

X8341-3 技術解説

(ワーキング・ドラフト)

日本規格協会 INSTAC

X 8341-3:2004 高齢者・障害者等配慮設計指針－情報通信における機器、ソフトウェア及びサービス－ 第3部：ウェブコンテンツ

技術解説

Working Draft 2.11

3月24日

この技術解説は作成中のものであり、
5月末日まで改変される可能性があります。

<作成>

日本規格協会情報技術標準化研究センター
アクセシビリティ国際標準化に関する調査研究開発委員会（WG2）

目次

目次.....	2
1. はじめに	4
2. 目的.....	4
3. 記述方針	4
4. 技術解説の使い方	5
5. JIS X8341-3 第 5 章 "開発・制作に関する個別要件" に関する解説.....	6
5.1 規格や仕様への準拠.....	6
5.1.a 規格・仕様・文法への準拠.....	6
5.1.b アクセシブルなオブジェクトの使用	9
5.2 構造及び表示スタイル	11
5.2.a 文書構造のマークアップ	11
5.2.b 構造と表示スタイルの分離.....	14
5.2.c データテーブルの適切なマークアップ	21
5.2.d レイアウトテーブルの禁止.....	23
5.2.e 識別可能なページタイトル.....	25
5.2.f フレームの多様の禁止	28
5.2.g 階層構造とサイトマップの提示	30
5.3 操作及び入力.....	31
5.3.a 非デバイス依存.....	31
5.3.b 入力欄のわかりやすさ	33
5.3.c 入力時間制限	35
5.3.d 制限時間の延長・解除.....	36
5.3.e ページの自動更新、移動の禁止	37
5.3.f 一貫性のある基本操作部の提供	39
5.3.g リンクの識別と操作.....	40
5.3.h ナビゲーションスキップ	42
5.3.i 誤操作の復帰	44
5.4 非テキスト情報	45
5.4.a 画像の代替テキスト.....	45
5.4.b ハイパリンク画像の代替テキスト	47
5.4.c 音声の代替	49
5.4.d 動画の字幕と状況解説.....	51
5.4.e オブジェクトの代替形式.....	54
5.5 色及び形.....	56
5.5.a 色の上に依存する情報提示.....	56
5.5.b 形・位置の上に依存する情報提示.....	57
5.5.c 背景色と前景色のコントラストと配色.....	59
5.6 文字	60

5.6.a 文字サイズの変更	60
5.6.b フォントの読みやすさ	66
5.6.c フォント色	67
5.7 音	68
5.7.a 音の自動再生の禁止	68
5.7.b 利用者による音量調整	72
5.8 速度	75
5.8 a 変化する画像・テキスト	75
5.8 b 早い周期の点滅禁止	77
5.9 言語	78
5.9.a 自然言語の指定	78
5.9.b 外国語の多用禁止	80
5.9 c 省略語、専門用語、流行語、俗語の多用禁止	80
5.9 d 読みの難しいと考えられる言葉の多用禁止	82
5.9 e 単語途中のホワイトスペースの禁止	83
5.9.f 図記号とイラストレーション	83
6. 情報アクセシビリティの確保・向上に関する全般的要件	85
6.1 企画・制作に関する要件（JIS X 8341-3:6.1）	85
6.2 保守及び運用に関する要件（JIS X 8341-3:6.2）	88
6.3 検証に関する要件（JIS X 8341-3:6.3）	89
6.4 フィードバックに関する要件（JIS X 8341-3:6.4）	90
6.5 サポートに関する要件（JIS X 8341-3:6.5）	91
7. 要素別索引	92
7.1 メタ要素	92
7.2 基本構造要素	92
7.3 リスト	92
7.4 テキスト構造マークアップ	92
7.5 テキスト修飾マークアップ	93
7.6 リンク	93
7.7 フォーム	93
7.8 テーブル	93
7.9 フレーム	93
7.10 画像とイメージマップ	94
7.11 オブジェクト	94
7.12 スタイルシート	94
Appendix 1 アクセシブルな Macromedia Flash を作成するためのテクニック	94
Appendix 2 アクセシブルな Adobe PDF を作成するためのテクニック	94
（参考資料：html4.01 elements list）	94

1. はじめに

「高齢者・障害者等配慮設計指針—情報通信における機器、ソフトウェア及びサービス— 第3部：ウェブコンテンツ - JIS X8341-3:2004」は2004年6月に公示、出版されたウェブコンテンツのアクセシビリティガイドラインである。このJISは、ウェブコンテンツをどのように制作すればよりアクセシブルなものにすることができるかをウェブの開発・制作者の立場でまとめたもので、多くの例示によってその具体的な実現方法の例を示し、使いやすいガイドラインを目指して作成されたものである。しかし、JISは頻繁に改定されるべきでないとの考え方から、規定はできるだけ技術に依存しないように記述されている。また、規格発効後、日本規格協会を通じていくつかの質問や意見が寄せられた。それらの疑問に答えることもこの技術解説の目的のひとつである。そこで、本体規格を技術に即した視点からよりわかりやすくし、JIS X8341-3:2004の正しい理解を促進するために技術解説を策定することとなった。この文書は、日本規格協会情報技術標準化研究センター（INSTAC）に平成16年度に設置された、情報アクセシビリティの国際標準化委員会ウェブアクセシビリティ国際規格調査研究部会（WG2）において作成された。

なお、この技術解説はJIS X8341-3:2004を補足するものであり、規格ではないことに注意されたい。

2. 目的

本体規格を技術に即した視点からよりわかりやすくし、JIS X8341-3:2004の正しい理解を促進するために
⇒この文章少し直す。

この技術解説は、JIS X8341-3:2004 の第5章及び第6章を対象に、その内容の技術的背景を明らかにする。
その実現方法の例を技術に即して具体的に示すことを目的としている。

3. 記述方針

次の点を明確にして記述した。

- a) ハイパertextマーク付け言語(HTML、XHTML)、及び段階的スタイルシート(CSS)のバージョンを明確にして、ウェブコンテンツの製作現場でJIS X8341-3:2004を用いようとしたときに、どのようなテクニックが利用可能かを示す。
- b) 規格票を理解するうえで誤解なく、正しく理解できるようにするための情報を提供する。
- c) 可能な場合には、チェック方法を示す。
- d) 可能な限り具体的な例示をおこなう。

なお、この技術解説は規格票を理解し運用するための参考であって、規格ではない。また、本書の各々の技術解説は規格票の示す基準を満たそうとすることに用いることができる技術を例示的に列挙したものであって、それらを用いて開発されたウェブコンテンツがJIS X8341-3:2004に準拠していることを保障するものではない。

4. 技術解説の使い方

この技術解説では、JIS X-8341-3 の第 5 章，第 6 章の要件を「背景と問題点」「関連する要素」「関連項目」「技術解説」「ソリューション」「注意事項」に分けて解説している。

- **背景と問題点**

その要件に関連して高齢者や障害のある人々がどのような問題に遭遇する可能性があるか，またその背景となる技術について解説する。

- **関連する要素、属性及び宣言**

HTML，XHTML の関連する要素、属性宣言で、関連するものを列挙している。

- **関連項目**

同時に参照して欲しい要件を列挙してある。

- **技術解説**

要件を理解するための補足的な解説を記述している。JIS 本文には書かれていない技術的な背景や数値などを含む。

- **ソリューション**

この要件を満たすための技術的な方法を示している。この中から一つ，あるいは複数の手法を選択する。

- **注意事項**

関連する注意事項が記載してある。

また，付録として「用語索引」「要素索引」を用意した。

5. JIS X8341-3 第5章 “開発・制作に関する個別要件” に関する解説

5.1 規格や仕様への準拠

5.1.a 規格・仕様・文法への準拠

ウェブコンテンツは、関連する技術の規格や仕様、および文法に準拠して作成しなければならない。(対応：JIS X8341-3:5.1 a)

背景と問題点

一般によく使われているブラウザでは、HTMLなどの文法に厳密に準拠していなくても、文法のエラーを自動的に回避して表示する機能を持っている場合がある。しかし、このような文法解釈の自動的な修正機能はウェブブラウザに依存した機能であり、あらゆる環境で有効である保証はない。音声ブラウザやスクリーンリーダーなどの高齢者障害者支援技術は、これらの問題のあるHTMLを一般的なブラウザと同等に解釈できるとは限らない。また、アクセシビリティチェックツールは文法が正しくないと、判断を誤ってしまう可能性がある。したがって、アクセシビリティの改善を行う前提条件として仕様や文法に正しく則ってコンテンツが作成されている必要があるのである。

関連する要素、属性及び宣言

meta

DOCTYPE 宣言

注 DOCTYPE 宣言は HTML の要素ではなく、文書型宣言である。

関連項目

技術解説 HTML、CSSなどのウェブコンテンツを記述するための言語やソースコード、技術は、W3C、ISO、又はJISが定めた規格、あるいは勧告に準拠することを求める。ただし、このガイドラインはW3CのWCAGに準拠することを求めているわけではなく、一般的な文法等の使用に準拠することをもとめている点に注意が必要である。あくまでも、一般的な技術の仕様や文法についていっているものであって、アクセシビリティに関する基準については、X8341-3:2004を用いる。準拠すべき規格、勧告の代表的なものを以下に示す。この中から、利用している技術だけを選択して、準拠しているかどうかを確認すべきである。

- HTML 4.01 Specification

W3C Recommendation (1999年12月24日)

<http://www.w3.org/TR/1999/REC-html401-19991224>

- JIS TR X0033: ハイパテキストマーク付け言語 (HTML) 4.0

この標準情報 (TR) は、HTML 4.0 Specification W3C Recommendation, revised on 24-Apr-1998を翻訳し、技術的内容を変更することなく作成されたもので、TR X0033:2000 に、HTML 4.01 Specification W3C Recommendation との差分が、技術的内容を変更することなく附属書 1 として追加されている。

- XHTML™ 1.0 The Extensible HyperText Markup Language (Second Edition)

A Reformulation of HTML 4 in XML 1.0

W3C Recommendation (2000 年 1 月 26 日、2002 年 8 月 1 日に改定)

<http://www.w3.org/TR/xhtml1/>

- XHTML™ 1.1 - Module-based XHTML

W3C Recommendation (2001 年 5 月 31 日)

<http://www.w3.org/TR/2001/REC-xhtml11-20010531>

- Modularization of XHTML™

W3C Recommendation (2001 年 4 月 10 日)

<http://www.w3.org/TR/2001/REC-xhtml-modularization-20010410>

➤ JIS TR X0080:2003 XHTML 1.1 – モジュールに基づく XHTML

この標準情報は、XHTML™ 1.1 - Module-based XHTML W3C Recommendation (2001 年 5 月 31 日) と一致している。

- Cascading Style Sheets, level 2 revision 1

CSS 2.1 Specification

W3C Candidate Recommendation 2004 年 2 月 25 日

<http://www.w3.org/TR/2004/CR-CSS21-20040225>

➤ JIS TR X 0032:2000 段階スタイルシート水準 2 (CSS2)

この標準情報は、Cascading Style Sheets, level 2 CSS2 Specification W3C Recommendation (1998 年 5 月 12 日) と一致している。

- Synchronized Multimedia Integration Language (SMIL 2.0) - [Second Edition] (2005 年 1 月 7 日)

<http://www.w3.org/TR/2005/REC-SMIL2-20050107/>

➤ JIS TR X0093:2003 同期化マルチメディア統合言語 (SMIL2.0)

Synchronized Multimedia Integration Language (SMIL 2.0) W3C Recommendation (2001 年 8 月 7 日) が、この標準情報(TR)と一致している。

- Scalable Vector Graphics (SVG) 1.1 Specification

W3C Recommendation (2003 年 1 月 14 日)

<http://www.w3.org/TR/2003/REC-SVG11-20030114/>

これら以外の規格、勧告、技術を用いる場合にもできる限りそれらの仕様に合致するよう開発するとよい。

XHTML など XML を基本にした仕様の場合、その文法は DOCTYPE 宣言によって示された DTD で指定されるが、その場合ローカルな DTD を指定することは好ましくない。ウェブブラウザや高齢者障害者支援技術が DTD を正しく解釈する能力を持っているとは限らず、プログラムによって固定的な解釈をおこなう可能性があるからである。したがって、やむをえない事情がある場合を除いて、上記のマーク付け言語などで定義された DTD をそのまま使うほうが良い。

本ガイドラインは、必ずしも最新の仕様を用いることを要求してはいない。例えば、段階的スタイルシートに現時点で最新の CSS2.1 を使うことを要求しているわけではない。最新の規格、勧告、または仕様を用いることは好ましいが、それらをウェブブラウザや高齢者障害者支援技術が正しく解釈、処理できない場合には、かえって混乱を招くこともあるからである。

また、マーク付け言語の仕様において本ガイドラインを用いる場合に、Strict な要素だけを用いるべきか、あるいは Transitional な要素を用いても良いと考えるべきかと迷う場合があるかもしれない。本ガイ

ドラインでは、仕様に準拠することだけを求めていると考えることができるので、Strict でも Transitional でも許容するというのが原則だ。しかし、一般的に Transitional な要素の中には、アクセシビリティの観点から見て好ましくない要素が残されており、Strict な要素だけを用いたほうがアクセシブルなページを容易に開発できることが分かっている。したがって、Transitional を用いる場合にも、Strict を基準として用いて、やむをえない場合だけ Transitional な要素を使うようにするとよい。

文字コードの指定を正しくおこなうことも重要である。HTML 文書でマーク付けされた文字コードの指定と、実際に用いた文字コードが一致しないと、利用者の環境によっては文字化けをおこしてしまう。この問題も、高齢者・障害者だけでなく、広く一般の利用者が混乱する原因となるので、注意が必要である。

ソリューション 次の方法を選択して、あるいは組み合わせて用いる。

1) 使用する言語の明確化

制作、開発に当たっては、使用する HTML 等のバージョンをあらかじめ決定しておき、それらを用いることを制作担当者、開発担当者に徹底するとともに、正しい文法や仕様で記述できるよう教育する。また、このことがアクセシビリティの確保・向上に役立つことを理解させる。

2) DTD の定義

HTML、XHTML、XML においては DOCTYPE 宣言を用いて DTD を定義する。

例 1 HTML4.01 Strict を用いる場合

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN"
"http://www.w3.org/TR/html4/strict.dtd">
```

例 2 HTML4.01 Transitional を用いる場合

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
```

例 3 XHTML1.0 Strict を用いる場合

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
```

この例にないローカルな DTD を用いる場合は、代表的な高齢者障害者支援技術によって利用可能なことを確認する必要がある。

3) 文法の妥当性

DTD 宣言の内容にしたがって文法が妥当(valid)であるかどうかを確認する。確認の方法には、文法チェッカーを用いる方法とオーサリングツールに内蔵されたチェッカーを用いる方法とがある。オーサリングツールのチェッカーを用いる場合には、そのチェッカーがチェック可能な言語や仕様を確認しておく必要がある。

例 1 W3C の Markup Validation Service を用いる。

<http://validator.w3.org/>

例 2 W3C CSS 検証サービスを用いる。

<http://jigsaw.w3.org/css-validator/>

例 3 市販のオーサリングツールを用いて文法を点検する。

4) 文字コード

文字コードの宣言と実際に記述した文字コードが一致することを確認する。

⇒JIS X0208:1997 「附属書 1（規定）シフト符号化表現」を参照してもいい

例 1 言語コードは、meta 要素を用いて head 内で次のように定義する。

