

# SVG 2 – ein (Er)Wartungsbericht

tekom-Jahrestagung 2014 – Stuttgart, 13. November

**Dr. Thomas Meinike**

Hochschule Merseburg | FB Informatik und Kommunikationssysteme

<http://www.iks.hs-merseburg.de/~meinike/>

[thomas.meinike@hs-merseburg.de](mailto:thomas.meinike@hs-merseburg.de)

# SVG – Einstieg

→ Scalable Vector Graphics, ein XML-basiertes 2D-Vektorgrafikformat

→ Existiert bereits 15 Jahre:

- 1999 – Pressemitteilung von Adobe / Übergabe an das W3C zur Standardisierung
- 2001 – Spezifikation 1.0
- 2003 – Spezifikation 1.1 mit den Profilen Tiny, Basic und Full
- 2008 – Spezifikation 1.2 Tiny für mobile Anwendungen
- 2011 – Spezifikation 1.1 Second Edition (Fehlerbereinigungen)
- 2012 – Arbeitsgruppe startete die Weiterentwicklung zur Version 2(.0)
- 2014 – 4. Working Draft im Februar, aktueller Editor's Draft im Oktober erschienen:

## Scalable Vector Graphics (SVG) 2

W3C Working Draft 11 February 2014

**This version:**

<http://www.w3.org/TR/2014/WD-SVG2-20140211/>

## Scalable Vector Graphics (SVG) 2

W3C Editor's Draft 10 October 2014

**This version:**

<https://svgwg.org/svg2-draft/>

# SVG – Grundlagen kompakt

## Objekte und Effekte in SVG

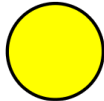
[Der rote Kreis, die Erklärungstexte und die Textlinks sind mit JavaScript-Funktionen verknüpft.]



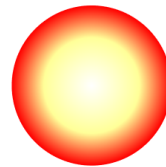
Rechteck



linearer Gradient



Kreis



radialer Gradient



Gruppe+Transformation



Ellipse



Opazität (Durchlässigkeit)



externes Bild



Polygon



Polylinie

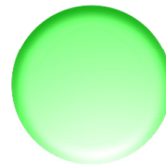


Animation

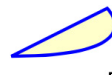
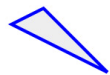


