

## SVG 資料第 3 回目 (その 1) 複雑な図形を描く

メディア専門ユニット I(SVG)

2017/5/9

いくつかの直線をつなげて折れ線や多角形を描くことができる

- ▶ 折れ線は<polyline>要素、多角形は<polygon>要素を用いる
- ▶ 各頂点は属性 `points` で指定する
- ▶ 折れ線も多角形も内部が塗られる。塗りたくなければ属性 `fill` に `none` を指定する
- ▶ 内部の塗り方を指示する属性 `fill-rule` がある
- ▶ 各頂点における線分の交わりの形状を定める属性 `linejoin` がある (配布資料 41 ページ)

## 多角形の例-(配布資料 39 ページ)

第 3 回目 (その 1)

メディア専門ユニット I(SVG)

折れ線と多角形

直線で構成される  
図形を描く

曲線を描  
く—Bézier 曲線

正五角形や星形を描いてみました。

## 多角形の例—ソースコード

第 3 回目 (その 1)

メディア専門ユニット I (SVG)

折れ線と多角形

直線で構成される  
図形を描く

曲線を描く  
— Bézier 曲線

```
1<?xml version="1.0" encoding="UTF-8" ?>
2<svg xmlns="http://www.w3.org/2000/svg"
3    xmlns:xlink="http://www.w3.org/1999/xlink"
4    height="100%" width="100%">
5  <title>polygon 要素の例</title>
6  <g transform="translate(110,110)">
7    <g transform="scale(1,-1)">
8      <polygon
9        points="0,100 -95.1,30.9 -58.8,-80.9 58.8,-80.9 95.1,30.9"
10       stroke="black" stroke-width="4" fill="blue" />
11    <g transform="translate(210,0)">
12      <polygon
13        points="0,100 -58.8,-80.9 95.1,30.9 -95.1,30.9 58.8,-80.9"
14        stroke="black" stroke-width="4" fill="yellow" />
15    </g>
16    <g transform="translate(420,0)" fill="red" fill-rule="evenodd" >
17      <polygon
18        points="0,100 -58.8,-80.9 95.1,30.9 -95.1,30.9 58.8,-80.9"
19        stroke="black" stroke-width="4"/>
20    </g>
21  </g>
22 </svg>
```

## 多角形の例-ソースコード (解説)

第 3 回目 (その 1)

メディア専門ユニット I(SVG)

折れ線と多角形

直線で構成される  
図形を描く

曲線を描く  
— Bézier 曲線

- ▶ 7 行目で  $y$  座標の向きを数学で使われる方向に直している
- ▶ 左の正五角形は 8 行目から 10 行目で定義
- ▶ 座標は前もって計算しておいた。
- ▶ 中央の星形は 12 行目から 14 行目で定義
- ▶ 正五角形の頂点を一つ置きに取っている
- ▶ 右の図形は 17 行目から 19 行目で定義
- ▶ 頂点の並びは中央と同じ
- ▶ 違いは属性 `fill-rule` が `evenodd`
- ▶ 領域内の点と遠方を直線で結んだとき、図形の縁と交わる回数が奇数のところだけ塗られる

## <polygon>要素と<polyline>要素の注意

第 3 回目 (その 1)

メディア専門ユニット I(SVG)

折れ線と多角形

直線で構成される  
図形を描く

曲線を描  
く—Bézier 曲線

属性 `linejoin`(配布資料 41 ページ)

- ▶ 折れ線の曲がる場所の処理の指定
- ▶ 値の変化が激しいグラフを書く時などには `miter` は使わないように

<path>要素は多角形や曲線を含む図形を描ける。

- ▶ 図形の形は属性 `d` で指定
- ▶ 記述するパラメータは大文字と小文字がある
- ▶ 大文字は絶対座標
- ▶ 小文字は直前の点からの相対座標
- ▶ `M,m`: 指定した位置に移動 (moveto)
- ▶ `L,l`: 指定したまで道のりを指定 (lineto)。これらは省略可能
- ▶ `z`: 初めの位置と最後の位置を結んで閉じる
- ▶ 座標は空白またはコンマ (,) で区切る  
テキストでは座標の対応をはっきりさせるため、ここでは  $x$  座標と  $y$  座標の間はコンマにしている

折れ線と多角形

直線で構成される  
図形を描く

曲線を描く  
— Bézier 曲線

# 長方形をいろいろな方法で描く

第 3 回目 (その 1)

メディア専門ユニット I (SVG)

折れ線と多角形

直線で構成される  
図形を描く

曲線を描く  
—— Bézier 曲線

```
1<?xml version="1.0" encoding="UTF-8" ?>
2<svg xmlns="http://www.w3.org/2000/svg" height="100%" width="100%">
3  <title>長方形の描き方</title>
4  <g fill="red" stroke="blue" stroke-width="10">
5    <g transform="translate(20,10)">
6      <rect x="0" y="0" width="150" height="50"/>
7    </g>
8    <g transform="translate(20,110)">
9      <polygon points="0,0 150,0 150,50 0,50" />
10   </g>
11   <g transform="translate(210,110)">
12     <polyline points="0,0 150,0 150,50 0,50"/>
13   </g>
14   <g transform="translate(20,210)">
15     <path d="M0,0 150,0 150,50 0,50 0,0"/>
16   </g>
17   <g transform="translate(210,210)">
18     <path d="M0,0 150,0 150,50 0,50 z"/>
19   </g>
20   <g transform="translate(20,310)">
21     <path d="M0,0 1150,0 0,50 -150,0 z"/>
22   </g>
23 </g>
```



