
- ・対象色数
- ・文字サイズ指定単位
- ・パーツ切り出し工程の分担及びその方策(切り出し方、ファイル名、etc.)

パーツ切り出しは減色工程を含んでおり、ビジュアルデザイナーによる判断が求められる可能性があるが、ベースコーディング(後述)作成に着手した時点で基本設計/詳細設計担当者が行う

### ●基本設計(3)ビジュアルデザインの多様性

ビジュアルデザインが多様でリッチであればあるほど、スタイルシートの設計は複雑化・長期化するだけでなく、サイト全体のユーザビリティの低下も懸念される

逆に簡素過ぎてしまうと、スタイルシートの設計は楽になり全体的な工数を下げることができるが、サイトの見た目は退屈になってしまう

多様性のバランスコントロールが、ビジュアルデザイナーにとって押さえるべきポイント

### ●基本設計(4)概念としてのCMSの重要性

どんなに巨大なWebサイトであっても、同じトーン&マナーでもって統一されていれば、大抵の場合、同じ見た目のコンポーネント(部品)が繰り返し使われているはず

コンテンツマネジメントシステム(CMS)は、あらかじめコンポーネント化されたコンテンツを組み合わせてページを生成するもの

その発想はあらゆるWebサイトに適用が可能であり、特にCSSレイアウトでサイトを構築する場合にはうまくフィットする

実際のCMSツールを導入せずとも、「概念としてのCMS」をワークフロー全体に適用することで、効率的な構築作業が可能

ビジュアルデザイナーがPhotoshop等で作成したカンプをモジュール化(部分化)し、それらを個々にCSSレイアウトで実現しておくことで、後の制作(マークアップ)プロセスを効率化

模型を作るのと同様、モジュールの組み合わせや繰り返しで、サイト内に登場するすべてのビジュアルを簡単に実現

### ●基本設計(5)アクセシビリティ担当者との連携

アクセシビリティの向上が、クライアントがWeb標準準拠に最も期待するメリットである場合には特に、CSSやJavaScriptの有効/無効、あるいは画像の表示/非表示にかかわらず、伝えるべき情報がしっかりと伝わるよう配慮する必要がある

そのようなアクセシビリティは、後付けでは実現が難しい場合が多く、設計段階からしっかりと意識して組み込む

必要に応じ、下記事項を情報共有しておく

- ・準拠対象(Webコンテンツ)JIS/WCAG 1.0/  
クライアントオリジナルのガイドライン)
- ・対応アクセシビリティレベル  
(WCAG 1.0であればAA/AAA)

この時点で既に実装に特に留意すべきアクセシビリティ事項があれば確認しておき、ベースコーディング(後述)作成時に反映

### ●基本設計(6)

#### Web標準準拠ルールセットの作成

Web標準準拠ルールセットの定義と位置づけは下記の通り

- ・サイト構築及び準拠後の運用にとっての指針
- ・必要であればビジュアルデザインとは分冊
- ・複数人が構築や運用に参加する際の共通認識の醸成
- ・準拠後の安定した品質の維持が最大の目的

Web標準準拠ルールセットに盛り込むべき内容とは、基本的には要件定義と現状把握から導出される全て

### ●詳細設計(1)詳細設計の作業細目

詳細設計では以下の項目を作業を想定

- ・ビジュアルデザインとのすり合わせ
  - ・クライアントの確認前に社内確認
  - ・実装上の潜在的な問題点の有無を確認(もしあれば、事前に解消を図る)
- ・ディレクトリ設計
- ・ベースコーディングの作成
  - ・文書構造の策定
  - ・読み上げ順序の策定(コンテンツユニライズ)
- ・モジュール設計
  - ・スタイルフック(id属性値/class属性値)の設定
  - ・基本モジュール
  - ・見出しモジュール
  - ・レイアウトモジュール
  - ・リスト(序列有/序列無/定義)モジュール
  - ・インライン系モジュール
  - ・etc.
- ・複数のモジュールによる組み合わせテスト
- ・ブラウザのバグや仕様の解釈の違いへの対処