Linie

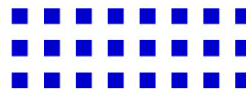
Spezialfilter



<http://www.datenverdrahten.de> Textlink



Pfad



Muster

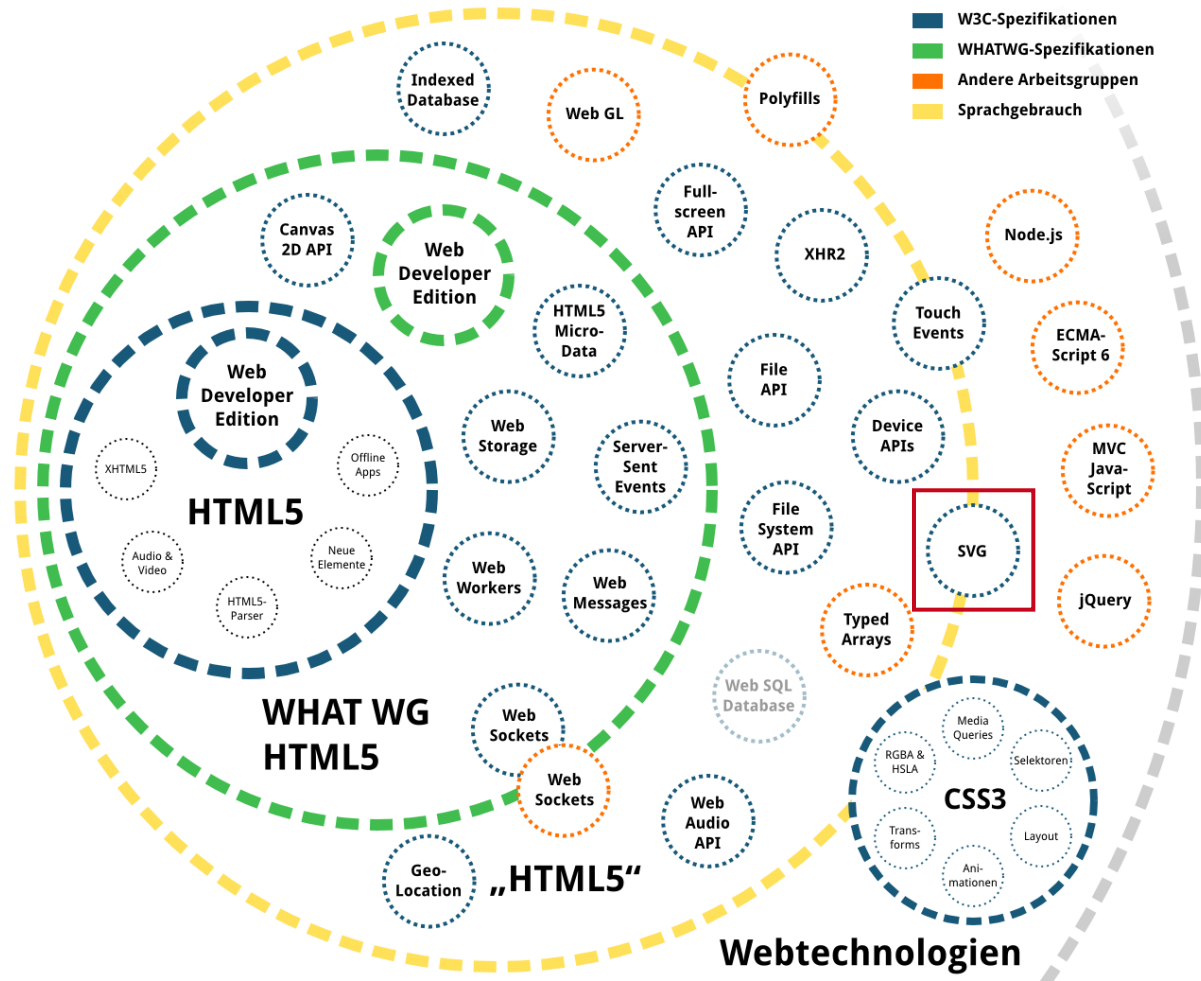
normaler Fließtext

© by Dr. Thomas Meinike 2002

TMs10kSVGDemo.svg

Darstellung unter iOS 8 – Quelle (Autor):  
[svgjbc.datenverdrahten.de/svg/TMs10kSVGDemo.htm](http://svgjbc.datenverdrahten.de/svg/TMs10kSVGDemo.htm)

# SVG – Teil des HTML5-Universums

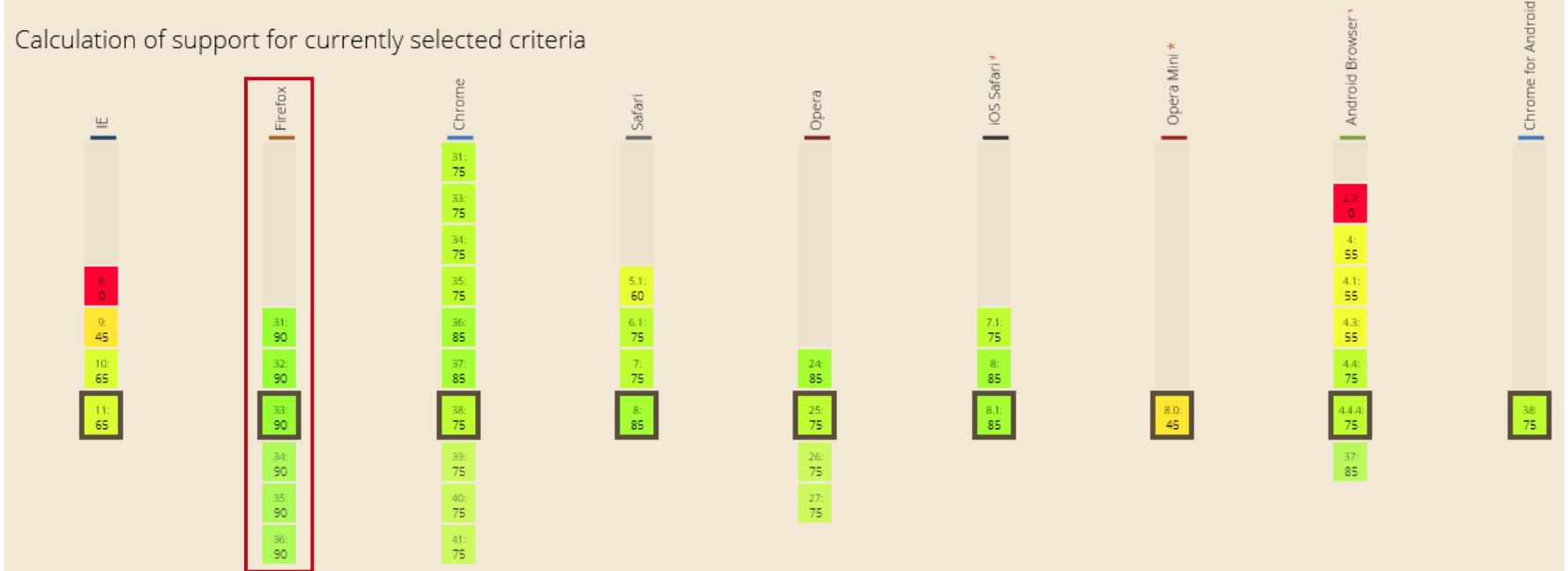


Quelle:  
[peterkroener.de](http://peterkroener.de)

# SVG – Browser-Unterstützung → gut++

## Summary

Calculation of support for currently selected criteria



**Firefox 33**  
Oktober 2014

Quelle: [caniuse.com/#cats=SVG](http://caniuse.com/#cats=SVG)  
Alternative: [codedread.com](http://codedread.com) (bis 2011)

# SVG 2 – Recherche (1)

→ Studium der veröffentlichten Entwürfe:

[www.w3.org/TR/SVG2/](http://www.w3.org/TR/SVG2/) und [svgwg.org/svg2-draft/](http://svgwg.org/svg2-draft/)



**Me**  
@estellew

 Folgen

Never follow a TR spec when there's an Editor's Draft. The TR spec is an outdated Editor's Draft.



