

Графика

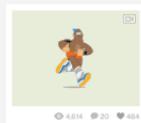
Симонова Екатерина, Рыбалка Екатерина

What are you working on? Dribbble is where designers get inspired and hired.

[Continue →](#)

Looking to hire a designer? [Learn more](#)

Popular ▾ Shots ▾ Now ▾



Markus Magnussen



Eddie Lobanovskiy



Sochnik



uixNinja



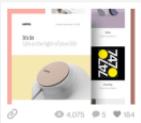
UBB



Outcrowd



UIGREAT



meno.



Dropbox Design



Focus Lab



Netguru



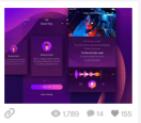
B&B



B&B



Glenn Thomas

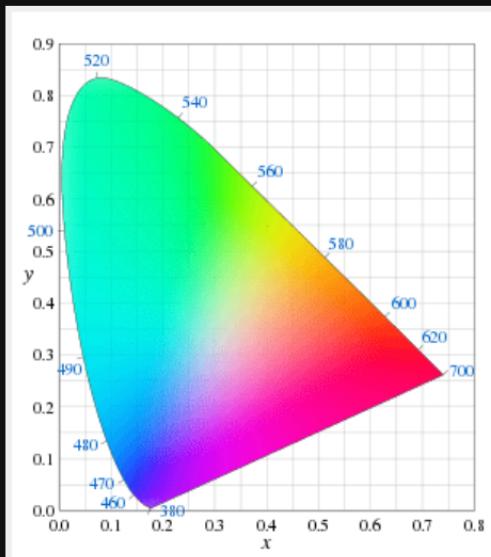


Radesign

Цветовые модели

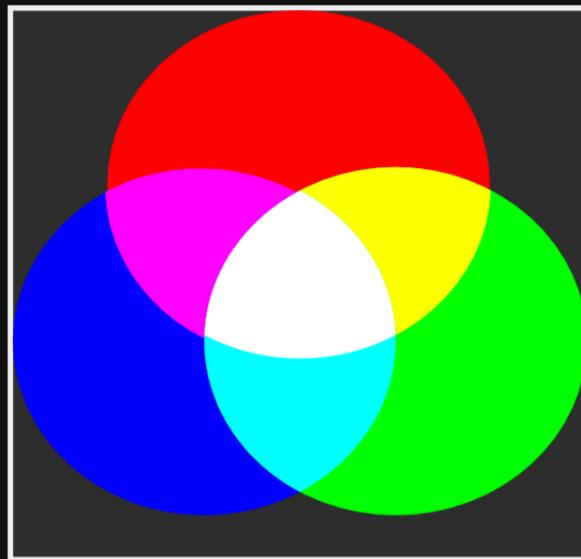
Цветовая модель

— математическая модель описания представления цветов в виде кортежей чисел (обычно из трёх, реже — четырёх значений).



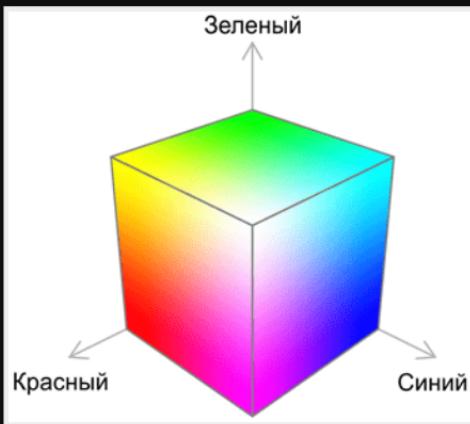
RGB

- кодирует любой цвет, представляя в виде комбинации трех цветов (каналов): красного (Red), зеленого (Green) и синего (Blue), с разной интенсивностью.



Интенсивность каждого из трех цветов – это один байт (т. е. число в диапазоне от 0 до 255):

- Белый цвет $r=255; g=255; b=255$
- Красный цвет $r=255; g=0; b=0$
- Желтый цвет $r=255; g=255; b=0$
- Черный цвет $r=0; g=0; b=0$



RGB в CSS

```
a:hover {  
    /*      rgb(red, green, blue); */  
    color: rgb(255, 0, 0);  
  
    /*color: rgb(100%, 0, 0);*/  
}
```

$$255_{10} = ff_{16}$$

```
a:hover {  
    /*Шестнадцатиричная запись*/  
    color: #ff0000;  
}
```

```
a:hover {  
    /*Короткая запись*/  
    color: #f00;  
}
```

RGBA

Похож по синтаксису на RGB, но включает в себя альфа-канал, задающий непрозрачность элемента (0 – прозрачный, 1 – непрозрачный).

```
red-box:hover {  
    background: rgba(255, 0, 0, .5);  
}
```

```
red-box:hover {  
    background: #ff000050;  
}
```



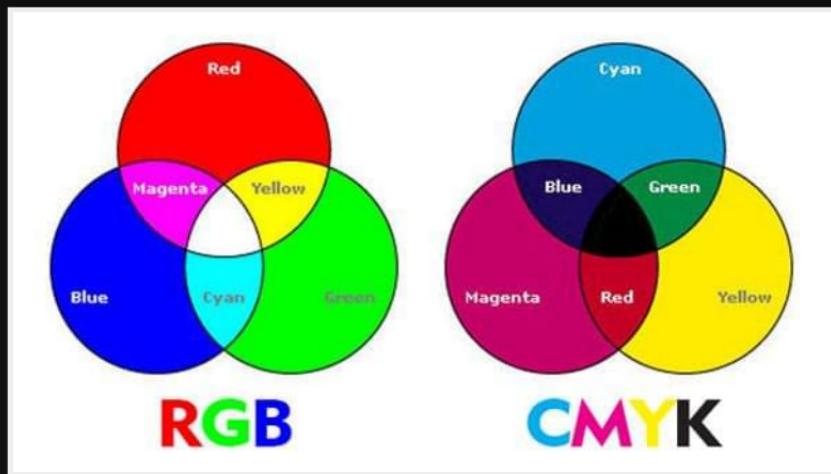
Имена цветов

В CSS существует 140 зафиксированных имен

- red
- black
- green
- darkmagenta
- orangered
- springgreen
- yellowgreen
- slategrey
- transparent

CMYK

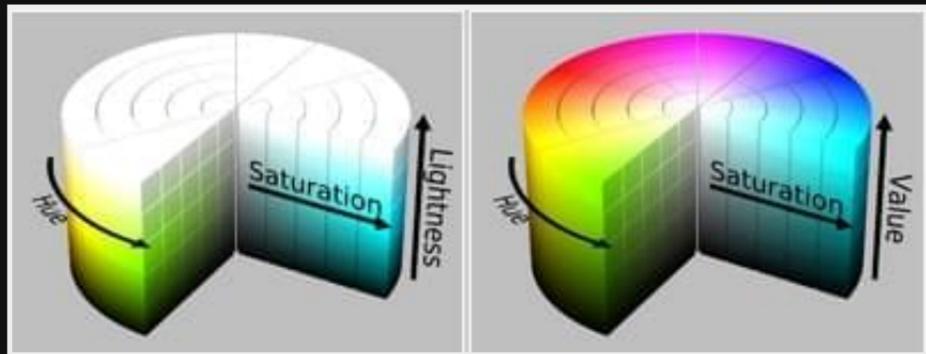
Данная модель является вычитающей, поскольку цвета в ней образуются путем вычитания из черного цвета базовых цветов: голубого (Cyan), пурпурного (Magenta), желтого (Yellow).



HSV

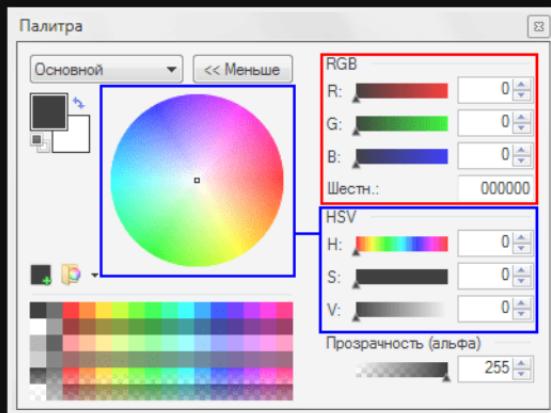
(Hue, Saturation, Value)

- Hue — цветовой тон (оттенок цвета, от 0 до 360).
- Saturation — насыщенность. Чем выше этот параметр, тем «чище» будет цвет, а чем ниже, тем ближе он будет к серому (0 – 100%).
- Value — яркость цвета. Чем выше значение, тем ярче будет цвет. А чем ниже, тем темнее (0% — черный, 100% – яркий)



HSV

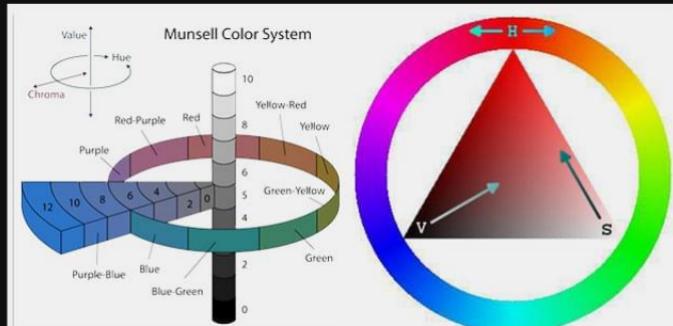
Чаще всего используется в простой обработке изображений.



HSL

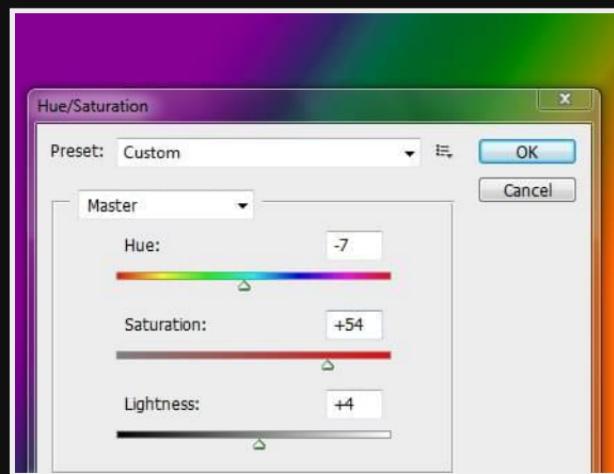
(Hue, Saturation, Lightness)

- Hue — цветовой тон (оттенок цвета, от 0 до 360).
- Saturation — насыщенность. Чем выше этот параметр, тем «чище» будет цвет, а чем ниже, тем ближе он будет к серому (0 – 100%).
- Lightness — это светлота цвета (не путать с яркостью). Чем выше параметр, тем светлее цвет (100% — белый), а чем ниже, тем темнее (0% — черный).



HSL

Преимуществом HSL перед RGB является то, что HSL является гораздо более понятным. Кроме того, с помощью него легче создать палитры подходящих цветов (сохраняя оттенок и меняя яркость и насыщенность)



HSL B CSS

```
.span-1 {  
    color: hsl(0, 0%, 100%);  
}  
.span-2 {  
    color: hsl(0, 100%, 50%);  
    background-color: hsl(120, 100%, 50%);  
}  
.span-3 {  
    color: hsl(30, 100%, 0%);  
    background-color: hsl(60, 100%, 100%);  
}
```

span-1
span-2
span-3

EXAMPLE

HSLA в CSS

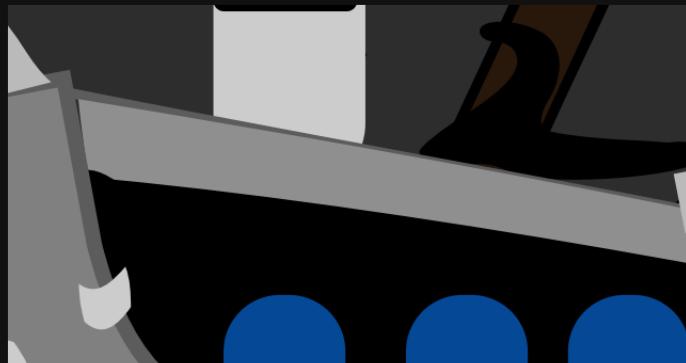
Формат HSLA похож по синтаксису на HSL, но включает в себя альфа-канал, задающий непрозрачность элемента.

```
.box {  
    background-color: hsla(120, 100%, 50%, 1);  
}  
.box:hover {  
    background-color: hsla(120, 100%, 50%, .5);  
}
```



