

Формат SVG в HTML5
Минакова Е.А., гр. 11-ТБ
рук. Абашин В.Г.

SVG (Scalable Vector Graphics standard, стандарт масштабируемой векторной графики) — формат на основе XML как средство обеспечения векторной графической поддержки для Web-браузеров.

Технология SVG позволяет объединить в одном формате текст, графику, анимацию и интерактивные компоненты и базируется на трех типах графических изображений: векторных формах, рисунках и тексте. Формы, как это принято в векторной графике, представлены либо прямолинейными и криволинейными контурами, либо графическими примитивами (прямоугольниками, эллипсами и др.), а рисунки представляют собой импортированные растровые изображения. Помимо этого формат SVG поддерживает различные виды анимационных (напоминающих GIF- и flash-анимацию) и интерактивных объектов, таких как ролловеры, карты ссылок и прочие элементы навигации.

SVG допускает использование CSS, что дает возможность манипулировать визуализацией данных очень легко. Он имеет важное преимущество перед растровыми форматами в том, что изображение в SVG может быть изящно преобразовано до большего или меньшего масштаба.

Поскольку этот стандарт основан на XML, SVG файл может содержать не только элементы, предназначенные для визуального представления, но и метаданные, предназначенные для описания альтернативного текстового представления изображения. В настоящее время при создании Web-страниц в качестве текстовой альтернативы рисунков используется свойство alt тега . В отличие от этого, SVG рисунки могут содержать метаданные (т.е. данные о данных) в самых разных форматах - таких как Resource Description Framework (RDF), Meta Content Framework (MCF) и других. Другая особенность SVG формата заключается в поддержке людей с ограниченными возможностями.

SVG предоставляет все преимущества XML:

- Возможность работы в различных средах.
- Интернационализация (поддержка Юникода).
- Широкая доступность для различных приложений.
- Лёгкая модификация через стандартные API — например, DOM. SVG

поддерживает стандартизированную W3C объектную модель документа DOM, обеспечивая доступ к любому элементу, что даёт широкие возможности по динамическому изменению элементов, их атрибутов и событий.

- Лёгкое преобразование таблицами стилей XSLT. Как любой основанный на XML формат, SVG дает возможность использовать для его обработки таблицы трансформации (XSLT). Преобразуя XML-данные в SVG с помощью простого XSL, можно легко получить графическое представление любых данных, например визуализировать химические молекулы, описанных на языке CML (Chemical Markup Language).

Стандарт SVG разработан и поддерживается консорциумом World Wide Web и в данный момент имеет статус рекомендации. Практически все графические редакторы позволяют работать с рисунками в формате SVG. Однако в Internet Explorer и Netscape Navigator эта поддержка реализована только на уровне plug-in'ов.

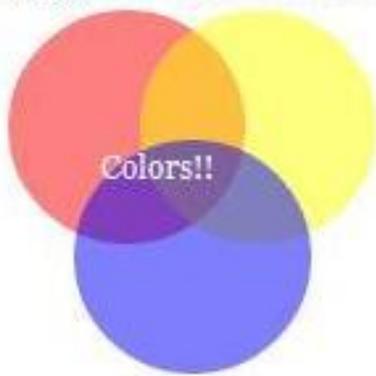
SVG формат поддерживают большое количество редакторов, начиная с таких распространенных, как Adobe Illustrator, Adobe Photoshop, Corel Draw, и кончая некоторыми специализированными редакторами. Главное преимущество нового формата заключается в создании средства обмена векторной графикой в текстовой форме, которую смогут понимать самые различные устройства.

В HTML5 SVG может встраиваться непосредственно в код HTML. Текущие веб-страницы могут быть обновлены для включения SVG с небольшим изменением общей структуры.

SVG неразрывно связан с форматом документа. Сегодня SVG используется в статических документах. Формат SVG подходит для сложных инженерных схем и иллюстраций, учитывая их требования к масштабируемости, высокому качеству печати и переносимости.

В HTML5 будущее SVG связано со следующим поколением интерактивной графики в Интернете, в которой SVG будет использоваться значительно шире.

Здесь можно вставить векторные изображения с помощью строчного HTML! Они наследуют стили CSS от родительских элементов, равно как и другие элементы HTML5.



```
<text x="40" y="70" fill="white">Colors!!</text>
</svg>
</p>
```

```
<!DOCTYPE html>
<p style="font-family:Georgia;font-size:9pt;">Здесь можно вставить векторные изображения с помощью строчного HTML! Они наследуют стили CSS от родительских элементов, равно как и другие элементы HTML5.
```

```
<BR>
<svg width="200" height="200">
<circle cx="50" cy="50" r="45" fill-opacity=".5" fill="red"/>
<circle cx="100" cy="50" r="45" fill-opacity=".5" fill="yellow"/>
<circle cx="75" cy="100" r="45" fill-opacity=".5" fill="blue"/>
```

Плюсы и минусы применения формата SVG.