RETWEETS

2

FAVORIT

1



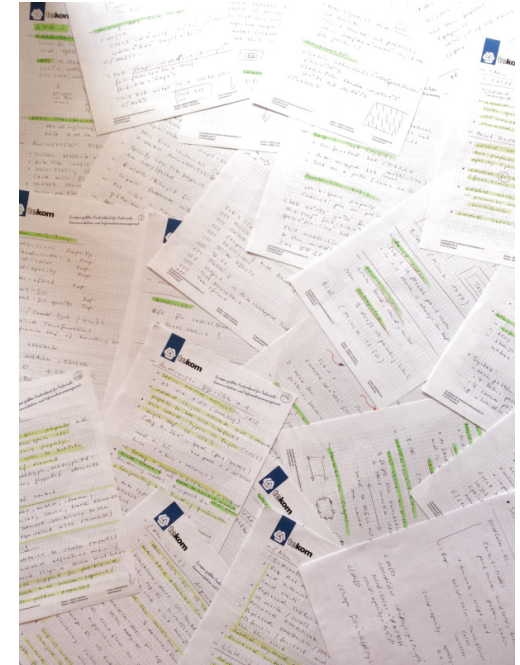
12:30 - 3. Nov. 2014

→ Material ist seit August 2014 auch auf GitHub verfügbar:

[github.com/w3c/svgwg](https://github.com/w3c/svgwg)

# SVG 2 – Recherche (2)

- ➔ Informationen im W3C-Wiki zur Arbeitsgruppe:  
[www.w3.org/Graphics/SVG/WG/wiki/Main\\_Page](http://www.w3.org/Graphics/SVG/WG/wiki/Main_Page)
- ➔ Öffentliche Protokolle von Telekonferenzen:  
[www.w3.org/Graphics/SVG/WG/wiki/Agenda](http://www.w3.org/Graphics/SVG/WG/wiki/Agenda)
- ➔ Verfolgen von (privaten) Mitteilungen:
  - Blogs
  - Twitter
  - Google+
  - Konferenzbeiträge



## Editors:

Nikos Andronikos, Canon, Inc. <[nikos.andronikos@cisra.canon.com.au](mailto:nikos.andronikos@cisra.canon.com.au)>  
Tavmjong Bah, Invited Expert <[tavmjong@free.fr](mailto:tavmjong@free.fr)>  
Brian Birtles, Mozilla Japan <[bbirtles@mozilla.com](mailto:bbirtles@mozilla.com)>  
Cyril Concolato, Telecom ParisTech <[cyril.concolato@telecom-paristech.fr](mailto:cyril.concolato@telecom-paristech.fr)>  
Erik Dahlström, Opera Software <[ed@opera.com](mailto:ed@opera.com)>  
Chris Lilley, W3C <[chris@w3.org](mailto:chris@w3.org)>  
Cameron McCormack, Mozilla Corporation <[cam@mcc.id.au](mailto:cam@mcc.id.au)>  
Doug Schepers, W3C <[schepers@w3.org](mailto:schepers@w3.org)>  
Dirk Schulze, Adobe Systems <[dschulze@adobe.com](mailto:dschulze@adobe.com)>  
Richard Schwerdfeger, IBM <[schwer@us.ibm.com](mailto:schwer@us.ibm.com)>  
Satoru Takagi, KDDI Corporation <[sa-takagi@kddi.com](mailto:sa-takagi@kddi.com)>  
Jonathan Watt, Mozilla Corporation <[jwatt@jwatt.org](mailto:jwatt@jwatt.org)>

# SVG 2 – Allgemeines (1)

- Keine DOCTYPE-Deklaration mehr! Ob und welche Datenmodelle es für Entwickler geben wird, ist noch nicht bekannt (RELAX NG?).
- Bisheriger Namensraum <http://www.w3.org/2000/svg> bleibt erhalten.
- CSS3-Spezifikationen werden integriert (u. a. Compositing and Blending, Filter Effects, Masking, Shapes, Transformations).
- DOM-Interfaces werden überarbeitet, das `script`-Element selbst erhält eine größere Reichweite, auch außerhalb des `defs`-Elements nutzbar.
- Universalattribute `tabindex` (Fokussierung) und `z-index` (Überlagerung).
- Glyph\*-Elemente werden zugunsten von WOFF-Schriftarten entfernt.