Виды графики

Векторная графика



Растровая графика

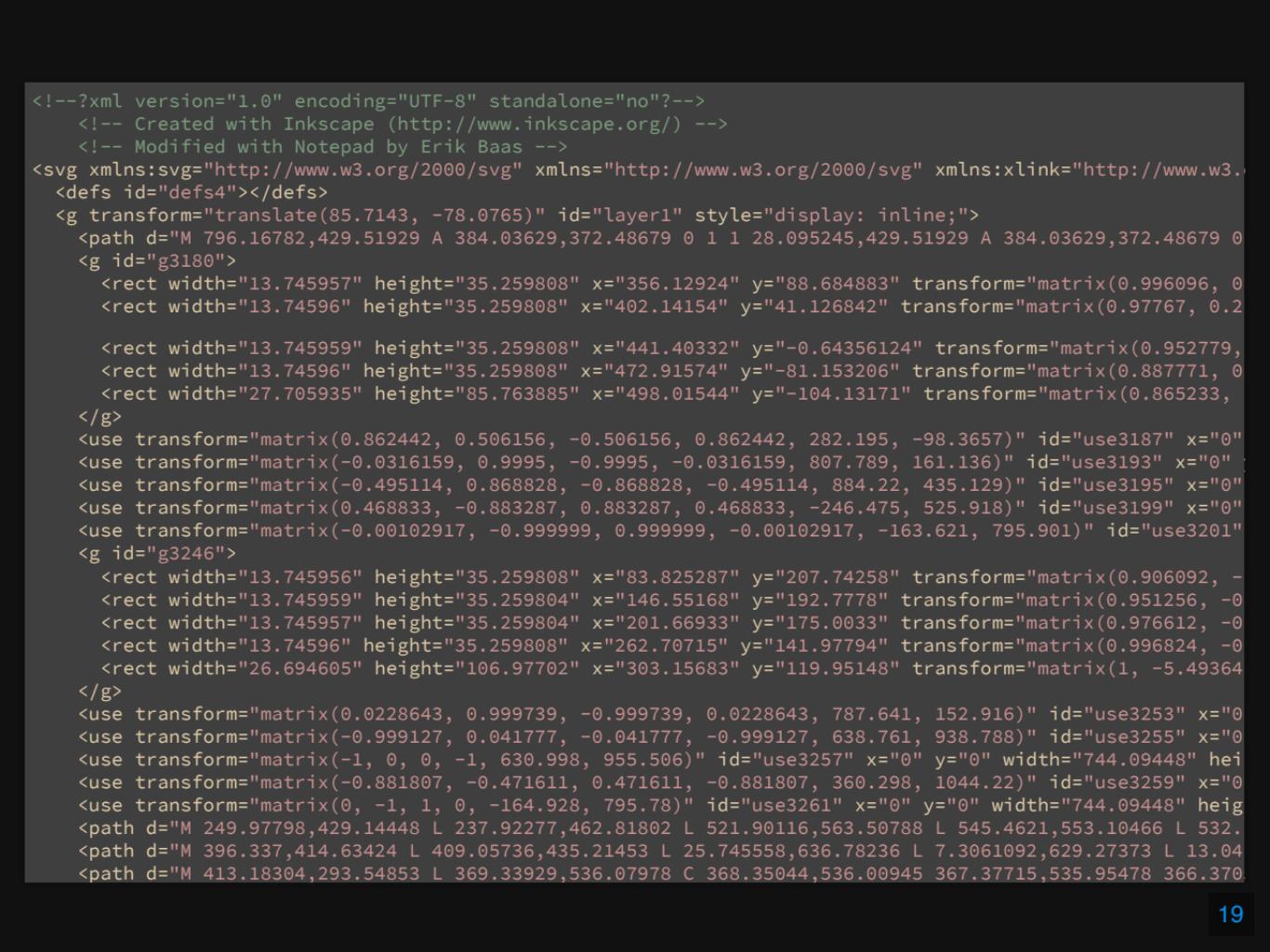
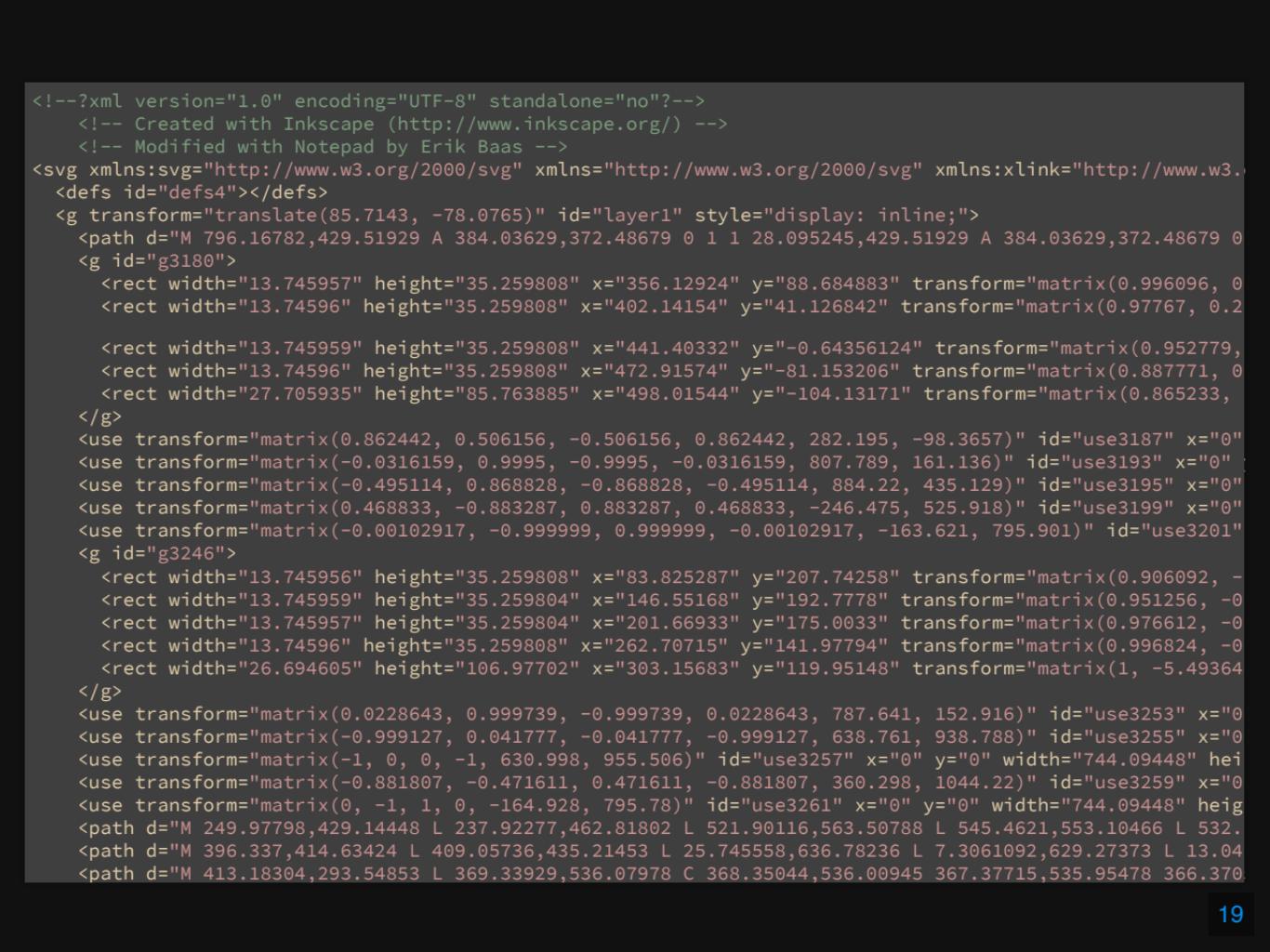
