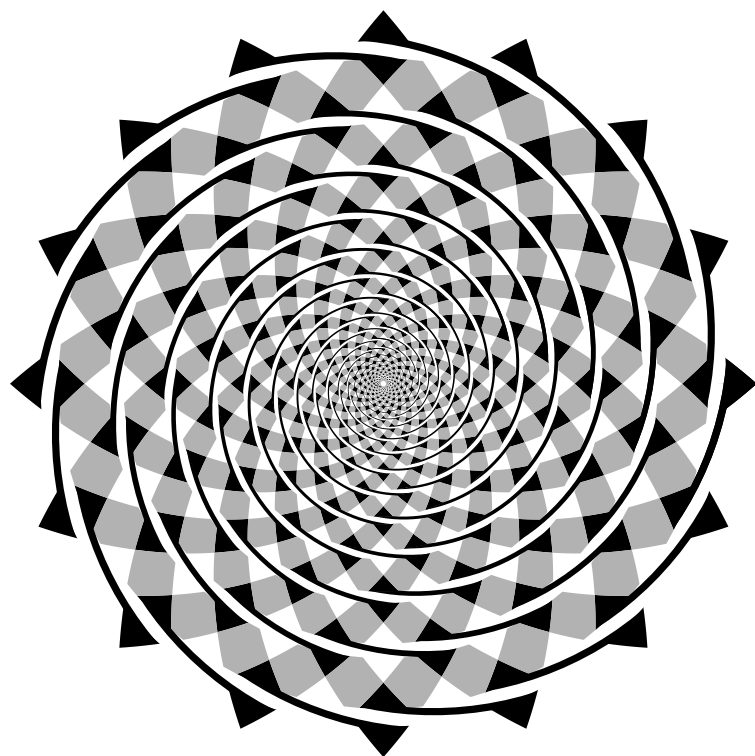


SVG ではじめる Web Graphics 2016 年度版
(情報メディア専門ユニット I)
平成 29 年 2 月 27 日改訂



フレイザーの渦巻き錯視

この文書についてお問い合わせは下記のメールまでご連絡ください。

©2006 – 2016 平野照比古

e-mail: hilano@ic.kanagawa-it.ac.jp

URL: <http://www.hilano.org/hilano-lab>

目 次

第 1 章 SVG について	1
1.1 SVG とはなにか	1
1.2 XML とは	2
1.3 SVG の画像を表示する方法	5
1.3.1 SVG が取り扱えるソフトウェア	5
1.4 HTML5 について	5
1.4.1 HTML5 までの道程	5
1.4.2 HTML5 における新機能	5

第1章 SVG について

1.1 SVG とはなにか

Web 上での各種規格は World Wide Web Consortium(W3C) と呼ばれる組織が中心になって制定しています。その Web Design and Applications¹ (図 1.1 参照) には次のように書かれています。

