SVGDrawer User Guides

Chunfeng Yang

Some Place

July 22, 2018

Overview

Introduction

VISON

Precisely Drawing SVG with Script Programming Language

Technology Path: Creating svg file with program
Programming Language: Script language (JavaScript)
Programming Environment: Node.js

File Scheme

STEP 1 – Initialization

```
var SVGDrawer = require('drawsvg');
var drawer = new SVGDrawer();
```

STEP 2 – Draw something

```
theAttr={};
theAttr['x'] = 25;
...
drawer.element('text', theAttr );
drawer.popData();
```

STEP 3 – Save in svg file

```
var fileName = "grad.svg";
drawer.setFileName( fileName );
// draw something
drawer.close();
```

Example – Draw Circle

```
var SVGDrawer = require('drawsvg');
var drawer = new SVGDrawer();

var fileName = "foo.svg";
drawer.setFileName( fileName );
drawer.header();

var origin_x = 300;
var origin_y = 300;

 // svg
theAttr={};
theAttr['width'] = '100%';
theAttr['height'] = '100%';
theAttr['version'] = '1.1';
theAttr['xmlns']="http://www.w3.org/2000/svg";
drawer.element('svg', theAttr, '');

// line
theAttr={};
theAttr['x1'] = origin_x;
theAttr['y1'] = origin_y;
theAttr['x2'] = parseInt( origin_x ) + 300;
theAttr['y2'] = origin_y;
theAttr['style'] = 'stroke-width:1;stroke:rgb(0,0,0)';
theAttr['marker-end'] = 'url(#Triangle)';
drawer.element('line', theAttr )
drawer.popData();

drawer.popData();
drawer.close();
```

Basic Methods

Setting Attributions

```
theAttr={};
theAttr['width'] = '100%';
```

Adding New Tag

```
drawer.element('line', theAttr )
drawer.pop();
```

Basic Methods

Setting File Name

```
var fileName = "grad.svg";
drawer.setFileName( fileName );
```

Save to SVG File

```
drawer.svg();
```

Show JSON Formate Data

```
drawer.show();
```

SVG Attributions

```
theAttr={};
theAttr['width'] = '100%';
theAttr['height'] = '100%';
theAttr['version'] = '1.1';
theAttr['xmlns'] = "http://www.w3.org/2000/svg";
theAttr['xmlns:xlink']="http://www.w3.org/1999/xlink";
drawer.element('svg', theAttr, '');
// ...
drawer.popData();
```

SVG Code

```
<svg width="100%" height="100%" xmlns="http://www.w3.org/2000/svg"
   xmlns:xlink="http://www.w3.org/1999/xlink">
   ...
</svg>
```

SVG Description Tag

```
drawer.writeComment("---This is a comment---");
```

SVG Code

```
<desc>----------------------------------------------------------
<desc>---This is a comment---</desc>
```

Draw Rectangle


```
theAttr={};
theAttr['id'] =  'rectElement';
theAttr['x'] =  '250';
theAttr['y'] =  '300';
theAttr['width'] =  '250';
theAttr['height'] =  '200';
theAttr['fill'] =  'rgb(255,255,0)';
drawer.element('rect', theAttr );
drawer.popData();
```

Draw Curve


```
var path = 'M 540 140 Q 490 180 510 240';
theAttr={};
theAttr['id'] = 'myCurve';
theAttr['d'] = path;
theAttr['fill'] = 'none';
theAttr['stroke'] = "black";
theAttr['stroke-width'] = 1;
drawer.element('path', theAttr );
drawer.popData();
```

Draw Curve


```
var path = 'M 540 140 Q 490 180 510 240';
theAttr={};
theAttr['id'] = 'myCurve';
theAttr['d'] = path;
theAttr['fill'] = 'none';
theAttr['stroke'] = "black";
theAttr['stroke-width'] = 1;
drawer.element('path', theAttr );
drawer.popData();
```