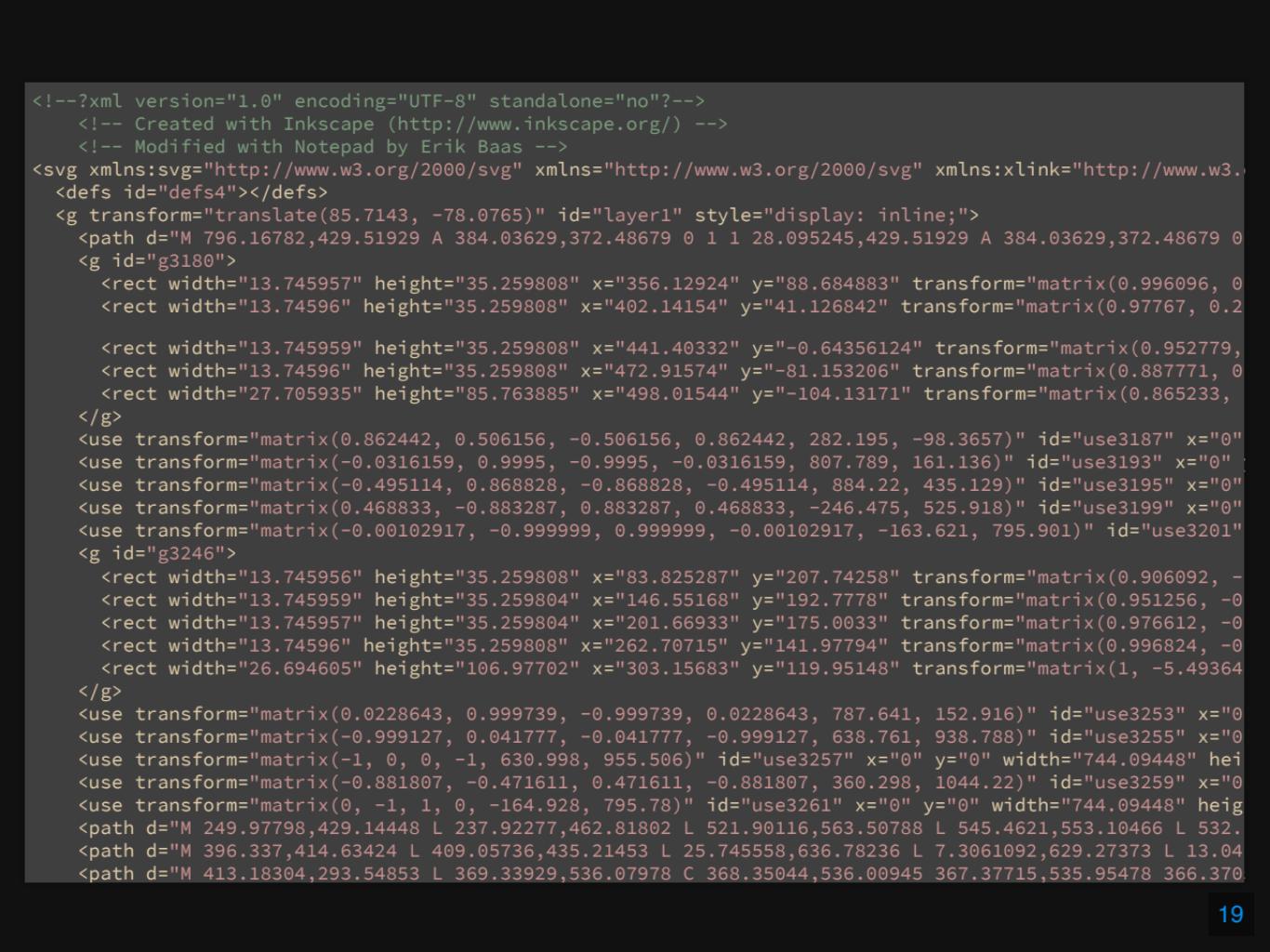
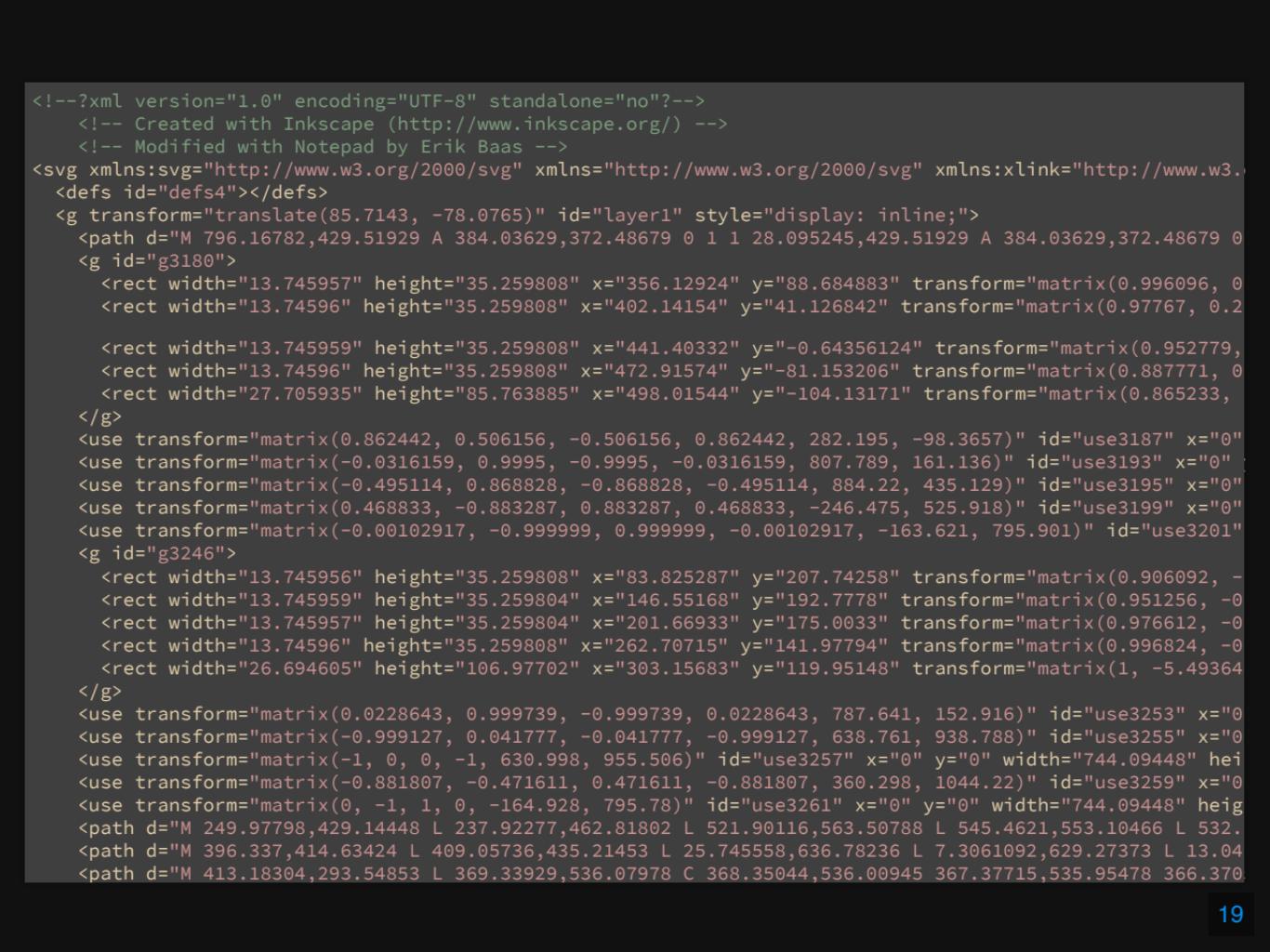
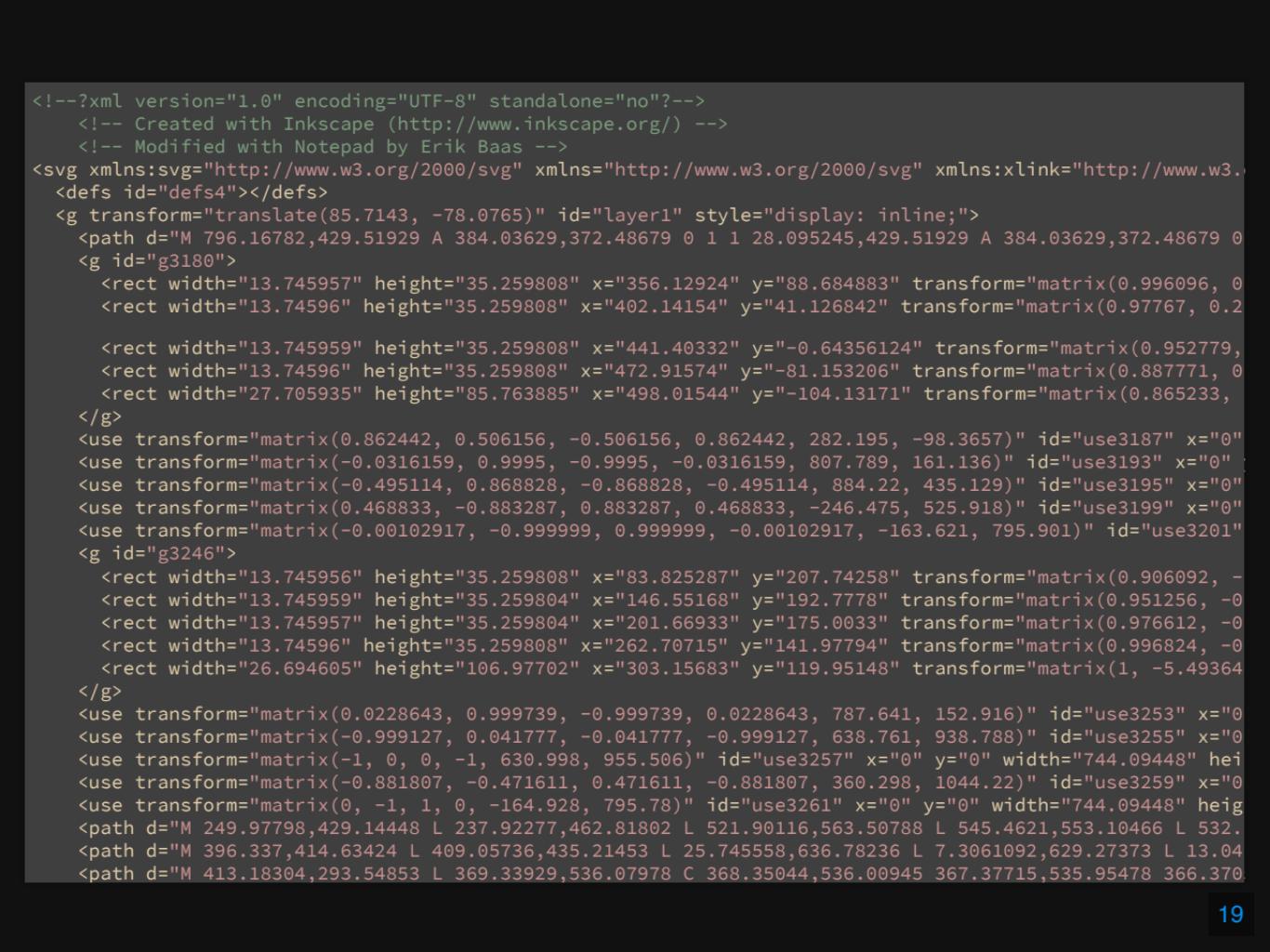
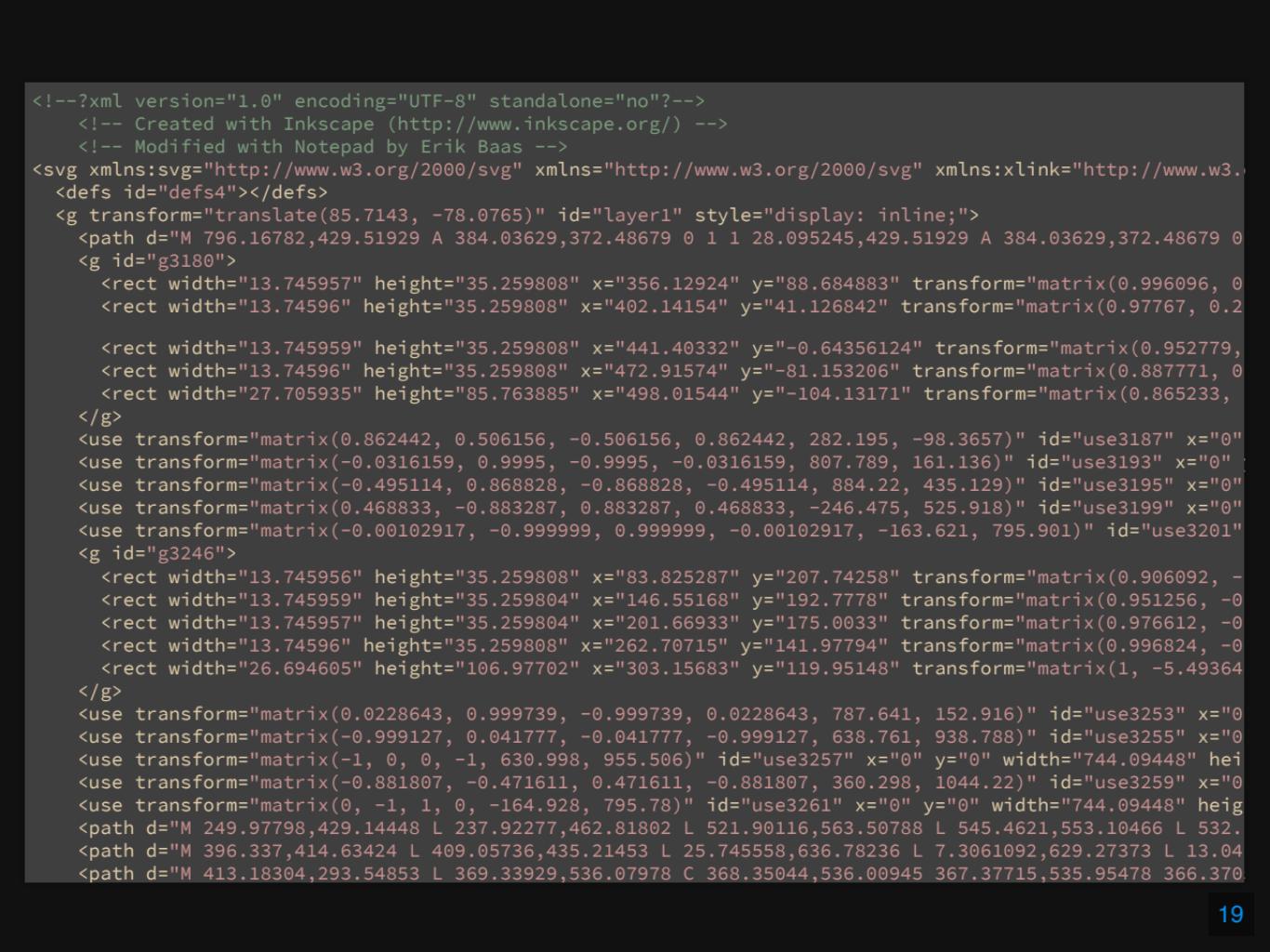
