

Responsive Web Design

Kailashkumar V. Natda

Assistant Professor, Department of Information Technology,
Om Vindhya Vasini College of IT & Management, Morbi
Email : kevin@omvvim.edu.in

Abstract:

One of the revolutionary and newest web design trend developing at this time is responsive web design. We have seen last couple of years the growth of handheld devices like Smart Phone, tablet and other gadgets in. According to research and sales statistics of tablet and Smart Phone has taken over desktop and notebook in 2012. This growth will continue through next some years. increasing more consumers adopt mobile internet technologies in the coming years, those increase people are consume online web content from various devices like Smart Phones, tablets, and other connected devices. Therefore each marketing or technical manager need to know about this web design trend, and this is right time to adopt this approach for growth of your business. Fixed screen size website design is no longer great this time. Responsive Web Design is an approach to web design in which a site is built to deliver an ideal viewing experience, easy reading and navigation with a minimum of resizing, panning, and scrolling across a wide range of devices like desktop, notebook, tablet, Smart Phones and other gadgets.

Keywords:

Breakpoint, Flexible Image, Fluid Layout, HTML

What is responsive design?

Your website looks great on the desktop screen but may not be true for every size of screen or display. Once you have developed the responsive design, the website will look decent and readable on every size of screen or display, not just for desktop.

The basic idea behind responsive design is that a website should be gracefully "respond to" every size of screen or display of devices like Smart Phone, tablet, notebook or any size desktop monitor.

The term responsive design was invented by Ethan Marcotte who identifies three fundamental techniques for responsive design fluid grids, media queries and flexible images & media.

There are three basic technologies HTML, CSS & JavaScript behind responsive design. And there are fundamental techniques for responsive design fluid grids for flexible layouts, media queries to help you adapt content to specific screen sizes, and flexible images and media that respond to changes in screen sizes as well.

Why responsive design? We have seen during last couple of years an explosion of hand held devices beating the market including tablets, e-readers, and Smart Phones, each consuming its own screen resolutions, orientations, and layouts. As this growth continues through next some years, people expect to view online web content as an ideal viewing experience, easy reading and navigation with a minimum of resizing and scrolling across a devices like tablet, Smart Phones and other gadgets and responsive web design will persevere and continue to expand. According to the latest NPD Display Search figures, tablet shipments in 2016 will top 35 Crores units, putting them ahead of notebook shipments for the first time ever you can see from below given chart.

In survey on internet-enabled Smart Phones in India Mid 2012 probable that the number of internet-enabled Smart Phones in India would reach 2.5 Crores in 2012, which is more than double of 2011 and five time more 5.8 Crores in 2013 growth is

continues through 2016 you can see from below given chart. According to Cyber Media research India's tablet PC market is expected double to around six million units in 2013, firm Cyber Media. From Microsoft Tag mobile internet usage is projected to overtake desktop internet usage by 2014 (Source: Microsoft Tag)