# SVG 2 – Allgemeines (2)

- **Layout Properties** beschreiben Position und Dimension von grafischen Objekten und stehen somit auch als CSS-Eigenschaften zur Verfügung: [svgwg.org/svg2-draft/layout.html](http://svgwg.org/svg2-draft/layout.html)

Zuerst auf Twitter gesehen, später im Editor's Draft enthalten:



 **Dirk Schulze**  
@dirkschulze Folgen

x, y, width and height are **#SVG** "presentation attributes" in **@webkit** now. Means you can set them with CSS. r, cx, cy, rx, ry to follow.

Antworten Retweeten Favorisieren Mehr

RETWEETS  
**2**

23:59 - 14. Aug. 2014

## Chapter 7: SVG Layout Properties

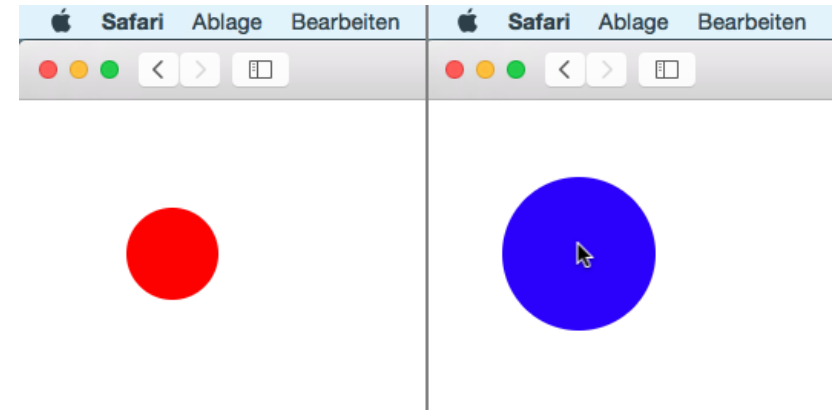
### Contents

- 7.1. Horizontal center coordinate: The 'cx' property
- 7.2. Vertical center coordinate: The 'cy' property
- 7.3. Radius: The 'r' property
- 7.4. Horizontal radius: The 'rx' property
- 7.5. Vertical radius: The 'ry' property
- 7.6. Horizontal coordinate: The 'x' property
- 7.7. Vertical coordinate: The 'y' property

# SVG 2 – Allgemeines (3)

➔ Kreis mit `circle`-Element + CSS konstruiert (funktioniert mit WebKit-Nightly-Builds unter Mac OS X seit Mitte August | [nightly.webkit.org](http://nightly.webkit.org)):

```
<?xml version="1.0" encoding="UTF-8"?>
<svg xmlns="http://www.w3.org/2000/svg">
  <defs>
    <style type="text/css">
      #kreis1 { cx: 100px; cy: 100px; r: 30px; fill: #F00; }
      #kreis1:hover { r: 50px; fill: #00F; }
    </style>
  </defs>
  <circle id="kreis1"/>
</svg>
```



# SVG 2 – Audiovisuelle Medien

- Die Elemente `audio` und `video` werden gemäß der HTML5-Spezifikation (W3C Rec. vom 28.10.2014) in den Funktionsumfang einbezogen:

```
<audio src="sound.mp3" type="audio/mpeg" controls="controls">
  <!-- Fallback: Alternativinhalt / Download ... -->
</audio>
```

```
<audio controls="controls">
  <source src="sound.mp3" type="audio/mpeg"/>
  <source src="sound.ogg" type="audio/ogg"/>
  <source src="sound.wav" type="audio/x-wav"/>
  <!-- Fallback: Alternativinhalt / Download ... -->
</audio>
```

```
<video src="film.mp4" width="640" height="480" type="video/mpeg">
  <!-- ggf. weitere Format-Alternativen mit source-Element analog zu audio und Fallback ... -->
</video>
```

Weitere Attribute möglich: `autobuffer`, `autoplay`, `loop`, `poster`

# SVG 2 – Bilder

➔ Mit `image` eingebundene Rasterbilder benötigten bisher Angaben zu `width` und `height`, diese Attribute durften nicht leer sein oder fehlen. Künftig können beide Attribute alternativ den Wert **auto** annehmen, der jeweils andere Wert wird automatisch (proportional) angepasst.

➔ Beispiel (`#tekomp11`):  
Cover einer Diskografie mit festen Breiten (hier 300px) und variablen Höhen – CDs vs. DVDs

Praktisch: Bei Zuweisung über JavaScript `width` bzw. `height` als Konstanten vorgebar.