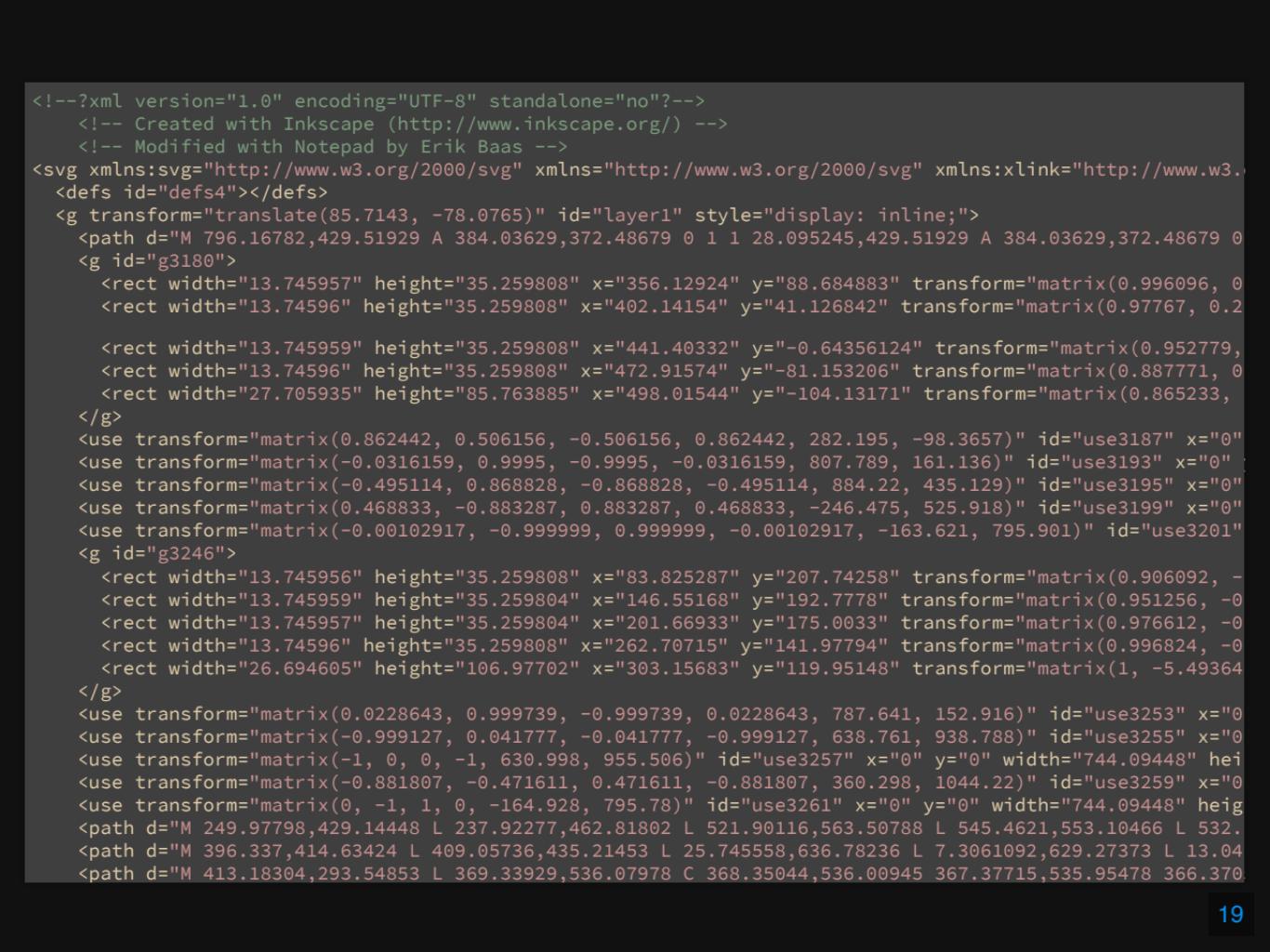
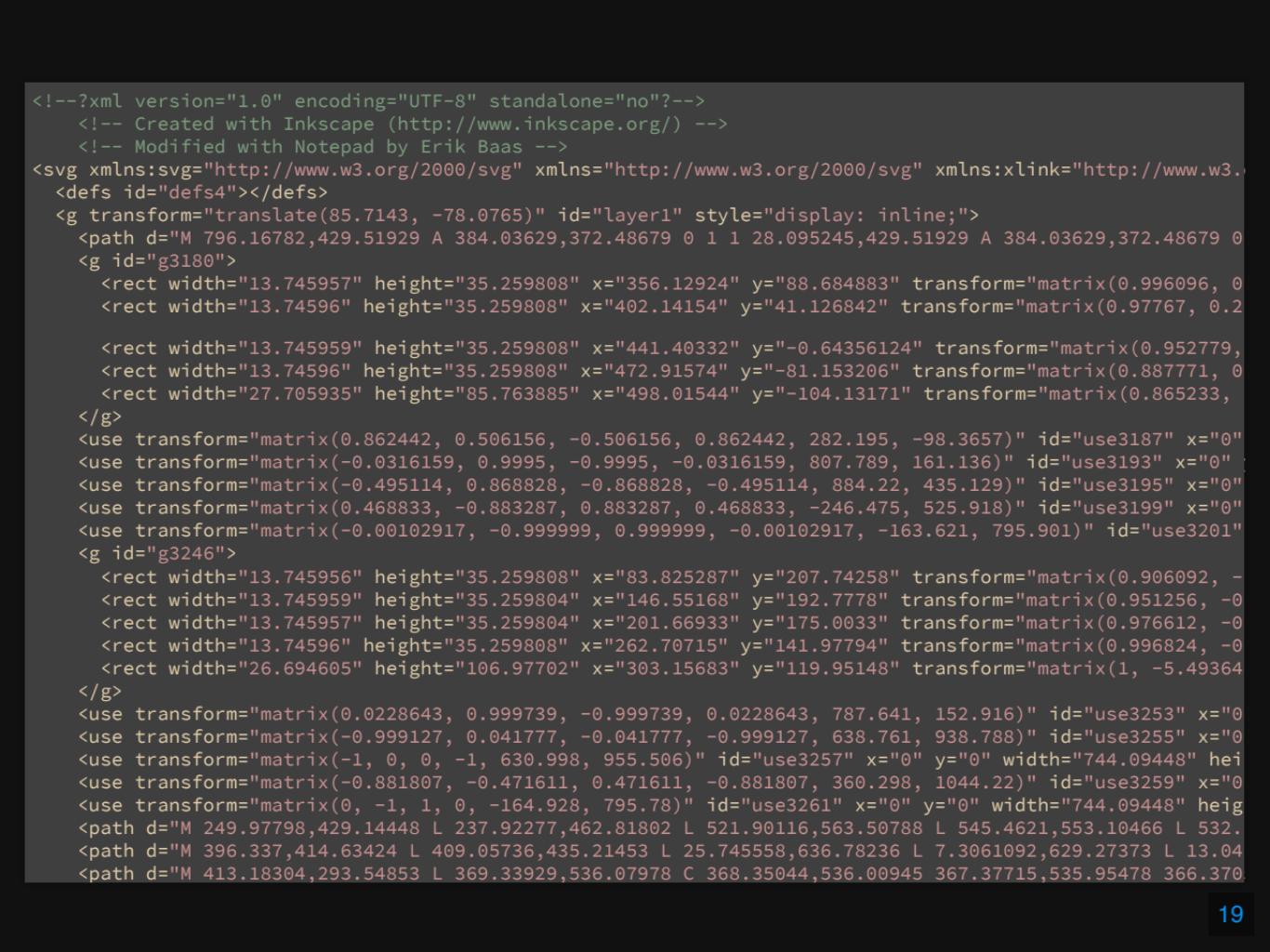
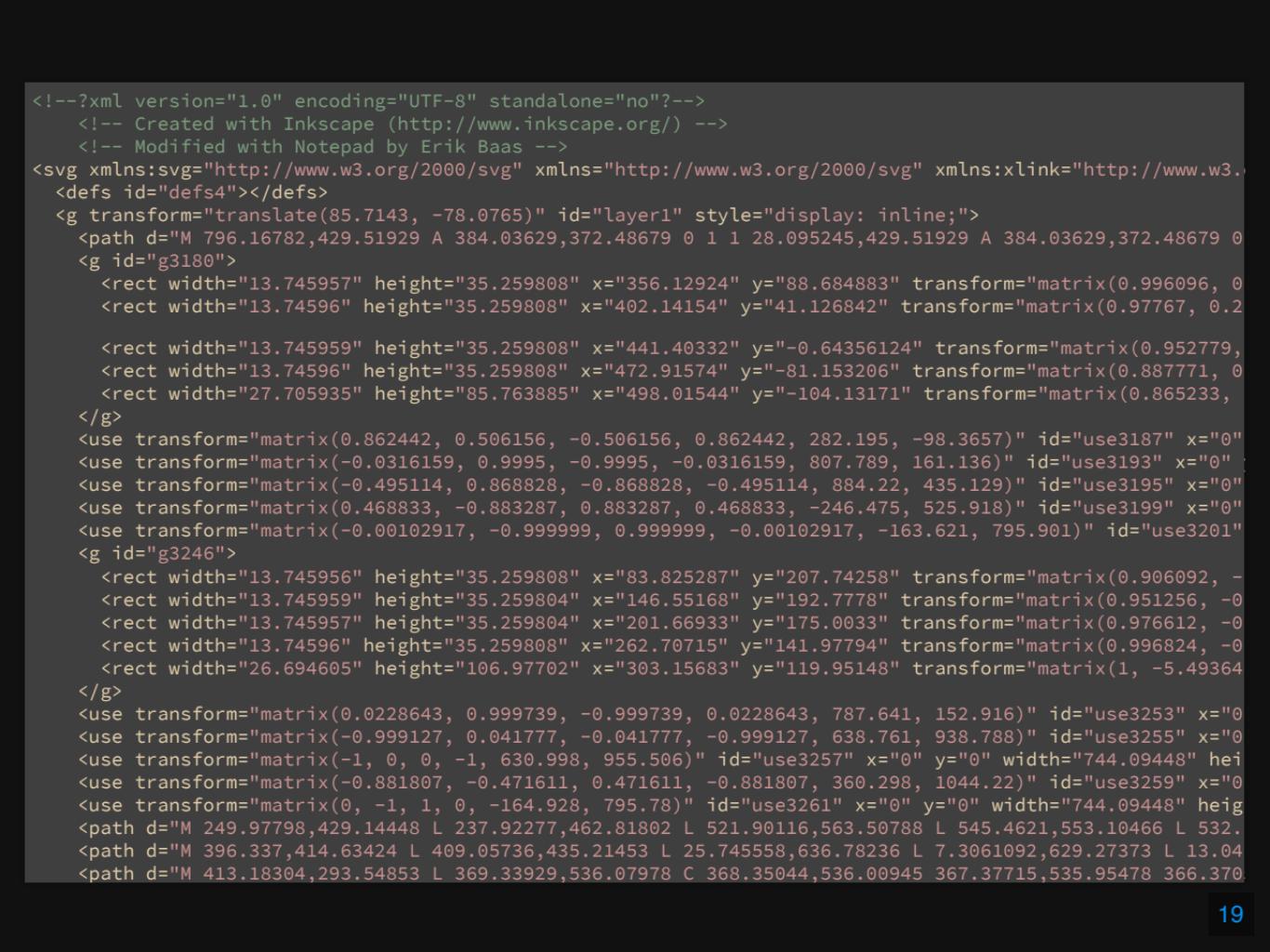
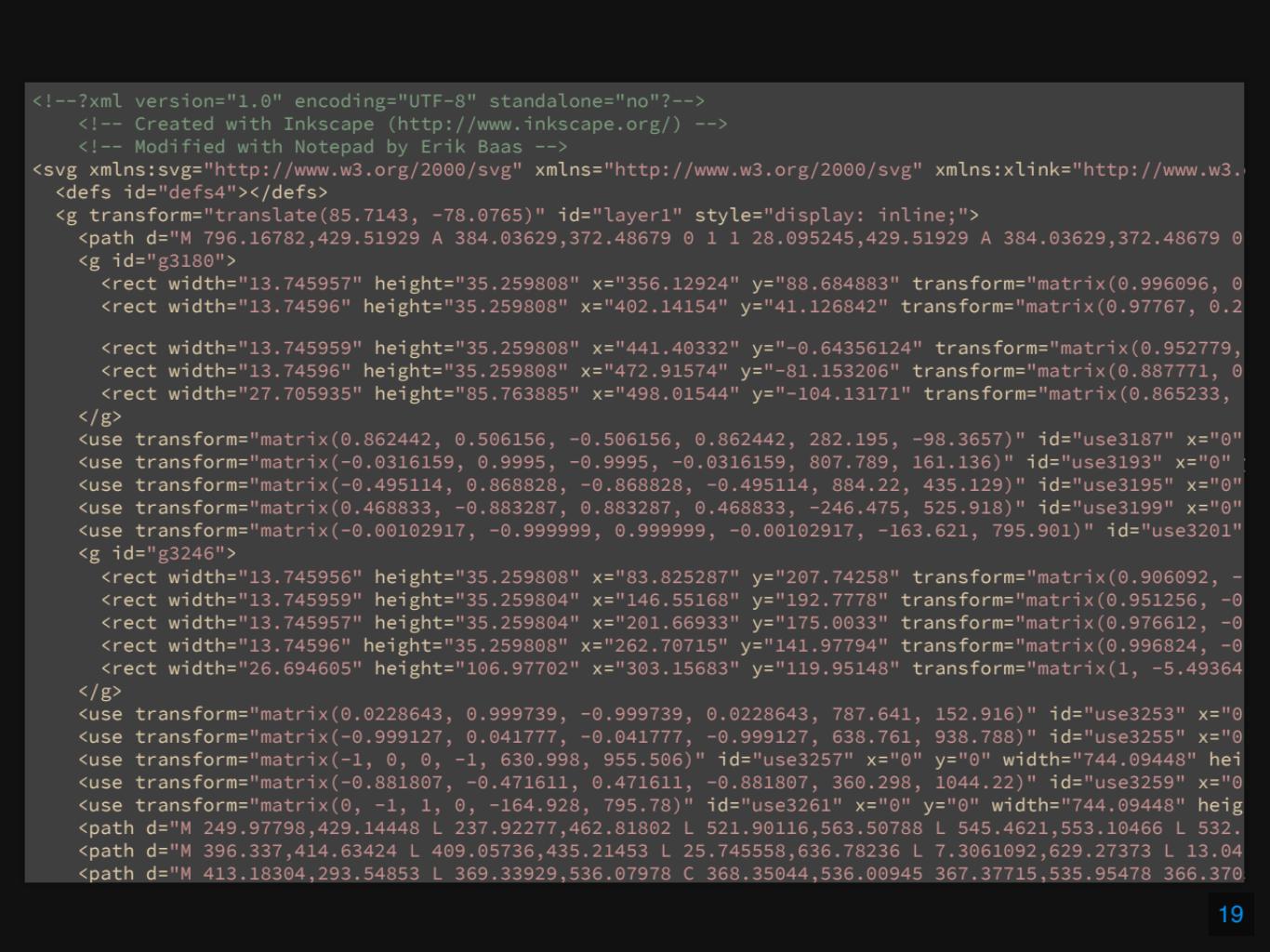