```
<meta http-equiv="Content-Type" content="text/html; charset=EUC-jp">  
<meta http-equiv="Content-Type" content="text/html; charset=iso-2022-jp">  
<meta http-equiv="Content-Type" content="text/html; charset=Shift_JIS">  
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
```

5) 機種依存文字

機種依存文字や携帯電話で用いられる絵文字（携帯端末用コードでよく使われるもの）が用いられていると、音声ブラウザ等で正しく読み上げられないばかりでなく、一般的なウェブブラウザでも正しく表示されないことがある。これらは、アクセシビリティというよりはむしろユーザビリティにおける問題と考えられるが、高齢者などが混乱することが考えられるため、配慮が必要である。機種依存文字は”丸付き文字”、“ローマ数字”、“単位記号”及び“外字”などがある。図 1 に機種依存しない文字の例を示す。

例 1 機種依存しない文字の例(ISO-2022-JP)

（ここに機種依存でない文字、JIS コード番号のリストを入れる）

注意事項 規格、または勧告等にウェブブラウザが対応できていない場合がある。また、一般的なウェブブラウザは対応していても、高齢者障害者支援技術が未対応の場合もある。例えば、XForms、SVG はアクセシビリティの確保・向上には有効な技術だが、対応が進んでいない。したがって、これらの技術を使う場合には、それがたとえ標準規格や勧告であっても、一般的なウェブブラウザで正しく表示され、代表的な高齢者障害者支援技術を用いて利用可能であることを確認しておく必要がある。

5.1.b アクセシブルなオブジェクトの使用

ウェブコンテンツには、アクセス可能なオブジェクトなどの技術を使うことが望ましい。

(対応：JIS X8341-3:5.1 b)

背景と問題点

HTML やスタイルシート以外に、動画、音声、アニメーションのためのプラグインなど、多様なオブジェクトがウェブコンテンツの一部に用いられつつある。特に、サイトの重要な情報や基本的な操作のためのメニューなどに、非 HTML の技術を用いられていてアクセシブルでないと、そのサイト全体が利用できず致命的な問題となることがある。

関連する要素、属性及び宣言

object applet script noscript embed noembed

関連項目

5.4.c 5.4.d 5.4.e 5.3d

技術解説 オブジェクトなどの技術とは、主としてプログラムを用いて作成されたものを指す。このガイドラインでは、これらの技術が高齢者障害者支援技術で利用できることを保障することを求めている。オブジェクトなどの技術は、object、applet 要素を用いて html ページに埋め込まれるものと、アンカーを使って直接リンクされたファイルを含む。たとえば、Java、JavaScript、VBScript のようにプログラムによって制御されウェブブラウザ上に HTML とともに表示されるもの、ActiveX 技術などを使った専用のプラグインを用いる Macromedia FLASH など、外部のプログラムを起動して表示される Adobe PDF などを含む。マイクロソフトワード、エクセル、アクセスなどの文書ファイルがコンテンツとして提供される場合には、それらのファイルが高齢者障害者支援技術を用いてアクセス可能であることを確認しておく必要がある。なお、本ガイドラインはオブジェクトを積極的に用いることを推奨しているわけではなく、単に用いる際にアクセス可能であることを求めているだけである。

ソリューション 次の方法を選択して、あるいは組み合わせて用いる。

1) object/applet 要素

これらの要素を用いているときは、その技術がアクセシブルであることを確認する。確認の方法としては、次の方法が考えられる。

- オブジェクトの開発、実行環境等を提供している企業がアクセシビリティのガイドラインを持っている場合には、それらが JIS X8341-3:2004 を満たしていることを確認したうえで、そのガイドラインを使用する。

例1 マクロメディア社の FLASH を使う場合には、同社のアクセシビリティ基準を用いて、作成する。

<http://www.macromedia.com/jp/macromedia/accessibility/> を参照する。

例2 アドビ社の Adobe PDF を使用する場合には、同社のアクセシビリティ基準を用いる。

<http://www.adobe.co.jp/enterprise/accessibility/> を参照する。

例3 ActiveX を用いる場合には、マイクロソフト社の Microsoft Active Accessibility に対応していることを確認する。 <http://www.microsoft.com/japan/enable/> (ユーザ向け情報)、

<http://www.microsoft.com/japan/msdn/accessibility/> (開発者向け情報) を参照する。

例4 Java™を用いる場合には、Java Foundation Classes (JFC)に含まれている [Java Accessibility API](#) を用いる。また、ウィンドウズにおいては、Java Access Bridge For Windows Operating System を用いる。ただし、日本における主要なスクリーンリーダーは現状ではこれらの対策によっても、Java プログラムにアクセスすることができない点に注意が必要である。

例5 アップル社 Mac OS X で動作する Carbon プログラムは同社のアクセシビリティガイドラインを用いること。 <http://developer.apple.com/documentation/Carbon/Accessibility-date.html> を参照する。ただし、現在のところ、Mac OS に対応した日本語のスクリーンリーダーは発売されていない。

例6 Linux や UNIX におけるアクセシビリティについては???

- ガイドラインがない場合には、JIS X8341-2 及び 3 を用いてアクセシブルかどうかを確認する。特に、高齢者障害者支援技術を用いた評価、利用者を交えた評価が有効である。

2) object 要素の代替テキスト

object 要素には代替テキストを提供する。代替テキストは、オブジェクトのもつ情報と同等の内容を提供するか、もしくはそのオブジェクトを用いることができない場合の代替手段に関する情報を提供する。

例 1 `<object data="logo.mpg" type="application/mpeg"> やさしい工業株式会社のロゴ </object>`

3) プラグインに関する情報提供

高齢者や初心者などはオブジェクトの表示に必要なプラグインの入手に手間取る可能性がある。したがって、プラグインの入手とインストールに関する情報を提供するとわかりやすい。

4) script 要素があるとき、そのスクリプトはアクセシブルであること。

Script に関する具体的なソリューションは、XXXXXXXXXXXXXXXXX を参照すること。

5) script 要素があるとき、noscript 要素を使って script が使えないときにも、アクセスできること

例 1 `<script type="text/javascript"> メニューのためのスクリプトが書いてある </script>
<noscript> メニューのページを開く </noscript>`

6) embed および noembed 要素は非標準の要素であり用いない

オブジェクトは object 要素を用いる。

注意事項

なし

5.2 構造及び表示スタイル

5.2.a 文書構造のマークアップ

ウェブコンテンツは、見出し、段落、リストなどの要素を用いて文書の構造を規定しなければならない。
(JIS X8341-3: 5.2 a)

背景と問題点

紙の文書を作成する際、太字にしたり、大きい文字にしたり、あるいは、先頭に点や丸などの記号置くなどして、目立たせることで、見出しを示すことがある。また、箇条書きも同様に、先頭に点や丸などの記号を置くなどして表すことがある。これは、文書の構造をわかりやすく（明確に）するための工夫である。つまり、※文書構造※を明確にすることで、文書の概要や必要な情報を探すのに役に立つからである。同様に、Web ページでも、視覚的に文書構造をわかりやすくするために、見出しや箇条書きを点や丸などの記号、あるいは字体に変化をつけて作成されている場合が多い。しかし、紙の文書と同じように、文字種や文字の大きさ、あるいは、行頭に記号を置くなどして、視覚的な方法で文書構造を表してしまうと、音声ブラウザやスクリーンリーダーなどの支援技術は適切な文書構造を理解することができず、ユーザー

に適切な形で情報を伝えることができなくなる可能性がある。

関連する要素、属性及び宣言

h1, h2, h3, h4, h5, h6, p, div, span, dl, ol, ul, li, big, small, em strong, blockquote, q, font, style

関連項目

5.2 b 構造と表示スタイルの分離

技術解説

HTML では、文書構造をあらわすための要素が準備されている。たとえば、見出し(h1～h6),や箇条書き(ul, ol, dl)など。これらの要素を利用することで、見た目だけでなく、HTML 的に文書構造を明確に表すことができる。

HTML の要素を利用して文書構造をあらわすことで、統一された形で視覚的に文書構造を示すことができる。また、視覚に頼らないで文書を利用する場合、たとえば、音声ブラウザなどで文章を読み上げる場合や点字ディスプレイで文章を表示する場合、※見出し※や※箇条書き※であることを知らせることができると、支援技術が適切に情報を提供したり、適切な形に変換したりしてユーザーに提供することが可能となる。また、検索エンジンなどのコンピュータが文書を理解し処理するためにも役立つ情報である。なお、視覚的にどのように表現するか指定は、CSS（段階的スタイルシート）で表現することができる。

ソリューション

次の方法を選択して、あるいは組み合わせて用いる。

1) 見出し要素 h1 ～ h6 を利用する

文書の構造を示すために、見出しをつける場合は、太文字などのフォント変更ではなく、見出し要素 h1 から h6 を利用して記述する。そうすることで、h1 から h6 までに対応した見出しとして表現できる。見出しはできるだけ簡潔に記述し、文書構造や内容が理解できるようにする。見出し要素を書式の変更や見出しを表現すること以外に利用すべきでない。

2) 見出し要素は可能な限り正しい階層構造で記述する

h1 は大見出し、h2 は副題、h3 は副々題、といった文書の階層構造を正しく表現できるように見出し要素を利用する。文字の書式（字体、大きさ）の違いによって、h1 から h6 を選択するべきではない。

3) リスト要素を利用する

箇条書きは、順序付きリスト(ol)、順序なしリスト(ul)、定義リスト(dl) をそれぞれ用途にあった形で利用する。これらのリスト要素を用いると、結果的にリストのマーク（「・」や数字など）が表示されるが、それらはブラウザの表示形式に依存する。それらの表示形式を制御したいときには、スタイルシートを用いる必要がある。また、テーブル要素を用いて箇条書きを記述すると視覚的に同様な効果が得られる場合があるが、文書構造を正しく示しているとはいえない。リスト要素を字下げ等の箇条書きを表す目的以外で利用すべきでない。

4) 段落要素 <p> や<div> を利用する

段落を表現する場合は、段落要素 <p> を利用する。また、本文の一部の表示スタイルを変更（引用する意図ではなく単に見栄えのために字下げしたり、右寄せ左寄せなど）する場合は、<div> や を利用する。字下げの目的のために、リスト要素や引用要素（<blockquote>、<q>）等を利用すべきでない。

5) 論理的な字種や字体を現す要素 <big>, <small>, 等を用いる

段落の中で特定の箇所を強調する場合は、論理的な字種や字体を現す要素 <big>, <small>, 等を用いる。字体を直接変更するような などは用いるべきでない。字体や書式などの表示スタイルをする場合はスタイルシートを用いる。

5.2.b 構造と表示スタイルの分離

ウェブコンテンツの表示スタイルは、文書と分離して、書体、サイズ、色、行間、背景色などを※スタイルシート※を用いて記述することが望ましい。ただし、利用者がスタイルシートを使用できない場合、又は意図的に使用しないときにおいても、ウェブコンテンツの閲覧及び理解に支障が生じてはならない。(JIS X8341-3: 5.2 b)

背景と問題点

前項と同様に、視覚的な方法で文書構造を表してしまうと、音声ブラウザやスクリーンリーダーなどの支援技術は適切な文書構造を理解することができず、ユーザーに適切な形で情報を伝えられなくなる可能性がある。一方で、HTML の※文書構造※をあらわすための要素のみ利用すると、視覚的な表現が十分得られない場合があり、文書構造を視覚的に示すために、書体、サイズ、色、行間、背景色などを用いて、表示上目立たせたりする必要がある。

関連する要素、属性及び宣言

h1, h2, h3, h4, h5, h6, p, div, span, dl, ol, ul, li, big, small, em, del, th, strong, blockquote, q, font, style
font-size, font-weight, text-align, color, background-color, margin-top, margin-bottom, margin-left, margin-right,
list-style-type, list-style-image, line-height, font-size, position:, top, left

関連項目

なし

技術解説

文書構造を視覚的に示すために、書体、サイズ、色、行間、背景色などの表示上の表現は CSS（段階的スタイルシート）で記述する。

HTML では文書構造をただしく記述し、スタイルシートが利用できないブラウザや、利用者が意図的にスタイルシートを無効にしたり、利用者独自のスタイルシートを設定したりしている場合でも、文書構造が最低限の内容が正しく伝わり、理解できるようにしておく必要がある。そのためにも、 JIS X8341-3: 5.2 a を守ることが必要になってくる。

ソリューション

次の方法を選択して、あるいは組み合わせて用いる。

1) 見出し要素 <h?> の書式変更スタイルシートを用いる

見出し要素は、ブラウザによって異なるが、大抵は<h1>が大きい文字で、<h6>に行くにしたがって、小さい文字になっていく。また、少し広めの改行幅で改行される。

しかし、ページの構成上、文字書式（字体・大きさ）や改行幅などを変更したい場合がある。その場合は CSS を利用して書式設定を行うことができる。

例 1) すべての見出しを標準の 1.5 倍の文字の大きさに太文字にする

```
<style type="text/css">
h1,h2,h3,h4,h5,h6{
    font-size:1.5em;
    font-weight:bold;
}
</style>
```

例 2) 見出し 1 を中央そろえて表示し、見出し 2 から 6 の文字色と背景色をつける。

```
h1{
    font-size:1.5em;
    font-weight:bold;
    text-align:center;
}
h2,h3,h4,h5,h6{
    font-size:1.5em;
    font-weight:bold;
    color:#ffffff;
    background-color:#00ff00;
}
```

例 3) どうしても画像を見出しにしなければいけない場合、余白をゼロにして画像と画像の間に余白ができないようにする。

```
<style type="text/css">
h1{
    font-size:1.5em;
    font-weight:bold;
    margin-top:0;
    margin-bottom:0;
}
</style>
<h1></h1>

```

2) リスト要素の書式変更スタイルシートを用いる

リスト要素は、各項目の先頭に点や丸などの記号、または数字をつけて表示される。

しかし、点や丸などの記号を変更したり、数字の始まりを変更したりしたい場合がある。その場合は、CSS を利用して書式設定を行うことができる。

例 1) 箇条書きの記号を画像に変更する。

```
<style type="text/css">
```



```

ul.disc{list-style-type:disc;}
ul.circle{list-style-type:circle;}
ul.square{list-style-type:square;}
ul.myface{list-style-image:url(myface.jpg)}
</style>
<ul class="myface">
...
</ul>

```

例 2) 箇条書き全体のインデント（左余白）を半分にする。

```

ul{
    margin-left:0.5em;
    padding-left:0.5em;
}

```

例 3) 箇条書きの改行幅や箇条書き記号との間隔などを変更する。

```

li{
    margin-bottom:1em;
    line-height:1.5em;
    padding-left:2em;
}

```

例 4) 箇条書きの文字書式を変更する（ LI 要素または UL 要素に設定することができる）。

```

li{
    font-size:1.5em;
    font-weight:bold;
    color:#00ff00;
}

ul{
    font-size:1.5em;
    font-weight:bold;
    color:#00ff00;
}

```

3) 段落やブロックレベルでの書式変更 (P 要素 DIV 要素) にスタイルシートを用いる

各段落において、字下げしたり、右寄せをしたりしたい場合がある。その場合は、CSS を利用して書式設定を行うことができる。

例 1) 段落(P)を字下げする場合

```
p{
    margin-left:2em;
}
```

例 2) ある特定の範囲を字下げする場合

```
<div style="margin-left:4em">
....
</div>
```

例 3) ある範囲を右寄せする場合

```
<div style="text-align:right">
....
</div>
```

例 4) 字間や行間を調節する場合

```
<p style="letter-spacing:0.25em;line-height:1.5em;">
...
</p>
```

4) 文字書式の書式変更にスタイルシートを用いる

段落の中で特定の箇所を強調する場合がある。その場合は、CSS を利用して書式設定を行うことができる。

例 1) ある箇所を強調したい場合

```
<em> ... </em>
<strong> ... </strong>
<span style="text-decoration: underline;"> ... </span>
<span style="font-style:italic;"> ... </span>
```

注意： 要素で文字書式を変更する場合は、まず論理的要素が利用可能かどうかを検証し、どうしても難しい場合に利用する。

例 2) ある箇所について論理的強調を視覚的に変化させたい場合

```
<em style="color:green"> ... </em>
<strong style="color:red;font-size:2em"> ... </strong>
```

例 3) ある範囲を中央そろえにしたい場合

```
<div style="text-align:center">
...
</div>
```

例 4) フォントサイズを調整する

```
<span style="font-size:1.5em"> ... </span>
```

例 5) 取り消し線を入れる

```
<del> ... </del>
<del style="text-decoration:line-through"> ... </del>
<span style="text-decoration:line-through"> ... </span>
```

注意：取り消し線の場合、大抵の音声ブラウザでは、取り消線があることを知らせないため、通常に書いてあるとおりに読み上げるので、ブラウザが対応するまでは利用しないほうが無難である。

例 6) 字間を調節する場合

```
<!-- 2 文字分の字間を設定 -->
<span style="letter-spacing:2em">日時</span>：2 月 14 日<br>
<!-- 字間等は設定せず -->
集合場所：駅前噴水広場<br>
<!-- 0.5 文字分の字間を設定 -->
<span style="letter-spacing:0.5em">目的地</span>：高尾山<br>
<!-- 0.5 文字分の字間を設定 -->
<span style="letter-spacing:0.5em">雨プロ</span>：博物館<br>

<!-- 字間を-0.3 文字分に縮める設定 -->
<h1 style="letter-spacing:-0.3em">ネットサーフィンを始めましょう</h1>
```

5) JIS X8341-3: 5.2 a を遵守し、文書構造を正しく記述する。

JIS X8341-3: 5.2 a を遵守し、文書構造を正しく記述することで、CSS によって指定している書式や表示スタイルがなくなった場合やユーザーによって CSS が変更された場合でも、最低限の基本的な情報をユーザーに伝えることができるようになっているはずである。

6) CSS による表示位置の指定を行う場合は、HTML の記述の順番が理解できる順番で配置するように記述する。

CSS による表示位置の指定をむやみに多用することは避けるべきである。それは、CSS が有効である場合と無効である場合で表示される順番が異なる可能性があるからである。

もし表示位置の指定をする場合は、CSS が無効になった場合のことを考慮し、HTML の記述する順番どおりに表示しても、最低限の基本的な情報をユーザーに伝えることができるようになっているようにすべきである。

例 CSS で直接表示位置を指定している場合の表示例

```
<html>
<head>
<title>
CSS での位置指定
</title>
```

```

</head>
<body>
<style type="text/css">
div.1{
    position:absolute;
    top:60px;
    left:100px;
}
div.2{
    position:absolute;
    top:78px;
    left:100px;
}
div.3{
    position:absolute;
    top:96px;
    left:100px;
}
div.4{
    position:absolute;
    top:114px;
    left:100px;
}
p.ranking{
    position:absolute;
    top:60px;
    left:20px;
}
</style>
<body>
<h1>今週の血液型ランキング</h1>
<p class="ranking">
第一位:<br>
第二位:<br>
第三位:<br>
第四位:<br>
</p>
<div class="2">A 型</div>
<div class="3">B 型</div>
<div class="4">A B 型</div>

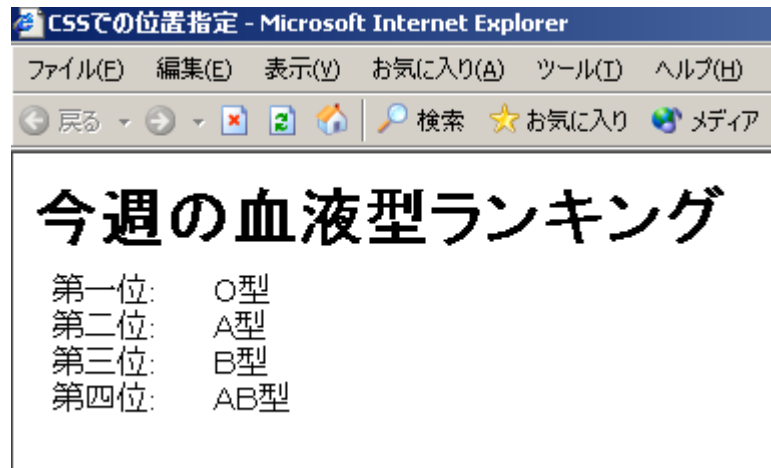
```

```
<div class="1">O型</div>
```

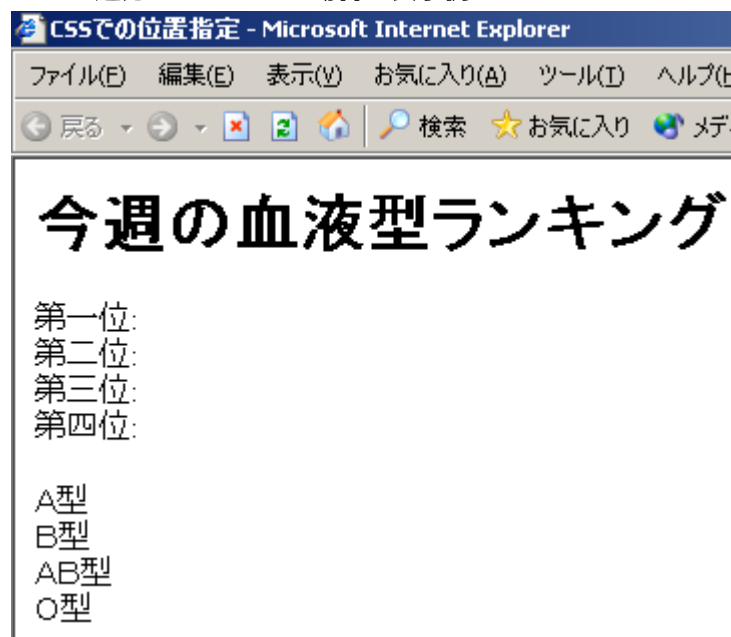
```
</body>
```

```
</html>
```

CSS が適応されている場合の表示例



CSS が適応されていない場合の表示例



7) テーブルの見出し TH 要素の書式の変更（自動的に太文字になるのを抑制する）