## どのような図形が描かれるでしょうか？

第 3 回目 (その 1)

メディア専門ユニット I (SVG)

折れ線と多角形

直線で構成される  
図形を描く

曲線を描く  
— Bézier 曲線

- ▶ 一番上に基本の長方形 (6 行目)
- ▶ その下に<polygon>要素 (9 行目) と<polyline>要素 (12 行目)
- ▶ さらに下に<path>要素で位置を絶対座標で与え、最後に z なし (15 行目) とあり (18 行目)
- ▶ 21 行目は相対座標 (1) を用いた場合

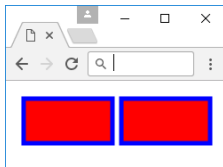
デモに移る

## 次のような図形を描いてみよう。

第 3 回目 (その 1)

メディア専門ユニット I (SVG)

- ▶ 一つの<path>要素で二つの長方形を描く

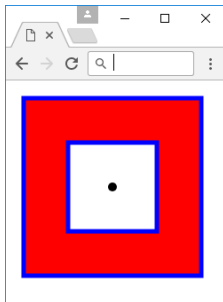


折れ線と多角形

直線で構成される  
図形を描く

曲線を描く  
— Bézier 曲線

- ▶ 穴が開いた正方形を一つの<path>要素で描く<sup>1</sup>



<sup>1</sup>中が塗られていないことを示すために初めに小さな円を描いている

Bézier 曲線は PhotoShop などドロー系のソフトで曲線を描くために利用される  
フリーなドロー系のソフトである GIMP で曲線を描いているところをデモする。

- ▶ (3 次の) Bézier 曲線を定めるためには 4 点  $P_0, P_1, P_2, P_3$  が必要
- ▶ この曲線は始点が  $P_0$  で 終点が  $P_3$
- ▶  $P_1$  は点  $P_0$  における接線の方角を定める
- ▶  $P_2$  は点  $P_3$  における接線の方角を定める
- ▶ 曲線はこの 4 点からなる凸包の中に含まれる

具体的な性質についてはデモで行う

## デモ内容の説明

第 3 回目 (その 1)

メディア専門ユニット I (SVG)

折れ線と多角形

直線で構成される図形を描く

曲線を描く  
— Bézier 曲線

1. 4 点の座標がそれぞれ右に表示
  - ▶ 各点はドラッグ可能
  - ▶ 各点の座標を直接設定可能 (入力後、「設定」ボタンを押す)
  - ▶ 「次」のボタンを押すことで Bézier 曲線の性質が確認できる
2. Bézier 曲線が表示
3. 4 点を含む凸包が表示
4.  $P_0, P_1$  の中点  $P_{01}$  が表示
5.  $P_1, P_2$  の中点  $P_{12}$  が表示
6.  $P_2, P_3$  の中点  $P_{23}$  が表示
7.  $P_{01}, P_{12}$  の中点  $P_{012}$  が表示
8.  $P_{12}, P_{23}$  の中点  $P_{123}$  が表示
9.  $P_{012}, P_{123}$  の中点  $P_{0123}$  が表示。この点は Bézier 曲線上にある
10.  $P_{0123}$  の前後で 2 つに分けた曲線もまた、Bézier 曲線になっている

# 面白い曲線をデザインしてみよう

第 3 回目 (その 1)

メディア専門ユニット I(SVG)

折れ線と多角形

直線で構成される  
図形を描く

曲線を描  
く—Bézier 曲線

これを使って曲線の一部を作成してみよう。  
半端な座標は手動で直すことを忘れないこと

## <path>要素における指定方法

第 3 回目 (その 1)

メディア専門ユニット I(SVG)

折れ線と多角形

直線で構成される  
図形を描く

曲線を描く  
— Bézier 曲線

- ▶ 4 点のうち初めの点  $P_0$  はそれまでに描いた図形の最終位置が使用される
- ▶ 新規に定めたい場合には  $M$  または  $m$  で指定
- ▶ 残りの 3 点は  $C$  または  $c$  の後に指定する。
- ▶ 2 つの Bézier 曲線を滑らかにつなぐためには共通点の接線方向を合わせる

次のコードは (配布資料 58 ページ) の問題 3.8 のハートのマークのソースコード

```
<?xml version="1.0" encoding="UTF-8" ?>
<svg xmlns="http://www.w3.org/2000/svg"
      xmlns:xlink="http://www.w3.org/1999/xlink"
      height="100%" width="100%">
  <title>スーツマークを描く</title>
  <g transform="translate(150,300)">
    <path d="M0,0 C220,-150 100,-320 0,-240 C-100,-320 -220,-150 0,0z"
          fill="pink" stroke-width="4" stroke="red" />
  </g>
</svg>
```

左右対称なので  $x$  座標の値を符号が逆になるように座標系をとっている