Векторная графика

– это изображения, состоящие из элементарных геометрических фигур:

- Точки
- Линии и ломаные линии
- Многоугольники
- Окружности и эллипсы
- Кривые Бéзье
- Текст

```

<!--?xml version="1.0" encoding="UTF-8" standalone="no"?-->
<!-- Created with Inkscape (http://www.inkscape.org/) -->
<!-- Modified with Notepad by Erik Baas -->
<svg xmlns:svg="http://www.w3.org/2000/svg" xmlns="http://www.w3.org/2000/svg" xmlns:xlink="http://www.w3.org/1999/xlink" id="image1" width="1000" height="1000" viewbox="0 0 1000 1000">
<defs id="defs4"></defs>
<g transform="translate(85.7143, -78.0765)" id="layer1" style="display: inline;">
    <path d="M 796.16782,429.51929 A 384.03629,372.48679 0 1 1 28.095245,429.51929 A 384.03629,372.48679 0 1 1 796.16782,429.51929" id="use3180"/>
    <g id="g3180">
        <rect width="13.745957" height="35.259808" x="356.12924" y="88.684883" transform="matrix(0.996096, 0.0865233, -0.0865233, 0.996096, 0, 0)" id="use3187"/>
        <rect width="13.74596" height="35.259808" x="402.14154" y="41.126842" transform="matrix(0.97767, 0.200000, -0.200000, 0.97767, 0, 0)" id="use3193"/>
        <rect width="13.745959" height="35.259808" x="441.40332" y="-0.64356124" transform="matrix(0.952779, 0.0468833, -0.0468833, 0.952779, 0, 0)" id="use3195"/>
        <rect width="13.74596" height="35.259808" x="472.91574" y="-81.153206" transform="matrix(0.887771, 0.468833, -0.468833, 0.887771, 0, 0)" id="use3199"/>
        <rect width="27.705935" height="85.763885" x="498.01544" y="-104.13171" transform="matrix(0.865233, 0.00102917, -0.00102917, 0.865233, 0, 0)" id="use3201"/>
    </g>
    <use transform="matrix(0.862442, 0.506156, -0.506156, 0.862442, 282.195, -98.3657)" id="use3187" x="0" y="0" width="13.745956" height="35.259808" style="display: inline-block; position: absolute; left: 356.12924px; top: 88.684883px; transform-origin: left top; transform: matrix(0.996096, 0.0865233, -0.0865233, 0.996096, 0, 0);"/>
<use transform="matrix(-0.0316159, 0.9995, -0.9995, -0.0316159, 807.789, 161.136)" id="use3193" x="0" y="0" width="13.745959" height="35.259808" style="display: inline-block; position: absolute; left: 402.14154px; top: 41.126842px; transform-origin: left top; transform: matrix(0.97767, 0.200000, -0.200000, 0.97767, 0, 0);"/>
<use transform="matrix(-0.495114, 0.868828, -0.868828, -0.495114, 884.22, 435.129)" id="use3195" x="0" y="0" width="13.74596" height="35.259808" style="display: inline-block; position: absolute; left: 441.40332px; top: -0.64356124px; transform-origin: left top; transform: matrix(0.952779, 0.0468833, -0.0468833, 0.952779, 0, 0);"/>
<use transform="matrix(0.468833, -0.883287, 0.883287, 0.468833, -246.475, 525.918)" id="use3199" x="0" y="0" width="27.705935" height="85.763885" style="display: inline-block; position: absolute; left: 472.91574px; top: -81.153206px; transform-origin: left top; transform: matrix(0.887771, 0.468833, -0.468833, 0.887771, 0, 0);"/>
<use transform="matrix(-0.00102917, -0.999999, 0.999999, -0.00102917, -163.621, 795.901)" id="use3201" x="0" y="0" width="26.694605" height="106.97702" style="display: inline-block; position: absolute; left: 498.01544px; top: -104.13171px; transform-origin: left top; transform: matrix(0.865233, 0.00102917, -0.00102917, 0.865233, 0, 0);"/>
</g>
<g id="g3246">
    <rect width="13.745956" height="35.259808" x="83.825287" y="207.74258" transform="matrix(0.906092, -0.0468833, 0.0468833, 0.906092, 0, 0)" id="use3253"/>
    <rect width="13.745959" height="35.259804" x="146.55168" y="192.7778" transform="matrix(0.951256, -0.0865233, 0.0865233, 0.951256, 0, 0)" id="use3255"/>
    <rect width="13.745957" height="35.259804" x="201.66933" y="175.0033" transform="matrix(0.976612, -0.0468833, 0.0468833, 0.976612, 0, 0)" id="use3257"/>
    <rect width="13.74596" height="35.259808" x="262.70715" y="141.97794" transform="matrix(0.996824, -0.0865233, 0.0865233, 0.996824, 0, 0)" id="use3259"/>
    <rect width="26.694605" height="106.97702" x="303.15683" y="119.95148" transform="matrix(1, -5.49364, 5.49364, 1, 0, 0)" id="use3261"/>
</g>
<use transform="matrix(0.0228643, 0.999739, -0.999739, 0.0228643, 787.641, 152.916)" id="use3253" x="0" y="0" width="13.745956" height="35.259808" style="display: inline-block; position: absolute; left: 83.825287px; top: 207.74258px; transform-origin: left top; transform: matrix(0.906092, -0.0468833, 0.0468833, 0.906092, 0, 0);"/>
<use transform="matrix(-0.999127, 0.041777, -0.041777, -0.999127, 638.761, 938.788)" id="use3255" x="0" y="0" width="13.745959" height="35.259804" style="display: inline-block; position: absolute; left: 146.55168px; top: 192.7778px; transform-origin: left top; transform: matrix(0.951256, -0.0865233, 0.0865233, 0.951256, 0, 0);"/>
<use transform="matrix(-1, 0, 0, -1, 630.998, 955.506)" id="use3257" x="0" y="0" width="13.745957" height="35.259804" style="display: inline-block; position: absolute; left: 201.66933px; top: 175.0033px; transform-origin: left top; transform: matrix(0.976612, -0.0468833, 0.0468833, 0.976612, 0, 0);"/>
<use transform="matrix(-0.881807, -0.471611, 0.471611, -0.881807, 360.298, 1044.22)" id="use3259" x="0" y="0" width="13.74596" height="35.259808" style="display: inline-block; position: absolute; left: 262.70715px; top: 141.97794px; transform-origin: left top; transform: matrix(0.996824, -0.0865233, 0.0865233, 0.996824, 0, 0);"/>
<use transform="matrix(0, -1, 1, 0, -164.928, 795.78)" id="use3261" x="0" y="0" width="26.694605" height="106.97702" style="display: inline-block; position: absolute; left: 303.15683px; top: 119.95148px; transform-origin: left top; transform: matrix(1, -5.49364, 5.49364, 1, 0, 0);"/>
<path d="M 249.97798,429.14448 L 237.92277,462.81802 L 521.90116,563.50788 L 545.4621,553.10466 L 532.396.337,414.63424 L 409.05736,435.21453 L 25.745558,636.78236 L 7.3061092,629.27373 L 13.04<path d="M 413.18304,293.54853 L 369.33929,536.07978 C 368.35044,536.00945 367.37715,535.95478 366.370

```



Недостатки векторной графики

- большое количество описательной информации для сложных объектов
- недостаточная реалистичность векторного изображения
- грузит CPU

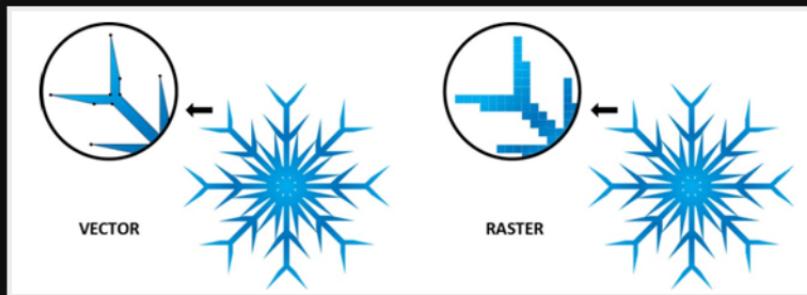
```
<svg version="1" xmlns="http://www.w3.org/2000/svg" width="400" height="200" viewBox="0 0 300.000
<g fill="#567660">
    <path class="node" d="M147.5 61l-5 3.7-3-2.3c-1.7-1.2-4.2-2.7-5.7-3.3-2.6-1-2.7-.9-2.3 1.7z
</g>
</svg>
```



EXAMPLE

Растровая графика

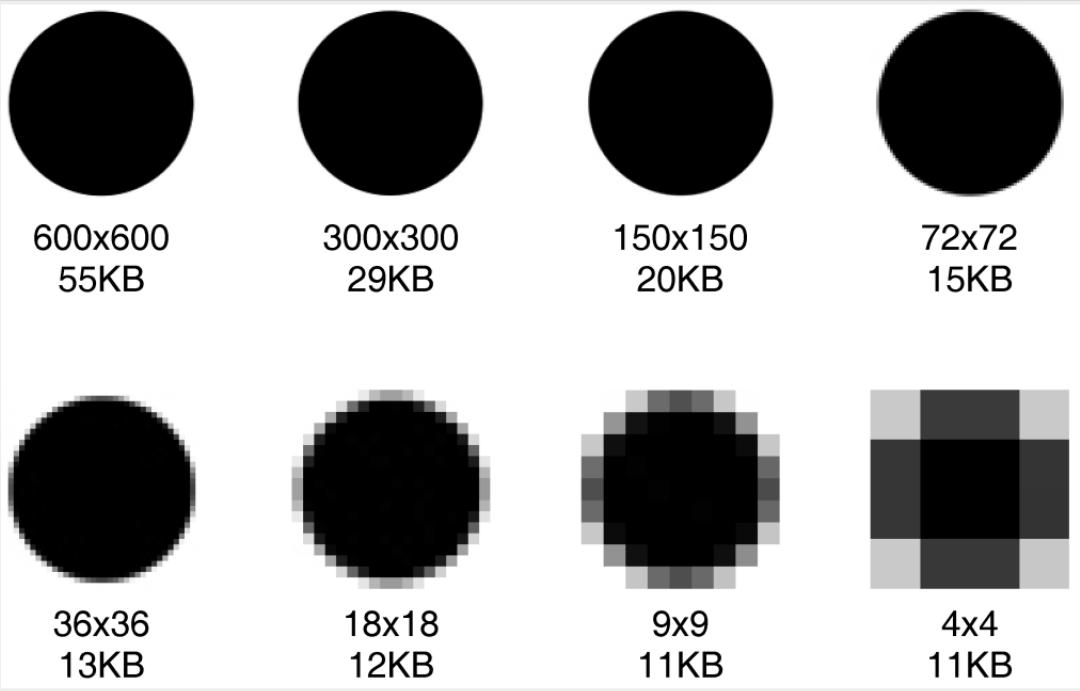
Изображение по вертикали и горизонтали разбивается на достаточно мелкие прямоугольники – пиксели.



Характеристики

растровых изображений

Разрешение



Глубина цвета

- определяет то количество оттенков, в диапазоне которых точка может изменять свой цвет.

	1 0 0 1 0 1 1 0 0 1 0 0 1 0 0 1
	00 11 00 01 01 00 10 00 00 10 00 01 01 00 11 00

Глубина цвета (битов) Количество цветов

2	$2^2 = 4$
---	-----------

4	$2^4 = 16$
---	------------

8	$2^8 = 256$
---	-------------

16	$2^{16} = 65\ 536$
----	--------------------

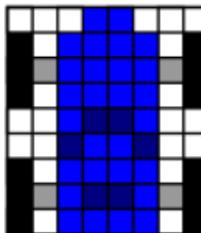
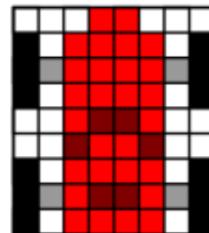
24	$2^{24} = 16\ 777\ 216$
----	-------------------------

1 бит / 24 бита



Индексированный цвет

0	0	0	1	1	0	0	0
3	0	1	1	1	1	0	3
3	4	1	1	1	1	4	3
3	0	1	1	1	1	0	3
0	0	1	2	2	1	0	0
0	0	2	1	1	2	0	0
3	0	1	1	1	1	0	3
3	4	1	2	2	1	4	3
3	0	1	1	1	1	0	3



Плюсы растрового изображения:

- высокая реалистичность изображения
- позволяет воспроизвести изображение любой сложности

Минусы растрового изображения:

- плохо масштабируются
- изображение можно редактировать только целиком
- зачастую может иметь больший размер, чем аналог векторного изображения



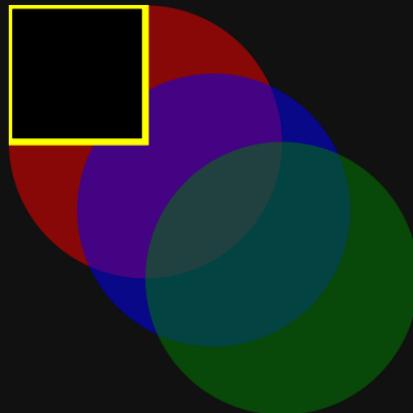
в вектор – 1.98 КБ, а в растре – 51 КБ

SVG

SVG (Scalable Vector Graphics)

— язык разметки масштабируемой векторной графики

```
<!--?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" style="fill-opacity: 0.5"/>
    <rect x="0" y="0" width="100" height="100" fill="black" stroke="yellow" stroke-width="5px"/>
  </svg>
```



SVG в HTML5

В настоящий момент формат SVG поддерживается 95.83% браузеров.

```
<svg width="150" height="100" viewBox="0 0 3 2">
  <rect width="1" height="2" x="0" fill="#008d46" />
  <rect width="1" height="2" x="1" fill="#ffffff" />
  <rect width="1" height="2" x="2" fill="#d2232c" />
</svg>
```



EXAMPLE

Форматы растровой графики

BMP (Bitmap Picture)

- с форматом BMP работает огромное количество программ
- можно сохранять черно-белые, серые полутонаовые, индексные цветные и цветные изображения системы RGB
- большой объем



15.6 MB

GIF (Graphics Interchange Format)

- анимации
- однобитовая прозрачность
- индексная палитра
- стандарт разрабатывался только для поддержки 256-цветовой палитры



EXAMPLE



BMP

GIF

15.6 MB

860 KB

JPEG (Joint Photographic Experts Group)