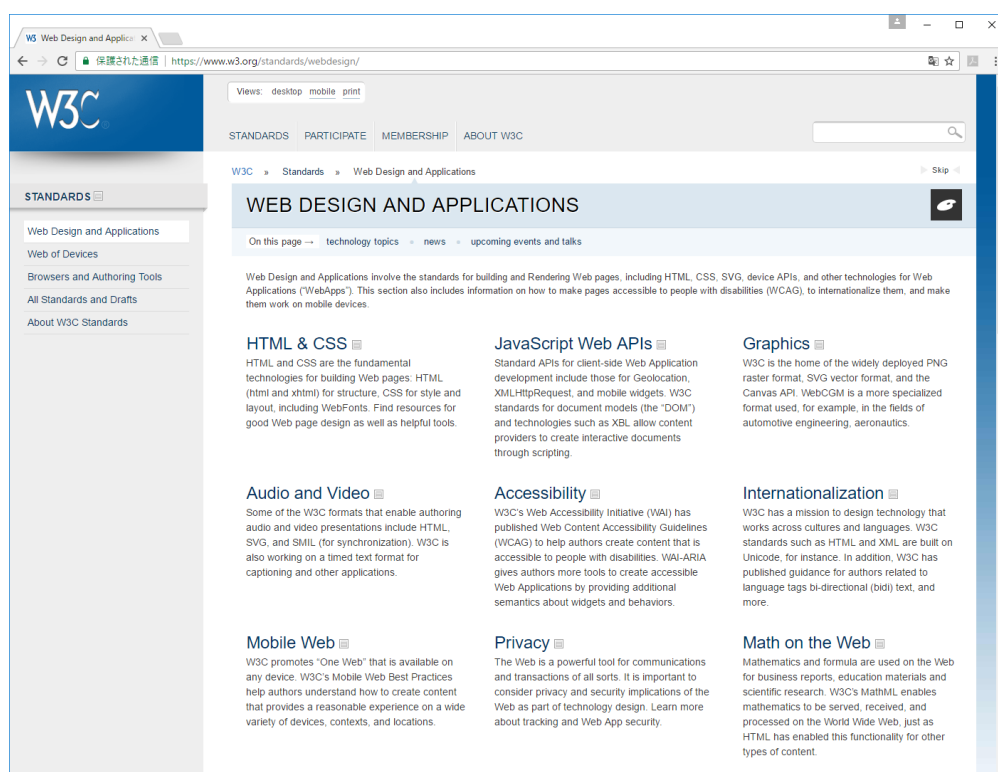


図 1.1: World Wide Web Consortium の Web Design and Application のページ (2017/2/27 参照)

Web Design and Applications involve the standards for building and Rendering Web pages, including HTML, CSS, SVG, device APIs, and other technologies for Web Applications (“ WebApps ”). This section also includes information on how to

¹<http://www.w3.org/standards/webdesign/>

make pages accessible to people with disabilities (WCAG), to internationalize them, and make them work on mobile devices.

このページの Graphics をクリックすると W3C のグラフィックス関係の規格の解説のページへ移動します (図 1.2)。

このページの下に SVG の簡単な説明があります。

Scalable Vector Graphics (SVG) is like HTML for graphics. It is a markup language for describing all aspects of an image or Web application, from the geometry of shapes, to the styling of text and shapes, to animation, to multimedia presentations including video and audio. It is fully interactive, and includes a scriptable DOM as well as declarative animation (via the SMIL specification). It supports a wide range of visual features such as gradients, opacity, filters, clipping, and masking.

The use of SVG allows fully scalable, smooth, reusable graphics, from simple graphics to enhance HTML pages, to fully interactive chart and data visualization, to games, to standalone high-quality static images. SVG is natively supported by most modern browsers (with plugins to allow its use on all browsers), and is widely available on mobile devices and set-top boxes. All major vector graphics drawing tools import and export SVG, and they can also be generated through client-side or server-side scripting languages.

SVG の正式な規格書は図 1.2 のページの右側にある CURRENT STATUS の部分の SVG をクリックするとみることができます (図 1.3)。

画像を作成し、保存する形式を大きく分けるとビットマップ方式とベクター方式があります。

ビットマップ方式は画像を画素 (ピクセル) という単位に分け、その色の情報で画像を表します。したがってはじめに決めた大きさで画像の解像度が決まります。Adobe 社の Photoshop や Windows に標準でついてくるペイントはビットマップ方式の画像を作成するツールです。

これに対し、ベクター方式では線の開始位置と終了位置 (または長さや方向)、線の幅、色の情報などをそのまま持ちます。したがって画像をいくら拡大しても画像が汚くなることはありません。ベクター方式のソフトウェアはドロー系のソフトウェアとも呼ばれます。代表的なものとしては Adobe 社の Illustrator があります。