```
<style type="text/css">  
th{  
    font-weight:normal;  
}
```

</style>

5.2.c データテーブルの適切なマークアップ

表は、わかりやすい表題を明示し、できる限り簡単な構造にして、適切なマーク付けによってその構造を明示しなければならない。(JIS X8341-3: 5.2 c)

背景と問題点

新聞や雑誌などに載っているテレビ番組表やカレンダーのように、行と列の位置関係に情報の意味があるような場合がある。たとえばテレビ番組表の場合は、行と列の位置関係に時間とテレビ局の情報が関連付けられているし、カレンダーの場合は、曜日や第何週目かといった情報が関連付けられている。

このような表のデータは、視覚的に確認する場合非常に便利であるが、音声読み上げで情報を理解しようとすると非常に困難である。音声ブラウザ上から順番に読み上げていくだけだからだ。たとえばカレンダーの場合、曜日を一通り読み上げて、次に1から30まで読み上げるという状態になり、曜日と日付の対応や、第何週目かといった情報を理解することが困難である。

関連する要素、属性及び宣言

table, hd, th, caption, style, scope, headres, id, summary,

関連項目

5.2 b 構造と表示スタイルの分離(CSS)

技術解説

音声ブラウザの中には「テーブル読み上げモード」があり、位置情報と一緒に読み上げる機能が用意されている。これを活用するためには、あらかじめ行と列の見出しなど、表の構造を明示しておかなければならない。

「テーブル読み上げモード」は、表全体を眺めることはできず、1つの操作では、現在の読み上げ一にあたるセルの中身だけであるため、表の全体の情報を知るためには、表の中のたくさんのセルを上下左右に、まるで迷路のように動き回る必要がある。このとき、セルが結合されていたりすると、利用者が表の中の位置を見失ってしまい、セルからセルへの移動が非常に困難になる場合がある。そのため、表は必要以上にセル結合せず、単純な構造にする。

どうしても、セル結合なので、表の構造が複雑になってしまう場合は、id headers scope などの属性で、セルの関係性を示しておく。

また、その表に対する表題や概要を記述することで、すばやくその表を理解したり、重要度を判断したりする助けとなる。

ソリューション

次の方法を選択して、あるいは組み合わせて用いる。

1) 表の見出しセルには th を利用する

表の見出し、たとえば、テレビ番組表であれば、時間帯やテレビ局名が記述されているセルには、th 要素

を利用する。th 要素を利用する。 th 要素を利用すると、字体が太字等に変更されてしまう場合がある。字体を変更したい場合は、CSS を利用する。

2) セル結合が本当に必要かどうか検討する。

3) セルの結合などで、複雑な構造のため th のみでは十分に表の構造を表現できない場合は、 scope headers, id 属性を利用する

4) 表題をつけるには caption 要素を利用する

caption 要素で表と表題を構造的に関連付けする。表題は、その表がどんな表であるか明確にわかるようなものにする。

5) 表を理解するうえで必要な概要や構造に関する情報を summary 属性として記述する

summary 属性には、表の理解の助けとなるような情報、たとえば、何を伝えようとしている表なのかといった概要や、表の大きさ、行や列に関する情報などを記述しておく、表を理解するうえで大きな助けとなる。

5.2.d レイアウトテーブルの禁止

表組みの要素をレイアウトのために使わないことが望ましい。(JIS X8341-3: 5.2 d)

背景と問題点

現在 table 要素（表組み）を利用して視覚的な画面※レイアウト※を行っている Web ページが多い。これは、※CSS※が無かった時代から、一般的に利用されてきた、視覚的な画面レイアウト手法であった。しかし、表組みで、視覚的な画面レイアウトを行うと、製作者が意図した画面表示の順番と、音声ブラウザが読み上げる順番が異なる場合があるため、音声ブラウザでは、正しく情報を理解できない可能性がある。

関連する要素、属性及び宣言

table, td, th

関連項目

5.2 b 構造と表示スタイルの分離(CSS)

技術解説

レイアウトは CSS を利用することが望ましいが、表を利用して画面レイアウトする場合は、音声ブラウザが読み上げる順番を常に意識して記述していくことが必要である。音声ブラウザは、一般的に HTML に記述されている順番に読み上げる。そのため表組みの場合は、1 行目の左から右へ、2 行目の左から右へ、3 行目の左から右へ、・・・そして最終行の左から右へと読み上げていく。さらに、このような順番で読み上げても、内容が理解できるようにする必要がある。さらに、セル結合などをさせた表の場合は、無意識のうちに思い込んでいる順序と、まったく異なる場合がある。また、テーブルの構造を表すような要素 たとえば th などを利用してしまうと、音声ブラウザなどの支援技術が間違った解釈をしてしまい、ユーザーに正しく情報が伝わらないことがある。

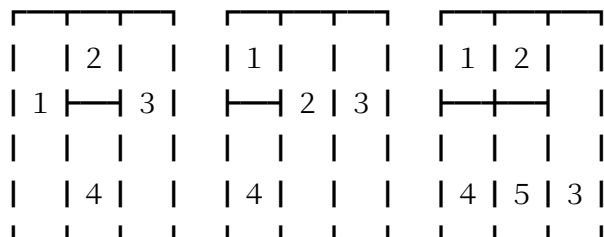
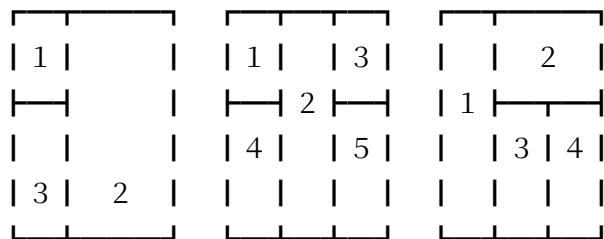
ソリューション

次の方法を選択して、あるいは組み合わせて用いる。

1) 表組みによるレイアウトにおける音声ブラウザの読み上げ順番を配慮する

次のいくつかの表は、音声ブラウザがどのような順番で読み上げるかを示したものである。この読み上げ順序を考慮しながら、表組みによるレイアウトを行う。

					1		2		3
	1		2		3				
					4		5		6
					2		3		4



2) CSS を使ったレイアウト

表組みを利用せずに、CSS を利用して表示位置を調整することもできる。HTML では、音声ブラウザの読み上げ順序に配慮した形で記述し、CSS で表示レイアウトを指定する。

5.2.e 識別可能なページタイトル

ページのタイトルには、利用者がページの内容を識別できる名称を付けなければならない。(JIS X8341-3: 5.2 e)

背景と問題点

ページの※タイトル※は、そのページの概要を理解したり、目的のページかどうか判断したりするのに利用されることが多い。またページを保存したり、※ブックマーク※やお気に入りに登録したりする場合に、ページのタイトルがその名前として利用される。このページのタイトルが記述されていなかったり、ページの内容と異なっていたり、あるいは、複数のページにまったく同じタイトルが記述されていると、開いたページが本当に目的のページかどうか、すぐに判断できなかったりする場合がある。また、せっかくブックマークに保存して後から参照しようとしても、タイトルがわかりやすすないと、ブックマークそのものが、見つけ出せなくなってしまう場合がある。

関連する要素、属性及び宣言

title

関連項目

なし

技術解説

多くの音声ブラウザなどは、最初にページのタイトルを読み上げてから、内容を読み上げる。もし、ページのタイトルが適切でないときや、記述されていない場合、そのページを上から下まで全部読み上げても、そのページが目的のページかどうかの判断ができないことがある。音声ブラウザに限らず、同じようなデザインの場合、同様の問題が発生する可能性がある。このような問題を避けるために、ページのタイトルにそのページの概要がわかるような固有のタイトルを記述する。

さらに、目的のページを探すために、ページを進めたり、戻ったりしている場合、それぞれに内容を表す固有のタイトルが記述されていることで、ページを移動している範囲や現在地が確認しやすくなる。

ソリューション

次の方法を選択して、あるいは組み合わせて用いる。

1) サイトを設計するときにタイトルの付け方の指針を決める。

ホームページやサイトを設計するときに、あらかじめ同一タイトルが存在しないようタイトルの付け方を決める。

2) 利用者が経験する可能性のあるページのパス（流れ）の中で同一タイトルがないか確認する。

音声ブラウザを利用者はページが移動したかどうかを確認するときに最初に確認するのが、ページタイトルである。そのため同一のタイトルであるとページの移動が確認できない場合がある。最低限、利用者が経験する可能性のあるページのパス（流れ）の中で同一タイトルがないか確認する必要がある。

3) ページの内容を最も端的に表す単語やフレーズを利用する。

単にサイト名等だけでなく、ページの内容を端的に表す単語やフレーズを含めることで、利用者の必要かどうかの判断を助けることになる。

4)長い文字列にならないようにする。

これはブラウザによって表示しきれない部分は省略されることを考慮するためである。また、音声でタイトル情報を聞く場合、あまり長いとタイトルを聞くだけで時間がかかってしまう場合がある。

5) サイト名など同じ文字列を入れる必要がある場合は、タイトル文字列の最後に入れるようにする。

小さいが画面でタイトルの後半が省略されてしまう場合、先頭の同じ文字列(サイト名)などしか見えず、結果的にユーザーにとっては同じタイトルしか見えないという事態を避けることができる。また、音声ブラウザを利用する場合も、最初の段階でタイトルの違いがわかるため、効率よく情報を得ることができる。

例)

目次 - HTML4.01 の基礎講座

P 要素 - HTML4.01 の基礎講座

BODY 要素 - HTML4.01 の基礎講座

DIV 要素 - HTML4.01 の基礎講座

4) サイト名など必ず同じ文字列をタイトルの最初に入れる場合は、なるべく最初の文字列は、省略するなど長くならないようにし、その文字列の後は必ず固有のタイトルを付ける。

例)

HTML4.01 の基礎講座 - 目次

HTML の基礎 - P 要素

HTML の基礎 - BODY 要素

HTML の基礎 - DIV 要素

5) どうしても同じタイトルになってしまう場合は、数字などを入れるなど、ページの推移がわかるようにする。

例 1)



例 2)

HTML4.01 の基礎講座 – 目次 1

HTML4.01 の基礎講座 – 目次 2

HTML4.01 の基礎講座 – 目次 3

例 2)

HTML4.01 の基礎講座 – 目次 1/5

HTML4.01 の基礎講座 – 目次 2/5

HTML4.01 の基礎講座 – 目次 3/5

5.2.f フレームの多様の禁止

フレームは、必要以上に用いないことが望ましい。使用するときは、各フレームの役割が明確になるように配慮しなければならない。(JIS X8341-3: 5.2 f)

背景と問題点

フレームは適切な場面で利用すれば、有効な表現手段となる。ただし、音声ブラウザでは、それぞれのフレームを切り替えないと全体が確認できない場合もあるため注意が必要である。また、フレーム切り替えに、キーボード操作をしている場合、切り替え操作が煩雑になる可能性がある。

関連する要素、属性及び宣言

frameset, frame, noframes, iframe, link

関連項目

なし

技術解説

音声ブラウザによっては、1 度に確認できる範囲が1つのフレームに限られ、キーボード操作により、次のフレームまたは、前のフレームといった具合にその都度切り替えながらでないと、全体が確認できない場合がある。そのため、複雑なフレーム構成になっていると、ページ構成が理解できない場合がある。さらに、支援技術によってはフレームに対応していない場合もある。

また、リンクをクリックしたときに、別のフレームに内容が表示されている場合、音声ブラウザによっては、別のフレームに新しいページが開いたことがわからないため、目的の情報が獲られない場合や、次の操作ができなくなってしまう場合も考えられる。

したがって、フレームを利用する場合は、本当にふさわしいかどうか、検討する必要がある。フレームを利用する場合は、各フレームの役割を明確に記述することが大切である。

ソリューション

次の方法を選択して、あるいは組み合わせて用いる。

1) フレームの利用が本当に必要かどうか検討する。

サイトロゴやバナー広告のためだけのフレーム基本的に作成しないほうがよい。特に XHTML1.1 では、フレームは廃止されているため、どうしても必要な場合は内部フレーム (iframe) 要素 や、link 要素の利用もあわせて検討する。

2) frame 要素には title 属性を用いて各フレームの役割を記述する。さらに、参照先の HTML ファイルにも title 要素を用いてフレームの役割を記述する。

Title 属性を記述することで、音声ブラウザなどの支援技術がユーザーにそのタイトルの情報を示してくれる。また、タイトルを付けることで、おのずと不要なフレームを見分けることができるかもしれない。不要なフレームは省略し、できるだけ少なくする。

3) フレームの参照先 HTML がフレームであるような構造は避ける

frameset の src で指定されたファイルが、さらにフレームであるようなページは、音声ブラウザを利用するユーザによっては、非常に複雑な階層構造の中を移動することになるため避ける。

4) noframes 要素には、フレーム未対応のブラウザで表示した場合でも最低限の情報やナビゲーションを記述する。

noframes 要素は必ず記述するようにする。さらに、noframes 要素の内容は、「フレーム対応のブラウザをご利用ください」といった内容であってはならない。必ずフレームが利用できない場合のことを考慮した最低限の内容を記述する。最低限フレームへのリンクやナビゲーションなどを記述する。

5.2.g 階層構造とサイトマップの提示

閲覧しているページがウェブサイトの構造のどこに位置しているか把握できるように、階層などの構造を示した情報を提供することが望ましい。(JIS X8341-3: 5.2 g)

背景と問題点

ページを閲覧していて、次から次へとページを移動していったり、目的のページを探すために、ページを戻ったり進んだりしていくと今現在のページが、サイトの中のどのあたりに位置するのか、わからなくなる場合があったり、作業の中断などによってそれまでの作業があいまいになって、現在位置や作業目的がわからなくなる場合がある。または、作業をやり直すために、スタート地点や基準となる場所へ戻りたい場合、ブラウザの戻る機能を利用しても、なかなか適切な場所に戻れないことがある。

関連する要素、属性及び宣言

なし

関連項目

なし

技術解説

Web の閲覧者がページの中で迷子にならないように、あるいは迷子になっても適切な場所からやり直せるようにする必要がある。特に他のサイトから直接リンクされているようなページの場合、そのサイトのトップページへリンクがあると有効である。また、※ナビゲーションバー※や※パンくずリスト※の採用、あるいはサイトマップを作成して、どのページからも参照できるようにするとよい。

ソリューション

次の方法を選択して、あるいは組み合わせて用いる。

1) トップページおよび、後方、前方へのリンク

各ページには、最低限トップページのリンクを設定することで、いつでもサイトの先頭に戻ってやり直すことができる。また、前のページや次のページがある場合は、後方または前方へのページに移動するリンクを設定する。

2) サイトマップ

サイト全体の構造を現すサイトマップを作成して、各ページのわかりやすいところへリンクを設定しておく。目立たないところにリンクを設定すると、サイトマップは利用されないので注意が必要である。

3) ナビゲーションバーやパンくずリスト

サイト全体が複雑な場合やより快適なナビゲーションを提供する場合は、ナビゲーションバーの採用や、いわゆるパンくずリストなどのナビゲーションの採用を検討する。

5.3 操作及び入力

5.3 a 非デバイス依存

ウェブコンテンツは、特定の単一のデバイスによる操作に依存せず、少なくともキーボードによってすべての操作が可能でなければならない。

背景と問題点

全盲の利用者はマウスポインタが見えないためマウスを利用できない。また弱視の利用者や加齢により視力低下した利用者はマウスポインタが見つからないため、マウスではなくキーボードを利用することがある。また、肢体不自由者や加齢により手の震える利用者は、正確にマウスポインタを制御することが難しい、また筋力の低下した利用者は大きくマウスを動かすことが難しいため、マウスではなくキーボードを利用することがある。多くの支援技術は OS のキーボードショートカットやキーボード操作のエミュレーションによって自動的な処理を行っているので、キーボードで操作できないコンテンツは利用しづらい、あるいはできないことがある。

関連する要素、属性及び宣言

`tabindex`, `accesskey`, `onClick`, `onDblClick`, `onMouseDown`, `onMouseup`, `onMouseover`, `onMouseout`, `onDragDrop`, `onChange`

関連項目

5.2 a, 5.3 f, 5.4 a, 5.4 b, 5.4 e

技術解説

「キーボードによってすべての操作が可能」というのは、コンテンツが提供している機能やサービスなどをキーボードだけで操作できることを求めており、画面上の個々のオブジェクト（アイコンやボタン）すべてを操作可能であることを意味していない。すなわち、キーボードで操作できないボタンをの代わりとなるプルダウンメニューを用意して、同等の機能を提供する方法も許容されている。キーボードアクセスが可能で、Script に依存しない（ページ移動のための）プルダウンメニューの作り方は以下のサイトが参考になる。

Project UDON : http://alfasado.net/udon/accessible_html20/03.html

ソリューション

1) 全てのリンク及び入力項目は、理にかなった順序で移動できるようにする。

通常は要素に `tabindex` 属性を付記しなくてもフォーカスは表示の順序と一致するが、スクリプトを用いる場合や、テキストフィールドやラジオボックスなどが多いフォーム、スタイルシートを用いて表示位置を制御している場合などには、フォーカスが適切な順序で移動しない場合がある。そのような場合には、タブキーによる移動順序が、表示順や操作手順に対応していることを確認する。

- ・画面スクロールは「上矢印」「下矢印」キーで、フォーカスの移動は「Tab」キーで行うなどキーと動作は一貫させる。

`tabindex` 属性で、Tab キーによる移動順序を設定できる。

<要素名 `tabindex=n`>

Tab キーを押した時に `tabindex` で指定した数値の小さなものの順にフォーカスが移動する。

accesskey 属性で、リンクへの移動、ボタンの押下、入力フォームへのフォーカス移動などを行うことができる。Windows の「ファイル(F)」の F と同じような働きをする。

<要素名 accesskey="t">

この例ではリンクボタンに T のキーを割り当てている。詳細の動作は OS やブラウザに依る。

- ・フォーカスの存在するリンクやコマンドは、「Enter」キーを押すまで、実行しない。

2)次に示す JavaScript のイベントハンドラはプルダウンメニューに使用されることが多いが、マウスでの操作を前提にしているため、キーボードでの操作の可否に注意する。

onClick, onDblick, onMousedown, onMouseup, onMouseover, onMouseout, onDragDrop, onChange

例えば、onMouseover だけで、メニューを表示する場合、キーボードの利用者は、メニュー内のリンクを選択できないことがある。

- ・附属書 1 図 14 は、決定ボタン(<input type="button">)を加え、そのボタンの onClick 属性でページ変更の JavaScript を指定している。ドロップダウンコントロールに限らず、ラジオボタンなど他のフォーム要素でも注意が必要である。
- ・附属書 1 図 15 は、ドロップダウンコントロール(<select>)の onChange 属性で、ページ変更の JavaScript を指定しているためである。

3)キーボードだけで操作できるアクセシブルなオブジェクト（プラグイン）を使用する。できない場合は HTML による代替形式を提供する。

代表的なオブジェクトである Macromedia Flash と Adobe PDF を利用する場合には、下記のアクセシビリティ関連サイトの情報を利用し、よりアクセシブルな配慮を行う。Flash MX 2004 からはデフォルトの状態では、Tab キーによるフォーカスが可能になっている。また、Acrobat 6.0 および Adobe Reader 6.0 以降のバージョンには、メニューやツールバーを選択・操作できるショートカットキーが提供されている。

注意事項

なし

※keyword※

キーボード，全盲，肢体不自由

5.3 b 入力欄のわかりやすさ

入力欄を使用するときは、何を入力すればよいかを理解しやすく示し、操作しやすいよう配慮しなければならない。

背景と問題点

- ・ 本項目は、理解（認知）と操作（入力）の2つの観点での配慮を求めたものである。
- ・ 加齢により認知・理解力の低下した利用者や、操作に不慣れな初心者にとって、コントロールに説明がないと何を意味しているのかわからず、またコントロールが複数個ある場合、どれとどれを入力すべきかわかりにくい。
- ・ 入力欄の後ろにラベルが書かれていると、音声ブラウザを使用している全盲の利用者はどこに入力すれば前後関係がわかりにくい（5.3 b）図 6,7,9,10,12,13）。さらに、ラベルと入力欄が離れて書かれていると、拡大表示ソフトを使用している弱視の利用者はその両方を拡大しながら見るのが難しい。

関連する要素、属性及び宣言

label, fieldset, legend

関連項目

5.3 c

技術解説

- ・ ラベルには、入力する文字種（半角文字か全角文字か、カタカナかひらがなか、字数制限があるかなど）を表示すると利用者にわかりやすくなる。
- ・ 上肢に障害のある利用者や加齢による細かな操作が難しい利用者にとって、チェックボックスなど表示面積の小さいコントロールを、マウスで選択することが難しい場合がある。
- ・ ラベルとコントロールを関連づけるとラベルとコントロールが一体化するため、ラベル部分をクリックしてもコントロールを選択することができる。
- ・ ラベルは入力欄の“前”に置くと、音声ブラウザで読み上げたときに入力欄に何を入力すべきがわかりやすい（5.3 b）図 5,8,11）。またラベルと入力欄の位置を近くすると、拡大表示ソフトで拡大したときに、それらを同時に見ることができ、わかりやすい。
- ・ フォームとアクセシビリティに関しては、以下のサイトが参考になる。

The Web KANZAKI : <http://www.kanzaki.com/docs/html/htminfo33.html>

ソリューション

- 1) label 要素を使用し、ラベルとコントロールを関連づける。

例：<label for="uname">名前：</label>

<input type="text" name="username" id="uname" />

- 2) コントロールが多くなる場合は、グループ化する。fieldset 要素を使って、グループ化し、legend 要素でグループのタイトルをつければよい。

ソースプログラムは、JIS 28 ページの 2.3 b) の「記述例」を参照のこと。

(参考) 人が一度に記憶できる数は 7 ± 2 と言われており (認知心理学の分野では「マジカルナンバー」と言われる)、これ以下の数にすると覚えやすい。

附属書 1 図 16 は、その範囲内に選択肢を抑えているが、附属書 1 図 17 は、すべての選択肢が羅列されていて見にくく、覚えにくい。

3) 入力する文字種など入力書式に、自由度を持たせる。

例：プログラムの、英数字の半角、全角文字のどちらでも入力を受け付けるようにしておく。

例：サーバ側で、入力された文字属性を“全角-半角”変換する。

参考：XForms 1.0 Appendices E Input Modes <http://www.w3.org/TR/2003/REC-xforms-20031014/sliceE.html>

(※梅垣注：XForms はブラウザが未対応！！！！！！！！)

4) 文字の入力フィールドには、入力すべき文字種 (漢字、全角文字など) を記述する。(3) のような手段で対応できず文字種に制限がある場合には、その旨を記述する。

5) 必須入力項目と任意入力項目との違いを、明確に示す。音声ブラウザでの読み上げを考慮し、必須であることは文字色や記号だけで表現しない。

6) 入力に関する指示、説明、注意事項などは、入力項目の近くに表示する。音声ブラウザの使用を考慮し、コントロールの前に記述する (これにより、入力操作を行う前に、入力方法が把握できる)。

7) 入力しやすく、また入力した結果を確認しやすくするために、テキストボックス内の文字サイズは特別な理由がない限り、ラベルの文字サイズより小さくしない。また、入力領域の文字サイズは利用者がブラウザによって文字サイズを変更した場合でも連動するようにしておく。なお、入力領域は十分な幅を持たせておくと、文字サイズを大きくした場合でもスクロールすることなく入力した文字列を容易に確認できる。

注意事項

なし

※keyword※

入力、フォーム、コントロール、ラベル、認知

5.3 操作及び入力 (JIS X 8341-3:5.3)

5.3 c 入力時間制限

入力に時間制限を設けないことが望ましい。制限時間があるときは事前に知らせなければならない。

背景と問題点

加齢などによる認知・理解力の低下、視力低下による画面の見えにくさ、筋力低下、麻痺（痺）、手の震え、不随意運動などのために、応答の入力に時間がかかる場合がある。

関連する要素、属性及び宣言

関連項目

5.3 d

技術解説

ウェブコンテンツによっては、一定時間内にアクセスしないと、システム側で自動的にログアウトする機能を設けている場合がある。しかし、利用者によっては短時間で入力することができず、時間内に操作を完了できない場合がある。したがって原則的に入力に要する時間に制限を設けないことが必要である。ただし、コンテンツ利用時の離席中の第三者による成りすましや情報漏えいなどから利用者を保護するために時間制限を設けざるを得ない場合もある。その場合でも、事前に時間制限がある旨を告知すると、利用者が操作する際にあわててることを少なくできる。

ソリューション

- 1)可能な限り時間制限は設けない。
- 2)セキュリティなどの理由から、やむを得ない場合は、時間制限があることを表示する。その場合でも、操作をするために十分な時間設定にし、経過時間、残り時間または受付終了時刻をページ内で表示する。利用者側での設定時間延長については、5.3 d を参照。

注意事項

なし

※keyword※

時間、制限、セキュリティ

5.3 操作及び入力 (JIS X 8341-3:5.3)

5.3 d 制限時間の延長・解除

時間制限があるときは、利用者によって制限時間を延長又は解除できることが望ましい。これができないときは、代替手段を用意しなければならない。

背景と問題点

加齢などによる認知・理解力の低下、視力低下による画面の見えにくさ、筋力低下、麻痺（痺）、手の震え、不随意運動などのために、応答の入力に時間がかかる場合がある。

関連する要素、属性及び宣言

関連項目

5.3 c

技術解説

5.3 c では“時間制限を設けない、設ける場合は事前に告知する”ことを求めたが、本項目では更に拡張し、利用者の保護のためなどで設けた制限時間を延長したり、解除できるようにすることを求めている。時間設定を行う場合でも、ウェブコンテンツ提供者側であらかじめ十分な時間を設けている場合が多いが、それでもすべての利用者が入力できるとは限らないからである。制限時間を延長したり、解除したりできれば、入力に非常に時間のかかる重度の身体障害者も確実にウェブコンテンツを利用できるようになる。ただし、(1)前述の利用者の情報保護との兼ね合いもあり、操作に要する制限時間の延長や解除が難しい、(2)ウェブコンテンツの提供者側に利用終了時間がある、などの場合はインターネットを使用しない代替手段を用意する。これは通信回線が使用できない環境、利用者にとっても有効である。

ソリューション

- 1)ウェブコンテンツ内に「利用延長ボタン」を設けておき、利用者がそれを押すことで1回分の時間延長ができるようにする。その場合は、どのくらいの時間が延長されるかを明示しておく。
- 2)ウェブコンテンツ内に「時間制限解除」を設けておき、利用者がそれを押すことで制限時間を撤廃できるようにする。時間制限をなくすことによって、利用者に不利益が生じる可能性がある場合は、その旨を事前に告知する。
- 3)時間制限を設けている場合は、経過時間、残り時間または受付終了時刻をページ内で表示する。
- 4)代替手段として、電子メール、Fax、電話でも手続きができるようにする。

注意事項

なし

※keyword※

時間、制限、セキュリティ

5.3 操作及び入力 (JIS X 8341-3:5.3)

5.3 e ページの自動更新、移動の禁止

利用者の意図に反して、又は利用者が認識若しくは予期することが困難な形で、ページの全部若しくは一部を自動的に更新したり、別のページに移動したり、又は新しいページを開いたりしてはならない。

背景と問題点

- ・ 新しいページを開いた直後、またはページを参照しているときに、そのページが自動的に更新されたり、他のページに移動してしまうと、加齢により認知・理解力が低下した利用者や視力低下による画面を見にくい利用者は、新しいウィンドウやタブが開いたことに気づかなかったり、意図しないページが勝手に開いたことに戸惑ったりする場合がある。
- ・ 新しいウィンドウが開いた場合、筋力低下、麻痺（痺）、手の震え、不随意運動のある利用者は、不要なウィンドウを閉じる操作が困難であり、どちらが元のページであったかわからなくなる。
- ・ 音声ブラウザ・点字ディスプレイの利用者は、読み上げ時にページが自動的に更新されると、再度最初から読まなければならなくなる。

関連する要素、属性及び宣言

a, area

関連項目

技術解説

- ・ 本項目は、“利用者の意図や予期に反して”新しいページが開くことを禁止するものであり、利用者が意図的に新しいページを開くことを禁止しているわけではない。たとえば、(1)ヘルプページを別ページとして新たに開きたい、(2)数値を比較対照したい、などの場合、複数ページを開くことは、操作を容易にしたり理解しやすくできるので有効である。
- ・ ウェブコンテンツの表示や操作の都合上、別ページを開くことが必要な場合は、あらかじめ新しいページが開くことを告知する。なお、指定したページの URL が更新した場合は利用者に告知せず新しいページを開いても良い。

ソリューション

1)リンクのある文字列上でマウスの右クリック操作し「新しいウィンドウで開く」を選択すれば、利用者は自分の意思で、そのページを新しいウィンドウで開くことができる。したがって、ページには新しいウィンドウを開くことを目的として、a 要素もしくは、area 要素の target 属性は、_blank, _new を指定しない。

2)あらかじめリンク元で新しいウィンドウが開くことを明示する方法としては、

- ・ 「ニュース (新しいウィンドウで表示)」などと表記する
- ・ 新しいページが開くことを示す図記号を付記する

などがある。(JIS 5.3e) 例 2.)

3)やむをえず更新する場合は、あらかじめそのことを告知しておく。

例：「このページは 15 分ごとに最新情報を更新します」などとする。

4)リダイレクトすることが利用者の利便性に供与する場合は、告知は必ずしも必要ではない。

例：ウェブサイトの URL が変更（移動）になった場合のサーバによるリダイレクト。

参考：WCAG ではリダイレクトの要する時間を 0 秒としている。

5)情報がその性質的に時々刻々に変化するもので、最新情報を提供することが利用者の利便性に供与するものはリフレッシュする際に、毎回告知する必要はない。

例：気象情報，株価情報。

注意事項

なし

※keyword※

ページ，更新，移動

5.3 操作及び入力 (JIS X 8341-3:5.3)

5.3 f 一貫性のある基本操作部の提供

ウェブサイト内においては、位置、表示スタイル及び表記に一貫性のある基本操作部を提供することが望ましい。

背景と問題点

- ・ 基本操作部（ページ共通のナビゲーションバー、「トップページ」「サイトマップ」などへのリンク、およびページ内リンクなど）がページごとに異なると、利用者は新しいページを参照するたびに、基本操作の方法を理解しなおさなければならない。特に、加齢により認知・理解力が低下した利用者や視力低下による画面を見にくい利用者は、その理解に時間がかかる。
- ・ 音声ブラウザの利用者は、同じ操作部であれば読み飛ばすことができるが、毎回操作部を最初から読まなければならない。
- ・ キーボードやマウスの操作が難しい筋力低下、麻痺（痺）、手の震え、不随意運動のある利用者は、ページ送りボタンなどが同じ位置に表示されていないと、操作に時間がかかる。

関連する要素、属性及び宣言

関連項目

5.2 a

技術解説

- ・ 本項目はページ共通の基本操作部について規定したものであるが、ページ固有の操作部がある場合も見やすさ、使いやすさ（ユーザビリティ）の観点から同様な配慮を行う。
- ・ ウェブサイト管理者がテンプレートを積極的に活用することによって、ウェブサイトの統一感を実現すると同時に、ウェブページ制作者のページ制作工数を低減することもできる。

ソリューション

- 1) 基本操作部分の表現は、ウェブサイト内あるいは同一カテゴリ内で一貫性をもたせる。
- 2) [戻る]ボタンなどに使用する画像は、ウェブサイト内で統一し、画像の alt 属性も必ず統一する。
- 3) 基本操作部分の表現と反応の対応関係は、ウェブサイト内で統一する。例えば「サイトマップ」と書かれたリンクからは、常に同じウェブページにジャンプさせる。

注意事項

なし

※keyword※

基本操作、一貫性

5.3 操作及び入力（JIS X 8341-3:5.3）

5.3 g リンクの識別と操作

ハイパリンク及びボタンは、識別しやすく、操作しやすくすることが望ましい。

背景と問題点

- ・ 本項目は、理解（認知）と操作（入力）の2つの観点での配慮を求めたものである。
- ・ 加齢により認知・理解力が低下した利用者や視力低下による画面を見にくい利用者は、写真のような画像や下線のないテキストをリンクであることに気が付かない場合がある。すなわち、利用者は必要なリンクを選択することが困難になる。
- ・ リンクのある文字や画像が隣接している場合、リンクの区切りを把握できない場合がある。
- ・ 音声ブラウザはリンク部分のみを読み上げる機能がある。「ここ」「こちら」などのように指示代名詞だけにリンクを付けると、視覚に障害のある利用者は、正しくリンクを理解できない場合がある。
- ・ リンクのある文字や画像が隣接していたり、それぞれの面積が小さすぎたりすると、筋力低下、麻痺（痺）、手の震え、不随意運動のある利用者は、意図したリンクを選択するのが困難な場合がある。

関連する要素、属性及び宣言

関連項目

5.4 b

技術解説

- ・ ハイパリンクやボタンなどを独自にデザインする場合は、それがリンクやボタンであることがわかるよう形状や配色に配慮すると同時に、操作しやすい十分な大きさを確保する。

ソリューション

- 1) 適度なリンク範囲を確保できるように、文字列全体にリンクを設定するか、大きな画像・文字などを用いる。
- 2) 隣接するリンクの間は、十分な間隔を設ける。
- 3) スタイルシートを用いる場合には、テキストや画像の配置などを考慮する。
- 4) 指示代名詞だけでリンク先を指定しない。
例：「より詳細な情報はこちら」の文字列のうち、「こちら」ではなく、「より詳細な情報」をリンクとする。
- 5) 「クリック!」「click here!」など、リンク先の内容を推測できない表現は避ける。
- 6) リンクする範囲を広げるだけで、わかりやすくなることもある。
例：「x xに関するより詳細な情報」の文字列のうち、「情報」だけをリンクとするのではなく、「より詳細な情報」をリンクする。
- 7) ブラウザのデフォルトのリンクの表示スタイルは、特に必要でなければむやみに変更しない。
（※梅垣注：リンクだけではなく、visited など明記したほうがいい）
- 8) テキストリンクが横に並ぶ場合、各テキストリンクの間に縦線（|）や斜線（/）などを入れる。
- 9) 行間を設定する場合は、狭くしない。

- 10) 画像に表示された文字や絵文字（アイコン）は，ボタンの機能を正しく推測できるものを用いる。
- 11) リンクのある画像は，枠をつける，影をつけるなどして押せそうな（選択・押下できそうな）形のデザインにする。ロールオーバー機能を付加しても良い。なお，表現はサイト内で一貫して用いる。
- 12) 特定の技術やプラグインで，ボタンやラジオボタンなどを独自にデザインする場合は，操作方法が見ただけでわかるように作成する。

注意事項

なし

※keyword※

リンク，ボタン，識別

5.3 操作及び入力 (JIS X 8341-3:5.3)

5.3 h ナビゲーションスキップ

共通に使われるナビゲーションなどのためのハイパリンク及びメニューは、読み飛ばせるようにすることが望ましい。

背景と問題点

ナビゲーションバーやメニューなどをページの最初につける場合は、音声ブラウザはページを表示するたびに、それらを必ず読み上げる。そのため、音声ブラウザの利用者は本文を読み上げるまでの時間がかかり効率が悪い。

関連する要素、属性及び宣言

関連項目

5.3 f

技術解説

- ページの最初の部分に、ハイパリンクやメニューを読み飛ばせるスキップ機能を付記しておく、利用者が必要に応じて本文を直接読み上げることができる。
(参考) 音声ブラウザ利用者のみを対象にする要件であるため、隠しリンク (ソースコードには記述する (= 音声ブラウザでは読み上げる) が画面表示はしない) にしておく場合もある。
- ナビゲーションバーやメニューなどの分量が多いと、ウィンドウに最初に表示される本文の量が少なくなる。そのため、ページの下の方を表示させたい場合、スクロール操作が必要になるが、筋力低下、麻痺 (痺)、手の震え、不随意運動がありキーボードやマウス操作が難しい利用者にとっては、特に効率が悪い。一般の利用者の使いやすさ向上 (ユーザビリティ) のためにも、ナビゲーションバーやメニューなどは必要最小限にする。

ソリューション

- 各ページで使用している共通のナビゲーションバーやメニューなどは、音声ブラウザの使用時にスキップできるよう、本文へのページ内リンクを設ける。

例 1: ナビゲーションが始まる箇所に透過 gif 画像を埋め込み、その画像に本文の開始位置に設定したアンカーへのリンクを設定する。なお gif 画像には音声読み上げソフト用に「alt=""ここからメニュー""」など、ページ内の各エリアに関する補足説明を記述する。

例 2: 背景色と同じ色のテキストで記述する、

例 3: スタイルシートで非表示の設定をしたテキストを用いる。

例 4: 非常に小さなフォントで書く。

なお、例 2~例 4 は、ブラウザの仕様によって正しく表示されないことがある。

- ページのヘッダーに該当するナビゲーションバーや、画面の左側に表示されるメニューは、できるだけ

小さくする。

3) スタイルシートを使用し，本文を先に読み上げ，メニューを後で読み上げる方法もある。この場合は逆にメニューに移動するためのページ内リンクを設ける。

注意事項

なし

※keyword※

ナビゲーション，メニュー，スキップ，音声ブラウザ

5.3 操作及び入力 (JIS X 8341-3:5.3)

5.3 i 誤操作の復帰

利用者がウェブコンテンツにおいて誤った操作をしたときでも、元の状態に戻ることができる手段を提供しなければならない。

背景と問題点

- ・ 前のページの内容を十分確認しないまま次のページに移動してしまった場合、前のページに容易に戻れないと、利用者はページを表示させるための操作を再度行う必要がある。加齢などによる認知・理解力が低下したり、視力低下によって画面が見えにくい利用者は画面表示を確認するための時間がかかり効率が悪い。また、筋力低下、麻痺（痺）、手の震え、不随意運動がありキーボードやマウス操作が難しい利用者にとっては、操作の効率が悪い。
- ・ 特に利用者が情報を入力するフォームの場合、入力済みのフォームに戻れない場合や、入力済みのデータが消去されている場合は、入力操作を再度繰り返す必要が生じる。
- ・ 直前のページに戻れる機能がないと、加齢により認知・理解力が低下した利用者や初心者は、入力ミスが許されない不安感を感じさせる。

関連する要素、属性及び宣言

関連項目

5.3e, 5.4b

技術解説

- ・ ウェブコンテンツ自身に前のページに戻る機能を付記するか、ブラウザの「戻る」機能を利用できるようにする。
- ・ フォームのような入力操作を生じさせる場合、ウェブコンテンツ上に前のページに戻れることを明記しておくで安心感が高まる。

ソリューション

- 1) ブラウザの[前に戻る]ボタンなどで、前に表示した画面を、簡単に表示できるようにする。特に、フォームの場合、ブラウザの[前に戻る]ボタンを押しても、「セッションが切れました」などのメッセージを表示しない。
- 2) ブラウザの[前に戻る]ボタンなどで、フォームに戻ったとき、入力済みのデータをそのまま表示する(消去しない)。
- 3) 入力後、データの確定前に、利用者自身が入力内容を確認できるようにする。
例：データの確定前に入力内容の確認画面を表示する。
- 4) 申し込み内容を表示し、その内容を印刷できるようにする。
- 5) サイトで自動的に入力内容をチェックする場合、文字種などの入力エラーメッセージは、こまめにその都度表示する。

- 6) 入力内容のエラー画面を表示するときは、問題のある入力項目と問題のない入力項目を明確に示す。さらに、問題の修正方法を明確に示す。
- 7) フォームの送信後に、利用者への適切なフィードバックを行う。

注意事項

なし

※keyword※

誤操作

5.4 非テキスト情報

5.4 a 画像の代替テキスト

画像には、利用者が画像の内容を的確に理解できるようにテキストなどの代替情報を提供しなければならない。(対応：JIS X8341-3:5.4 a)

背景と問題点

視覚障害のため、音声ブラウザなどを用い音声によって情報を得ている場合がある。その場合、音声ブラウザなどは、画像を音声化できない代わりに、その画像の代替テキストを読み上げる。したがって、代替テキストがないと利用者は内容を理解することができない。

また、テキストブラウザの利用者など、画像を表示しないでウェブコンテンツを閲覧しているときも、代替テキストがないと利用者は内容を理解することができない。

関連する要素、属性及び宣言

img

関連項目

技術解説 画像に代替テキストを提供することで、その画像が伝えている情報は、視覚系ブラウザで画面に表示されるだけでなく、音声あるいは点字などに変換することが可能になる。また、画像を非表示にしている視覚系ブラウザあるいは画像を表示しないテキストブラウザなどでは、画像を表示できないかわりに代替テキストをその画面に表示することができる。

Web ページに画像を用いることで視覚的な訴求力を高めることができるが、同時にその画像を見ることができない利用者のために代替テキストを記述することで、その画像が伝えている情報をより多くの利用者に伝えることができるようになる。

ソリューション 次の方法を選択して、あるいは組み合わせて用いる。

ポイントとなるのは以下の 2 点である。

- (1) 必ず `img` 要素には `alt` 属性をつけること。
- (2) 画像が用いられている意図や目的に応じて、代替テキストを簡潔に記述すること。ただし、その画像が情報を伝えていない場合は、`alt` 属性を空 (`alt=""`) にすること。

その画像を使用する目的あるいはその画像がページで果たしている役割、前後および周囲との文脈により、以下から適切な方法を用いて、画像の代替テキストを記述する。なお、W3C が公開している(X)HTML の勧告（仕様）によれば代替テキストが不要な画像の `img` 要素にも `alt` 属性は必須であり、代替テキストが不要であることを示すためにも `alt` 属性を空 (`alt=""`) にしなくてはならない。

1) `img` 要素の `alt` 属性値に代替テキストを記述する

画像が伝えている情報を簡潔にテキストで記述する。

例 1 画像の代替テキストを、`img` 要素の `alt` 属性に記述する。

