



argonavis.com.br

SVG 1

modelo de apresentação
estilos, cores e transparência
traços e preenchimento

Helder da Rocha (helder@argonavis.com.br)



Modelo de apresentação gráfica

- Objetos em SVG são desenhados em contextos gráficos individuais
 - Grupos, figuras, etc. são desenhados como se cada um fosse uma tela (canvas) própria
 - No processo, efeitos de máscara, transparência, etc. são levados em conta
- Há três tipos de elementos gráficos em SVG
 - figuras (curvas, linhas, etc.)
 - texto
 - imagens matriciais (bitmaps, raster images)



Pintura de figuras e texto

- Figuras e texto podem ser pintadas com dois atributos
 - **preenchimento** (fill)
 - contorno, ou **traço** (stroke)
- O traço é desenhado pelo centro da borda do objeto (ou no meio da linha)
 - Metade do traço está dentro da imagem, metade está fora
- Pode-se variar a **transparência** de traços e preenchimentos
- Há três tipos de "tinta" que podem ser usadas para pintar traços e preenchimentos
 - Cores sólidas (sRGB)
 - Gradientes (degradê)
 - Padrões (texturas)
- **Adicionalmente, qualquer operação de pintura pode ser**
 - filtrada (com algoritmos de filtro diversos)
 - recortada e composta (com máscaras, clipping e operações de composição gráfica)



Valores e tipos de dados

- Valores em CSS e SVG têm unidades
 - SVG usa **defaults** e **contextos** para descobrir unidades de valores absolutos
 - Ex: pixels, graus, segundos são defaults usados em algumas situações (em outras pode ser obrigatório usar unidades)
- Tipos de dados comuns usados em SVG e unidades
 - *ângulo* (deg, grad, rad), *tempo* (s, ms, h, d, etc.)
 - *cor* (nome, #fff, etc.), *corICC*
 - *percentagem* (%), *freqüência* (Hz, kHz),
 - *inteiro* (-2147483648 a 2147483647), *p. flutuante* (-3.4e-38F a+3.4e38F)
 - *IRI, URI*
 - *comprimento* (px, cm, pt, em, etc.)
 - *tinta* (cores, gradientes e padrões)
 - *listas de valores*



Cores

- Toda a especificação de cores em SVG é herdada do CSS.
- Há três maneiras de definir uma cor
 - Por nome*. Ex: **red**, **lightblue**, **maroon**
 - Por código hexadecimal. Ex: **#F93AB7**, **#A09**, **#AA0099**
 - Por função RGB: **rgb(255,0,0)**, **rgb(128,64,32)**, **rgb(0%,50%,100%)**
- Abaixo várias formas de especificar cores básicas
 - **red** **#FF0000** **#F00** **rgb(100%, 0%, 0%)** **rgb(255, 0, 0)**
 - **green** **#00FF00** **#0F0** **rgb(0%,100%, 0%)** **rgb(0,255, 0)**
 - **blue** **#0000FF** **#00F** **rgb(0%, 0%,100%)** **rgb(0, 0,255)**
 - **yellow** **#FFFF00** **#FF0** **rgb(100%,100%, 0%)** **rgb(255,255, 0)**
 - **magenta** **#FF00FF** **#F0F** **rgb(100%, 0%,100%)** **rgb(255, 0,255)**
 - **cyan** **#00FFFF** **#0FF** **rgb(0%,100%,100%)** **rgb(0,255,255)**
 - **white** **#FFFFFF** **#FFF** **rgb(100%,100%,100%)** **rgb(255,255,255)**
 - **black** **#000000** **#000** **rgb(0%, 0%, 0%)** **rgb(0, 0, 0)**



CSS e estilos em SVG

- Os estilos usados para cores, fontes, largura de traço, etc. são derivados do CSS (Cascading Style Sheets)
 - Se você conhece CSS, será fácil aplicar cores, traços, fontes, etc. em SVG

- Regras do CSS podem ser aplicados a vários elementos do SVG usando o atributo **style**

```
<rect x="50" y="20" width="120" height="100"  
style="stroke-width: 4; stroke: blue; fill: yellow" />
```

- Vários atributos próprios do SVG recebem declarações e funções do CSS, como **fill**, e **stroke**

```
<rect x="50" y="20" width="120" height="100" fill="yellow"  
stroke="blue" stroke-width="4"/>
```



Atributos usados em SVG e CSS

- Propriedades de fonte e texto
 - 'font'
 - 'font-family'
 - 'font-size'
 - 'font-size-adjust'
 - 'font-stretch'
 - 'font-style'
 - 'font-variant'
 - 'font-weight'
 - 'direction'
 - 'letter-spacing'
 - 'text-decoration'
 - 'unicode-bidi'
 - 'word-spacing'
- Outras propriedades
 - 'clip'
 - 'color'
 - 'cursor'
 - 'display'
 - 'overflow'
 - 'visibility'
- SVG possui muitos outros atributos que podem ser usados tanto como **atributos XML** como como **propriedades CSS**