SVG はその名称からもわかるようにベクター形式の画像を定義するフォーマットのひとつです。

予習問題 1.1 次の事柄について調べなさい。

1. 画像の保存形式を調べ、それがビットマップ方式かベクター方式か調べなさい。
2. ビットマップ方式とベクター方式の画像形式の利点と難点をそれぞれ述べなさい。

1.2 XML とは

XML は eXtensible Markup Language の略です。“Markup Language” の部分を説明します。

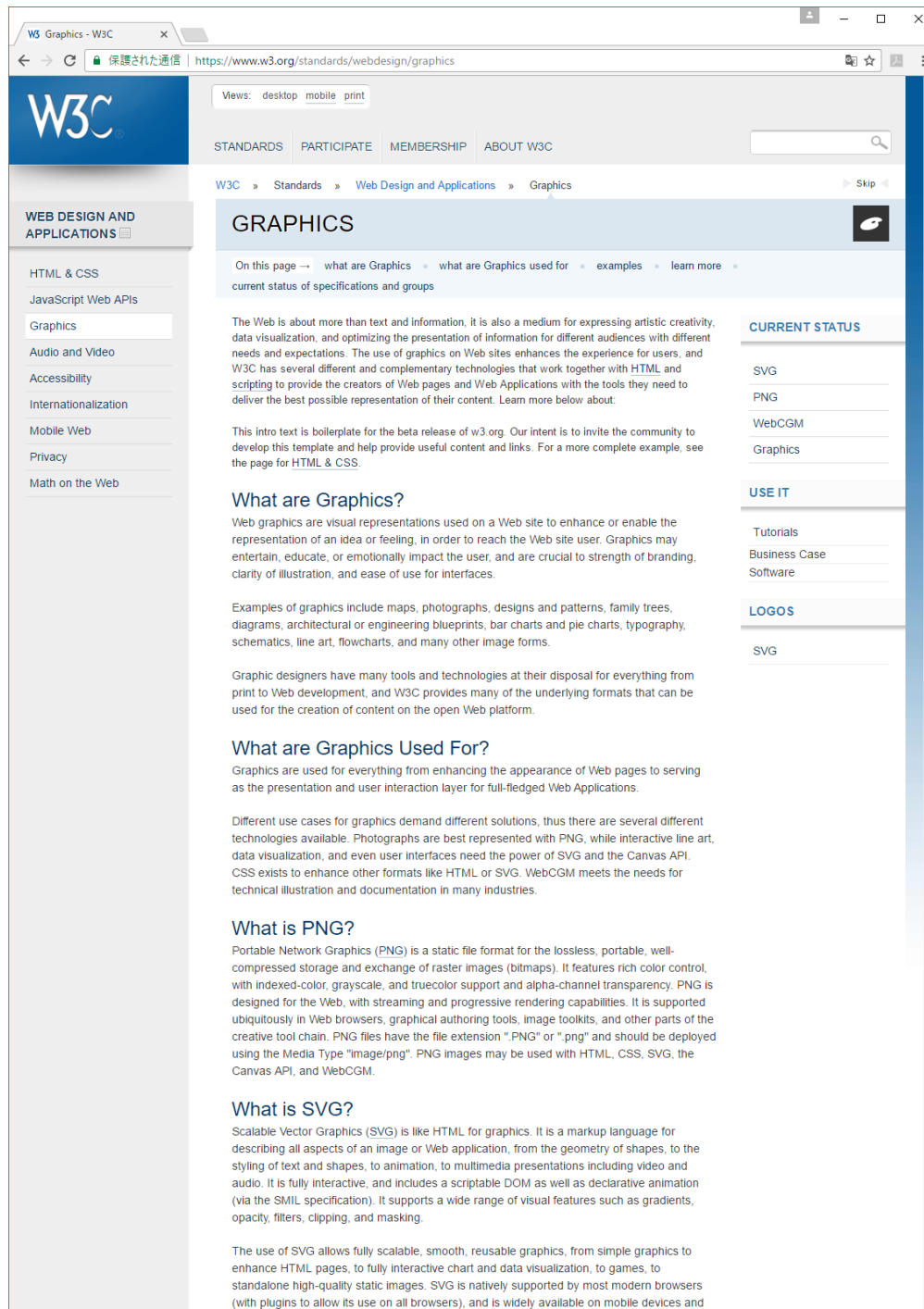


図 1.2: W3C の SVG に関するトップページ (2017/2/27 参照)



図 1.3: SVG の規格のページ (2017/2/27 参照)

ワープロなどで文書を作成するとき、ある部分は見出しなので字体をゴシックに変えるということを行います。このように文書には内容の記述の部分とそれを表示するための情報を含んだ部分があることになります。このように本来の記述以外の内容を含んだ文書をハイパーテキストと呼びます。Microsoft Word のようなワープロソフトが作成する文書もハイパーテキストです。このようなハイパーテキストを表現するために文書の中にタグと呼ばれる記号を挿入したものが Markup Language です。