SVG Code

```
<path id="myCurve" d="M 540 140 Q 490 180 510 240"
   fill="none" stroke-width="1" stroke="black">
</path>
```

Draw Arc


```
var style = {'stroke': "black",
'stroke-width': "1",
'stroke-dasharray':"4 2 "};
var path = 'M 500 350 A 150
150 0 0 0 575 330 ';
theAttr={};
theAttr['id'] = 'M';
theAttr['d'] = path;
theAttr['fill'] = 'none';
theAttr['style'] = style;
drawer.element('path', theAttr );
drawer.popData();
```

SVG Code

```
<path id="M" d="M 500 350 A 150 150
0 0 0 575 330 " fill="none" stroke="black"
stroke-width="1" stroke-dasharray="4 2 " >
</path>
```

Draw Line


```
theAttr={};
theAttr['x1'] = 0 ;
theAttr['y1'] = 0 ;
theAttr['x2'] = 0 ;
theAttr['y2'] = 70 ;
theAttr['stroke-width'] = 1;
theAttr['stroke'] = 'black';
theAttr['marker-end'] = 'url(#T)';
drawer.element('line', theAttr );
drawer.popData();
```

SVG Code

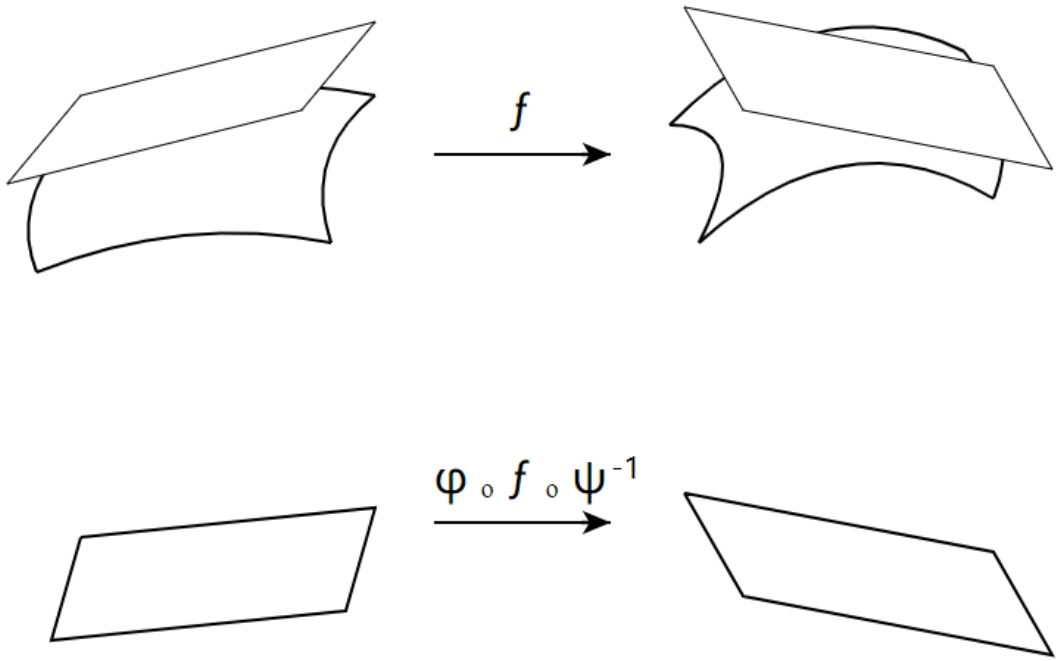
```
<line x1="0" y1="0" x2="0" y2="70" stroke-width="1"
   stroke="black" marker-end="url(#T)">
</line>
```