## Web標準時代に求められるサイト構築法

### ●詳細設計(2)ベースコーディング

ベースコーディングとは、そのサイト上で使用するビジュアルデザインをモジュール化し、その全て(もしくは「ほとんど全て」)を単一のページに含めたもの

ベースコーディングでもって視覚表現保証ブラウザによる検証を行い、問題がなければ、制作後に個々のページの画面確認を割愛したとしても、すべての設計用ターゲットブラウザに対し見た目の再現性は十分期待できる

ベースコーディングの作成(=詳細設計)に着手する前には、サイト上のあらゆるビジュアルデザインが社内的な実装チェックとクライアントの確認の双方をクリアしていなければならない(たとえ100%が無理であっても、90%以上は目指すべき)

さもなければ、実装プロセスの最中に手戻りが発生したり、新しいモジュールを追加する毎に視覚保証ブラウザによるチェックが求められ、効率が格段に下がる

ベースコーディングを基にサイト内の代表的なページを数パターン、テンプレートとして作成

どのようにマークアップを行えばどのようにレンダリングされるかをまとめておく

### ●実装(1)コンテンツ作成

このプロセスは他のプロセスと同時並行で進めることが可能

以下の点に配慮し、特に(X)HTMLで記述することを意識して、コンテンツの準備を行う

- ・Wordなど、使用するツールは何でも可
- ・HTMLの基本要素(hn, p, ul, ol, dl, table, etc.)を意識した構造的なコンテンツライティング
- ・可能であればどのコンポーネントを使うかも原稿上で指定
- ・文字素材、画像素材のファイルの準備
- ・画像ファイルやマルチメディアコンテンツについてはalt属性値や代替コンテンツについて指示
- ・リンク先指定の明記

### ●実装(2)マークアップ

基本的には実装担当者(マークアップエンジニア)が担当するプロセス

以下の項目を遵守して作業を進める

- ・ベースコーディングやそれを基にしたテンプレートを参考にしながら、各ページを生成
- ・Dreamweaverなど、使用するツールには何でも可(ただし、Web標準準拠ルールセットには厳密に従うこと)
- ・基本はMozilla Firefoxで画面確認と文法チェックを同時並行で進める(他ブラウザでの見た目は詳細設計段階のベースコーディングで担保)
- ・ソースはシンプルかつわかりやすく記述(ソフトウェアばかりでなく人間にとっての可読性にも配慮)
- ・1バイトでもファイル容量を軽くする努力を怠らない
- ・ビジュアルデザインの再現性が確保できない問題が発見された場合には設計担当がスタイルシートの修正を行う

### ●品質保証

品質保証プロセスでは、主に以下の点につき検品を行う。

文法的妥当性(実装担当者/品質保証室)

目視確認によるビジュアルデザインの再現性

文法的妥当性についてはW3C Validator、Html Validator for Firefox and Mozilla、Another HTML lintなどを使用

必要に応じリンクチェック、文言チェック、アクセシビリティ検品を同時並行で進行

ビジュアルデザインの再現性については、設計用ターゲットブラウザと検品用ターゲットブラウザは同義でない点に注意

あらかじめクライアントとの間で合意・策定した検品用ターゲットブラウザでもって、品質保証担当者が確認

場合によっては特定のブラウザに対し意図的に他と異なる見た目を提供する場合には、あらかじめその旨を実装担当者が品質保証室に連絡しておき、実装上のミスでないことを明確にしておく

### ●納品

検品を経て、基本的には以下の2点を納品

- ・制作したファイル一式
- ・Web標準準拠ルールセット

### ●準拠後の運用

特にインハウスで運用を行っている場合、Web標準準拠後の運用に注意が必要

Web標準に準拠した状態を維持できないようではプロジェクトとしては失敗

初期のヒアリングを通じ、維持が難しそうな見通しが立っていれば、それを前提に、設計や納品物を検討

### ●想定されるサポート

- ・定常更新
- ・Dreamweaver/Contribute用テンプレート提供
- ・新規スタイルの追加とガイドラインの改訂
- ・勉強会やトレーニングを通じたナレッジシェア
- ・CMS(含Blogツール)の導入

### ●Q&A