К основным достоинствам графического формата SVG можно отнести следующие:

- высокое качество изображений независимо от их размеров, что объясняется векторной природой SVG-формата.;

- гораздо меньший размер файлов по сравнению с форматами GIF, JPG, PNG и animation GIF, а тем более с форматом Flash.

Для разработчиков важными плюсами являются:

- возможности совмещения в одном формате разработки статичных, анимационных и интерактивных элементов, а также сочетания векторных и растровых объектов;

- улучшенная работа с текстом, включая кернинг, текст по кривой и неограниченное использование шрифтов;

- более эффективное управление точностью передачи цветов и широчайшие возможности в плане использования в web-изображениях градиентных заливок высокого разрешения, теней, фильтров и т.п.;

- текстовая природа SVG-формата и поддержка им каскадных таблиц стилей, что значительно упрощает процесс обновления web-сайта и позволяет при необходимости вносить в него изменения без обращения к специальным программам;

- интеграция с построенными на стандартах XML (Extensible Markup Language) и CSS (Cascading Style Sheets) базами данных, что позволяет сохранять SVG-изображения в базе данных и создавать с их использованием динамические web-страницы — различные для разных платформ, персональных настроек и т.д.;

- отсутствие проблем индексации — SVG-файлы индексируются любыми поисковыми машинами.

Недостатки технологии SVG:

- SVG-изображения слабо поддерживаются производителями Интернет-браузеров. В результате для просмотра SVG-графики из браузера пользователи вынуждены дополнительно устанавливать обеспечивающий данную возможность плагин от сторонних производителей, например SVG Viewer от компании Adobe;

- SVG наследует все недостатки XML, такие как большой размер файла (впрочем, последний компенсируется существованием сжатого формата SVGZ).

- Сложность использования в крупных картографических приложениях из-за того, что для правильного отображения маленькой части изображения документ необходимо прочитать целиком.

- Чем больше в изображении мелких деталей, тем быстрее растёт размер SVG-данных. Предельный случай — когда изображение представляет собой белый шум. В этом случае SVG не только не даёт никаких преимуществ, но и даже обладает чрезмерно избыточным по отношению к растровому формату размером. На практике, SVG становится невыгоден уже задолго до того, как изображение дойдёт до стадии белого шума.

Преимущества формата SVG.

1. Масштабируемость. Формат SVG идеально подходит для карт, диаграмм и графиков. В формате SVG четкость графики поддерживается при любых разрешениях. Как правило, карты просматриваются как единое целое изображение.

2. Поддержка сценариев. Формат SVG поддерживается моделью DOM, что упрощает создание сценариев и использование JavaScript в элементах SVG. Поскольку фигуры SVG являются графическими объектами, родственными

элементам HTML, при создании сценариев SVG используются общие шаблоны. Элементы SVG могут принимать обработчики событий. Их изменяют с помощью модели DOM или методов `setAttribute()` и `getAttribute()` DOM L2 Core.

3. Поддержка стилей. Существует несколько способов добавления элемента SVG на веб-страницу. SVG-файл может быть указан в тегах `<embed>`, `<object>`, `<iframe>` и ``. Изображение SVG можно добавить с помощью CSS в качестве фонового изображения или изображения стиля списка. Элемент SVG рассматривается как один из элементов HTML, поэтому к нему применяются блоки CSS, которые используются для задания стиля всей страницы. Содержимое веб-сайтов, вместе со значками, эмблемами, элементами навигации и остальными графическими объектами SVG, поддерживает отдельные таблицы стилей. Для управления выводом изображения SVG на печать используются таблицы стилей печати.

4. Поддержка поиска. SVG — это формат, основанный на XML. Он состоит из элементов и атрибутов, которые удобны для восприятия человеком. Текст в SVG-файле отображается как текст в разметке. Текст в изображениях SVG доступен для индексирования поисковыми системами.

5. Доступность. Элементы `<title>` и `<description>` могут применяться к любому элементу SVG. Документ SVG имеет четкую и понятную структуру.

6. Декоративный текст

Формат SVG может использоваться на веб-сайтах с ресурсоемкой графикой. В тексте SVG можно использовать градиенты и шаблоны, он может содержать штриховку. Текст SVG переворачивают или располагают в требуемой плоскости.

7. Формат SVG, управляемый данными и создаваемый сервером.