```

```

2) `alt` 属性値を空 (`alt=""`) にする

代替テキストが不要な場合は、`alt` 属性値を空 (`alt=""`) にする。また、その画像の周囲にあるテキストで説明している場合も同様である。

例 1 以下に挙げるような画像は、`img` 要素の `alt` 属性を空 (`alt=""`) にする。

- ・ スペース画像（レイアウトのために使用する透明の GIF 画像）
- ・ 装飾だけを目的にした画像

```

```

3) その画像の周囲にあるテキストを代替テキストとする

例 1 画像の周囲にあるテキストが画像の内容を説明している場合、あるいは画像の周囲に説明するテキストを置く場合は、繰り返しを避けるために `alt` 属性を空 (`alt=""`) にする。

```

```

```
<p>（…画像の内容を説明しているテキスト…）</p>
```

4) 代替情報としてデータテーブルを提供する

数値データを円グラフあるいは棒グラフなどの画像にして提供する場合は、そのグラフが伝えている数値データの内容をアクセシブルなデータテーブルであわせて提供する。

例 1 `img` 要素の `alt` 属性には代替テキストとして、グラフがあることを記述する。

```

```

```
<table>
```

```
<caption>表 1. 過去 10 年間の利用者人口推移</caption>
```

```
<tr>
```

```
<th></th>
```

```
.....
```

5) 別ページに代替テキストを記述する

alt 属性値に記述する代替テキストが長くなってしまう場合には、別に説明ページを設けて longdesc 属性でリンクを提供する。また、longdesc 属性をサポートしていないユーザーエージェントのために、画像のすぐ近くに同じ説明ページへのリンクをあわせて提供する必要がある。

例 1 img 要素の longdesc 属性で説明ページの URL を指定する。あわせて、longdesc 属性の代替手段として、同じリンク先を指定したテキストリンクを画像のすぐそばに置く。

```

<a href="detail.html">組織図の詳細</a>
```

注意事項 画像の代替テキストは、その画像の用途あるいは目的、役割などにより、ある程度パターン化できるが、実際には前後の文脈や周囲にあるテキストなどにより、ケースバイケースであることが少なくない。重要なのは、その画像を画面で見ることのできない利用者にも同等の情報が伝わることであり、ここで挙げているソリューション以外にもその方法は考えられる。何のために代替テキストを提供するのかをふまえて、適宜その提供の方法を検討する必要がある。

※keyword※ 代替テキスト、alt 属性、longdesc 属性

5.4 非テキスト情報 (JIS X 8341-3:5.4)

5.4 b ハイパリンク画像の代替テキスト

ハイパリンク画像には、ハイパリンク先の内容が予測できるテキストなどの代替情報を提供しなければならない。(対応：JIS X8341-3:5.4 b)

背景と問題点

視覚障害のため音声ブラウザなどを用い音声によって情報を得ている場合がある。その場合、音声ブラウザなどは、ハイパリンク画像（リンクを設定している画像）を音声化できないので、代わりに代替情報（テキスト）を読み上げる。したがって、代替情報がないと利用者はハイパリンク先を識別・理解することができない。

関連する要素、属性及び宣言

img, area, input

関連項目

技術解説 リンク画像の img 要素に alt 属性がないと、音声ブラウザなどではそのリンク先に指定している URI (a 要素の href 属性値) がそのまま読み上げられてしまう。また、画像を表示できないブラウザでは何も表示されないため、利用者はリンクであることは認識できてもそのリンク先は全く分からない。

また、イメージマップを使用する際には、マップ内のリンク領域を定義できない場合を除いて、クライアントサイド・イメージマップを用いる。その際、イメージマップそのものを説明する代替テキストと、マップ内の各リンク領域の代替テキストとをあわせて記述する必要がある。

どうしてもサーバサイド・イメージマップを使用するしかない場合だが、サーバサイド・イメージマッ

ブは利用者がマウスを操作できることが前提となる。マウスを操作できない利用者のために、マップ内に設定されている全てのリンクの代替手段となるテキストリンクをマップ画像のすぐそばに置かなければならない。

ソリューション 次の方法を選択して、あるいは組み合わせて用いる。

どの方法を用いるにしても、リンク画像の代替テキストは、テキストリンクと同様に、その部分だけでリンク先の内容が分かるように記述する必要がある。

1) img 要素の alt 属性値に代替テキストを記述する

その部分だけでリンク先の内容が分かるように記述する。

例 1 リンク画像の代替テキストを、img 要素の alt 属性に記述する。