применяемый для хранения фотоизображений и подобных им изображений



BMP

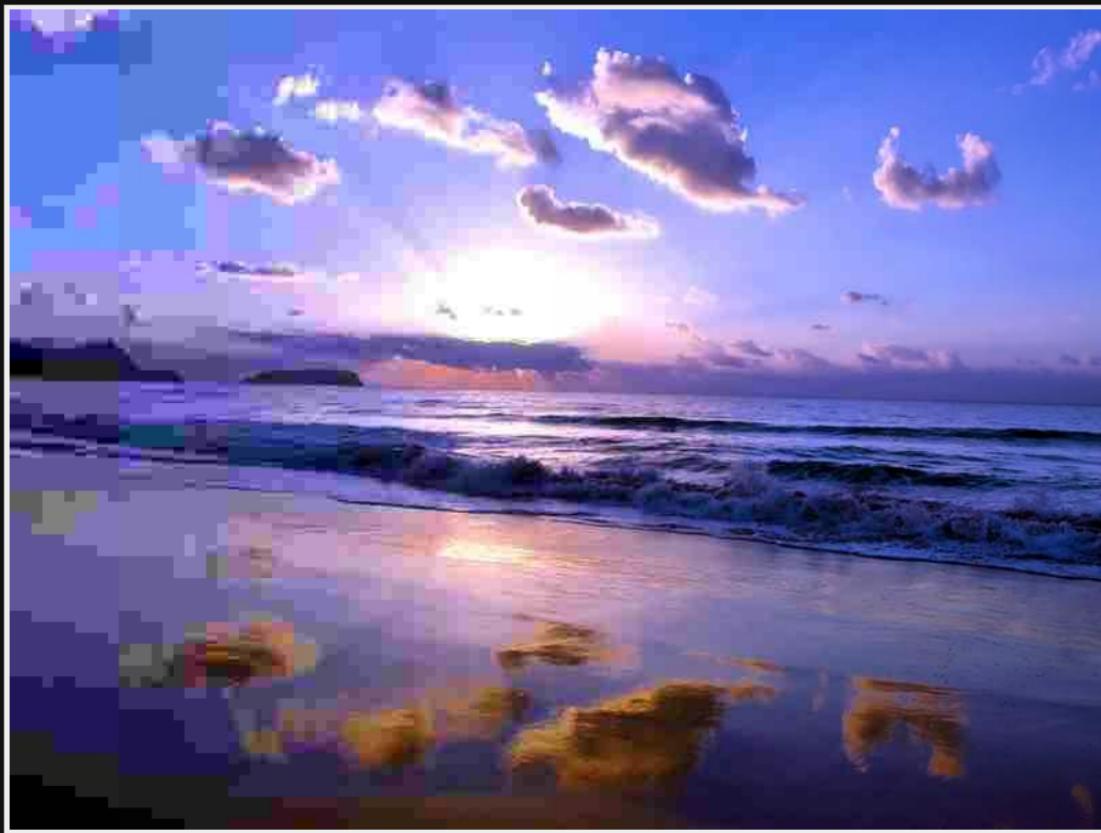
GIF

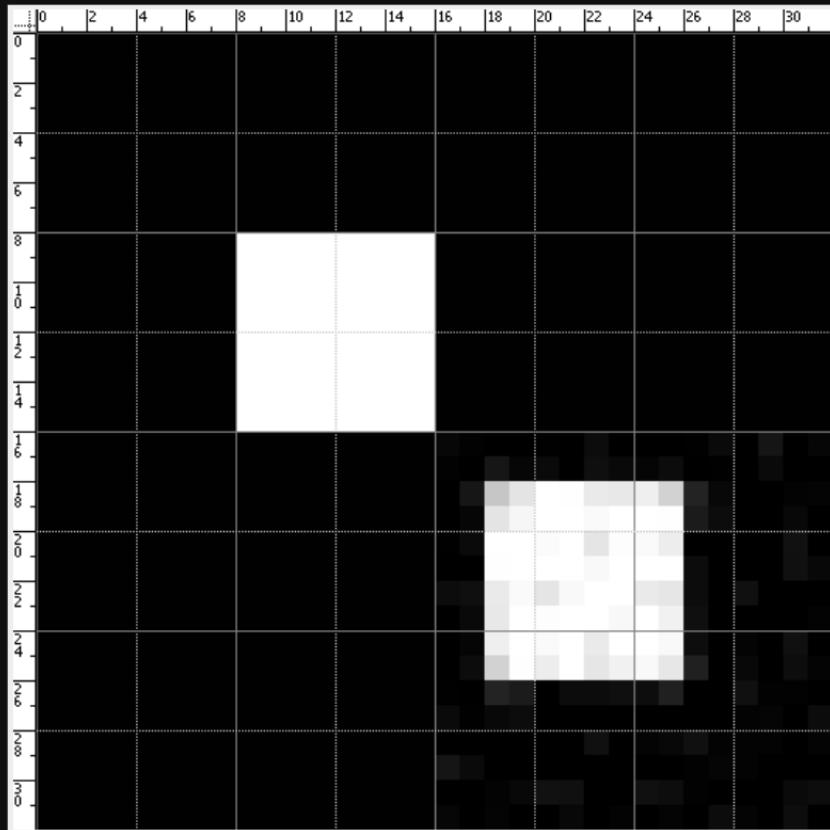
JPEG

15.6 MB

860 KB

980 KB





Множественное сохранение



<https://clck.ru/AMF7p>

PNG (Portable Network Graphics)

- альфаканал
- обладает более высокой степенью сжатия для файлов с большим количеством цветов
- пересохранение изображения проходят без потерь в качестве.
- цветное индексированное изображение (палитра 8 бит для цвета глубиной 24 бит)
- не подходит для фотографий

PNG vs JPEG





PNG vs GIF

png-8



16 KB

gif



17 KB

WebP

– обеспечивает сжатие изображений как с потерями, так и без потерь.

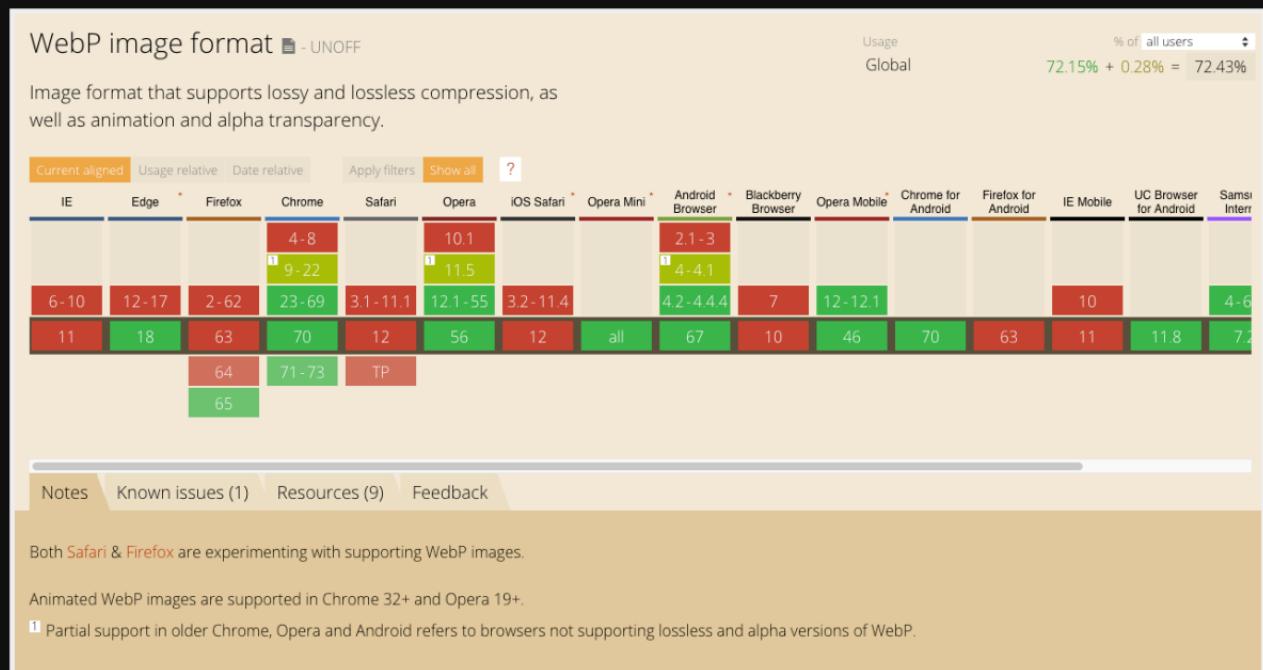


BMP	GIF	JPEG	PNG	WebP
15.6 MB	860 KB	980 KB	4.9MB	450KB

WebP

- Меньший размер файла
- Улучшенный алгоритм сжатия
- Более качественный переходы цвета
- Маска альфа-канала
- Искажение имеет пластиковый вид

Поддержка



Инструменты сжатия



JPEGmini



imageOptim



Kraken.io (online)

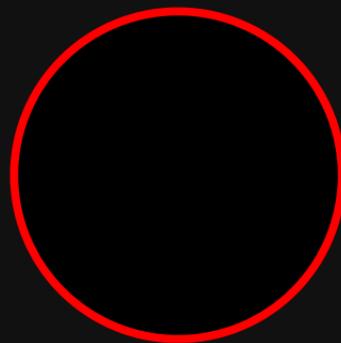
Оптимизация SVG

– заключается в удалении «ненужной» информации из SVG, тем самым уменьшается размер самого файла.

- SVG Cleaner
- SVG Scour (для работы необходимо установить Python)
- SVG Optimizer (для работы необходимо установить Node.js)
<https://clck.ru/8qdWP>

До сжатия

```
<!--?xml version="1.0" encoding="utf-8"?-->
    <!-- Generator: Adobe Illustrator 17.1.0, SVG Export Plug-In . SVG Version: 6.00
<svg version="1.2" baseProfile="tiny" id="Layer_1" xmlns="http://www.w3.org/2000/svg" xmlns:xlink="http://www.w3.org/1999/xlink" width="100%" height="100%" viewBox="0 0 1000 1000">
<g id="XMLID_1_>
    <g>
        <circle fill="black" stroke="red" stroke-width="2" stroke-miterlimit="10" cx="50" cy="50" r="50"/>
    </g>
</g>
</svg>
```

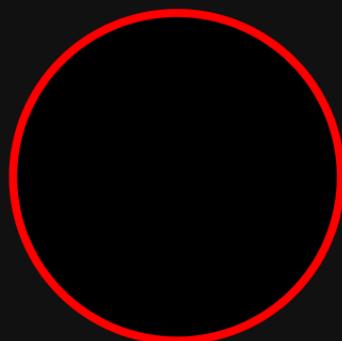


EXAMPLE

После сжатия

```
<svg version="1.2" baseProfile="tiny" xmlns="http://www.w3.org/2
```

$$0.614 \text{ KiB} - 68.2\% = 0.195 \text{ KiB}$$



EXAMPLE

img

```

```

Retina дисплей



Изображение

нормального человека



человека с Apple



Изображение

нормального человека



человека с Apple



srcset

```

```

```

```



Picture

```
<picture>
  <source srcset="img@2x.webp 2x, img@1x.webp 1x">
    
</picture>
```

Picture

```
<picture>
  <source media="(min-width: 800px)"
         srcset="images/filter-cat.jpg" type="image/jpeg">
  <source media="(max-width: 799px)"
         srcset="images/cat.png" type="image/png">
  
</picture>
```



object-fit

fill



contain



cover



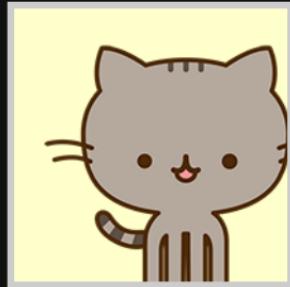
none



Background

```
.box1 {  
    background-image: url(pusheen.png);  
}
```

Картинка 250x250, содержащий блок 200x200

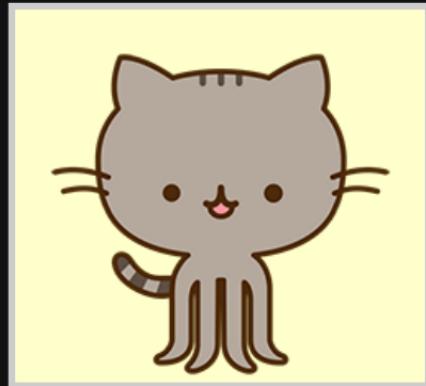


EXAMPLE

Размеры фона

```
.box1 {  
    background-image: url(pusheen.png);  
    background-size: 100%;  
  
    /*background-size: 100% auto;*/  
}
```

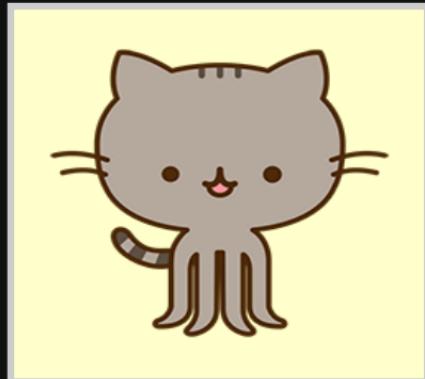
Картинка 250x250, содержащий блок 200x200



Размеры фона

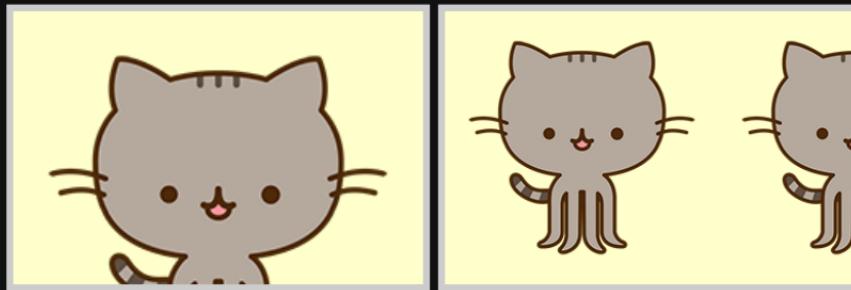
```
.box1 {  
    background-image: url(pusheen.png);  
    background-size: 100% 100%;  
  
    /*Можно указывать значение в px*/  
}
```

Картинка 250x250, содержащий блок 200x200



cover/contain

```
.box1 {  
    background-image: url(pusheen.png);  
    background-size: cover;  
}  
.box2 {  
    background-image: url(pusheen.png);  
    background-size: contain;  
}
```