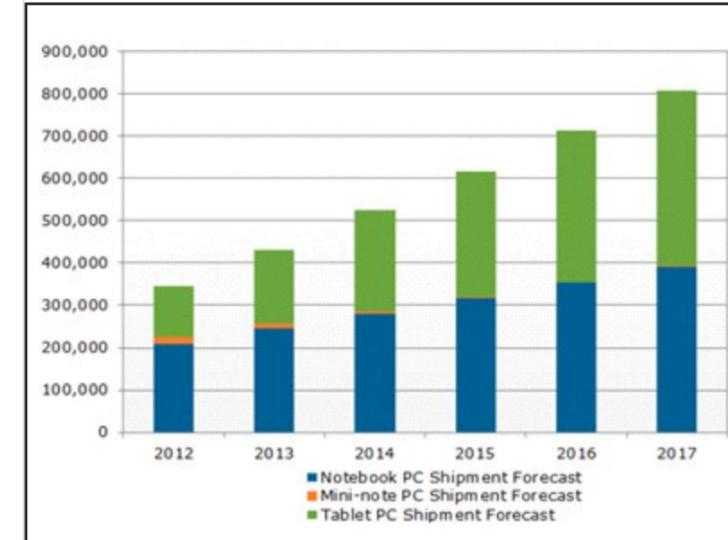


Fig. 1

Source: NPD Display Search Quarterly Mobile PC Shipment and Forecast Report

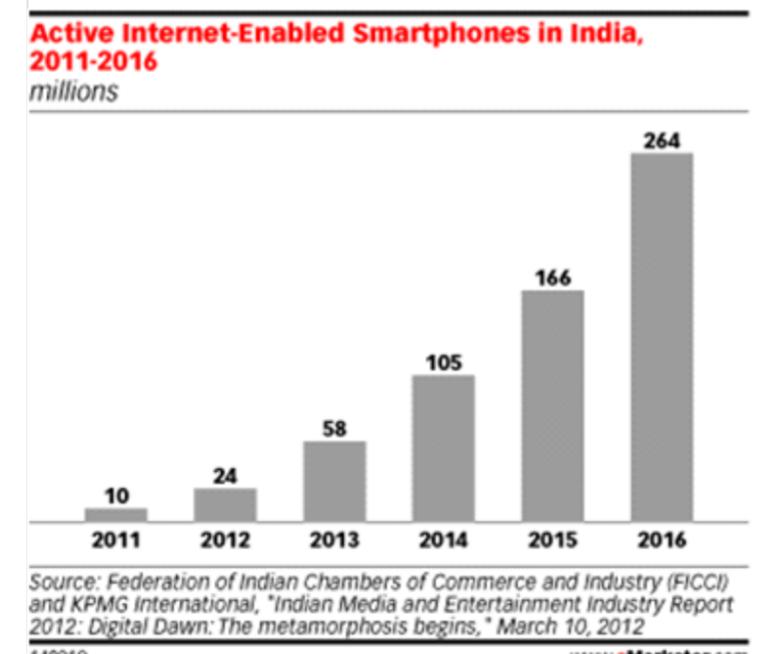


Fig. 2

From above specified research and statistics shown, the coming era will be of hand held devices like Smart Phone, tablet and other gadgets. So you should develop online contents and must be responsive and site is

constructed to provide an optimal viewing experience, easy reading and navigation with a minimum of resizing, slating, and scrolling across a wide range of devices. Thus responsive design sign is much needed in this time.

Fluid grids layout

Fluid grids layout is the first technique behind responsive design. Fluid Grid layouts allow you to use media queries to determine the break points where a site begins to resize the different elements, such as headers, images, etc. Fluid Grid Layout enables you to specify the look of your site across different devices. You can adopt the structure of your site for desktop screens, for tablets, and for smaller mobile devices.

Fixed-width layout is constructed by specific pixel measurement. This type of layout is commonly used for web design. Fluid grid layout is constructed by percentage, instead of pixel. How fluid grid layout works against fixed-width layout we can understand with the help of this side by side example

Fixed Layout css by example	Fluid Layout css by example
div { width: 900px; }	div { width: 96%; } (if parent is 1000px wide) $100 / 1000 = .1$

You have to calculate the proportions for each page element, you must divide the target element by its context. First you should create a high fidelity model in a pixel based imaged editor, like Photoshop or Illustrator. Then you can measure a page element and divide it by the full width of the page. In order to calculate the extents for each page element, you must divide the target element by its context can calculate with this formula given below:

target / context = result

How this formula is used, we can understand by this example

```
body {  
font: normal 100%;  
}
```

With 100% font size all elements in our web page relative to the browser's default font size which is normally 16px. Let us assume that the body default font size to be 16px, we can apply such font-size on this formula. We have to divide the target value (32px) by the font-size of its container (16px):

$$32 \div 16 = 2$$

So header H1 is two time of its default body font. We can denote by em (1em is equal to the current font size), we can put into our css.

```
h1 {  
font-family: font-family: Helvetica, Arial, sans-serif;  
font-size: 2em; /* 32px / 16px = 2em */  
}
```

Fluid grids are the very key parts of building a responsive design. When the width of the browser becomes too small, the design can start to breakdown harshly. This problem is solved by another technique of responsive design known as media queries. Before going on media queries let us see what is breakpoint? And viewport?

A **breakpoint** indicates the moment your layout changes from one layout to another, and is generally generated by the width of a screen.

Let us sight below example. When I resize the screen, you can see the containing block changes both when the viewport reaches a certain size. The screen sizes are based on the typical screen sizes for Smart Phones, tablets, notebook and desktop devices. While many layouts will have more than these four basic breakpoints, almost all responsive designs will feature some combination of those four screen size ranges. This means that one of the first things that you need to do when planning a responsive design, is define the breakpoints for your layout. Here you see a list of the most common breakpoints. Mobile styles will target any screen smaller than 480 pixels wide, while tablet styles will usually target screens between 481 and 768 pixels, tablet screen (wide) between 768 pixels and 960 pixels. Desktop styles, will target any screen wider than 960 pixels.



Source: <http://dustycartridge.com/>

Viewport

A viewport is the area in which your web page is displayed. The default viewport width on Mobile browser is 980px which is support to design layout for desktop. When you are creating a responsive web design for mobile or laptop or other device you want the viewport to be equal to the physical width of your device. That is where the convenient device-width property comes into

play. Device-width is the width of your device in pixels. You can use the following viewport tag:

```
<meta name="viewport" content="width=device-width">
```

If your device-width is 320px then your viewport's width has now been set to 320px. That means if your website's layout is wider than 320px then a portion of your layout will not be visible within the mobile device's viewport

Media Queries

A **media query** contains of a media type and as a minimum one expression that bounds the style sheets' possibility by using media features include width, height, max-width, max-height, device-height, orientation, aspect-ratio, resolution and more

W3C improved in CSS3, media queries let the performance of content be tailor-made to a specific range of output devices without having to change the content itself.