## Diskografie / Einstürzende Neubauten

- 2014 • Lament (CD)
- 2012 • Live at Rockpalast 1990 (DVD)
- 2010 • 3 Decades Live – Paris, Cité De La Musique, 16.11.2010 (USB-Stick) ××× Strategies Against Architecture IV (2002 – 2010) (DCD)
- 2008 • The Jewels (CD)
- 2007 • Alles wieder offen (CD) ××× Weil Weil Weil (SCD)
- 2006 • Palast der Republik (DVD)
- 2005 • 25th Anniversary Tour 2005 (Budapest, March 31st, 2005) (CD) ××× Grundstück (CD)
- 2004 • Kalte Sterne Early Recordings (CD) ××× Perpetuum Mobile (CD)
- 2002 • 9-15-2000 Brussels (Live) (CD)
- 2001 • Berlin Babylon Soundtrack (CD) ××× Strategies Against Architecture III (1991 – 2001) (DCD)
- 2000 • Silence is Sexy (CD) ××× Supporters' Album #1 (CD)
- 1999 • Total Eclipse of the Sun (SCD)
- 1996 • Ende Neu (CD)
- 1995 • Faustmusik (CD)
- 1993 • Interim (SCD) ××× Malediction (SCD) ××× Tabula Rasa (CD)
- 1992 • Die Hamletmaschine (CD)
- 1991 • Strategies Against Architecture II (1984 – 1990) (DLP)
- 1990 • Feurio! (SCD)
- 1989 • Haus der Lüge (LP)
- 1987 • Fünf auf der nach oben offenen Richterskala (LP)
- 1985 • Halber Mensch (LP)
- 1984 • Strategies Against Architecture (1980 – 1983) (LP)
- 1983 • Zeichnungen des Patienten O.T. (LP)
- 1981 • Kollaps (LP)



# SVG 2 – Farben und Füllungen (1)

- Bisher nur Hex- und RGB-Farbwerte sowie Farbwörter, nun weitere Notationen aus CSS3 nutzbar: RGBA, HSL, HSLA, LAB, LABA (A für den Alpha-Kanal = 0 ... 1)
- Beispiel **rot** in HSL/A (Hue in Grad, Saturation in %, Luminance in %): *hsl(0, 100%, 50%)* bzw. *hsla(0, 100%, 50%, 1)*

## RGB-Umrechnung nach HSL

Theorie:

[en.wikipedia.org/wiki/HSL\\_and\\_HSV](http://en.wikipedia.org/wiki/HSL_and_HSV)

[en.wikipedia.org/wiki/Lab\\_color\\_space](http://en.wikipedia.org/wiki/Lab_color_space)


Eingabe-Daten					
R	<input type="text" value="255"/>	G	<input type="text" value="0"/>	B	<input type="text" value="0"/>
Berechnung					
		HSL	<input type="text" value="hsl(0, 100%, 50%)"/>		

- Einsetzbar für typische Farb-Attribute und CSS-Eigenschaften: **fill**, **stroke**, **solid-color** (neu), **stop-color**, **flood-color**, **lighting-color**

# SVG 2 – Farben und Füllungen (2)

- Eigene, benannte und somit wiederverwendbare Farbdeklarationen mit dem Element `solidColor` und zusätzlichen Attributen definieren (übernommen aus SVG 1.2 Tiny):

```
<?xml version="1.0" encoding="UTF-8"?>  
  
<svg xmlns="http://www.w3.org/2000/svg">  
  <defs>  
    <solidColor id="myblue" solid-color="#00F" solid-opacity="0.5"/>  
  </defs>  
  <circle cx="100" cy="100" r="20" fill="url(#myblue)"/>  
</svg>
```

- Das *sollte* so aussehen: 

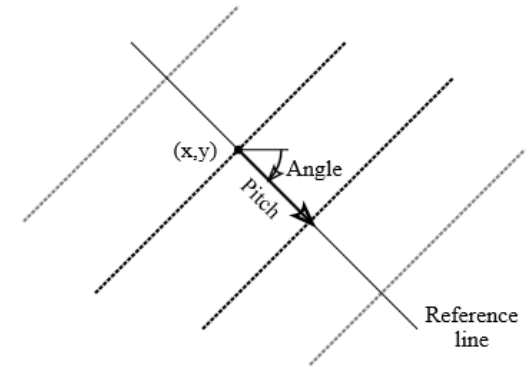
*simuliert mit `rgba(0, 0, 255, 0.5)`*

# SVG 2 – Farben und Füllungen (3)

→ Neue Elemente `hatch` / `hatchPath` für schraffierte Flächen mit höherer Genauigkeit gegenüber `pattern`-Techniken (u. a. für Gravuren):

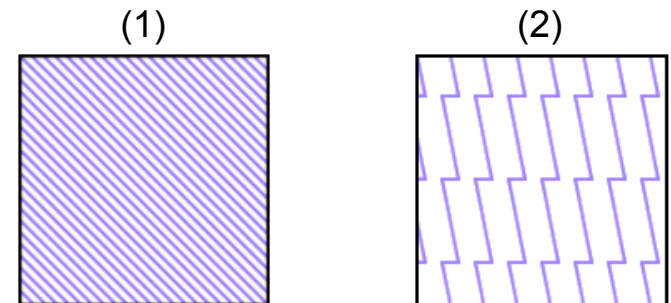
Beispiel aus der Spezifikation (1):

```
<hatch hatchUnits="userSpaceOnUse" pitch="5" rotate="135">  
  <hatchPath stroke="#a080ff" stroke-width="2"/>  
</hatch>
```



Beispiel aus der Spezifikation (2):

```
<hatch hatchUnits="userSpaceOnUse" pitch="20">  
  <hatchPath stroke-width="2" d="L 0,0 10,50"/>  
</hatch>
```