Animate

```
var thePathes = [];
thePathes.push("M -10 200 C 150 100 300 300 450 200")
thePathes.push("M -10 200 C 150 300 300 100 450 200")
thePathes.push("M -10 200 C 150 100 300 300 450 200")

theAttr={};
theAttr['id', 'P');
theAttr['d', "M 0 200 C 100 0 200 400 300 200");
theAttr['fill', "none" );
theAttr['stroke', "#804");
theAttr['stroke-width', 3);
drawer.element('path', theAttr );

  theAttr={};
  theAttr['attributeName', 'd');
  theAttr['dur', '1s');
  theAttr['repeatCount', 'indefinite');
  theAttr['values', thePathes.join(";") );
  drawer.element('animate', theAttr );
  drawer.popData();

drawer.popData();
```

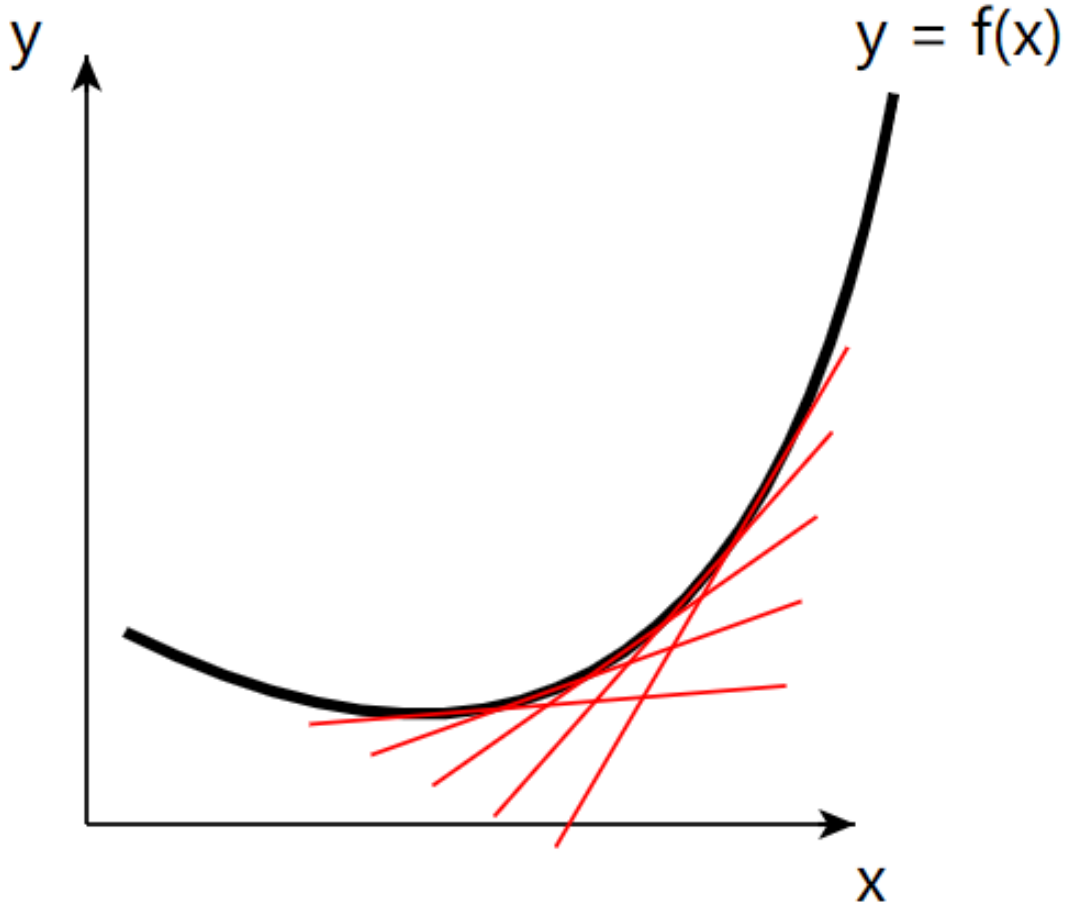
Gallery

Manifold



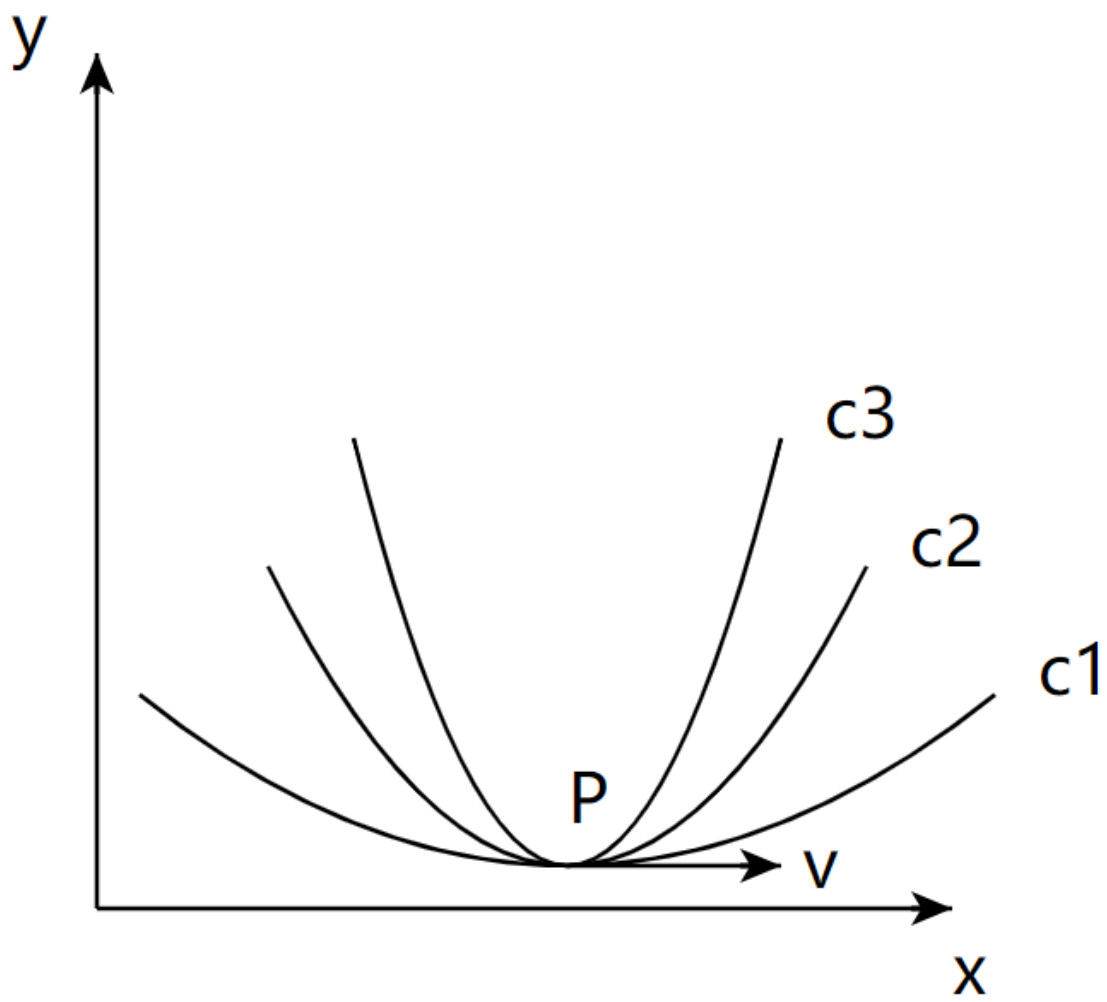
Gallery

Tangent Bundle