```
<a href="about.html">
  
</a>
```

2) alt 属性値を空 (alt="") にする

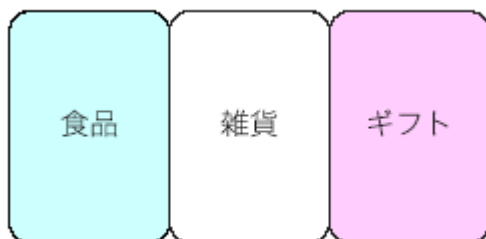
リンク画像のすぐそばに同じリンク先へのテキストリンクがある場合は、繰り返しを避けるために、リンク画像の alt 属性値を空 (alt="") にする。

例 1 img 要素の alt 属性を空 (alt="") にする。

```
<a href="about.html">
  
</a><br>
<a href="about.html">会社概要</a>
```

3) クライアントサイド・イメージマップの area 要素の alt 属性に代替テキストを記述する

例 1 マップ内のリンク領域の代替テキストを、area 要素の alt 属性に記述する。



```
<map name="menu">
  <area shape=rect coords="0, 0, 60, 90" alt="食品" href="../syokuhin.html">
  <area shape=rect coords="60, 0, 120, 90" alt="雑貨" href="../zakka.html">
  <area shape=rect coords="120, 0, 180, 90" alt="ギフト" href="../gift.html">
</map>
```

例 2 イメージマップ全体の代替テキストを、img 要素の alt 属性に記述する。

```

```

4) サーバサイド・イメージマップの代替手段としてテキストリンクを提供する

サーバサイド・イメージマップに設定してある全てのリンクと同じリンク先へのテキストリンクをマップ画像のすぐそばに置く。

例 1 イメージマップ画像の alt 属性に代替テキストを記述し、あわせて代替手段となるテキストリンクをすぐそばに置く。

```
<a href="cgi-bin/worldmap/database">  
  
</a>  
<br>  
<a href="asia.html">アジア</a>|<a href="europe.html">ヨーロッパ</a>|.....
```

5) フォームのボタン画像は、input 要素の alt 属性に代替テキストを記述する

フォームのボタンを画像にする際には、その input 要素に alt 属性をつけて、ボタンの操作結果をあらわす代替テキストを記述します。

例 1 リンク画像の代替テキストを、img 要素の alt 属性に記述する。

```
<input type="image" value="送信" src="images/submit.gif" name="submit" alt="送信">
```

注意事項 リンク画像の代替テキストは、そのリンク画像の用途あるいは目的、役割などにより、ある程度パターン化できるが、実際には前後の文脈や周囲にあるテキストなどにより、ケースバイケースであることが少なくない。重要なのは、その画像を画面で見ることのできない利用者にもそのリンク先の内容が分かるようなラベル（文言）を代替テキストとして記述することである。

※keyword※ 代替テキスト、alt属性、イメージマップ、クライアントサイド・イメージマップ、サーバサイド・イメージマップ

5.4 非テキスト情報（JIS X 8341-3:5.4）

5.4 c 音声の代替

ウェブコンテンツの内容を理解・操作するのに必要な音声情報には、聴覚を用いなくても理解できるテキストなどの代替情報を提供しなければならない。（対応：JIS X8341-3:5.4 c）

背景と問題点

聴覚障害があるとき、音だけで情報を提供していると提供されていることが認識できず、その内容も理解できない。音声情報は、利用者が聴覚を用いることでその情報を得ることができる。しかし、聴覚に障害があったり、PCの音声出力をオフにしていたり、周囲が騒がしい環境で利用していたりすると、その情報を全くあるいはほとんど得ることができない。

関連する要素、属性及び宣言

object

関連項目

技術解説

音声で提供する情報には代替テキストを提供して、利用者が聴覚を用いなくてもその内容を理解できるようにする必要がある。

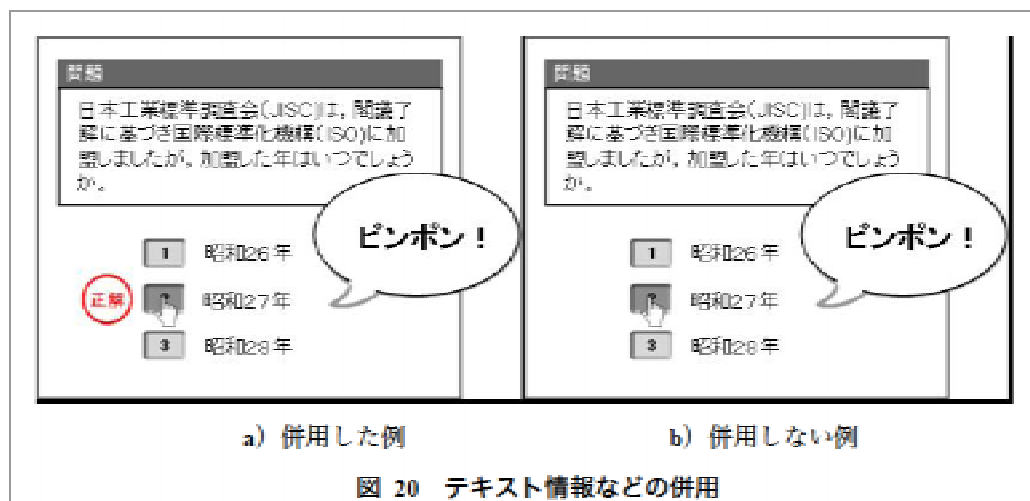
警告音などを用いる際は、その音が聞こえなくても分かるように、代替テキストなどにより画面で視覚的にもその情報を伝える。

ソリューション 次の方法を選択して、あるいは組み合わせて用いる。

1) 音声情報の代替テキストを提供する

音声情報が聞こえなくても、同じ情報が視覚的に伝わるようにする。

例 1 音声の再生に合わせて、画面に代替テキストも表示する。JIS X 8341-3:2004 規格書 P12 より引用。



※keyword※ 代替テキスト、音、音声

5.4 非テキスト情報（JIS X 8341-3:5.4）

5.4 d 動画の字幕と状況解説

動画など時間によって変化する非テキスト情報には、字幕又は状況解説などの手段によって、同期した代替情報を提供することが望ましい。同期して代替情報が提供できない場合には、内容についての説明を何らかの形で提供しなければならない。（対応：JIS X8341-3:5.4 d）

背景と問題点

聴覚障害がある場合、音によって情報を提供していると提供されていることが認識できず、その内容も理解できない。

視覚障害がある場合、どのような映像が表示されているのか理解できないことがあるために、内容を理解できない場合がある。

認知又は記憶に障害がある場合、字幕、状況説明などがあると繰り返し見たりすることによって、内容を理解しやすくなる場合がある。

関連する要素、属性及び宣言

object

関連項目

技術解説 音声付の動画には、画面で視覚的に伝えている情報と音声で聴覚的に伝えている情報とがあり、それぞれに代替情報を提供する必要がある。

まず、画面で視覚的に伝えている情報については、音声による補足説明を副音声で提供し、主音声と重ならないように同期させるのがベストだが、実際には困難なケースが多い。そこで、これが困難な場合は、音声や点字に変換できるように、主音声を書き起こしたテキストおよび補足説明となる代替テキスト（副音声の原稿にもなるテキスト）を何らかの手段で提供する。テキスト情報になっていれば、音声あるいは点字に変換することができるので、視覚に障害があっても画面で伝えられている情報を得ることができる。

次に、音声で聴覚的に伝えている情報については、キャプション（字幕）を提供し、画面上で同期させるのがベストだが、これもやはり困難なケースが少なくない。そこで、これが困難な場合は、キャプションと同じ要領で代替テキストを書き起こして別ページで提供するなど、音声を聞くことができない利用者にも同等の情報が伝わるようにする。テキスト情報になっていれば、それを点字に変換することができるので、例えば視覚と聴覚の両方に障害があっても、利用者はその情報を得ることができる。

ソリューション 次の方法を選択して、あるいは組み合わせて用いる。

1) 画面で視覚的に伝えている情報を同期させた副音声で提供する

副音声で、登場人物の表情および動作、背景、画面上の文字などを、主音声と重ならないように提供する。

例 1 以下の []部分を副音声のナレーションとして録音し、画面と同期させる。

シンバ「それっ！」

[シンバは外へ駆け出して行き、両親が後に続きます。サラビはやさしく微笑みながら、シンバを父の方へ行くようにと促します。2頭は並んで座りながら、黄金に輝く日の出を見つめています。]

ムファサ「見なさい、シンバ。太陽が輝かせているのは、私たちの王国だ。」
シンバ「ほんとだ！」

2) 画面で視覚的に伝えている情報を代替テキストで提供する

登場人物の表情および動作、背景、画面上の文字などを、副音声ではなく、代替テキストとして提供する。

例1 登場人物のセリフおよび[]部分の補足説明を代替テキストとして別ページなどで提供する。

シンバ「それっ！」
[シンバは外へ駆け出して行き、両親が後に続きます。サラビはやさしく微笑みながら、シンバを父の方へ行くようにと促します。2頭は並んで座りながら、黄金に輝く日の出を見つめています。]
ムファサ「見なさい、シンバ。太陽が輝かせているのは、私たちの王国だ。」
シンバ「ほんとだ！」

3) 音声で聴覚的に伝えている情報を同期させたキャプション（字幕）で提供する

登場人物のセリフ、効果音などをテキストに書き起こして、同期したキャプションとして画面上で提供する。

例1 登場人物のセリフ以外の音もキャプションとして提供する。

* 「E.T.」のあるシーンに対する字幕。電話のベルが3回鳴った後、声が聞こえる場面。

[電話の呼出音が鳴る]

[呼出音]

[呼出音]

「もしもし？」

関連情報

SMIL 1.0 仕様書（日本語訳）

<http://www.doranekeo.org/misc/smil10/19980615/Overview.html>

QuickTime と SMIL

http://www.apple.com/jp/quicktime/tools_tips/tutorials/qtsmil.html

マイクロソフト - PC マルチメディアのキャプションとオーディオの説明

http://www.microsoft.com/japan/msdn/accessibility/general/ATG_CCandAudioDesc.asp

マクロメディア - Macromedia Flash MX 2004 によるアクセシブルコンテンツの作成：キャプションと字幕

<http://www.macromedia.com/jp/macromedia/accessibility/features/flash/captions.html>

4) 音声で聴覚的に伝えている情報を代替テキストで提供する

登場人物のセリフ、効果音などをテキストに書き起こして、代替テキストとして提供する。これをトランスクリプトとよぶが、このトランスクリプトへのリンクは目立つところに配置すべきである。

例1 登場人物のセリフ以外の音も代替テキストとして記述する。この代替テキストは、

* 「E.T.」のあるシーンに対する字幕。電話のベルが3回鳴った後、声が聞こえる場面。

[電話の呼出音が鳴る]

[呼出音]

[呼出音]

「もしもし？」

※keyword※ 動画、音声、代替テキスト、キャプション、字幕、副音声、トランスクリプト

5.4 非テキスト情報 (JIS X 8341-3:5.4)

5.4 e オブジェクトの代替形式

アクセス可能ではないオブジェクト、プログラムなどには、利用者がその内容を的確に理解し操作できるようにテキストなどの代替情報を提供しなければならない。また、アクセス可能なオブジェクト又はプログラムに対しても、内容を説明するテキストなどを提供することが望ましい。(対応：JIS X8341-3:5.4 e)

背景と問題点

オブジェクトあるいはプログラムをアクセシブルでないと、そのオブジェクトあるいはプログラムが提供している情報および機能を利用できない場合がある。また、特定のプラグインを用いる場合には、そのプラグインをサポートしていない環境では、やはりユーザーは情報および機能を利用することができない。

関連する要素、属性及び宣言

object, script, a

関連項目

技術解説 5.1 b) にあるように、オブジェクトあるいはプログラムを使用する際は、それ自体をアクセシブルにするよう最大限につとめなくてはならない。そして、プラグインについても同様である。しかし、どうしてもこれらをアクセシブルにできない場合には、利用者が同等の内容を理解してその情報およびサービスを利用できるように代替情報を提供する必要がある。

また、オブジェクト、プログラム、あるいはプラグインがアクセシブルであっても、あわせてテキストなどの代替情報を提供することで、利用者は好きなほうを選択することができるようになり、その利便性が向上する。

ソリューション 次の方法を選択して、あるいは組み合わせて用いる。

1) object 要素の代替テキスト

object 要素 には代替テキストを提供する。代替テキストは、オブジェクトのもつ情報と同等の内容を提供するか、もしくはそのオブジェクトを用いることができない場合の代替手段に関する情報を提供する。

例 1 `<object classid="java:Press.class" width="500" height="500">`

`<object data="Pressure.mpeg" type="video/mpeg">`

`<object data="Pressure.gif" type="image/gif">`

`As temperature increases, the molecules in the balloon...`

`</object>`

`</object>`

`</object>`

2) script 要素があるとき、noscript 要素を使って script が使えないときにも、アクセスできること

例 1 `<script type="text/javascript">` メニューのためのスクリプトが書いてある `</script>`

<noscript>メニューのページを開く</noscript>

3) プラグインの最新バージョンをダウンロードできるページへのリンクを提供する

プラグインを使用する際は、その最新バージョンをダウンロードできるページへのリンクを提供する。

例 1 Adobe Player (PDF 閲覧) の場合：