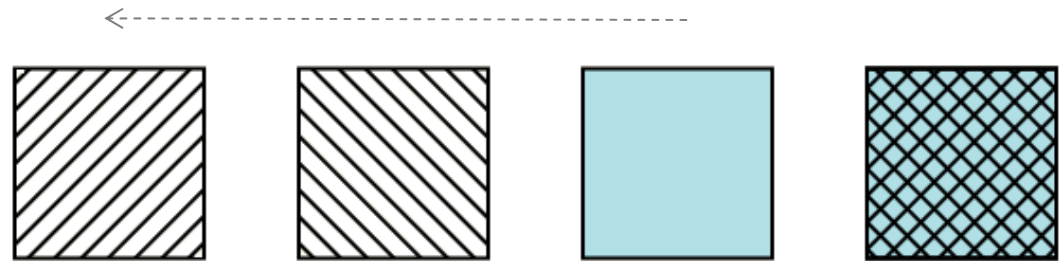
»The 'd' path data describes two line segments, the first starting at (10, 0). The repeating sections are joined.«

# SVG 2 – Farben und Füllungen (4)

→ Es wird möglich, mehrere **fill**-Angaben zu überlagern:

Beispiel aus der Spezifikation:

```
<rect width="100" height="100" fill="url(#MyHatch1), url(#MyHatch2), powderblue"/>
```



Ggf. im Code fehlende Angaben (**#MyHatch2**) werden nicht gerendert:



# SVG 2 – Filtereffekte

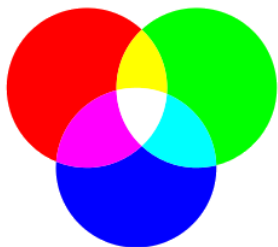
→ Separate Spezifikation *Compositing and Blending* beschreibt Möglichkeiten für Überblendungen, u. a. Blend-Modi:  
[dev.w3.org/fxtf/compositing-1/#ltblendmodegt](http://dev.w3.org/fxtf/compositing-1/#ltblendmodegt)

Beispiel aus der Spezifikation:

```
<circle cx="40" cy="40" r="40" fill="red"/>
```

```
<circle cx="80" cy="40" r="40" fill="green"/>
```

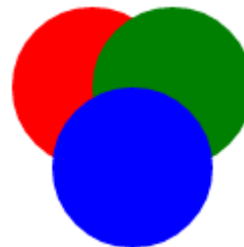
```
<circle cx="60" cy="80" r="40" fill="blue"/>
```



Spezifikation



Firefox 33



Opera Dev. 27

+ CSS:

```
circle { mix-blend-mode: screen; }
```

color	lighten
color-burn	luminosity
color-dodge	multiply
darken	normal
difference	overlay
exclusion	saturation
hard-light	screen
hue	soft-light

# SVG 2 – Gradienten (1)

→ Neues Attribut **fr** (focal radius) für radiale Gradienten:

```
<radialGradient id="radgra1" fx="0.7" fy="0.7"> <!-- im defs-Bereich -->
```

```
<stop offset="0%" stop-color="#FF0"/>
```

```
<stop offset="60%" stop-color="#FFF"/>
```

```
<stop offset="100%" stop-color="#00F"/>
```

```
</radialGradient>
```

```
<radialGradient id="radgra2" fx="0.7" fy="0.7" fr="0.2"> <!-- bzw. fr="20%" -->
```

```
<stop offset="0%" stop-color="#FF0"/>
```

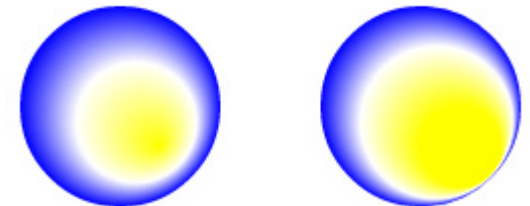
```
<stop offset="60%" stop-color="#FFF"/>
```

```
<stop offset="100%" stop-color="#00F"/>
```

```
</radialGradient>
```

```
<circle cx="100" cy="100" r="50" fill="url(#radgra1)"/>
```

```
<circle cx="250" cy="100" r="50" fill="url(#radgra2)"/>
```



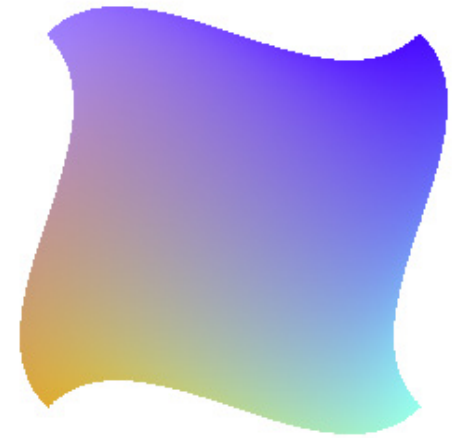
Opera Dev. 27

# SVG 2 – Gradienten (2)

→ Mesh-Gradienten (komplexeste Ergänzung):

»New in SVG 2. Added to allow *shadings along curved lines*. This is needed, for example, in *creating life-like drawings*.

