

данный момент трудно адаптироваться к чему-то или кому-то новому. *Зубчатые, неровные* линии выявляют дерзость и враждебность. А если линии *плавные, округлые* – то вы человек мягкий, спокойный, идущий на компромисс. *Грубые линии* характерны для человека, находящегося в эпицентре сложного конфликта или проблемы.

Сила нажима. С помощью нажима можно оценить характер внешних эмоционально-волевых проявлений автора (рис. 3).



Рис. 3. Анализ по нажиму

Как в почерке, так и в рисунке *сильный нажим* присущ людям эмоционально активным, энергичным. Те, кто рисует со *слабым нажимом*, пребывают в эмоционально сниженном настроении, не

особо активны в эмоциональных проявлениях. *Импulsивный, непостоянный* указывает на нервность, импульсивность, лишнюю эмоциональность при принятии жизненно важных решений, порывистость.

Заключение

Таким образом, мы можем видеть, что рисунок – это всегда какое-то сообщение, зашифрованное в образах. Каждый элемент и завиток несет в себе какую-либо информацию о характере автора и его эмоциональном настрое. Поэтому рисунок – это один из способов заглянуть во внутренний мир человека и глубже понять самого себя.

Литература

1. А.Л. Венгер Психологические рисуночные тесты. Иллюстрированное руководство, издательство ВЛАДОС-ПРЕСС, 2003;
2. Ваши рисунки как портрет личности [Электронный ресурс]. Режим доступа <http://mariapugacheva.ru> свободный.
3. Потемкина О.Ф., Потемкина Е.В. «Психологический анализ рисунка и текста», Издательство Речь, 2006.

COMPARATIVE REVIEW AND ANALYSIS OF VISUALIZATION FRAMEWORKS FOR JAVASCRIPT

A.P. Matveev, A.S. Korovin
Tomsk Polytechnic University
Lenina Avenue, 30, 634050, Tomsk, Russia
E-mail: rakch@sibmail.com

Introduction

It is undeniable that web technologies are very popular and they have very big impact on many sectors of human activity. These technologies allow us to send big arrays of data across the web and store and show it in a proper way. But it is very difficult for scientists to analyze raw data. So, we have to find ways to represent data in an aesthetically pleasing and comprehensible form. Instead of using boring excel tables, data can be presented in a more creative format with the help of visualization tools.

As the power of web browser increases, the creation of complex visualization of data via the web becomes possible. So, now we can easily visualize our data in a very effective way. But it is difficult to find an ideal framework, because each of them has its own specific features. People prefer libraries with the following features: reliability, quality of support and documentation, popularity, efficiency in information binding, appearance, performance, simple API, weight of library, etc.

The most recognized visualization frameworks in the world are RGraph, Canvas.js, Highcharts, Google charts, Dygraphs, D3JS. Let's consider pluses and minuses of each one in detail. Here you can see example of the standard chart (fig. 1).

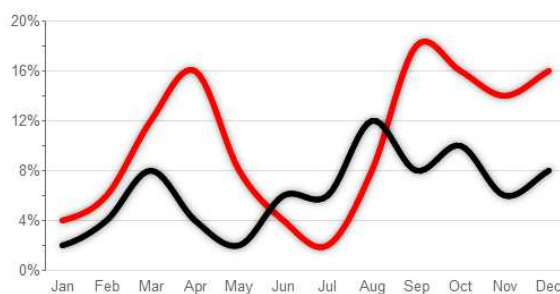


Fig. 1. Example

Rgraph

One of the most popular frameworks is RGraph. It is a free HTML5 and Javascript based library, it uses the HTML5 canvas tag and it leads to smaller page sizes, lower costs and faster pages loading. Rgraph provides high performance. This product is reliable, because it was created in 2008 and has been developing until now. It also provides many supported tech-

nologies, simple drawing API and a lot of tutorials, 20+ chart types.

One of the main features is dynamic update, because data changes every time and chart has to simultaneously change itself with the data. Also, it has integration with MySQL and PHP, it allows users to get data from databases and instantiate and draw a chart according to this data. By using AJAX requests users can easily fetch data from an XML or JSON file.

D3JS

A famous and efficient visualization library uses the wide range of technologies: HTML5, JavaScript, Scalable Vector Graphics (SVG), CSS3 technologies. This library is used by companies such as: “The New York Times”, “Datameer”, “OpenStreetMap Foundation” and others. D3 allows you to bind data to a Document Object Model (DOM), and then apply data-driven transformations to the document. As a simple example, you can use D3JS to create an HTML table from a JSON data. Or, you can use the same data to create an interactive SVG pie chart with visual effects.

It supports many charts, and a lot of effects, non-standard plots, but it has really low performance, which makes a really big minus to this library.

CANVAS.JS

CanvasJS is an easy to use JavaScript library, which was built on HTML5 canvases. It supports all devices: iPhone, iPad, Android, Windows Phone, Microsoft Surface, Desktops, etc. This allows you to create powerful charts that work on each device, without additional maintenance or functionality of your web application. CanvasJS has many beautiful themes and it is faster than Flash and SVG Charts – result of this is lightweight, beautiful and responsive dashboards.

CanvasJS has a lot of chart types: Area, Bar, Bubble, Column, Doughnut, Line, Pie, Spline, Spline Area, Scatter (Point), Stacked Area, Stacked Bar, Stacked Column, Step Line. Some features of CanvasJS are: the library has a simple API and beautiful themes, which are set, by changing one parameter; the performance is very high – CanvasJS processes 100,000 Data-Points per just around 100 milliseconds. It's really good if you need high speed of data processing.

To sum up, CanvasJS is very simple to use, well documented, it works across all devices and browsers, has awesome performance, and is supported directly by developers.