```
<a href=http://www.adobe.co.jp/products/acrobat/readstep2.html">

</a>
```

例 2 Flash Player の場合：

```
<a href=http://www.macromedia.com/jp/go/getflashplayer_jp/">

</a>
```

例 3 Windows Media Player の場合：

```
<a href="http://www.microsoft.com/japan/windows/windowsmedia/download/">

</a>
```

例 4 QuickTime の場合：

```
<a href=" http://www.apple.com/jp/quicktime/download/">
QuickTime のダウンロード
</a>
```

※keyword※ オブジェクト、スクリプト、PDF、Flash

5.5 色及び形

5.5 a 色のみに依存する情報提示

ウェブコンテンツの内容を理解・操作するのに必要な情報は、色だけに依存して提供してはならない。(対応：JIS X8341-3:5.5 a)

背景と問題点

ある情報を強調したり、他との差異を明確にしたりするために、文字色を変えるなどして色を用いることは視覚的により分かりやすく情報を伝えることができるという点ではとても有用である（例：必須項目のみ赤字で示す、新商品名のみ赤字で示す、など）。また、グラフなどでは複数の色を用いてデータを示す場合がある。

しかし、色を用いて情報を提供する場合、それらの色が識別できない場合でも情報が伝わるようにしなければならない。たとえば、音声読み上げソフトでは文字の色までは読み上げないため、利用者は文字の色だけで伝えられている情報が理解できない。また、色覚に特性のある利用者がその文字やグラフの色を識別できずに理解できないこともある。さらに、モノクロで印刷した場合には色は全て破棄されるため、特定の色で情報が表現されていると利用者はその色が識別できずに理解できない。

関連する要素、属性及び宣言

img, color, bgcolor

関連項目

技術解説 テキストの文字色で情報を示す場合は、あわせてテキストでも情報を提供する（例：「赤字の項目名（必須）」、「赤字の商品名（新商品）」、など）。そうすることで、音声読み上げソフトでも情報が伝わり、その色が識別できない場合でも情報が伝わるようになる。

また、グラフで複数の色を用いる際には、凡例だけではなく、引込み線を用いるなどして色が識別できなくても情報が伝わるようにする。

以下の手段を用いて、色が識別できなくても情報が伝わるかどうかを確認するとよい。

- ・グレースケールに変換してみる
- ・モノクロ印刷してみる
- ・音声ブラウザあるいはスクリーンリーダーで読み上げてみる

ソリューション 次の方法を選択して、あるいは組み合わせて用いる。

1) 文字の色だけでなくテキストの補足情報を提供する

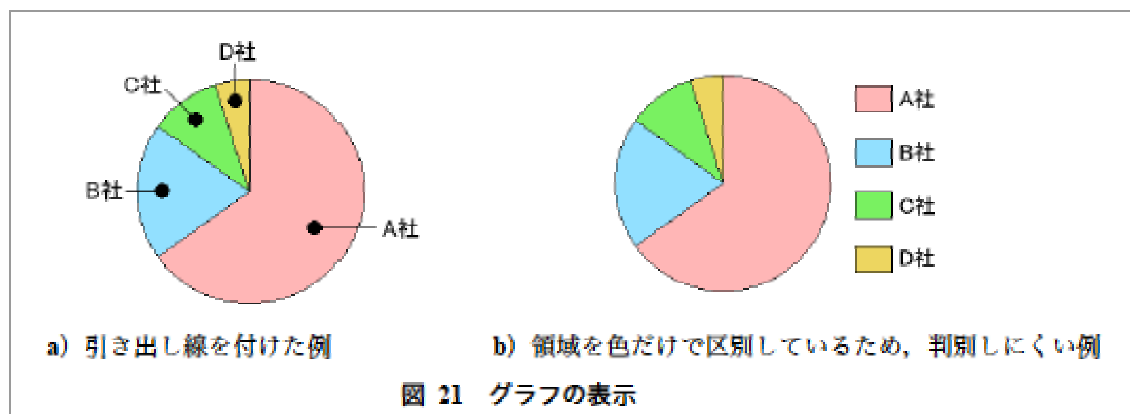
たとえば、文字色で情報を示す場合は、あわせてテキストでも情報を提供する。

例 1 `<p class="required">名前<em>（必須）</em></p>`

`p.required { color : red }`

2) グラフでは凡例だけでなく引込み線を用いて情報を提供する

例 1 JIS X 8341-3:2004 規格書 P13 より引用。



※keyword※ 色、文字色、グレースケール、モノクロ印刷、引込み線

5.5 色及び形 (JIS X 8341-3:5.5)

5.5 b 形・位置のみに依存する情報提示

ウェブコンテンツの内容を理解・操作するのに必要な情報は、形又は位置だけに依存して提供してはならない。(対応：JIS X8341-3:5.5 b)

背景と問題点

表組みなどで用いられる「○」「△」「×」といった記号文字は、視覚的により分かりやすく情報を伝えることができるという点ではとても有用である。

しかし、これらの記号文字は音声ブラウザやスクリーンリーダーでは意図したとおりに読み上げられないことがあるため、音声読み上げでは情報が伝わらないことがある。

また、「左のボタン」「右のボタン」というように、位置だけで示した情報も音声読み上げでは伝わらないため、利用者は画面上の位置が把握できず、操作することができない。

関連する要素、属性及び宣言

関連項目

技術解説 「○」「△」「×」などの記号文字で情報を提供する場合は、音声読み上げでもその情報が正しく読み上げられるように、テキストによる補足情報を提供する、あるいは記号文字を画像にしてその意味を代替テキストで提供する。

また、「左のボタン」「右のボタン」だけではなく、「左の送信ボタン」「右のキャンセルボタン」というように、位置が把握できなくてもどのボタンかが伝わるようにしなければならない。その際、ボタン自体にも「送信」「キャンセル」といったラベルをつけておく必要がある。

ソリューション 次の方法を選択して、あるいは組み合わせて用いる。

1) 「○」「△」「×」などの記号文字による情報にはテキストによる補足情報を提供する。

「○」「△」「×」「－」などの記号文字の直後に、テキストでその記号文字が伝えている意味を補足する。

例 1 「○（あり）」、「×（なし）」

「○（対応）」、「－（非対応）」

「○（開館日）」、「△（午前のみ開館）」、「×（休館日）」

2) 「○」「△」「×」などの記号文字を画像にして、代替テキストによる補足情報を提供する。

「○」「△」「×」「－」などの記号文字を画像にして、それぞれの代替テキストにその記号文字が伝えている意味を記述する。

例 1

3) 位置が把握できなくとも操作できるように補足情報を提供する。

ボタン自体に「送信」「キャンセル」といったそのボタンの機能を示すラベルをつけ、「左のボタン」「右のボタン」ではなく、「左の送信ボタン」「右のキャンセルボタン」あるいは「送信ボタン」「キャンセルボタン」と表現する。

※ keyword※ 形、位置、記号文字

5.5.c 背景色と前景色のコントラストと配色

画像などの背景色と前景色とには、十分なコントラストを取り、識別しやすい配色にすることが望ましい。

背景と問題点

文字の色とその背景の色、画像で表現された文字や記号、アイコンなどは背景色と前景色の間に十分なコントラストと配色における配慮がないと、見づらい、見えない利用者がいる。たとえば、弱視の人の場合、色のコントラストが十分でないと、色の差を識別できず表された情報を取得できないことがある。また、色覚障害のある人には、特定の色の識別が困難な場合がある。人間の色覚には、3色（赤、緑、青）の感覚受容体があり、それらの個々の色感覚が弱い場合と、まったく感覚がない場合がある。最も代表的なものは、第1色覚（赤）の障害である。

関連する要素、属性及び宣言

img object font

スタイルシートの色に関するプロパティすべて

関連項目

5.2.b 5.5.a

技術解説

この要件が求めているのは「コントラストに対する配慮」と「色の組み合わせに対する配慮」の2点である。コントラストへの配慮は、高齢者や弱視の人に必要なであると共に、色覚障害のある人でもコントラストが取れている色の組み合わせならば、識別可能になる。また、色の組み合わせに関する配慮は、一般には色覚障害の人のための配慮であるが、同時に色の効果的な使用によって識別性を高めることは一般の利用者のユーザビリティを高めるためにも有効である。具体的には、文字と背景色あるいは背景画像の関係、アイコンや図記号などと背景との関係、画像やオブジェクトなどの内部における表現したい文字や記号などと背景との関係に配慮が必要である。

（・WCAG 2.0 で提示されるかもしれない計算式も使える？）"

ソリューション 次の方法を選択して、あるいは組み合わせて用いる。

1) 文字の色を指定するときは、その文字が含まれる親要素の背景色とのコントラストを十分にとる。

コントラストを十分にとるということは、グレースケールで表示あるいは印刷しても識別可能になっているということを意味する。RGB で表現された色を YIQ 色空間に変換した Y 値（輝度）を基準としてその差を求めるとことで、ディスプレイに表示される色の組み合わせの近似的な輝度コントラスト比を求めることができる。RGB と YIQ の変換は、一般に次式で可能である。

$$R'G'B' \Rightarrow YIQ$$

$$Y = 0.299 \times R' + 0.587 \times G' + 0.114 \times B$$

$$I = 0.596 \times R' - 0.274 \times G' - 0.322 \times B$$

$$Q = 0.211 \times R' - 0.522 \times G' + 0.311 \times B$$

(この絵は書き直す必要あり)

W3C では、コントラストについて 125 以上と定めているが、変更されるかもしれない。

<http://www.w3.org/TR/AERT#color-contrast>

2) 文字の色を指定するときは、その文字が含まれる親要素の背景色との色差を十分にとる。

色差とは、次式で表される値で、色差を十分に確保することによって、より見やすい色の組み合わせが実現できる。

(式を書く)

W3C では、色差について 500 以上と定めているが、変更されるかもしれない。

<http://www.w3.org/TR/AERT#color-contrast>

3) 背景画像を使用しない。

html 要素やその他の要素の背景に画像を用いると、そこに表示される他の画像や文字の色とのコントラストや色差をすべて検討しなければならなくなるため、背景画像を使用しないほうがより見やすいページを容易に作成できる。また、利用者がユーザスタイルシートを提供したり、スタイルシートを利用せずに表示した場合、あるいは OS のハイコントラスト表示を利用した場合に、利用者の色の指定と背景画像の関係が似通っていると、コントラストを確保できない。したがって、背景画像は、使用しないほうがよい。

4) 画像のコントラストと色差を確保する。

文字やアイコン、その他の図記号などを含む画像では、1, 2 と同様にコントラストと色差に拝領する必要がある。

5) コントラストや色差の確保が難しいと思われる場合には、縁取りなどを利用する。

画像などで、コントラストや色差を確保できないときは、縁取りを利用することによって擬似的に背景色とのコントラストと色差を確保すると見やすくなる。

5.6 文字

5.6.a 文字サイズの変更

文字のサイズ及びフォントは、必要に応じ利用者が変更できるようにしなくてはならない。

背景と問題点

フォントに読みづらい小さなサイズのもの、読みづらい種類のフォントが用いられていると、弱視の人、加齢により視力の衰えた人には読みづらいことがある。そのような場合に、ウェブブラウザの表示文字の拡大、縮小機能を使って文字を大きく設定することがある。ウェブブラウザによる変更ができるように作成されていないと、この機能を利用することができないといった問題が発生してしまう。なお、この要件では制作者が積極的にフォントサイズや種類を指定することを推奨しているわけではない。あくまでも、指定する場合に配慮すべき事項について述べているだけである。

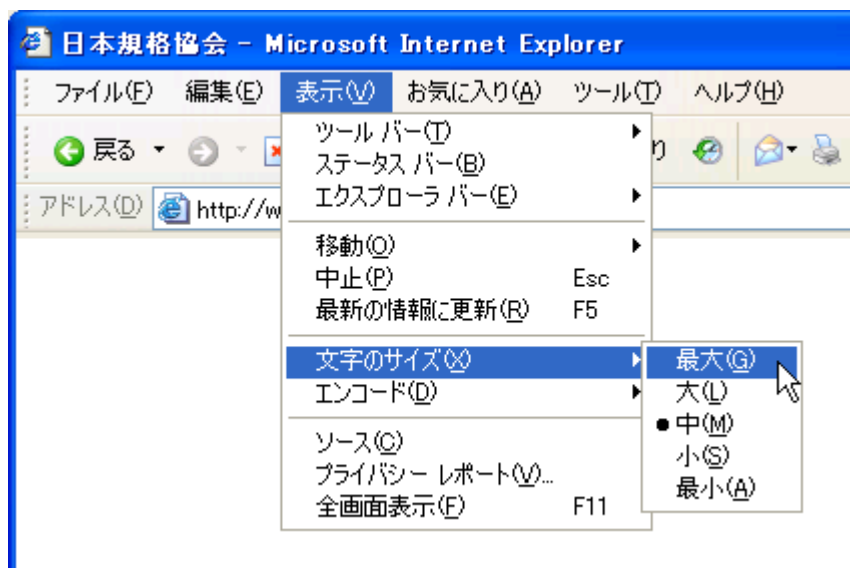


図 マイクロソフト社インターネットエクスプローラの文字サイズ変更メニュー

また、この文字の拡大機能を使っても一定程度しか文字を大きくできないため、それでも文字を読むことができない利用者は、拡大鏡を使ったり、画面を拡大する専用のソフトウェアを用いたりしている。画面を拡大するソフトウェアでは、画面を数倍から十数倍に拡大して表示することが可能であるが、画面をビットマップとして拡大するため、元の文字サイズが小さすぎると拡大した後も文字がつぶれて読みづらい。

また、デフォルトのスタイルシートを利用せず、利用者が自ら作成したユーザスタイルシートを用いて表示を見やすくすることも可能だが、スタイルシートの切り替えは操作が煩雑でウェブページの仕組みにある程度精通していないと使えないため、ユーザスタイルシートの使用を前提にしたデザインを行うべきではない。

関連する要素、属性及び宣言

関連項目

5.1.a 5.2.b

技術解説

この要件を実現するには、5.1 a) 仕様に関する準拠、5.2 b) 構造と表示スタイルの分離、にも注意を

払う必要がある。すなわち、HTML だけで文字のサイズやフォントを変更する場合と、CSS を用いる場合とで実現方法と、利用者に対する影響が異なる点に注意を払う必要がある。一般論として、フォントを指定する要素、あるいは CSS のプロパティでは相対値である “em”、“%”を用いる。

ソリューション 次の方法を選択して、あるいは組み合わせて用いる。

1) font 要素でサイズを指定したい場合には、相対値を使用する

font 要素は、非推奨の要素であり本来使用しないほうがよい。しかし、font 要素を使わざるを得ない場合には、絶対値指定せず、basefont からの相対値で指定するとよい。またこの場合には、basefont 要素で基準となるフォントサイズを指定しないほうがよい。なお、basefont の初期値は 3 であり、サイズ指定できるのは 1〜7 の範囲であることに注意する。

例) font 要素で文字を大きくする



文字のサイズを「中」にした状態



文字のサイズを「最大」にした状態

ソース

```

<ul>
<li><font size="-2">一番小さな文字(&lt;font size="-2"&gt; )</li>
<li><font size="-1">少し小さな文字(&lt;font size="-1"&gt; )</li>
<li>通常の文字</li>
<li><font size="+1">少し大きな文字(&lt;font size="+1"&gt; )</li>
<li><font size="+2">大きな文字(&lt;font size="+2"&gt; )</li>
<li><font size="+3">もっと大きな文字(&lt;font size="+3"&gt; )</li>
<li><font size="+4">いちばん大きな文字(&lt;font size="+4"&gt; )</li>
</ul>

```

2) スタイルシートを使用してサイズを指定する場合には、相対値を使用する

スタイルシートを用いて文字のサイズを指定する場合には、font-size または font プロパティで相対値をもちいて指定する。絶対値で指定する場合には、文字サイズを指定する次のキーワードを用いる。

xx-small、x-small、small、medium large、x-large、xx-large

これらのキーワードは CSS の絶対指定として位置づけられているが、ウェブブラウザで文字を拡大縮小することが可能である。medium を基準として、一般的には CSS1 では順に約 1.5 倍で変化し、CSS2 では、約 1.2 倍で変化するが、この値はウェブブラウザによって異なる。ただし、xx-small、x-small は非常に小さいフォントで表示されることがあるので、使用する際には注意が必要である。また、絶対値指定では、12pt、20px などの形式で数値＋単位での記述も許されているが、この方式でサイズを指定すると、利用者はフォントサイズを変更できない。

一方、相対値指定では、以下の方法で基準となるフォントから大きさを指定することが可能である。

・パーセントで指定する

例) font-size: 150%;

親要素のフォントサイズを基準として、150%（高さ、及び幅を 1.5 倍）にする。

・em で指定する

例) font-size: 2em;

親要素のフォントサイズを基準として、2 倍（面積は 4 倍）にする。

なお、pt(ポイント)、mm（ミリメートル）、cm（センチメートル）、pc（パイカ）、in（インチ）、px（ピクセル）は絶対指定となるため使用しない。また、ex（文字”x”を基準としたサイズ指定）もウェブブラウザによる対応が一律でないため使わないほうがよい。

・larger、smaller で指定する

例) font-size: larger;

font-size: smaller;

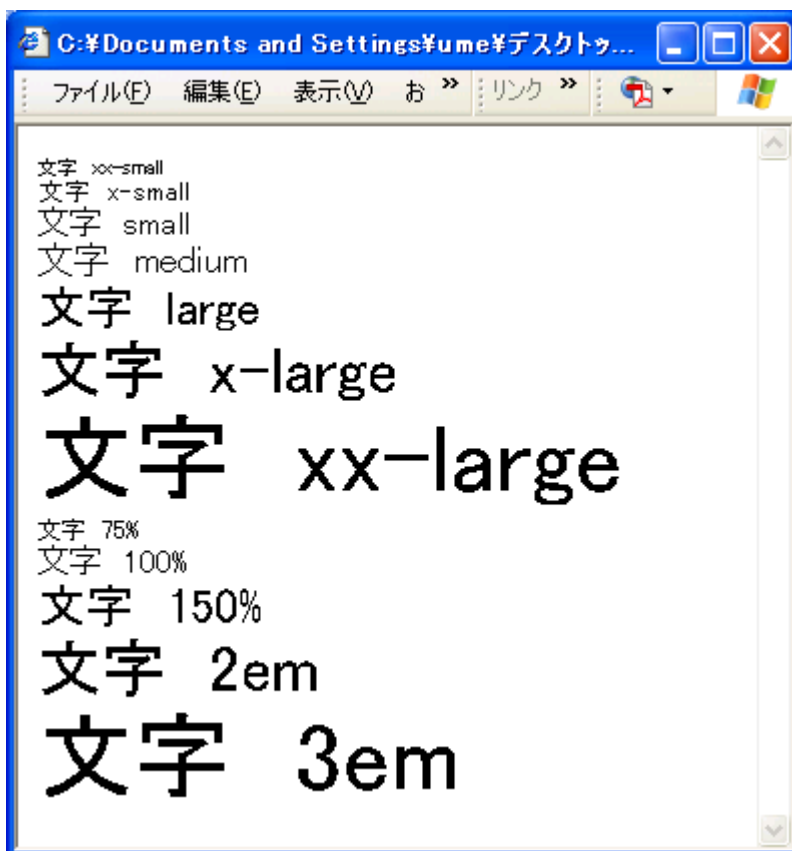
親要素が medium になっているときには、larger を指定すると large に、smaller を指定すると small を指定したのと同じになる。

以下の表が、この要件を満たすために用いることができるプロパティの記述法である。

	プロパティ値	使用する際の注意点
用いてもよいもの	x-small、small、medium、large、x-large、xx-large	xx-small は文字が小さくなりすぎ利可能性があるので避けたほうがよい。
	smaller、larger	smaller を用いる場合には、フォントが小さくなり過ぎないように注意する。
	“%”、“em”による指定	指定した文字が小さくなり過ぎないように注意する。
用いないほうがよいもの	“pt”、“mm”、“cm”、“pc”、“in”、“px” による指定	

表 プロパティ値の指定方法

例 フォントサイズの指定例



ソース

```
<p>  
<span style="font-size:xx-small">文字 xx-small</span><br>  
<span style="font-size:x-small">文字 x-small</span><br>  
<span style="font-size:small">文字 small</span><br>  
<span style="font-size:medium">文字 medium</span><br>  
<span style="font-size:large">文字 large</span><br>  
<span style="font-size:x-large">文字 x-large</span><br>  
<span style="font-size:xx-large">文字 xx-large</span><br>  
<span style="font-size:75%">文字 75%</span><br>  
<span style="font-size:100%">文字 100%</span><br>  
<span style="font-size:150%">文字 150%</span><br>  
<span style="font-size:2em">文字 2em</span><br>  
<span style="font-size:3em">文字 3em</span><br>  
</p>
```

3) フォントの種類は、スタイルシートで指定する

フォントの種類を指定する場合には、スタイルシートで指定し、font 要素で指定しない。font 要素で指定すると、利用者はユーザスタイルシートを用いてフォントを変えることができなくなる。なお、フォントの種類を指定する場合には、5.6 b) 読みやすいフォントの指定の要件も考慮する必要がある点に注意すること。

例 ゴシック系のフォントを指定する

```
font-family: "MS P ゴシック",Osaka,sans-serif;
```

4) 文字を画像化しない

文字を画像化すると、文字サイズを利用者が変更できなくなる。文字を画像化する代わりに、スタイルシートを使って表示したほうがよい。なお、ロゴマークなどその形に意味があるものに関しては画像化することはやむをえないが、その場合にも見易さに配慮する必要がある。

注意事項 フォントのサイズを絶対値で指定していても表示サイズを変更可能なウェブブラウザや、表示最小フォントサイズを指定できるブラウザもある。今後、絶対値でフォントサイズが指定されていても利用者が自由なサイズで表示できる機能が一般になってくると、この要件はそのような機能を有しないウェブブラウザの互換性のためだけのものになると思われる。

5.6 文字

5.6.b フォントの読みやすさ

フォントを指定するとき、サイズ及び書体を考慮し読みやすいフォントを指定することが望ましい。

背景と問題点

5.6 a) の要件により利用者がフォントを変更できるようになっていても、すべての利用者がその変更操作できるとは限らない。たとえば、ウェブブラウザの操作が不慣れな高齢者は、画面がよく見えない上に、文字を大きくする操作もできないということが考えられる。したがって、最初から読みやすいフォントのサイズと種類が使われているほうがよい。しかし、この要件はフォントを指定するときに配慮すべき事項を定めているだけなので、積極的にフォントを指定することを求めているわけではない点に注意が必要である。

関連する要素、属性及び宣言

関連項目

ソリューション 次の方法を選択して、あるいは組み合わせて用いる。

1) フォントサイズを小さくしすぎない

スタイルシートの font-size、font プロパティで xx-small、x-small を用いない。また、0.8em あるいは 80% 以下のサイズを指定しない。

2) フォントを指定するときは、できるだけゴシック系のフォントを選択する。

ゴシック系の日本語フォントには、マイクロソフトウィンドウズに搭載された「MS ゴシック」「MS Pゴシック」、マッキントッシュに搭載された「Osaka」などがある。欧文では、「sans-serif」がよく用いられる。

3) フォントスタイル等を制御する場合には、見易さに配慮する。

スタイルシートの font-style、font-stretch、font-size-adjust、font-weight、line-height、text-decoration、letter-spacing、word-spacing、text-shadow プロパティなどを用いてフォントのスタイルを調整したり、文字間や行間を制御したりする場合には、見易さへの配慮が必要である。

注意事項 フォントの大きさ、種類、書体に関する読みやすさは、色や利用者の使用しているコンピュータのモニタや照明環境などに左右されるため、明確な数値で評価することは難しい。しかし、一般論として小さな文字は読みづらく、太い文字、大きなもの字は読みやすい。また、明朝体よりもゴシック体のほうが読みやすいことが知られている。したがって、この要件を満たすための最もよい方法は、フォントを指定しないことである。

5.6 文字

5.6.c フォント色

フォントの色には、背景色などを考慮し見やすい色を指定することが望ましい。

背景と問題点

フォントの色と背景色の組み合わせによっては、文字が読みづらくなることがある。たとえば、ある種の色覚障害のある人には、赤と緑の区別が難しい。また、一般的に白内障や加齢による視力の低下のある人、弱視の人は、コントラストの低い文字は読みづらい。

関連する要素、属性及び宣言

関連項目

技術解説

これらの問題に対処するには、色の組み合わせについての配慮が必要である。文字色と背景色の間に十分なコントラストをとることで、見やすくすることができる。なお、この問題に対応する最もよい方法は、フォントの色を指定せず、背景色も指定しないことである。当然ながら、画像化したテキストがない場合、フォントの色、背景色を指定していない場合には、この要件を配慮する必要がない。

（ここに色の識別と見易さに関する一般的な説明を加えるか、appendix を用意してそれを参照する「新編 感覚・知覚心理学ハンドブック」が参考になりそう）

ソリューション 次の方法を選択して、あるいは組み合わせて用いる。

- 1) **背景色を指定する場合には、フォントの色との組み合わせ、コントラストを評価する**
- 2) **スタイルシートを外した状態でも色の組み合わせ、コントラストに問題がないかどうかを評価する。**
- 3) **文字を画像化するときは、文字と背景の色の組み合わせ、コントラストを評価する**
- 4) **背景画像を使用しない**
- 5) **OS の機能や支援技術の提供する「ハイコントラスト表示」を用いた場合にも、色の組み合わせ、コントラストを評価する**

注意事項

5.7 音

5.7.a 音の自動再生の禁止

自動的に音を再生しないことが望ましい。自動的に再生する場合には、再生していることを明示しなければならない。(JIS X8341-3: 5.7 a)

背景と問題点

ページを開いたときに※自動的※に※音※（音楽等）が流れると、意図しない出来事にユーザーが驚いてしまうことがある。たとえば、子供の寝ている横や、深夜の家族が寝ている横でページを参照しないといけないかもしれないし、あるいは静かな図書館などでページを利用しなければいけないことも考えられる。このような、静かな場所や静かにしないといけない場所で Web 閲覧をしている場合に、ページを開いたら、自動的に音が鳴り出したら困るだろう。そのような場合は、慌てて音量を下げたりする。しかし、聴覚障害の利用者の場合、音が出ていることに気づかず、音が再生され続けてしまい、寝ている子供を起こしてしまうといったことが起こってしまうかもしれない。音声ブラウザを利用している場合、読み上げの音声（音楽など）と重なって聞き取りづらくなる可能性がある。音量を下げようとすれば、本文の読み上げの音量も下がってしまい、そのページの閲覧が事実上できなくなってしまう。

関連する要素、属性及び宣言

bgsound, object, embed

関連項目

5.4 c 音声の代替

技術解説

ページを開いたときに自動的に音声を再生するような設定は基本的にしないことが大切である。自動再生ではなく、再生ボタンを押すなど、ユーザーの意思で再生の開始を行うようにすべきである。どうしても、ページを開いたときに自動再生する必要がある場合は、そのページのわかりやすいところに、現在音が再生されていることを表示しなければならない。仮に、自動再生されているページを開く1つ前のページに、自動再生されるページであることを明示されていたとしても、他の場所から直接その自動再生されるページにリンクされていたり、ブックマークに登録されていたりする場合のことも考慮し、現在音が再生されていることを表示しなければならない。

ソリューション

次の方法を選択して、あるいは組み合わせて用いる。

1) 音の自動再生がどうしても必要か再度検討する。

音を自動再生が許容されるか否かは、利用者の特性やページの利用する時間帯、利用する場所によって異なると考えられる。音声出力の利用できない利用者もいるかもしれない、あるいは、深夜に子供の寝ているそばでページを参照しないといけないかもしれない。あるいは静かな図書館などでページを利用しなければいけないことも考えられる。そのようなことも考慮に入れて、それでもなお、音の自動再生が必要なのかどうか、再度検討する。

2) ページの先頭部分やタイトルの中に、音が自動再生されていることをわかりやすく表示する。

音を自動再生する場合は、そのページのわかりやすいところ、ページ先頭やタイトル部分に、現在音が再生されていることを表示する。または、音楽を再生するプラグインのコントローラーを表示することで、音の再生を伝えることも可能と思われる。

タイトルや本文中にテキストで表示する場合：

「このページは音楽が自動的に再生されます」

「音楽自動再生中」

画像やアイコンで示す場合：

``

``

3) bgsound 要素は利用せず object 要素および embed 要素を利用する。

bgsound は、Internet Explorer の独自の拡張属性のため、Internet Explorer 以外のブラウザでは音が再生できない。さらに bgsound は利用者が音の制御をできなくする恐れがあるため、音を再生する場合は object 要素を利用する。ただし object 要素の未対応のブラウザ、あるいは不完全なブラウザが多いことに配慮し object 要素と同時に embed 要素も記述するほうがよい。

例 1) Windows Media Player で再生する例