The mesh gradients in SVG are based on an array of *Coons Patches*. A Coons Patch is a shading defined by colors place at the corners of an area enclosed by four *Bézier curves*.«  
(Editor's Draft, Oktober 2014)



**Figure 83.** A single Coons-Mesh patch.

# SVG 2 – Gradienten (3)

→ Mesh-Gradienten (Code mit neuen Elementen aus der Spezifikation):

```
<meshGradient x="100" y="100">
```

```
<meshRow>
```

```
<meshPatch>
```

```
<stop ... /> <!-- Up to four stops in first patch. -->
```

```
</meshPatch>
```

```
<meshPatch>
```

```
<!-- Any number of meshPatches in row. -->
```

```
</meshPatch>
```

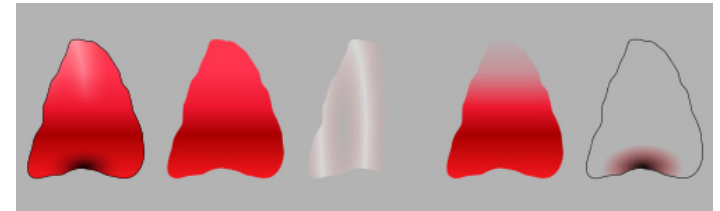
```
</meshRow>
```

```
<meshRow>
```

```
<!-- Any number of meshRows, each with the same number of meshPatches as in the first row. -->
```

```
</meshRow>
```

```
</meshGradient>
```

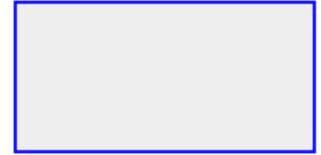


→Tavmjong Bah (SVG WG und Inkscape-Entwickler): [tavmjong.free.fr/blog/](http://tavmjong.free.fr/blog/)

# SVG 2 – Linien und Umrandungen (1)

- Größenänderungen von Rahmenlinien beim Zoomen vermeiden:  
**vector-effect** (normal | non-scaling-stroke)

```
<rect ... stroke="#00F" stroke-width="2px" vector-effect="non-scaling-stroke"/>
```



- Steuerung von Rahmenlinien mit **stroke-dashcorner** (none | <length>) und **stroke-dashadjust** (none || [stretch | compress] [dashes | gaps])



Details siehe Spezifikation: Stroke properties

- Neuer Wert arcs für **stroke-linejoin**:



'miter' join



'round' join



'bevel' join



'arcs' join

# SVG 2 – Linien und Umrandungen (2)

→ Zeichenreihenfolge von Füllungen, Umrandungen, Markern mit der Eigenschaft **paint-order** steuern (normal || fill | stroke | markers):

**normal** = fill stroke markers | **stroke** = stroke fill markers

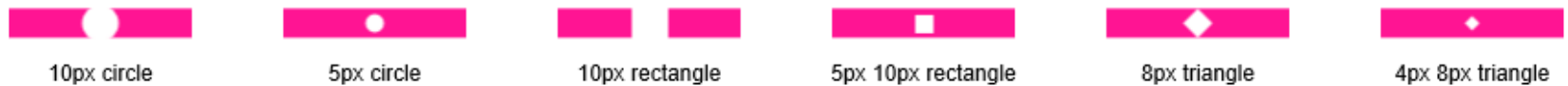
```
<text ... fill="#FF0" stroke="#00F" stroke-width="3px" paint-order="stroke">Beispieltext</text>
```

u. a. Firefox 33 und Opera Dev. 27

Beispieltext  
Beispieltext

→ Marker-Erweiterungen: **marker-knockout-left** + **marker-knockout-right**

Werte: <knockout-offset> | <knockout-shape> [ at <knockout-offset> ]



Details siehe Spezifikation: Markers | Auch neu: Repeating markers – the **marker-pattern** property

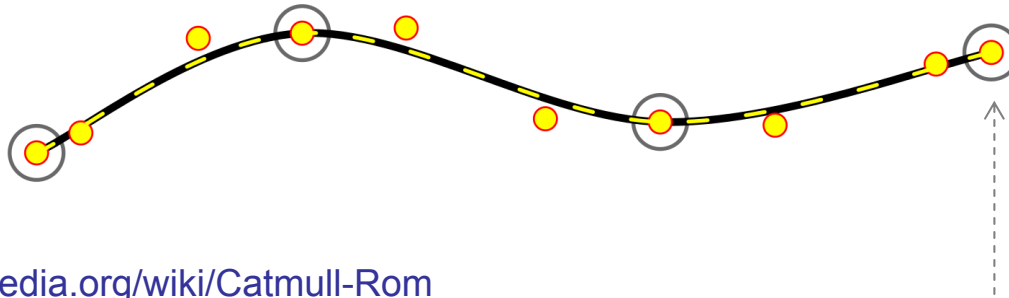
# SVG 2 – Pfaderweiterungen

- Neuer **d**-Parameter **B/r** (bearing), z. B. für reguläre Polygone  
Beispiel aus der Spezifikation (Werte sind **Winkel**):

```
<path fill="#eee" stroke="deeppink" stroke-width="8px" stroke-linejoin="round"  
      d="M 150,10 B 36 h 47 b 72 h 47 b 72 h 47 b 72 h 47 z"/>
```