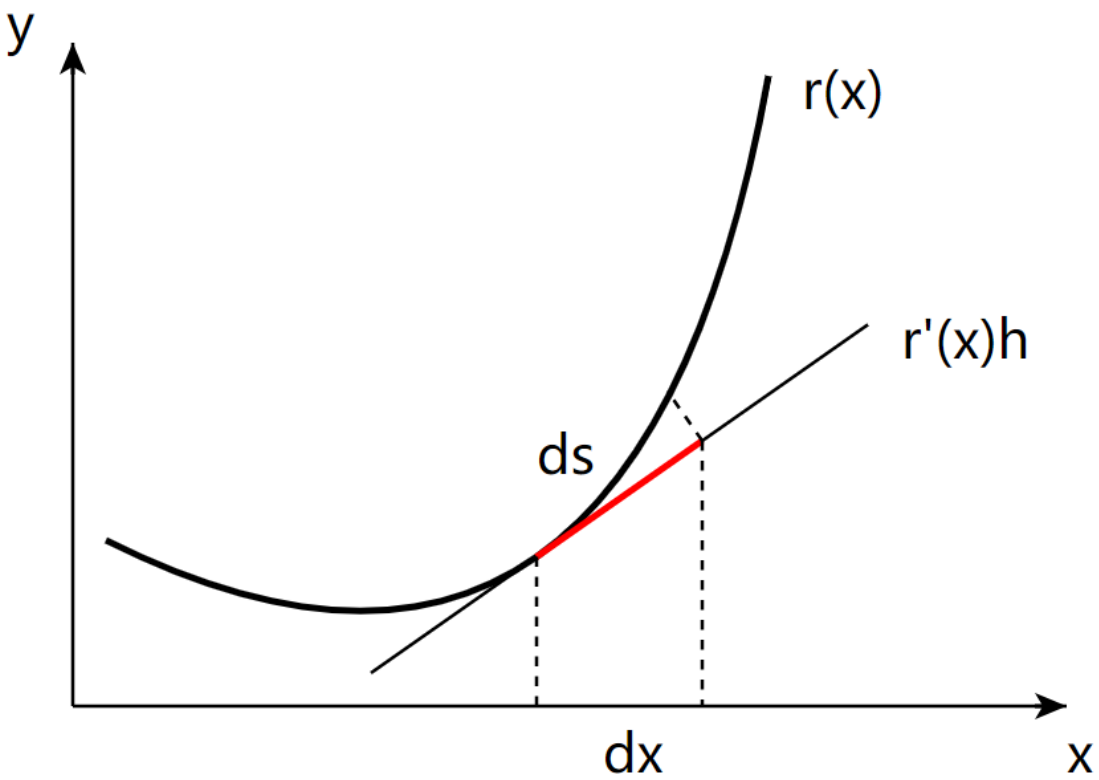
Gallery

Vector Equivalent



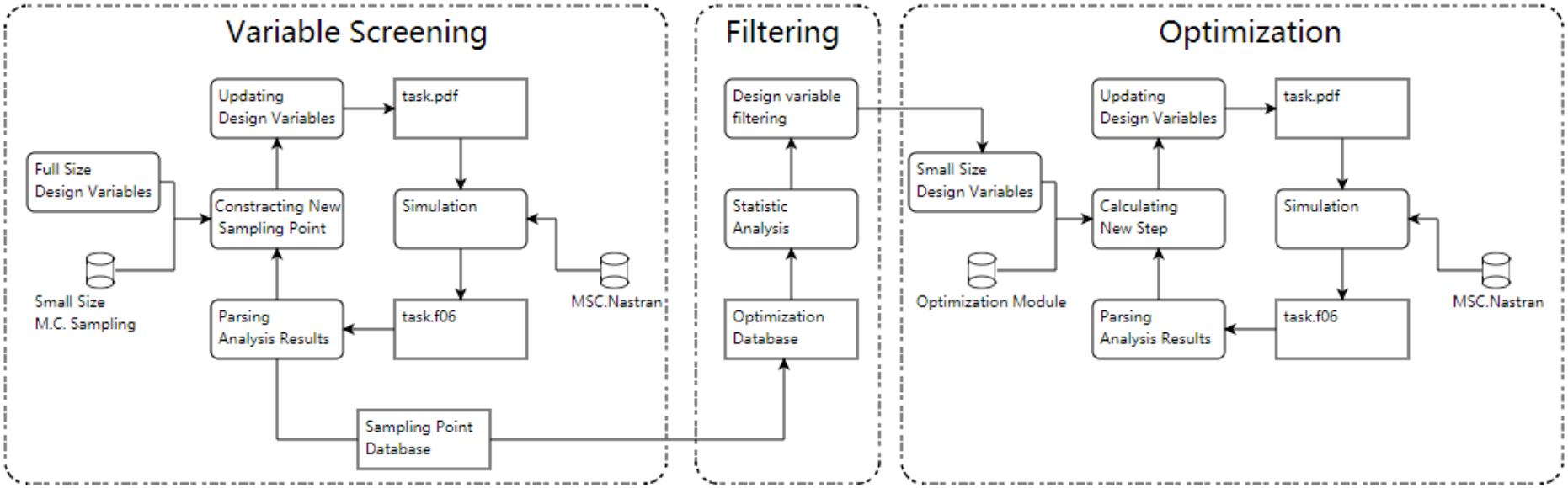
Gallery

Tangent Space



Gallery

Complicate Process



X-Y Coordinate System

```
function xyCoordSystem( length )
{
  drawer.desc("X Coordinate axis");
  var theLine = drawer.append("line");
  theLine.attr( 'x1', origin_x )
  theLine.attr( 'y1', origin_y )
  theLine.attr( 'x2', origin_x + parseFloat( length ) )
  theLine.attr( 'y2', origin_y )
  theLine.attr( 'marker-end', 'url(#Triangle)');
  theLine.attr( 'style','stroke-width:1;stroke:black; stroke-width: 1;');
  theLine.pop();

  drawer.desc("Y Coordinate axis");
  var theLine = drawer.append("line");
  theLine.attr( 'x1', origin_x )
  theLine.attr( 'y1', origin_y )
  theLine.attr( 'x2', origin_x )
