

Интерактивная кольцевая диаграмма, созданная с помощью Protovis

Большинство пользователей обратятся к Excel, чтобы построить кольцевую диаграмму. Я тоже в 2012 г. написал на эту тему заметку – [Excel. Круговая диаграмма с двумя наборами данных](#). Однако, около месяца назад я начал читать книгу Нейтана Яу Искусство визуализации в бизнесе. Автор работает в области инфографики, для которой стандартные средства Excel являются недостаточно выразительными. На страницах книги автор знакомит читателей с различными программами, которые расширяют возможности (см., например, [Создание столбчатой диаграммы в R](#)).

В настоящей заметке интерактивная кольцевая диаграмма будет построена с помощью программы Protovis. Protovis — бесплатный инструмент для визуализации с открытым исходным кодом. Protovis — это JavaScript-библиотека, позволяющая использовать возможности современных браузеров для работы с масштабируемой векторной графикой. Поскольку графические объекты генерируются динамически, это дает возможность делать их анимированными и интерактивными. А потому Protovis — отличный выбор для создания онлайн-графики.¹

На рис. 1 показано, к чему вы будете стремиться. Когда указатель мыши оказывается поверх того или иного сектора, вы видите, сколько именно человек проголосовало за данную категорию. Интерактив может быть и более продвинутым, но прежде чем давать волю фантазии, необходимо освоить азы.

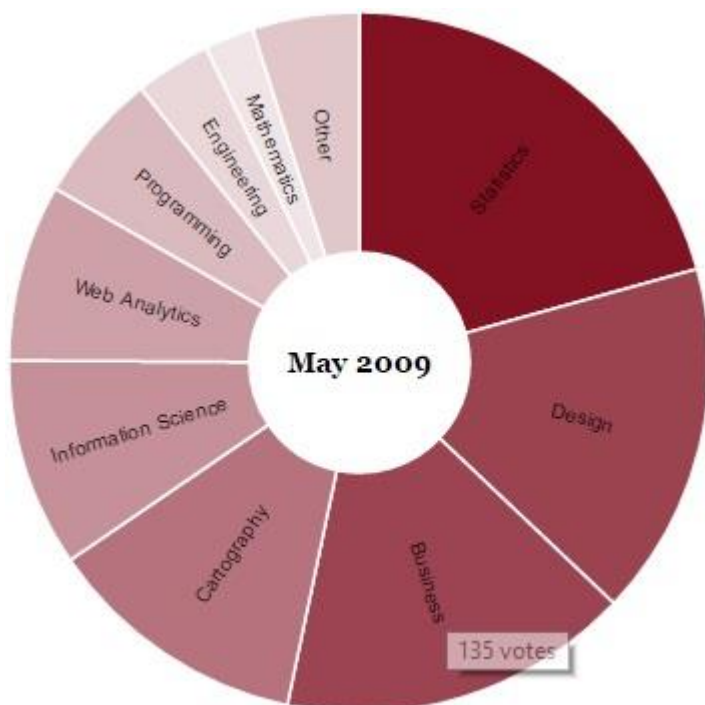


Рис. 1. Интерактивная кольцевая диаграмма, созданная с помощью библиотеки Protovis.js

Первым делом вам необходимо скачать библиотеку [Protovis](#). На сайте доступна версия v3.3.1. Распакуйте архив, и поместите файл `protovis.js` в рабочую папку. Создайте HTML-страницу, и разместите файл в той же папке. Назовите файл `donut.html` (рис. 2; код также можно найти в прикрепленном файле).

¹ Заметка написана на основе материалов книги Нейтан Яу. Искусство визуализации в бизнесе. – М.: Манн, Иванов и Фербер, 2013. – С. 151–156.

```

<html>
<head>
  <title>Donut Chart</title>
  <script type="text/javascript" src="protovis.js"></script>
  <style type="text/css">
    #figure {
      width: 400px;
      height: 400px;
    }
  </style>
</head>
<body>
  <div id="figure">

  </div><!-- @end figure -->

</body>
</html>

```

Рис. 2. Начало html-кода donut.html

Если вы когда-либо прежде создавали веб-страницы, то вам не надо ничего объяснять. А для тех, кто никогда не делал ничего подобного, поясню. Каждая страница начинается с тега `<html>`. За ним следует тег `<head>`, в котором содержится информация о странице, но который не отображается в окне вашего браузера. А вот все, что заключено в теге `<body>`, вполне видимо. В нашем примере страница получила название Donut Chart («Кольцевая диаграмма»), и на нее была загружена библиотека Protovis — это один JavaScript-файл — с помощью тега `<script>`. Далее идет фрагмент CSS для определения стиля страницы. Чем он проще — тем лучше. С помощью id "figure" ширина и высота блока `<div>` устанавливаются равными 400 пикселям. Это то пространство, где будет располагаться диаграмма. Приведенный фрагмент HTML, по сути, не является частью диаграммы, но он нужен, чтобы следующий далее код JavaScript правильно загрузился в веб-браузер. Если вы сейчас откроете файл donut.html в браузере, то увидите лишь чистую страницу.