- Neuer **d**-Parameter **R/r** für Catmull-Rom-Splinekurven  
(Werte sind die Kontrollpunkte der C-R-Kurve **x1 y1 x2 y2 (x y)+**):



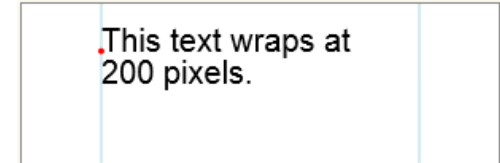
Theorie: [en.wikipedia.org/wiki/Catmull-Rom](https://en.wikipedia.org/wiki/Catmull-Rom)

Simulation mit JavaScript von Doug Schepers (SVG WG): [schepers.cc/getting-to-the-point](https://schepers.cc/getting-to-the-point)

# SVG 2 – Textdarstellung

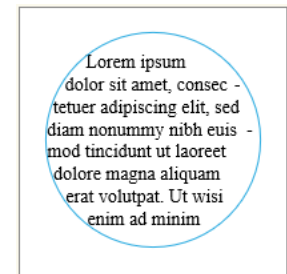
→ Textfluss wird über die Attribute **width** und/oder **height** möglich:

```
<text x="50" y="30" width="200" font-family="sans-serif"
      font-size="20px">This text wraps at 200 pixels.</text>
```



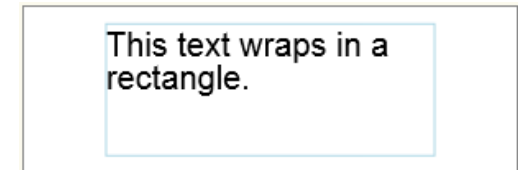
→ Text in Formen mit **shape-inside** (CSS-Shapes):

```
<text shape-inside="circle(100 at 100 100)" font-family="sans-serif"
      font-size="20px">Lorem ipsum ...</text>
```



→ Text in beliebigen Formen über ID-Referenz:

```
<text shape-inside="url(#wrap)" font-family="sans-serif"
      font-size="20px">This text wraps in a rectangle.</text>
```



(Beispiele aus der Spezifikation)

→ Weitere Eigenschaften: **shape-outside** (Bereiche ausschließen), **shape-margin** / **shape-padding** (Abstände), **white-space** (Leerraum)

# Fazit und Ausblick

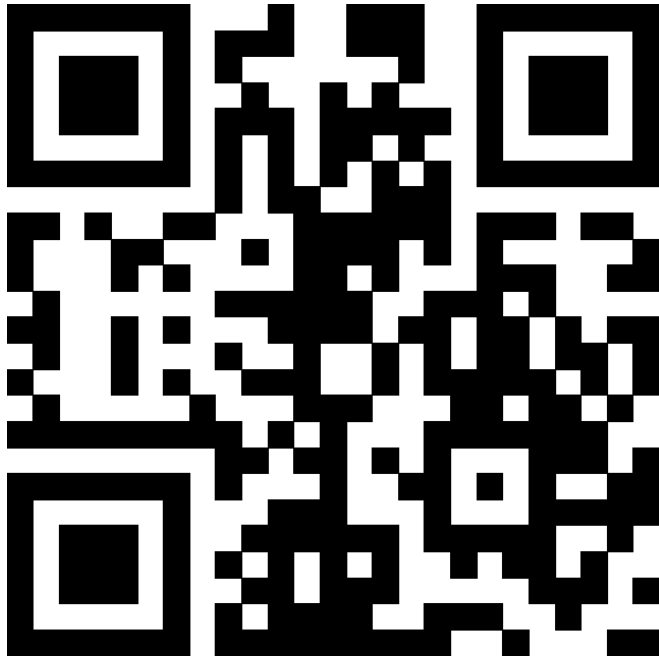
- ➔ SVG 2 baut auf den bestehenden Spezifikation auf, konsolidiert und erweitert den Funktionsumfang mit nützlichen Techniken.
- ➔ Vor allem Erweiterungen zur Darstellung von Linien und Pfaden sowie Texten bieten Potenzial für typische Anwendungen in der Technischen Kommunikation (z. B. Ersatzteilkataloge).
- ➔ Aktuell lässt sich noch sehr wenig ausprobieren. Insofern bleibt zu hoffen, dass die Entwickler von Browsern und Erstellungswerkzeugen die finale Spezifikation (2015?) zeitnah unterstützen.

# Feedback erwünscht ...

→ URL direkt aufrufen (auch nach der Tagung möglich):

[OTS2.honestly.de](https://OTS2.honestly.de)

→ Oder QR-Code scannen:



**Danke für Ihr Interesse!**