  theLine.attr( 'y2', origin_y - parseFloat( length ) )
  theLine.attr( 'marker-end', 'url(#Triangle)');
  theLine.attr( 'style','stroke-width:1;stroke:black; stroke-width: 1;');
  theLine.pop();
}
```

Quadratic Biezier Curve

```
function QBiezierCurve( p1, p2, p3, theID )
{
  var p1_svg = drawer.toUpwardCoord(p1[0], -p1[1], 0.0, origin_x, origin_y );
  var p2_svg = drawer.toUpwardCoord(p2[0], -p2[1], 0.0, origin_x, origin_y );
  var p3_svg = drawer.toUpwardCoord(p3[0], -p3[1], 0.0, origin_x, origin_y );

  var path = 'M ' + p1_svg[0] + ' ' + p1_svg[1] + ' ';
  path +=  'Q ' + p2_svg[0] + ' ' + p2_svg[1] + ' ';
  path +=  p3_svg[0] + ' ' + p3_svg[1] + ' ';

  // path
  var myPath = drawer.append("path");
  myPath.attr('id', theID );
  myPath.attr('d', path);
  myPath.attr('fill','none');
  myPath.attr('style','stroke-width:1;stroke:black; stroke-width: 2;');
  myPath.pop();
}
```

How

- Download node.js (https://nodejs.org/), and install
- Create working directory, for example "d://foo"
- Create a directory named as "node_modules" in working directory
- Copy svg.js into node_modules directory
- Create a file with textpad in working directory, and write the JavaScript codes into it
- Save the text file, for example "foo.js"
- Start => Windows PowerShell => Windows PowerShell
- Input "cd d://foo", and click enter key
- Input "node foo.js", and click enter key

References

John Smith (2012)

Title of the publication

Journal Name 12(3), 45 – 678.

SVG. https://www.w3.org/TR/SVG/

The End