Uso de folhas de estilo CSS

- Folhas de estilo CSS podem ser aplicadas a um SVG de duas formas
 - Carregando uma folha externa (isto é feito de acordo com a especificação XML)
`<?xsl-stylesheet type="text/css" href="estilo.css" ?>`
 - Embutindo a folha em bloco `<style>`
`<style type="text/css">`
 `g {font-family: sans-serif; stroke: 2mm}`
 `g.grafico rect, #bar {fill: rgb(100%,90%,10%)}`
`</style>`
- Pode-se incluir no bloco `<style>` qualquer atributo suportado em SVG



Classes e ids

- Objetos SVG podem ser referenciados em folhas de estilo CSS de várias formas
 - Através de um identificador unívoco (id)
 - Através de uma classe (class)
 - Através de outros seletores CSS (ex: nome do elemento)
- **Atributo class define uma classe.**
 - Vários elementos podem pertencer à mesma classe
`<g class="barra"...>` `<rect class="barra" ...>`
- **O atributo id é um identificador**
 - Não pode haver outro elemento com o mesmo id
`<g id="grafico">`
- **Referência a classes e ids em uma folha de estilo CSS**
`.barra {font-family: corbel}`
`#grafico {fill: yellow}`



Preenchimento

- Use **fill** (atributo ou regra CSS) para especificar uma **cor** que irá preencher um objeto
 - `<rect ... fill="rgb(100%,100%,0%)" />`
 - `<rect ... style="fill:rgb(100%,100%,0%)" />`
- O componente alfa, ou transparência é especificado separadamente na propriedade **fill-opacity**
 - Varia de 0 (transparente) a 1 (opaco).

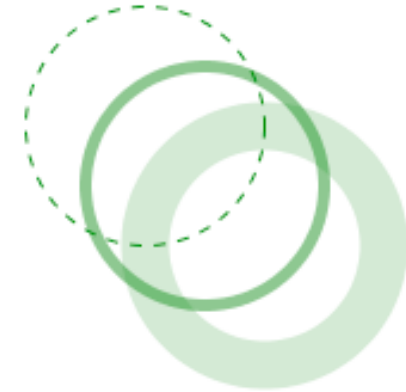
```
<?xml version="1.0" encoding="UTF-8"?>
<svg xmlns="http://www.w3.org/2000/svg"
  width="100%" height="100%">
  <circle r="50" cx="100" cy="100" fill="green"/>
  <circle r="50" cx="125" cy="125" fill="green"
    fill-opacity="0.5"/>
  <circle r="50" cx="150" cy="150" fill="green"
    fill-opacity="0.2"/>
</svg>
```





Traço

- **stroke**
 - Especifica a cor do traço
- **stroke-width**
 - Espessura
- **stroke-opacity**
 - Nível de transparência (alfa)
- **stroke-dasharray**
 - lista de valores para tracejado (seqüência de traços e vazios)



```
<?xml version="1.0" encoding="UTF-8"?>
<svg xmlns="http://www.w3.org/2000/svg" width="100%" height="100%">
  <circle r="50" cx="100" cy="100" stroke="green" fill-opacity="0"
    stroke-width="1" stroke-dasharray="5 5"/>
  <circle r="50" cx="125" cy="125" stroke="green" fill-opacity="0"
    stroke-width="5" stroke-opacity="0.5"/>
  <circle r="50" cx="150" cy="150" stroke="green" fill-opacity="0"
    stroke-width="20" stroke-opacity="0.2"/>
</svg>
```



Outros atributos de fill e stroke

- **fill-rule="nonzero | evenodd | inherit"**
 - O que acontece quando a figura tem mais de uma borda (ex: bordas internas)
- **stroke-linecap="butt | round | square | inherit"**
 - Como se apresentam as extremidades de uma linha
- **stroke-linejoin="miter | round | bevel | inherit" e stroke-miterlimit="valor"**
 - Como se apresentam os vértices de linhas, polígonos e figuras em geral
- **stroke-dashoffset="valor"**
 - posição onde começa o tracejado do contorno; esse atributo pode ser animado para criar o efeito de traços em movimento



Exercícios

- 1. Aplique folhas de estilo diferentes aos arquivos SVG e HTML com SVG localizados na pasta exemplos/01
 - Modifique algumas regras de estilo (ex: cores ou fontes) e veja o que acontece com a página e com o gráfico SVG