```
<object
classid="CLSID:22D6f312-B0F6-11D0-94AB-0080C74C7E95" >
<param name="filename" value="sound.mid">
<embed src="sound.mid">
<noembed>
Windows Media Player では音声再生できませんでした。<br>
下のリンクで音声ファイルを直接開くことができます。<br>
<a href="sound.mid">音声ファイル (sound.mid)</a><br>
</noembed>
</object>
```

例 2) Real Player で再生する例

```
<object
classid="clsid:CFCDAA03-8BE4-11cf-B84B-0020AFBBCCFA"
```

```

height=300 width=300>
<param name="src" value="sound.rm">
<embed src="sound.rm" height=300 width=300>
<noembed>
Real Player では音声再生できませんでした。<br>
下のリンクで音声ファイルを直接開くことができます。<br>
<a href="sound.mid">音声ファイル (sound.mid)</a><br>
</noembed>
</object>

```

4) 自動再生をしない場合は、自動再生しないように明示的に記述する。

環境によっては自動的に再生が開始されてしまう場合があるので、明示的に自動再生をしないように設定する。

例 1) Windows Media Player の場合

```

<object
classid="CLSID:22D6f312-B0F6-11D0-94AB-0080C74C7E95" >
<param name="autoStart" value="false">
<param name="filename" value="sound.mid">
<param name="loop" value="false">
<embed src="sound.mid" autostart="false" hidden="false" loop="false">
<noembed>
Windows Media Player では音声再生できませんでした。<br>
下のリンクで音声ファイルを直接開くことができます。<br>
<a href="sound.mid">音声ファイル (sound.mid)</a><br>
</noembed>
</object>

```

例 2) Real Player の場合

```

<object
classid="clsid:CFCDA03-8BE4-11cf-B84B-0020AFBBCCFA"
height=300 width=300>
<param name="src" value="sound.rm">
<embed src="sound.rm" height=300 width=300>
<noembed>
Real Player では音声再生できませんでした。<br>
下のリンクで音声ファイルを直接開くことができます。<br>
<a href="sound.mid">音声ファイル (sound.mid)</a><br>
</noembed>

```

</object>

<object

classid="clsid:CFCDAA03-8BE4-11cf-B84B-0020AFBBCCFA"

height=300 width=300>

<param name="src" value="sound.rm">

< param name="autostart" value="flase">

< param name="loop" value="false">

<embed src="sound.rm" autostart="false" loop="false" height="300" width=300>

<noembed>

Real Player では音声再生できませんでした。

下のリンクで音声ファイルを直接開くことができます。

音声ファイル (sound.rm)

</noembed>

</object>

5.7.b 利用者による音量調整

音は、利用者が出力を制御できることが望ましい。(JIS X8341-3: 5.7 b)

背景と問題点

音声を再生する場合、利用者のいる場所によって、※音量※を※制御※したい場合がある。喫茶店では音は小さめにしておきたいとか、聞こえづらい場合は大きくしたいとか、会議中なので消したい・・・など。そのような場合に、再生、停止（可能なら一時停止）、音量調整をユーザーが、操作できるようにしたいと思われる。

関連する要素、属性及び宣言

bgsound, object, embed,

関連項目

5.1 b アクセシブルなオブジェクトの使用

5.4 c 音声の代替

5.4 e オブジェクトの代替形式

技術解説

音を鳴らす場合、object 要素または embed 要素を利用して記述する。そのとき、再生の停止や一時停止、音量調節などためのコントロールパネルが表示できるプラグインを選択するべきである。また、意図的にコントロールパネル隠すようなことをしないことが重要である。

ソリューション

次の方法を選択して、あるいは組み合わせて用いる。

1) bgsound 要素は利用せず object 要素および embed 要素を利用する。

bgsound は、Internet Explorer の独自の拡張属性のため、Internet Explorer 以外のブラウザでは音が再生できない。さらに bgsound は利用者が音の制御をできなくする恐れがあるため、音を再生する場合は object 要素を利用する。ただし object 要素の未対応のブラウザ、あるいは、不完全なブラウザが多いことに配慮し object 要素と同時に embed 要素も記述するほうがよい。

2) プラグインを指定する場合は、ユーザーが Web ページで出力制御が可能な機能のあるプラグインを選択する。

再生の停止や一時停止、音量調節などのためのコントロールパネルが表示できるプラグインを選択する。

3) コントローラーなどは、非表示設定はせずに表示するようにする。非表示にする場合は最低限、音を停止または再生する機能を提供する。

例 1) Windows Media Player の場合

```

<object classid="CLSID:22D6f312-B0F6-11D0-94AB-0080C74C7E95" >
<param name="autoStart" value="false">
<param name="filename" value="sound.mid">
<param name="loop" value="false">
<param name=" " value="false">
<param name="hidden" value="false">
<embed src="sound.mid" autostart="false" hidden="false" loop="false">
<noembed>
Windows Media Player での音声再生できませんでした。<br>
下のリンクで音声ファイルを直接開くことができます。<br>
<a href="sound.mid">音声ファイル (sound.mid)</a><br>
</noembed>
</object>

```



例 2) Windows Media Player と JavaScript を利用して再生と停止リンクを作成する場合

```

<script type="text/javascript">
function Play(){
    MediaPlayer.play();
}
function Stop(){
    MediaPlayer.stop();
}
</script>
<object id="MediaPlayer" classid="CLSID:22D6f312-B0F6-11D0-94AB-0080C74C7E95" >
<param name="autoStart" value="true">
<param name="filename" value="sound.mid">
<param name="loop" value="false">
<param name="hidden" value="false">
<embed src="kid1.mid" autostart="false" hidden="true" loop="false">
<noembed>
Windows Media Player での音声再生できませんでした。<br>
下のリンクで音声ファイルを直接開くことができます。<br>
<a href="sound.mid">音声ファイル (sound.mid)</a><br>
</noembed>
</object>

<a href="javascript:Play()">音楽再生</A><br>

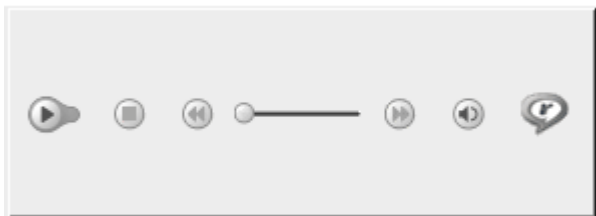
```

音楽停止



例 3) Real Player の場合

```
<object  
  classid="clsid:CFCDA03-8BE4-11cf-B84B-0020AFBBCCFA"  
  height=300 width=300>  
< param name ="src" value="sound rm">  
< param name ="autostart" value="flase">  
< param name ="loop" value="false">  
<embed src="sound.rm" autostart="false" loop="false" HEIGHT="300" WIDTH=300>  
< param name ="controls" value="all">  
</noembed>  
Real Player での音声再生できませんでした。<br>  
下のリンクで音声ファイルを直接開くことができます。<br>  
<a href="sound.mid">音声ファイル (sound.rm)</a><br>  
</noembed>  
</object>
```



5.8 速度

5.8 a 変化する画像・テキスト

変化又は移動する画像又はテキストは、その速度、色彩・輝度の変化などに注意して作成することが望ましい。 (対応：JIS X8341-3:5.8 a)

背景と問題点

画像あるいはテキストが変化したり移動したりすると、認知あるいは記憶に障害がある場合や高齢者などの利用者が認識できない可能性がある。たとえば、背景色および文字色が変わる際、背景と文字との輝度の変化が大きいほど目への負担が大きくなり、また点滅速度が早いと内容を認識できないことがある。

また、永続的に繰り返される変化、点滅、移動といった動きは、利用者のコンテンツへの集中力を奪うことにもなる。

関連する要素、属性及び宣言

img, object

関連項目

技術解説 バナー広告などのアニメーション画像では、背景の変化はなるべく緩やかにして、極端な色の変化を避ける。具体的には、複数の画像を順番に繰り返し表示する場合、画面ごとの明度、彩度および色相が大きく変わる画面を交互に点滅させ繰り返し表示してはいけない。文字を変化または移動させる場合は、利用者が認識しやすいようにそのコントラスト輝度の変化および速度を緩やかにする。また、点滅させる際は、認識しやすいようにその速度を遅くするなどの配慮も必要である。

また、スクロールや点滅といった動きのある表現は、利用者のコンテンツへの集中力をそぐため、できるかぎり避けるべきである。どうしてもこういった表現を用いる際には、一定時間経過後には静止させるなどの配慮が望まれる。

なお、文字をスクロールさせる **marquee** 要素、文字を点滅させる **blink** 要素は、いずれも W3C の勧告（仕様）にはない規格外要素であり、これらを用いるべきではない。

ソリューション 次の方法を選択して、あるいは組み合わせて用いる。

1) 複数の画像を順番に繰り返し表示する場合、画面ごとの明度、彩度および色相の変化を小さくする。各画像の背景色および文字色の明度、彩度および色相の変化をできるかぎり小さくする。

例 1

2) 文字を変化または移動させる場合は、そのコントラスト輝度の変化および速度を緩やかにする。

文字のコントラスト輝度の変化および移動の速度をできるかぎり緩やかにする。

例 1

3) スクロールあるいは点滅させる場合は、その速度を遅くして、一定時間が経過したら静止させる。

文字がある場合は、利用者がその文字が読めるように点滅する間隔を十分に空ける。また、利用者がコンテンツに集中できるように、スクロールあるいは点滅が数回繰り返したら停止させる。

例 1 たとえば、3 回同じ動きを繰り返したら停止させる。

※keyword※ 変化、移動、点滅、明度、彩度、色相、コントラスト輝度、速度

5.8 速度 (JIS X 8341-3:5.8)

5.8 b 早い周期の点滅禁止

早い周期での画面の点滅を避けなければならない。(対応：JIS X8341-3:5.8 b)

背景と問題点

光の明滅によって光感受性発作（光源性てんかん）を誘発することがある。20 Hz の時間周波数（1秒間に20回）にピークがあり、特に赤と青とを交互に点滅させると、光感受性発作を誘発しやすい。利用者の安全性に関することなので最大限の配慮が必要である。

関連する要素、属性及び宣言

img object

関連項目

技術解説 画面全体を明滅させると、光感受性発作を誘発することがあるので使用しない。どうしても明滅させたい場合は、画面全体を1秒間に2回以上明滅させないようにして、特に、彩度の高い赤の明滅およびコントラストの強い画面の反転を避けなければならない。

⇒まず、点滅をやめることを書いたほうがいいのではないか？

また、画面内で明滅する部分の領域は、できるだけ小さい範囲に留めるよう配慮するのがよい。しかし、たとえ画面の一部だとしても、中には弱視の利用者のように画面を拡大していることもあるので、できるかぎり画面の明滅は避けるべきである。

また、画面全体を占めるような、しま、渦巻き、同心円といった規則的なパターン模様も画面の明滅の場合と同様の危険があるため、そういったページデザインも避けるべきである。

なお、W3C では、この明滅に関する具体的な基準値に関して、「ITC Guidance Note for Licensees on Flashing Images and Regular Patterns in Television

(http://www.ofcom.org.uk/codes_guidelines/broadcasting/tv/vrs_code_notes/flsh_imgs/?a=87101)」

で定められている基準を WCAG 2.0 で採用することを検討している。

ソリューション 次の方法を選択して、あるいは組み合わせて用いる。

1) 画面全体を1秒間に2回以上明滅させない。特に彩度の高い赤の明滅およびコントラストの強い画面の反転は避ける。

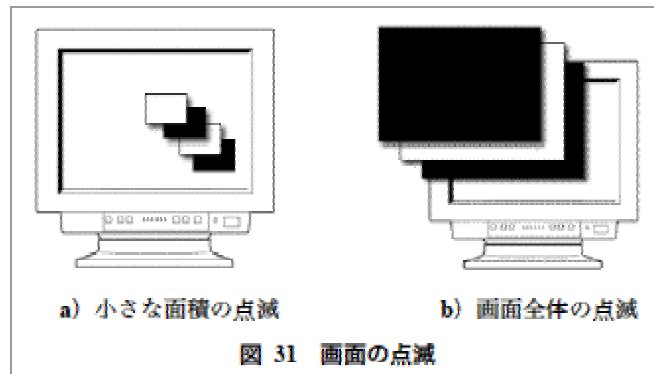
W3C の WCAG 1.0 では、チェックポイント 7.1 で「1秒間に20回の点滅をピークとして、1秒間に4～59回(Hz)の点滅」、米国のリハビリテーション法 508 条では、「2～55 ヘルツの周波数」（1秒間に2～55回）の点滅、を避けるようにとそれぞれ定めている。

例 1

- 2) 画面の一部を明滅させるときは、1024×768 ピクセルの画面解像度で 355×268 ピクセルの 1/4 以上を占める面積での明滅は避ける。

W3C の WCAG 2.0 で検討されている面積の基準です。ただし、弱視の利用者のように画面を拡大している場合がありますので、これ以上小さい面積でも上記 a) とあわせて配慮すべきです。

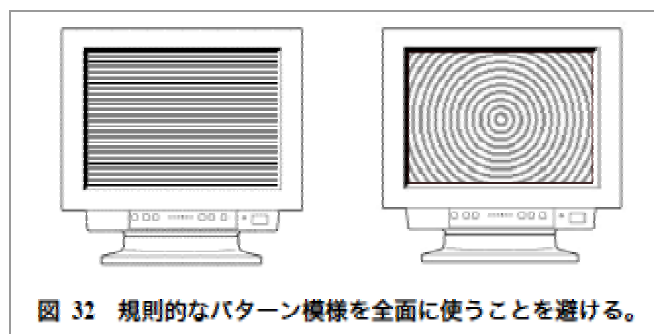
例 1 JIS X 8341-3:2004 規格書 P17 より引用。



- 3) 画面全体を占めるような、しま、渦巻き、同心円といった規則的なパターン模様を使用しない。

画面の明滅と同様の危険性があるため、このような表現は避ける。

例 1 JIS X 8341-3:2004 規格書 P17 より引用。



※keyword※ 明滅、光感受性発作、光源性てんかん、発作、てんかん、彩度、周波数、しま、渦巻き、同心円

5.9 言語

5.9.a 自然言語の指定

言語が指定できるときは、自然言語に対応した言語コードを記述しなければならない。

(対応：JIS X8341-3:5.9 a)

背景と問題点

自然言語の指定は、音声ブラウザが正しい言語辞書を用いて音声合成するために必要である。現在日本国内で使われている主な音声ブラウザやスクリーンリーダーは日本語にしか対応しないものが主流であるが、複数の言語を正しく合成できる環境が整備された際には、言語コードの指定によって複数の言語を用いて作成されたページを正しく読み上げることができるようになる。また、一般のウェブブラウザでも言語コードの指定が正しくされていないと、適切なフォントを用いて正しく表示できないことがある点に注意する。

関連する要素、属性及び宣言

html span

技術解説

言語コードは、各仕様にしたがって指定する。ブラウザが表示する際に使用する言語は、次の順序で解釈され表示される。

- 1) 要素の lang 属性で指定された言語コード
- 2) その要素の親になる要素において、lang 属性で指定された言語コード
- 3) HTTP の Content-Language ヘッダで指定された言語コード
- 4) ウェブブラウザのデフォルトの言語コード

したがって、HTML では html 要素で言語コードを指定しておくのが最もよい方法である。

ソリューション

- 1) HTML では、html 要素には lang 属性でデフォルトの言語を指定する。

例 1 <html lang="ja">

- 2) XHTML1.0 では、lang 属性に加えて xml:lang 属性を用いてデフォルトの言語を指定する。

例 1 <html lang="ja" xml:lang="ja">

- 3) XHTML1.1 では、xml:lang 属性を用いてデフォルトの言語を指定する。

例 1 <html xml:lang="ja">

- 4) HTTP の Content-Language ヘッダにおける言語指定がコンテンツのデフォルト言語と一致すること。
このヘッダは、ウェブサーバーの設定により指定する。

- 5) HTML 文書の途中で言語を切り替えたいときには、span 要素などで lang 属性を使って指定する。

例 1 English section of the document

注意事項

なし

5.9.b 外国語の多用禁止

日本語のページでは、想定する利用者にとって理解しづらいと考えられる外国語は、多用しないことが望ましい。使用するときは、初めて記載する時に、解説しなければならない。

(対応：JIS X8341-3:5.9 b)

背景と問題点

高齢者や認知障害のある人の中には、外国語を理解できない人や、外国語が出てくることで理解が困難になる人がいる。

関連する要素、属性及び宣言

なし

技術解説

日本語のコンテンツの場合で、外国語を用いなくても情報が伝えられる場合には、日本語で記述すれば必要な情報を伝えることができるようになる。また、外国語を使わざるを得ない場合でも、その解説を提供することで内容の理解を促進することができる。規格本文の「想定する利用者にとって理解しづらいと考えられる外国語」のくだりは、あらゆるコンテンツで外国語の禁止を求めているわけではないことを意図している。たとえば、学術研究に関する文書や外国語教育に関するもの、またあらかじめ外国語を十分に理解できると思われる利用者を対象として想定したコンテンツでは、外国語の使用を妨げない。

ソリューション

- 1) 想定する利用者が幅広くあらゆる利用者を対象にしたいときは、外国語の使用を最低限にし、ページの最初に出現した箇所でその単語の意味あるいは説明を提供する。文章で用いる場合には、その日本語訳を併記するか、日本語訳へのリンクを提供する。

例 1 `<span lang="en">foreign language </span>`(外国語)

- 2) 利用者が一定の外国語に関する知識を持っていると想定される場合は、そのレベルに合わせて理解の困難な言葉に限って説明を提供する。

注意事項

5.9 c 省略語、専門用語、流行語、俗語の多用禁止

省略語、専門用語、流行語、俗語などの想定する利用者にとって理解しにくいと考えられる用語は、多用しないことが望ましい。使用するときは、初めて記載されるときに定義しなければならない。

(対応：JIS X8341-3:5.9 c)

背景と問題点

省略語や専門用語などはそれを理解できない人がいるだけでなく、音声ブラウザなどでも正しく読み上げることができず理解できないことがある。

関連する要素、属性及び宣言

abbr acronym th td

技術解説

省略語、専門用語、流行語、俗語などを多用すると、利用者の記憶に負担をかけ文書を理解しづらくなる。

また、これらの言葉はスクリーンリーダーで正しく読み上げられない可能性がある。

規格本文の「想定する利用者にとって理解しづらいと考えられる用語」のくだりは、あらゆるコンテンツでこれらの用語の使用を禁止しているわけではないことを意図している。たとえば、学術研究に関する文書や、あらかじめその用語を十分に理解できると思われる利用者を対象として想定したコンテンツでは、それらの用語の使用を妨げない。

ソリューション

- 1) **省略語が最初に出現した箇所で、abbr 要素または acronym 要素によって正式な名称等を明示する。**

例 1 `<abbr title="Japanese Standards Association、日本規格協会">JSA</abbr>`のサービスを利用する

(@@@読み上げ要チェック@@@)

(@@@XHTML も OK か要チェック@@@)

- 2) **table のセルで省略語を定義する場合には、abbr 属性で指定する。**

abbr 属性を指定できる要素は、th、td のみである点に注意する。

例 1 `<th abbr=" JSA#">日本規格協会</th>`

(@@@読み上げ要チェック@@@)

(@@@XHTML も OK か要チェック@@@)

- 3) **想定する利用者にとって専門用語、流行語、俗語がわかりにくいとおもわれる場合には、少なくとも初めて出現した箇所でその説明を提供する。**

例 1 `<p><a name="return_netjunky" href="#netjunky">ネットジャンキー</a>だ。</p>`

：

：

`<a name=" netjunky">ネットジャンキーは、インターネットに熱中している人を指す言葉です<a>`

`<a href=" #return_netjunky">本文に戻る</a>`

(※梅垣注：HTML1.1 では name 属性ではなく id 属性なので、そのことを追記する)

- 4) **読みが難しいと思われる場合には、5.4 のソリューションを併用する。**

注意事項

なし

5.9 d 読みの難しいと考えられる言葉の多用禁止

想定する利用者にとって、読みの難しいと考えられる言葉（固有名詞など）は、多用しないことが望ましい。使用するときは、初めて記載されるときに読みを明示しなければならない。"

(対応：JIS X8341-3:5.9 d)

背景と問題点

読みの難しい言葉とは、姓名、社名、地名などの固有名詞、難読語などがある。これらの言葉を用いていると、読み方がわからず混乱することがある。また、スクリーンリーダーで正しく読み上げられず意味が伝わらないことが考えられる。

関連する要素、属性及び宣言

ruby

技術解説

読み方が難しい言葉に読み方に関する読みを提供すると、文書が読みやすくなり、スクリーンリーダーでの読み上げも改善される可能性がある。

ソリューション

1) XHTML1.2では、ruby を用いて読みを示す

Ruby Annotation W3C Recommendation 31 May 2001 <http://www.w3.org/TR/ruby/>を参照する。

例 1 (改変するひつようあり)