<https://clck.ru/9QAEL>

EXAMPLE

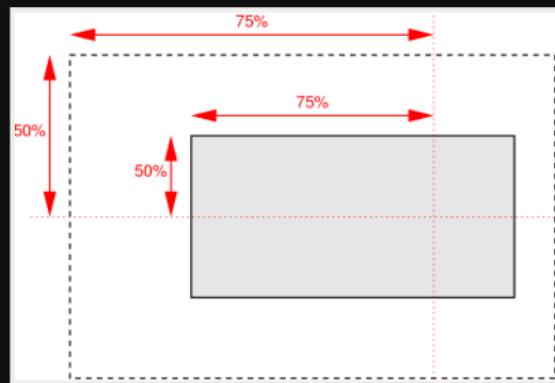
Повторение картинок

```
.box0 {  
    background-repeat: repeat; /*по умолчанию*/  
}  
.box1 {  
    background-repeat: no-repeat;  
}  
.box2 {  
    background-repeat: repeat-x;  
}  
.box3 {  
    background-repeat: repeat-y;  
}
```

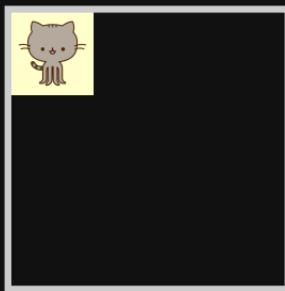


Положение фона

```
.box1 {  
    background-image: url(pusheen.png);  
    /*left | center | right | <проценты> | <значение>*/  
    /*top | center | bottom | <проценты> | <значение>*/  
    background-position: 75% 50%;  
}
```



<https://clck.ru/AMT8o>



```
background-position: 0 0;
```

EXAMPLE



```
background-position: 100% 0;
```

```
background-position: 100% 100%;
```

```
background-position: 0 100%;
```

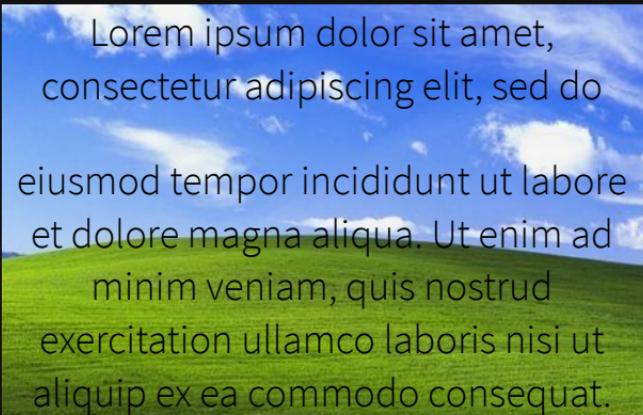
```
background-position: 0 0;
```

Фиксация фона

```
.box1 {  
    background-image: url(windows.png);  
    /*scroll | fixed | local*/  
  
    background-attachment: fixed  
}
```

```
<div class="background">  
    <p>Какой-то текст</p>  
</div>
```

scroll



 Lorem ipsum dolor sit amet,
consectetur adipiscing elit, sed do
eiusmod tempor incididunt ut labore
et dolore magna aliqua. Ut enim ad
minim veniam, quis nostrud
exercitation ullamco laboris nisi ut
aliquip ex ea commodo consequat.

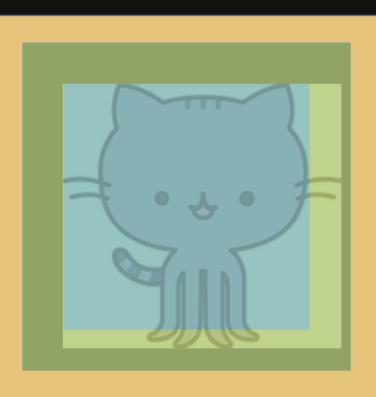
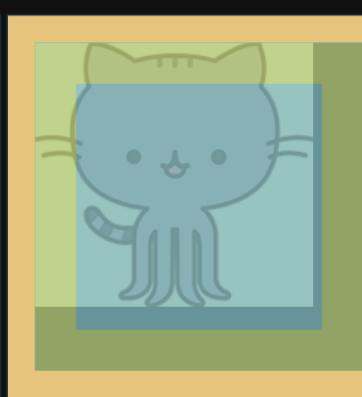
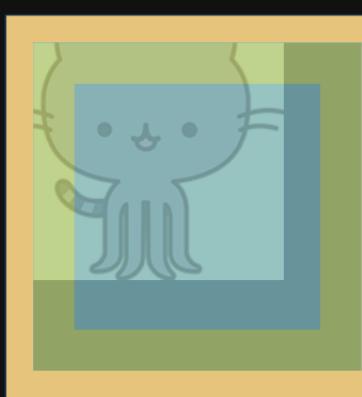
fixed

 Lorem ipsum dolor sit amet,
consectetur adipiscing elit, sed do
eiusmod tempor incididunt ut labore

Пример

Область позиционирования

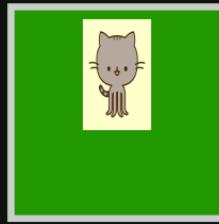
```
.box { background-repeat: no-repeat; padding: 20px; }  
.box1 { background-origin: border-box; }  
.box2 { background-origin: padding-box; /*по умолчанию*/ }  
.box3 { background-origin: content-box; }
```



EXAMPLE

Короче

```
/* background: <bg-image> ||  
<position> [ / <bg-size> ]? || <repeat-style> ||  
<attachment> || '&background-color'>*/  
.box {  
    background: url(pusheen.png) 50% 10% / 50px 80px  
            no-repeat #290;  
}
```



<https://clck.ru/AMKMc>

EXAMPLE

Линейные градиенты

```
.box {  
    background: linear-gradient(#f00, #0f0);  
}
```



EXAMPLE

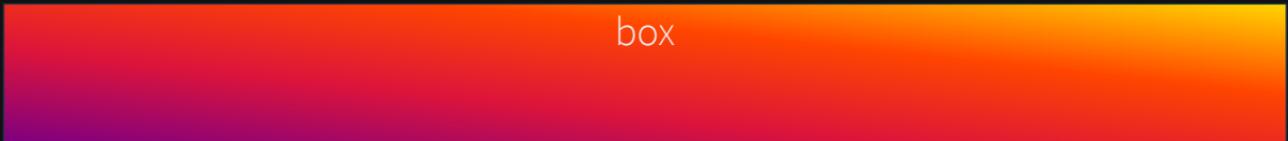
```
.box {  
    background: linear-gradient(45deg, #ff0 0, #f00 30%,  
                                #0f0 30%);  
}
```



EXAMPLE

Направление градиента

```
.box {  
    background: linear-gradient(to top right,  
                                purple, crimson, orangered, gold);  
}
```



box

EXAMPLE

```
.box {  
    background: linear-gradient(45deg,  
                                purple, crimson, orangered, gold);  
}
```



box

74

EXAMPLE

Радиальные градиенты

```
.box {  
    background: radial-gradient(gold, purple);  
}
```



box

EXAMPLE

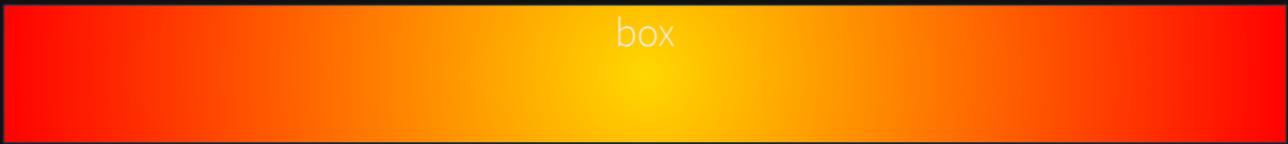
```
background: radial-gradient(100px 200px at 100px 10px,  
    white 0%, orangered 50%, purple 100%);
```



EXAMPLE

Форма

```
.box {  
    background: radial-gradient(circle, gold, red);  
}
```



box

EXAMPLE

```
.box {  
    background: radial-gradient(200px 30px, gold, red);  
}
```



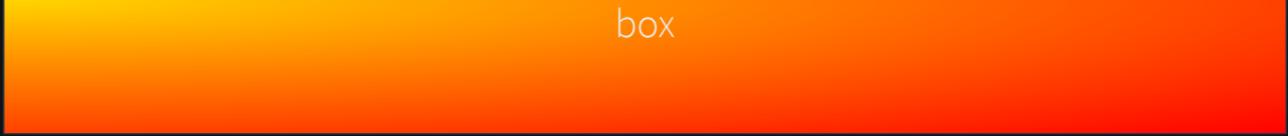
box

76

EXAMPLE

Положение

```
.box {  
    background: radial-gradient(at top left, gold, red);  
}
```



box

EXAMPLE

```
.box {  
    background: radial-gradient(at 600px 50px, gold, red);  
}
```

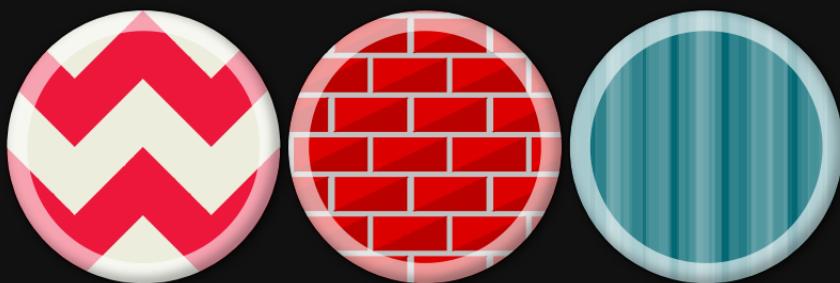


box

77

EXAMPLE

Шаблоны



EXAMPLE

```
background: linear-gradient(135deg, white 25%, transparent 25%) -50px 0,  
          linear-gradient(225deg, white 25%, transparent 25%) -50px 0,  
          linear-gradient(315deg, white 25%, transparent 25%),  
          linear-gradient(45deg, white 25%, transparent 25%);  
background-size: 100px 100px;  
background-color: red;
```

Фильтры

```
.cat {  
    opacity: .5;  
}
```



`filter: grayscale(1);`



`filter: sepia(1)`



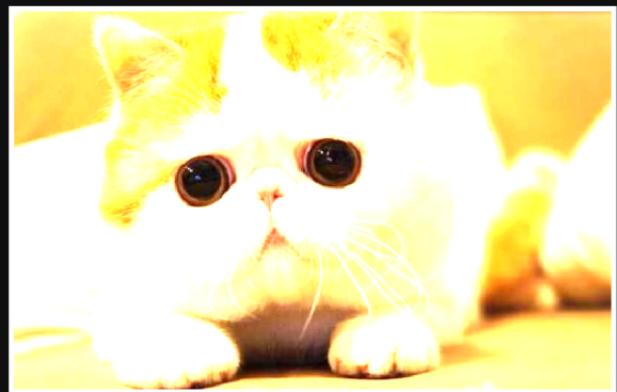
`filter: saturate(2);`

`filter: hue-rotate(180deg)`

filter: invert(1);



filter: brightness(2)



filter: contrast(2);

filter: blur(20px)

Объединение фильтров

```
.cat {  
    filter: invert(100%) sepia(100%);  
}
```

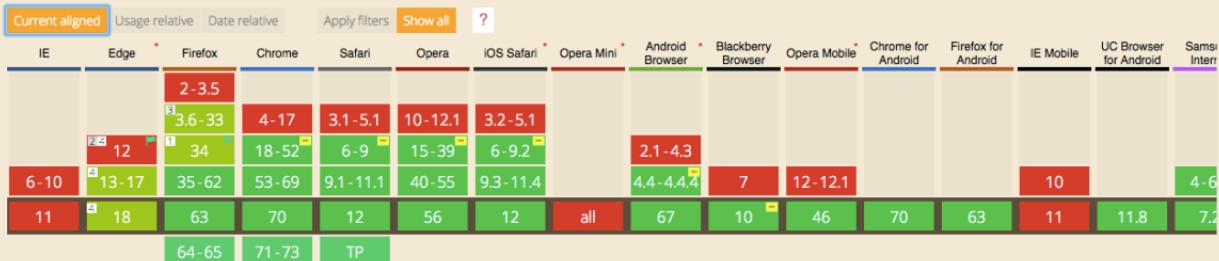


Поддержка

CSS Filter Effects - WD

Method of applying filter effects (like blur, grayscale, brightness, contrast and hue) to elements, previously only possible by using SVG.