SVG — это спецификация XML, которая позволяет использовать SVG в качестве формата изображений, создаваемых сервером. Благодаря простоте создания XML-кода SVG является форматом для создания информационных изображений.

8. Удобство работы с браузером. Встроенная поддержка формата обеспечивает удобство для пользователя и снижает риск потенциальных уязвимостей, возникающий при использовании надстройки. SVG обеспечивает тесную интеграцию с разметкой страницы, моделью DOM и сценариями для обеспечения совместимости программной модели и улучшения взаимодействия с пользователем.

9. Изучение и просмотр исходного кода.

Примеры кодов в HTML5 с тегом `<svg>`

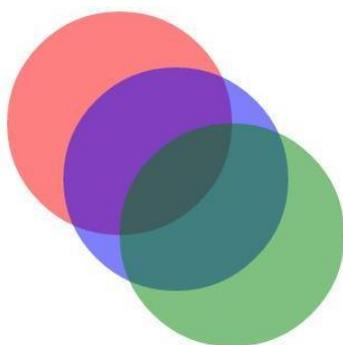


```
1. <?xml version="1.0" encoding="utf-8"?>
   <svg version="1.1" id="Layer_1"
        xmlns="http://www.w3.org/2000/svg"
        xmlns:xlink="http://www.w3.org/1999/xlink" x="0px" y="0px"
        width="100px" height="54.805px" viewBox="0 0 100 54.805" enable-
background="new 0 0 100 54.805" xml:space="preserve">
   <g id="rotator" transform="rotate(0)">
     <circle cx="14.466" cy="17.737" r="4.264" />
     <circle cx="77.027" cy="4.167" r="4.264" />
```

```

<path d="M 77.207 27.417 1 -1.003 4.827 1 4.959 1.267 1 4.274 -7.878 L 100 22.474
1 -1.143 -5.267 1 -5.006 1.085 c 1.573 -3.747 -1.045 -5.289 -1.045 -5.289 L 82.312
6.087 1 -4.217 3.005 1 -4.084 6.25 c 0 0 -1.974 3.023 -0.126 4.636 1 0.192 2.604 L
24.779 33.273 1 -0.903 -2.447 c 1.013 -2.233 -2.036 -4.167 -2.036 -4.167 1 -6.306 -
3.996 1 -5.082 -0.989 L 3.764 32.316 c 0 0 -1.747 2.488 1.239 5.246 L 0 38.648 1 1.142
5.269 1 14.562 -3.158 1 17.155 5.398 1 13.989 -3.207 1 -2.913 -3.978 L 77.207 27.417 Z M
20.043 34.301 1 -5.588 -2.679 1 1.379 -4.267 1 5.61 2.643 1 1.367 3.702 L 20.043 34.301
Z M 78.814 21.554 1 -2.771 0.602 1 -0.29 -3.938 1 4.014 -4.728 1 3.022 3.31 L 78.814
21.554 Z" />
</g>
<polygon points="49.999,33.319 44.807,37.859 44.807,54.804 50,54.804
55.193,54.804 55.193,37.859" />
</svg> </svg>

```



```

style="fill-opacity: 0.5" />
</svg>

```

```

2. <?xml version="1.0" encoding="utf-8"
standalone="yes"?>
<svg xmlns="http://www.w3.org/2000/svg" version="1.1"
height = "400px" width = "400px">
  <circle cx="100px" cy="100px" r="100px" fill="red"
style="fill-opacity: 0.5" />
  <circle cx="150px" cy="150px" r="100px" fill="blue"
style="fill-opacity: 0.5" />
  <circle cx="200px" cy="200px" r="100px" fill="green"

```

Литература

1. КомпьютерПресс [Текст] / Программное обеспечение // Обзор софта для работы с SVG графикой: ежемес. журн. – Молчанов Б.М.; под ред. Синева А.В. – М.: Москва, 2006, март № 3. — 128 полос. – 23-51 тыс. экз.
2. Изучаем HTML5. Библиотека специалистов [Текст] : компьютерная литература : [Web-разработки] / Б. Лоусон, Р. Шарп. – СПб.: Питер, 2011. – 272 с.
3. КомпьюАрт [Текст] / Веди // Традиционная и электронная типографика : ежемес. журн. – М.: Молчанов Б.М.; под ред. Синева А.В. – 2002, май.
4. SVG Essentials [Текст] / компьютерная литература : J. Eisenberg. – изд. O'Reilly Media, Inc . – 2002. – 364 с.
5. Designing SVG Web Graphics [Текст] / изд. New Riders Publishing. – Andrew H. Watt. – 2001, сент. – 592 с.