Web のページは普通 HTML(HyperText Markup Language) と呼ばれる形式で記述されています。HTML 文書は先頭が <HTML> 要素で最後が </HTML> で終わっています。ここで現れた <HTML> 要素がタグと呼ばれるものです。HTML 文書ではこのタグが仕様で決まっています。ユーザが勝手にタグを定義することができません。一方、XML では extensible という用語が示すようにタグは自由に定義できます。したがって、XML を用いることでいろいろな情報を構造化して記述することが可能になっています。XML についてもっと知りたい場合には文献 [?]などを参照してください。

1.3 SVG の画像を表示する方法

SVG に基づいた画像を作成する方法については次章で解説をします。SVG は XML の構造を持ったベクター形式の画像を定義するフォーマットです。したがって、SVG に基づいたファイルだけでは単なる文字の羅列にしか見えず、これだけでは画像を表示することはできません。画像を表示するためにはソフトウェアが必要になります。このテキストの題名である Web Graphics の観点からブラウザにより表示することを主に考えます。最近のブラウザは SVG の画像の表示をサポートしています。

しかし、SVG の機能の一つであるアニメーション (第?? 章で説明します) のうち色のアニメーションについてはサポートしていません。色のアニメーションは CSS で実現できるので 2017 年度からはこの方法で説明をします。

1.3.1 SVG が取り扱えるソフトウェア

ブラウザ以外のいくつかのソフトウェアでも SVG ファイルを取り扱うことができます。

Adobe 社のドローソフト Illustrator は 簡単な SVG のファイルを表示したり、結果を SVG ファイルとして保存することが可能です。

1.4 HTML5 について

2015 年 3 月現在、代表的なブラウザは最新の HTML5 機能をインプリメントしています。

1.4.1 HTML5 までの道程

W3C は文法上あいまいであった HTML4 の規格を厳密な XML の形式に合う形にするために XHTML[?] を制定しました。これとは別にベンダー企業は 2004 年に WHATWG (Web Hypertext Application Technology Working Group) を立ち上げ、XHTML とは別の規格を検討し始めました。2007 年になって W3C は WHATWG と和解をし、WHATWG の提案を取り入れる形で HTML5 の仕様の検討が始まり、2014 年 10 月に規格が正式なもの (Recommendation) になりました。

1.4.2 HTML5 における新機能

HTML5 では HTML 文書で定義される要素だけではなく、それに付随する概念も含めます。HTML5 における新機能としては次のようなものがあります。

- 文書の構成をはっきりさせる要素が導入されています。文書のヘッダー部やフッター部を直接記述できます。
- フォントの体裁を記述する要素が非推奨となっています。これらの事項は CSS で指定することが推奨されています。

- SVG 画像をインラインで含むことができます。

- WebStorage

ローカルのコンピュータにその Web ページに関する情報を保存して、あとで利用できる手段を与えます。機能としては Cookie と同じ機能になりますが、Cookie がローカルに保存されたデータを一回サーバーに送り、サーバーがそのデータを見て Web ページを作成するのに対し、WebStorage ではそのデータがローカルの範囲で処理されている点が一番の違いです。

Web ページの中には 2 度目に訪れた時に以前の情報を表示してくれるところがあります。これは Cookie に情報を保存しておき、再訪したとき、ブラウザが保存されたデータを送り、そのデータを用いてサーバーがページを構成するという手法で実現しています。WebStorage を用いると保存されたデータをサーバーに送ることなく同様のことが実現可能となるのでクライアントとサーバーの間での情報の交換する量が減少します。また、WebStorage では保存できるデータの量が Cookie と比べて大幅に増加しています。

- <canvas> 要素

インラインでの画像表示の方法として導入されました。画像のフォーマットはビットマップ方式で、JavaScript を用いて描きます。描かれた図形はオブジェクト化されないため、マウスのクリック位置にある図形はどれかなどの判断は自分でプログラムする必要があります。

また、アニメーションをするためには途中の図形の形や位置などを自分で計算する必要があり、複数の図形のアニメーションの同期も自分で管理する必要があります。

このテキストでははじめは単独で SVG による画像を描きますが、途中からは HTML 文書内で SVG の画像を表示し、それに対して構成要素を変えるプログラミングについて解説します。