Media queries help you deliver contents with different styles adapted to devices like desktop, notebook, tablet, Smart Phone and all other gadgets with different screen size

How to implement media queries,

The general form of a media query is,

```
@media [not | only] type [and] (expr)
```

```
{rules  
}
```

How media queries work for different screen size understand by following example

```
@media only screen and (min-width: 320px) { /* MOBILE  
PORTRAIT */
```

```
Some code .....
```

```
}
```

```
@media only screen and (min-width: 480px) { /* MOBILE  
LANDSCAPE */
```

```
Some code .....
```

```
}
```

```
@media only screen and (min-width: 600px) { /* SMALL  
TABLET */
```

```
Some code .....
```

```
}
```

```
@media only screen and (min-width: 768px) { /*  
TABLET/NETBOOK */
```

```
Some code .....
```

```
}
```

```
@media only screen and (min-width: 1024px) { /*  
LANDSCAPE TABLET/NETBOOK/LAPTOP */
```

```
Some code .....
```

```
}
```

```
@media only screen and (min-width: 1280px) { /*  
DESKTOP */
```

```
Some code .....
```

```
}
```

??

```
@media only screen and (min-width: 1400px) {/*
WIDESCREEN */
    Some code .....
}
@media print { /*PRINT */
    Some code .....
}
```

With CSS media queries, you can easily target particular screen sizes and provide excellent user experience on any type of browser or device used to visit your website. This technique is the center of responsive design, which is a developing responsive and device friendly web design and development experience. Visitors like to browse your website on tablets, phones, and e-readers. They would like to visit your website more.

Responsive graphics and other media

Using fluid layout, ensures that screen size or orientation changes and contents will reflow or reconfigure along with it. That works seamlessly for contents like text which naturally reflows along with its containing element. But what about content that is by default defined by fixed size, such as graphics or other media?

An image designed for a large desktop monitor is unlikely to fit into a layout designed for a 320-pixel-wide Smart Phone. Likewise, images designed to accompany content on a mobile screen would probably look like thumbnails when viewed within desktop layouts. This means that in the best of circumstances, these images will look out of context and in many cases, will break your carefully crafted responsive layouts. This problem underscores how important it is to find ways to make images and other media responsive. The solution is pretty simple: just replace the fixed dimensions with relative measurements or percentages and then boom, flexible responsive images. In example CSS-driven approach. Namely, all modern browsers get our max-width constraint.

Example 1.2 when you decrease screen size image or other media like video or flash object also decreases respectively *Image, embed, object, video*

```
{ max-width: 100%;  
}
```

Internet Explorer 6 and below don't support the max-width property. But in a separate IE6-specific style sheet, I'll include the following:

```
img,embed,object,video  
{ width: 100%;  
}
```

To utilize JavaScript or server-side scripting to examine the page, determine the context, and then serve the appropriate image. File, and add a query string on any image that requires a larger desktop version of that image. The script will then parse the page and replace

the smaller images with the larger ones when they are appropriate. There are many solutions like this one that have been developed over the last couple of years, and most of them share this basic approach have script determine the context, then parse the page and serve the appropriate images work within responsive designs.

Conclusion

This article will help you to know about responsive web design and how to implement its techniques in field of web design. Also described how you can provide an optimal user experience across a wide range of devices like desktop, notebook, tablet, Smart Phones and other gadgets. This article also provide an idea, how marketing manager or businessman adapting this techniques of responsive web design to develop their sites. Responsive Web Design is a great solution for website that provides great, user experience with the wide variety of resolutions.

References:

Offline:

Marcotte, E. (2011). *Responsive Web Design* by (1st edition ed.). New York: A Book Apart.
Kadlec, T. (2012). *Implementing Responsive Design: Building sites for an anywhere, everywhere web* (1st edition ed.). Berkeley: New Riders.

Online:

Responsive web design - Wikipedia, the free encyclopedia. (n.d.). Retrieved January 11, 2013, from http://en.wikipedia.org/wiki/Responsive_web_design
Responsive Design 101: How Your Business Can Benefit | Business 2 Community. (n.d.). Retrieved January 12, 2013, from <http://www.business2community.com/online-marketing/responsive-design-101-how-your-business-can-benefit-0370674#Zmj2E77Kiugs1S4m.99>
Getting Started with Responsive Web Design - A Dummies Guide. (n.d.). Retrieved January 14, 2013, from <http://www.labnol.org/internet/responsive-web-design-faq/21361/>
Beginner's Guide to Responsive Web Design - Treehouse Blog. (n.d.). Retrieved January 16, 2013, from <http://blog.teamtreehouse.com/beginners-guide-to-responsive-web-design>
HTML5 - Responsive Web Design. (n.d.). Retrieved January 18, 2013, from <http://msdn.microsoft.com/en-us/magazine/hh653584.aspx>
Dusty Cartridge - Game Reviews, Previews, Videos and Podcasts. (n.d.). Retrieved January 18, 2013, from <http://dustycartridge.com/>
The Growth of Mobile Marketing and Tagging | Blog | Microsoft Tag. (n.d.). Retrieved January 19, 2013, from http://tag.microsoft.com/community/blog/t/the_growth_of_mobile_marketing_and_tagging.aspx