Внутри `<div>` нужно внести уточнение о том, что код, который вы собираетесь написать, будет на JavaScript. Все остальное пойдет внутри тега `<script>`:

```

<script type="text/javascript+protovis">
  ...
</script>

```

Но начнем с данных. Перед вами результаты опроса, проведенного на сайте [FlowingData](#), которые записаны в виде двух последовательностей. Данные о числе голосов сохранены в одном массиве, а соответствующие им категории — в другом.

```

var data = [172,136,135,101,80,68,50,29,19,41];
var cats = ["Statistics", "Design", "Business", "Cartography", "Information Science", "Web Analytics",
"Programming", "Engineering", "Mathematics", "Other"];

```

Далее нужно задать ширину и высоту кольцевой диаграммы, а также радиус и шкалу для определения длины дуг.

```

var w = 350,
    h = 350,
    r = w / 2,
    a = pv.Scale.linear(0, pv.sum(data)).range(0, 2 * Math.PI);

```

Высота и ширина кольцевой диаграммы будут по 350 пикселей, а радиус (то есть расстояние от центра диаграммы до ее внешней кромки), равный половине ширины, составит 175 пикселей. Четвертая строка задает шкалу дуги. Расшифровывается она следующим образом: настоящие данные представлены на линейной шкале от нуля до суммы всех голосов, и эту шкалу необходимо преобразовать в шкалу кольца, которая имеет диапазон от 0 до 2π радиан (или от 0 до 360°).

Затем идет цветовая шкала. Чем больше голосов получила та или иная категория, тем насыщеннее должен быть красный цвет. Protovis способен подобрать цвета за вас. Вам только нужно определить, какую цветовую гамму вы предпочитаете.

```

var depthColors = pv.Scale.linear(0, 172).range("white", "#821122");

```

Теперь у вас есть цветовая шкала от белого до темно-красного (#821122), соответствующая линейному диапазону от 0 до 172 (наибольшего количества голосов, набранного одной категорией). Иными словами, категория, получившая 0 голосов, будет белого цвета, а категория с 172 голосами — темно-красного. Категории с промежуточным количеством голосов будут и по цвету где-то между белым и красным.

С переменными покончено. Размер и шкала определены. Чтобы нарисовать собственно диаграмму, сначала нужно создать чистую панель с размерами 350 (w, или ширина) на 350 (h, или высота) пикселей.

```
var vis = new pv.Panel()
    .width(w)
    .height(h);
```

Затем панель нужно наполнить содержимым, в данном случае сегментами. Следующий фрагмент может показаться вам несколько запутанным, но ниже мы разберем его строчка за строчкой.

```
vis.add(pv.Wedge)
    .data(data)
    .bottom(w / 2)
    .left(w / 2)
    .innerRadius(r - 120)
    .outerRadius(r)
    .fillStyle(function(d) depthColors(d)) .strokeStyle("#fff")
    .angle(a)
    .title(function(d) String(d) + " votes") .anchor("center").add(pv.Label)
        .text(function(d) cats[this.index]);
```

Первая строка сообщает, что вы добавляете сегменты на панель по одному для каждого пункта в массиве данных. Свойства `bottom()` и `left()` ориентируют сегменты так, чтобы их верхушки оказались в центре круга. `innerRadius()` определяет радиус дырки «бублика», а `outerRadius()` — радиус всей окружности. Этим задается структура кольцевой диаграммы. При указании стиля заливки цвета берутся не какие-то фиксированные, а определяются значением каждого конкретного элемента данных, после чего цветовая шкала сохраняется как `depthColors()`. Граница задана с помощью `strokeStyle()` — она белая (#fff). Круговая шкала сама способна определить угол каждого сегмента.

Для создания контекстного окна (тултипа), которое при наведении на сегмент указателя мыши должно сообщать, сколько человек отдали свои голоса соответствующей категории, используется `title()`. Можно было бы вместо этого создать событие «`mouseover`», чтобы определить, что именно должно происходить, когда указатель мыши оказывается над объектом. Но поскольку браузер автоматически показывает значение атрибута `title` (название), проще использовать именно его. В приведенном выше фрагменте название состоит из значений элементов данных и слова "votes" (голоса). Далее идут подписи к каждому сегменту. Единственное, что остается сделать — это добавить текст "May 2009" (май 2009 г.) в центре диаграммы:

```
vis.anchor("center").add(pv.Label)
    .font("bold 14px Georgia")
    .text("May 2009");
```

Этот фрагмент можно прочитать как команду вставить в центр диаграммы подпись полужирным шрифтом Georgia 14-м кеглем «Май 2009». Диаграмма выстроена, можно ее визуализировать:

```
vis.render();
```

Когда вы откроете файл `donut.html` в браузере, вы должны увидеть то, что изображено на рис. 1 (Internet Explorer не справится с задачей).

Если вы новичок в программировании, эта заметка могла показаться вам чересчур сложной. Но есть и хорошая новость: *Protovis* разработана так, чтобы ее можно было изучать на примерах. На сайте этой библиотеки вы найдете множество примеров кода, которые вы сможете использовать и для обучения, и в работе с вашими собственными данными. Там есть и традиционные формы статистической графики, и более продвинутые образцы интерактивной и анимированной графики.