Highcharts

Highcharts is a charting library written in HTML5/JavaScript, offering intuitive, interactive charts to your web application. Library works across many devices, but it has problems with mobile browsers.

It has a lot of charting tools, some of them: Line, Spline, Area, Area spline, Column, Bar, Pie, Scatter, Angular gauges, Arearange, Areasplinerange, Column range, Bubble, Box plot, Error bars, Funnel, Waterfall and Polar.

Charting tools use SVG for the graphics rendering. Additionally Internet Explorer graphics are drawn using VML.

Highcharts is solely based on native browser technologies and does not require client side plugins like Flash or Java. Setting the Highcharts configuration options is very easy; you can be a not experienced user to make it. Through working with your web application, you can add, remove and modify series and points or modify axes at any time after chart creation. Sometimes you want to compare variables that are not of the same scale - for example temperature versus air pressure. If you turn on exporting module, your users can export the chart to PNG, JPG, PDF or SVG format by the click of a button, or print the chart directly from the web page. By zooming in on a chart you can see directly what you need.

Google Charts

Google Charts is an extremely powerful library that can effectively visualize data on your web application. It is completely free for all purposes: commercial, personal, educational or even governmental. It provides many chart types like line charts, bar charts, pie charts, Venn diagrams and complex hierarchical tree maps.

Google Charts is highly customizable, so you can easily change the style of your chart by selecting its size and colors and ignore these options to draw a method. It makes charts more attractable and understandable for users. Charts are highly interactive and expose events to interact with users.

It has really full and readable documentation; it supports many technologies such as JSON, XML and AJAX to load data. It has a lot of tutorials, demos, big user community, so, it is not difficult to find a solution to your problem. Google Charts has a class DataTable which provides a lot of methods to work with databases and representation of data. Charts are rendered using HTML5 and SVG technology to create scalable charts with resolution independence and provide cross-browser compatibility (including VML for older IE versions) and cross platform portability to iPhones, iPads and Android.

Dygraphs

Dygraphs is a flexible open source JavaScript charting library that produces interactive, zoomable charts of time series. It allows exploring and interpreting dense data sets. It supports many technologies like CSV, AJAX, JSON and XML. Dygraphs is highly compatible. It means that it works with all popular browsers (including Internet Explorer 8) and modern mobile devices such as tablets or smartphones. But

there are few types of charts, just area, bar, candlestick and line charts.

There are many companies which use dygraphs such as NASA, Wikimedia Foundation, Eutelsat and Google. Although it has active developing community, the community is small. However, dygraphs library is fully tested in many cases, so there are a lot of examples. Furthermore, you can find well-documented API and many demonstrative tutorials on the site.

It is capable of handling huge data sets. You can easily bind millions of points without errors and bugs. Furthermore, it is highly interactive, so you can attach many events to charts like zoom, pan, mouseover. Finally, one of the main features is that the library is lightweight, because it allows pages to load faster.

Summary

To sum up, we have to say that all libraries have their own good sides. But, first of all, we want to underline two charting tools: Google charts, and rGraph,. They are more optimal in terms of all features, because other libraries, for example, Canvas.js can boast their performance, but its other aspects are very poor developed, D3 has awesome visualization but low performance. If we consider that javascript

and jquery are executed on the client side (browser), we can have big problems with the speed of operation. Google charts and rGraph are the most suitable in each aspect of the features. Google charts has date time axis and it can combine charts; another good feature is a data table class, which enables to work with data and change the view of this data easily, it supports query class for work with databases, which is not the feature of RGraph. However, if we need higher performance, RGraph is more suitable to solve this problem.

References

1. <http://www.rgraph.net/>
2. <http://canvasjs.com/>
3. <http://d3js.org/>
4. <http://www.highcharts.com>
5. <https://developers.google.com/chart/>
6. <http://dygraphs.com/>
7. <http://www.fusioncharts.com/javascript-charting-comparison/>
8. <http://blog.collideous.com/post/35361681763/highcharts-vs-google-charts-after-ive-been>
9. <http://socialcompare.com/en/comparison/javascript-graphs-and-charts-libraries>

СОЗДАНИЕ БУКЛЕТА ИК

Мухамадеев Р.Г., Ризен Ю.С.
Томский политехнический университет
634050, Россия, г. Томск, пр-т Ленина, 30
E-mail: flambergtheone@mail.ru

Введение

Институт кибернетики (ИК) образован в 2010 году в рамках структурных преобразований ТПУ в связи с присвоением ему статуса национального исследовательского университета. ИК основан на базе института «Кибернетический центр» и ряда подразделений машиностроительного и электрофизического факультетов. Изменения в организационной структуре института потребовали обновления рекламной продукции. Рекламная продукция ИК разрабатывается для нескольких категорий людей: абитуриенты, партнеры и т.д. В связи с чем в руководство ИК предложили разработать новый фирменный стиль и разработать «свежий» буклет для абитуриентов всех направлений института. Буклет должен быть узнаваем и достаточно прост в графическом отображении, а также отражать специфику работы (миссия Института кибернетики ТПУ – создание, распространение и применение знаний в области прикладной математики, вычислительной техники, информационных технологий и систем управления в процессе выполнения фундаментальных и прикладных исследований, подготовки и переподготовки инженерных и научных кадров).

Процесс создания

Буклет разрабатывался в графическом редакторе Corel Draw от начала, до конца. Важной частью создания буклета являлось сохранение цвета института и логотипа, но буклет не должен был иметь холодной гаммы или быть мрачным и серым. Обычно все проекты принято начинать с эскиза или зарисовки, но в данном случае все наброски целесообразнее было делать сразу в графическом редакторе, выбирая композиционные решения, меняя все элементы местами. Основной акцент в буклете ставился на максимальное количество наглядной информации.



Рис. 1.1. Начальный вариант буклета