```
<ruby xml:lang="ja">
  <rb>
    <rb>斎</rb>
    <rb>藤</rb>
    <rb>信</rb>
    <rb>男</rb>
  </rb>
  <rtc class="reading">
    <rt>さい</rt>
    <rt>と う</rt>
    <rt>のぶ</rt>
    <rt>お</rt>
  </rtc>
  <rtc class="annotation">
    <rt rspan="4" xml:lang="en">W3C Associate Chairman</rt>
  </rtc>
</ruby>
```

ただし、ruby は HTML では利用できない。また、主な日本の音声ブラウザでは ruby 要素を正しく扱うことができず、2 度読み上げてしまう等の問題があるため、今後の音声ブラウザの対応状況を見極めてこのソリューションを使うべきである。

2) HTML では、カッコ書きなどで読みを示す。

例 1 武者小路実篤（むしゃのこうじさねあつ）

注意事項

5.9 e 単語途中のホワイトスペースの禁止

表現のために単語の途中にスペース又は改行を入れてはならない。

(対応：JIS X8341-3:5.9 e)

背景と問題点

単語の途中にスペースや改行があると、スクリーンリーダーはそれを一つの単語として認識できず正しく音声合成することができない。

関連する要素、属性及び宣言

br

技術解説

日本語を一時的に縦書き表記するために、単語の 1 文字づつを br 要素で強制的に改行したり、文字間隔を調整するためにスペース（全角スペースを含む）を挿入すると、スクリーンリーダーは正しくその単語を音声合成できず、利用者は理解できないことがある。

ソリューション

1) **縦書きが必要な場合は、スタイルシートで制御する。**

ただし、現在の多くのブラウザでは縦書き表示を正しく表示できないことがあるので、正しく表示できるまでは 2 のソリューションをもちいる。

2) **縦書きが必要な箇所では、文字を画像化し、alt 属性を付加する。**

注意事項

5.9.f 図記号とイラストレーション

ウェブコンテンツは、文章だけでなく、分かりやすい図記号、イラストレーション、音声などを合わせて用いることが望ましい。

(対応：JIS X8341-3:5.9 f)

背景と問題点

知的な障害がある人がウェブページを使う場合だけでなく、一般の利用者が使う場合にも図記号やアイコン、イラストなどが適切だと使いやすく、認知的負担を軽減することができる。

関連する要素、属性及び宣言

img object

技術解説

ウェブページを使いやすく、わかりやすくするためにはテキストだけでなく、目印となる統一的なアイコンや画像、わかりやすい図記号などが有効である。

ソリューション

- 1) テキストのみのウェブページを避け、分かりやすい図記号、イラストレーション、アイコンなどの視覚的な目印を与える。
- 2) 報知音によってボタンをクリックしたことを知らせたり、適切な効果音によって操作を支援する。

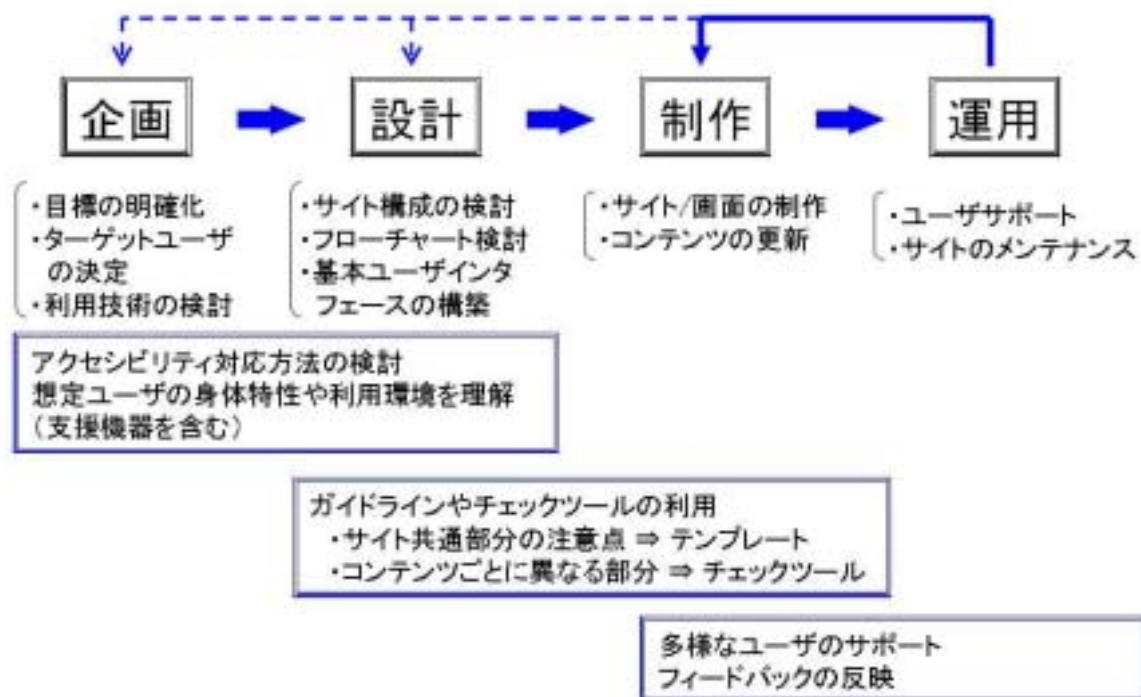
注意事項

画像や音声は、視覚、聴覚障害のある人にとっては扱いにくい場合もあるので、その使い方には十分注意する必要がある。

6. 情報アクセシビリティの確保・向上に関する全般的要件

ウェブサイトのアクセシビリティを向上させるためには、企画、設計、政策および運用までの幅広い工程での配慮が重要である。各工程での配慮事項の一部を以下に説明する。

- (1) 企画工程では、特定のプラグイン（JavaScript、Javaアプレット、Flash、PDFほか）などの利用が情報を表現する上で必要かどうかを検討し、使う場合には、それが使えない利用者に対して同等の情報を提示できるよう検討する必要がある。また、要素技術レベルの流れ作業的なアクセシビリティ対応に陥らないよう、利用者の心身特性や利用環境を理解することが重要である。
- (2) 設計工程では、ウェブサイト全体が分かりやすい構造になっているか、利用者の記憶に頼らず操作ができるか、認知面での配慮が必要である。また、サイト全体で共通に使用するテンプレートなどもアクセシビリティを考慮して設計する必要がある。
- (3) 制作工程では、制作者のスキルに依存せず、またチェック漏れを少なくするために、ウェブガイドラインを元にしたチェックリストを作成する。また、チェックツールを用いると文法チェックが効果的に行える。
- (4) 運用工程では、利用者から寄せられるウェブサイトへの要望や改善提案などは定期的に収集し、ページレベルで対応できるものは反映を行うことが必要である。



6.1 企画・制作に関する要件（JIS X 8341-3:6.1）

ウェブコンテンツの情報アクセシビリティが容易に維持できるよう企画・制作しなければならない。

ウェブコンテンツを自動的に生成するプログラムには、アクセス可能なウェブコンテンツを生成する機能を装備する。

背景と問題点

ウェブコンテンツの制作段階で、各ページ単位でアクセシビリティの対策を考えても、サイト全体のアクセシビリティを均一に保つことは困難である。また、効率的にすべてのアクセシビリティの要件を満たすことが難しくなる。また、後工程にいくほど個々のページをアクセシビリティ対応することになるので、コストもかかる。

効率的かつ効果的に実施するには、企画・制作工程などウェブサイト構築の早い段階で、アクセシビリティの配慮を検討する。企画工程では、サイト全体で共通とするアクセシビリティの方針を定め、できるだけ早いうちに具体的に反映する。

これにより、個々の工程ごとの単発的な対応でなく、サイト全体で共通のナビゲーションやフォーマットをアクセシブルにできる。

利用者情報などをデータベースで管理する場合には、データが集積すればするほどアクセシビリティ対応を後付するのは難しくなる。

電子商取引のような利用者の入力に対応して、ウェブページを自動的に生成するようなウェブプログラムを有するウェブサイトの場合は、生成されたコンテンツもアクセシビリティの要件を満たすように、自動生成を行うプログラムにアクセシビリティ機能を装備することが重要である。

こうした取組みを実現するために、ウェブサイトの関係者全員がアクセシビリティについて正しく理解することにより、効率的かつ効果的に対処できる。

サイト制作を外注する場合、最初のサイト構築は外注先が行っても、情報更新など運用は発注元が行う場合がある。その際、サイト全体を運用しやすいように企画段階から考慮しておかないと、アクセシビリティを維持することが難しくなる。

関連項目

なし

技術解説

- ウェブサイトの情報アクセシビリティが人や組織によらず容易に維持できるよう、デザインガイドラインを作成し、その中でアクセシビリティを実現・管理する方法を具体的に記述する。
- 画像情報をデータベースで管理する場合には、入力時に画像データとともにテキスト入力欄を付記して登録するようにしておく、alt 属性として活用できる。
- CMS (Content Management System) など、ウェブコンテンツを自動的に管理、生成するソフトウェアを採用する場合は、アクセシビリティの維持、管理を効率的に行えるものを採用する。例えば、コンテンツを登録する画面で、アクセシビリティ・チェック機能を利用できるソフトウェアを採用する。
- サイト全体で共通に使用するフォーマット（画面デザイン・HTML テンプレートなど）を制作する場合は、アクセシビリティについて考慮し、制作する。
- 自動的にアクセシビリティを確保できるような編集ツールが存在する場合は、その使用を制作者に義務づけるか、アクセシビリティのチェックツールが存在する場合は、その使用を制作者に義務づける。
- ウェブサイトにおけるアクセシビリティに関する基本的な考え方、方針、具体的な推進方法などを日付や作成部門名／作成者を付記して文書化し、ウェブサイトの関係者全員が参照できるようにする。
- ウェブサイトの関係者が、アクセシビリティに関する教育を受講する。

- サイト制作を外注する場合、発注元もアクセシビリティの基本的な要件を学び、企画段階から参画すると、サイト全体の構造などを理解しやすく、アクセシビリティに対応できるだけでなく、運用しやすいウェブサイトが構築できる。
- ウェブサイトを新規制作する場合は、サイト全体をアクセシブルなものにすべきであるが、既存のサイトを修正する場合には、利用者に閲覧されやすい上位階層から下位階層へ、情報の重要度が高いものから低いものへ、またアクセス数の高いものから低いものへ、など優先度の高いものからアクセシビリティ対応を行う。

6.2 保守及び運用に関する要件（JIS X 8341-3:6.2）

ウェブコンテンツを保守及び運用するときは、情報アクセシビリティの品質を確保し、向上させなければならない。

背景と問題点

ウェブコンテンツの保守及び運用工程で、アクセシビリティを考慮せずに情報を追加・削除・更新してしまうと、サイト構築時にはアクセシビリティに配慮されていたとしても、次第にサイト全体がアクセシブルでなくなってしまう。これは、(1)アクセシビリティを維持しにくいサイト構造・ページ構成である、(2)保守、運用担当者のアクセシビリティに対する理解やスキルが不足している（人事異動などで手法やノウハウが継承されないことも含む）、(3)アクセシビリティ維持のための予算や工数などが確保できていない、などさまざまな原因がある。

関連項目

なし

技術解説

- ウェブサイトの情報アクセシビリティが人や組織によらず容易に維持できるよう、保守及び運用のためのアクセシビリティチェック項目を検討し、チェックリストとして明文化する。
- ウェブサイトにおけるアクセシビリティに関する基本的な考え方、方針、具体的な推進方法などを日付や作成部門名／作成者を付記して文書化し、ウェブサイトの関係者全員が参照できるようにする。
- 保守及び運用作業において、アクセシビリティが確保しにくい場合は、ウェブサイトの企画・制作部門と連携し、より効率的に確実にアクセシビリティを確保できるようにする。
- アクセシビリティのチェックツールが存在する場合は、それを日々の作業で励行する。
- ウェブサイトの関係者が、アクセシビリティに関する教育を受講する。

6.3 検証に関する要件 (JIS X 8341-3:6.3)

ウェブコンテンツの企画・制作を行う者は、ウェブコンテンツがこの規格の要件を満たしていることを検証しなければならない。

背景と問題点

サイト全体で共通に使用するフォーマット (画面デザイン・HTML テンプレートなど) を制作する場合、そのフォーマットでアクセシビリティを確保する。そうでないと、各ページの制作段階で、個別に対応する必要が生じ、場合によっては、大きな修正作業が発生してしまうことになる。

企画・制作・保守・運用のすべての工程で、アクセシビリティの評価、検証を行うことで、効率的にアクセシビリティを実現できる。

関連項目

なし

技術解説

- ・ 評価・検証作業を効率的に実施するため、高齢者・障害者支援技術、チェックツールを適宜組み合わせて、利用するとよい。
- ・ 評価・検証作業を複数人で実施する場合は、アクセシビリティを一定のレベルにそろえるため、検証方法を定期的に調整し、チェックリストなどを作成する。また、チェックリストは、適宜、更新しておく。
- ・ 各工程に適した評価・検証手法を選択する。

(1)企画工程

評価できるウェブコンテンツがないので「質問法」が中心になる。

(2)設計工程

機能仕様書、ユーザインタフェース仕様書などを「洞察法」で評価できる。

(3)詳細設計段階

評価用のプロトタイプ、β 版、公開版ができれば、「洞察法」に加え、「観察法」、「実験的方法」が可能になる。

質問法 (ヒヤリング) には、1.質問紙調査(アンケート調査)、2.インタビュー/グループインタビュー、3.個人面接 などがある。

洞察法には、1.ヒューリスティック評価、2.認知的ウォークスルー などがある。

観察法には、1.行動観察、2.ユーザタスク分析 などがある。

実験的方法には、1.パフォーマンステスト、2.プロトコール解析 などがある。

- ・ チェックツールと人手による評価・検証手法を併用し、効率良く確実にアクセシビリティを向上させる。

(1)アクセシビリティ専用チェックツールとして、以下のものがある。

- ・ WebInspector (富士通) <http://design.fujitsu.com/jp/universal/assistance/>
- ・ Web Accessibility Toolbar (インフォアシア) <http://www.infoaxia.com/tools/wat/index.html>
- ・ aDesigner (日本 IBM) http://www.research.ibm.com/trl/projects/acc_tech/adesigner.htm
- ・ LIFT for Macromedia Dreamweaver (ソシオメディア) <http://www.sociomedia.co.jp/lift/>

- ・ウェブヘルパーASP (アライド・ブレインズ) <http://www.aao.ne.jp/author/webhelper/index.html>

例: alt 属性の有無など, HTML ファイルの記述を機械的に確認するときに効果的である。

参考: HTML 文法チェックツール (「Another HTML-lint」など) でもチェック可能な項目がある。

(2) 人手による評価・検証手法として, 以下のようなものがある。

- ・グレースケールで表示しても, 内容が伝わるか確認する。

解説: ウェブの画面を画像データとして保存し, 画像編集ソフトなどでグレースケールに変換する。これにより, 明度の違いだけで内容が把握できることを確認する。

- ・文字色と背景色の組み合わせをチェックする。

- ・キーボードのみで操作ができるか確認する。特に次の4つのポイントでチェックする。

1. 「上矢印」「下矢印」キーによる画面スクロールが可能か。

2. 「Tab」キーによるキーボード・フォーカスの移動が可能か。全てのリンク及び入力項目に, 正しい順序で移動できるようにする。

3. リンクやコマンドは, 「Enter」キーで実行できるか。フォーカスの移動ではなく, 「Enter」キーを押すまで, 実行しない。

- ・音声ブラウザを使用し, 正しく読み上げられるか確認する。

解説: alt 属性の内容の適切さなど, チェッカーツールで判定できない問題を確認するとき, 効果的である。

- ・ブラウザの設定を変更し, 内容が伝えられることを確認する。

1. グラフィックス表示をオフにし, 代替表示される alt 属性のテキストだけで情報が伝えられるか。

2. サウンド出力をオフにし, 重要な情報を伝えられるか。

3. スタイルシートをオフにし, ページが読めるか。

4. 画面表示を白黒反転などさせてページが読めるか。

参考: Microsoft Windows の場合, [コントロールパネル] の[ユーザー補助]の[画面]で[ハイコントラスト]を選択する。

6.4 フィードバックに関する要件 (JIS X 8341-3:6.4)

ウェブコンテンツの企画・制作を行う者は, 利用者の意見を収集する窓口を用意し, 利用者からの意見をウェブコンテンツの情報アクセシビリティの確保・向上に活かさなければならない。

背景と問題点

ウェブサイトは, 情報を追加, 削除, 更新しながら常に変化していくものである。

更新時のウェブサイトの情報が見やすく, わかりやすく, 操作しやすいかウェブサイト提供者側でチェックするとともに, 利用者からの意見や要望などを広く収集する必要がある。

さらに企画・制作の工程でそれらに的確に対応できる体制・手法などを確保しておく必要がある。

関連項目

なし

技術解説

- ・利用者が、ウェブサイトの管理者へ容易に意見、要望、質問ができるようにする。
- ・利用者から収集した意見を、関係者が適切に把握し、効率的かつ効果的に対処できるようにする。
- ・利用者の様々な状況に合わせ、電子メール、ウェブのフォーム、電話、ファックスなど、複数の意見収集窓口を用意する。

6.5 サポートに関する要件（JIS X 8341-3:6.5）

利用者とコミュニケーションが取れるよう、問い合わせ先をウェブコンテンツ上の分かりやすい位置に明示しなくてはならない。

背景と問題点

ウェブサイトは、情報を追加、削除、更新しながら常に変化していくものである。

更新時のウェブサイトの情報が見やすく、わかりやすく、操作しやすいかウェブサイト提供者側でチェックするとともに、利用者からの意見や要望などを広く収集する必要がある。

それには、利用者が容易に問い合わせできるようその方法をウェブサイトに明示にすることは重要である。

関連項目

なし。

技術解説

- ・利用者が、ウェブサイトの管理者へ容易に意見、要望、質問ができるようにする。
- ・利用者から収集した意見を、関係者が適切に把握し、効率的かつ効果的に対処できるようにする。
- ・利用者の様々な状況に合わせ、電子メール、ウェブのフォーム、電話、ファックスなど、複数の意見収集窓口を用意する。

7. 要素別索引

7.1 メタ要素

7.2 基本構造要素

html

body

head

h1～h6

title

p

7.3 リスト

ol

ul

dl/dd/dt

li

7.4 テキスト構造マークアップ

div

span

abbr

acronym

address

code

cite

dir

em

small

big

hr

ins

del

kbd

q

dfn

blockquote

sub

sup

em

strong

7.5 テキスト修飾マークアップ

b

basefont

font

i

u

tt

br

center

pre

7.6 リンク

a

link

7.7 フォーム

form

input

textarea

option

optgroup

button

select

label

legend

7.8 テーブル

table

tr

th

td

caption

tbody

thead

tfoot

7.9 フレーム

frame

noframes

7.10 画像とイメージマップ

img

map

area

7.11 オブジェクト

applet

object

7.12 スタイルシート

style

Appendix 1 アクセシブルな Macromedia Flash を作成するためのテクニック

(作成中)

Appendix 2 アクセシブルな Adobe PDF を作成するためのテクニック

(作成中)

(**参考資料**: [html4.01 elements list](#))

(作成中)