Usage	% of all users
Global	88.07% + 2.17% = 90.24%
unprefixed:	84.75% + 2.17% = 86.92%
Russian Federation	79.99% + 1.4% = 81.39%
unprefixed:	76.54% + 1.4% = 77.93%



Notes Known issues (1) Resources (7) Feedback

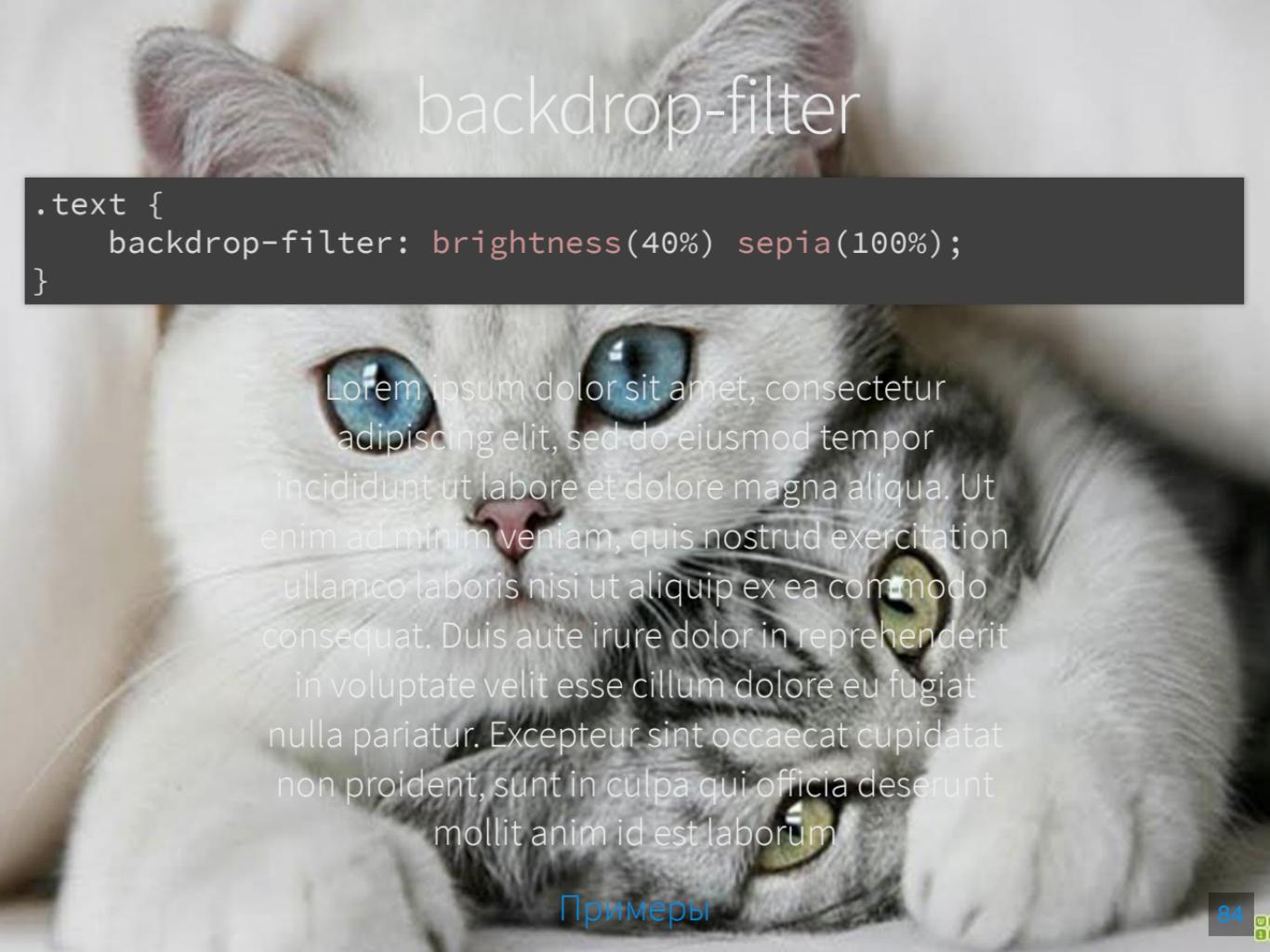
Note that this property is significantly different from and incompatible with Microsoft's [older "filter" property](#).

¹ Supported in Firefox under the `layout.css.filters.enabled` flag.

² Supported in MS Edge under the "Enable CSS filter property" flag.

³ Partial support in Firefox before version 34 [only implemented the url\(\)](#) function of the filter property

⁴ Partial support refers to supporting filter functions, but not the `ur1` function.

A close-up photograph of a cat's face, focusing on its large, bright blue eyes. The cat has white fur with some darker, greyish patches. The background is blurred.

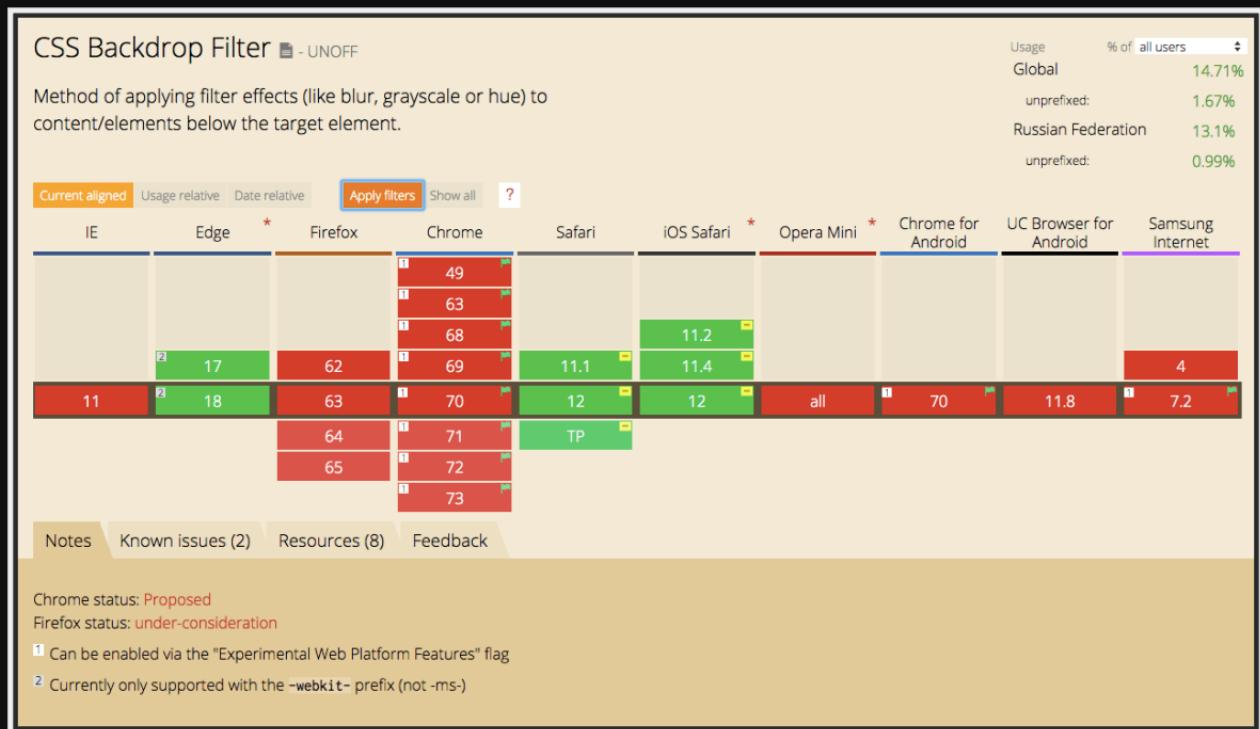
backdrop-filter

```
.text {  
    backdrop-filter: brightness(40%) sepia(100%);  
}
```

Lorem ipsum dolor sit amet, consectetur
 adipiscing elit, sed do eiusmod tempor
 incididunt ut labore et dolore magna aliqua. Ut
 enim ad minim veniam, quis nostrud exercitation
 ullamco laboris nisi ut aliquip ex ea commodo
 consequat. Duis aute irure dolor in reprehenderit
 in voluptate velit esse cillum dolore eu fugiat
 nulla pariatur. Excepteur sint occaecat cupidatat
 non proident, sunt in culpa qui officia deserunt
 mollit anim id est laborum

Примеры

Поддержка



Спрайты



Спрайты

– прием объединить много изображений в одно.

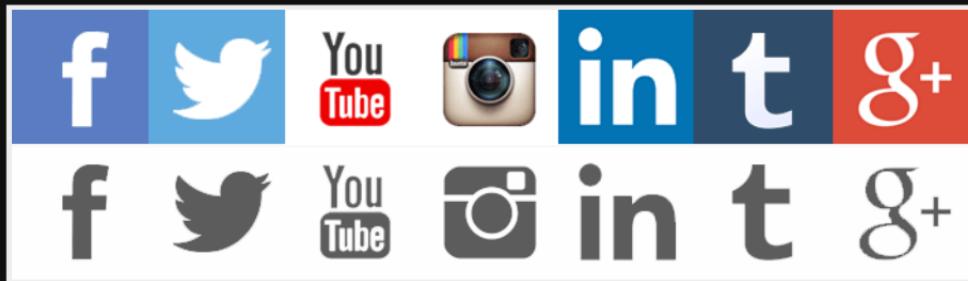


Спрайты в CSS

Нужно сделать кнопки "поделиться"



EXAMPLE



```
.social {  
    width: 100px;  
    height: 100px;  
    background-repeat: no-repeat;  
    background-image: url(share.svg)  
}  
  
.fb {  
    background-position: 0 0;  
}  
.fb:hover {  
    background-position: 0 -100px;  
}  
.twitter {  
    background-position: -100px 0;  
}  
.twitter:hover {  
    background-position: -100px -100px;  
}
```

Как это работает?



EXAMPLE



EXAMPLE

Особенности:

- Меньше запросов к серверу
- Смена по hover
- Сложно добавлять новые картинки (но можно автоматизировать)

Stitches (онлайн)
css-sprite

SVG-спрайты

```
<svg style="display: none;">
  <symbol id="instagram" viewBox="0 0 24 28">...</symbol>
  <symbol id="vk" viewBox="0 0 30 28">...</symbol>
</svg>
```

```
<div class="icons">
  <svg width="20px" height="20px">
    <use xlink:href="#instagram"/>
  </svg>
  <svg width="20px" height="20px">
    <use xlink:href="#vk"/>
  </svg>
</div>
```



EXAMPLE

SVG-спрайты

SVG: группировка и переиспользование элементов

Styling SVG <use> Content with CSS

Base64

– способ оптимизации загрузки изображения.

```


Colors

| Color      | Hex     |
|------------|---------|
| Black      | #000000 |
| White      | #ffffff |
| Yellow     | #ffe05e |
| Dark Gray  | #666666 |
| Light Gray | #f6f5f3 |

— А что если интернет подстроится под меня?

Интернет сегодня — это не только информация и технологии, это люди: их желания, предпочтения, задачи. Люди — вот атомы, из которых на самом деле состоит цифровая материя; и эти люди очень, очень разные. Сегодня поиск умеет подстраиваться, помогая людям быстрее приходить к тем решениям, которые им нужны.

Весь сайт в одном видео

Мы представляем Atom: совокупность инструментов, предназначенных научить интернет подстраиваться под разных

100%

Notes

96

# Экспорт изображений

The screenshot shows a Photoshop interface with the following elements:

- Top Bar:** Includes icons for file operations, a magnifying glass, and a search bar.
- Left Panel:** Shows a preview of the design, which features a large yellow 'Я' icon surrounded by white circles on a yellow background, with the text "— А что если интернет подстроится под меня?".
- Right Panel:** The "Assets" panel, titled "Artboard 3". It contains sections for "Download all assets" and "Download individual assets".
- Download all assets:** Includes options for "PNG" (87 bitmap images, 1x, 2x, 3x) and "JPG" (87 bitmap images, 1x, 2x, 3x).
- Download individual assets:** Lists items with their paths and dimensions:
  - path: 757px x 2453px
  - bg: 1416px x 757px
  - bag: 475px x 409px
  - palka: 228px x 622px
  - ipad: 304px x 453px
  - newspaper: 318px x 421px
  - notebook: 306px x 434px
- Bottom Bar:** Includes zoom controls (100%), a notes icon, and a help icon.

# Работа с элементом

Dashboard Artboard 3

Мы представляем Атом: совокупность инструментов, призванных научить интернет подстраиваться под разных людей, которые им нужны.

Максим      Алексей Быков      Георгий И. Дауге

Очень дорожит редкими отпусками и превыше всего ценит комфорт и качество обслуживания.

Юра сразу покупает тур: в его ситуациях так очевидно проще. Основные вводные - приемлемого качества.

The Landmark

Ellipse 43  
X: 1077px Y: 1196px  
Width: 138px Height: 138px

Colors  
#666666

Borders  
Inside, 2px  
#white #ffffff

CSS

```
.Ellipse-43 {
 width: 138px;
 height: 138px;
 background-color: #666666;
}
```

Notes

100%

98

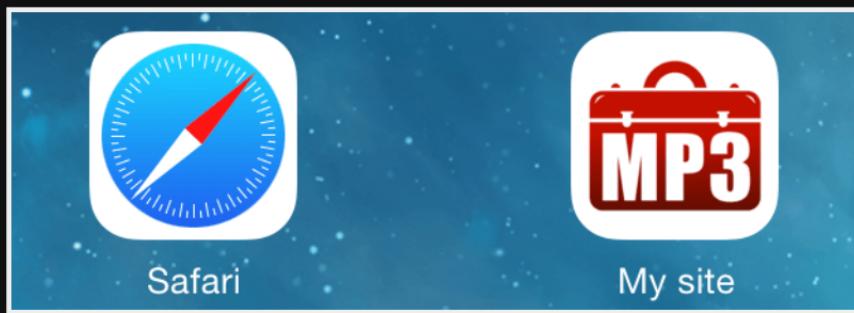
# Подробнее

Про Zeplin

Про Sketch

Про фигму

# favicon



# manifest.json

- name - название сайта
- short\_name - короткое название сайта
- icons - набор иконок разных размеров
- start\_url - url, который будет открываться при открытии сайта с рабочего стола
- display - как отображать сайт: с адресной строкой или без
- background\_color - цвет страницы до того как она загрузилась