SPEAKING A COMMON LANGUAGE:

Terminology & concepts in adaptive and responsive website design

Excellence in Software Engineering

June 2013

Introduction

The treatment and development of various iterations of HTML content presentation has been an ongoing area for reflection, conversation and debate. More so, the distinction and definition of fixed, fluid, adaptive and responsive layouts on the basis of a simple web page has received a lot of attention in the development community. Although a lot of great discussion has ensued, it's obvious that our community doesn't share a common understanding of the principal differences of these similarly-named approaches to web development. And often, we do not mean the same things when we say "responsive" or "adaptive". This problem is observable worldwide – to explore on your own, simply Google "adaptive vs. responsive web design" to see the diversity of terminology understanding in various posts.

This document will delineate the terminology of responsive and adaptive and explain the principal differences between layouts and web design approaches, in order to help our community achieve a common understanding and basis for discussion.

Types of HTML Layouts

> Note: for the purposes of this discussion, "layout" refers to an HTML page, not an image in PSD format.

Fixed Layout

Toolset: px

In a fixed layout, the content width is strictly defined in pixels and doesn't change depending on the size of the user's browser window.

Fixed layout development is mostly a relic of the past (common exceptions include small promotional websites), when the width of the page's content was determined by the resolution of the most popular screens (800x600, then 1024x768, and so on). Although websites with content width set to 960 pixels exist even today, these sites typically have not been updated for five (or more) years.

Fluid Layout

Toolset: %

Fluid layouts are often also called "liquid" or "stretchy", as the content resizes to any screen size due to the use of relative measures (most often, %) for structural elements (instead of static ones, in pixels).

This type of layout is also outdated as it takes into account only one type of device (the PC) and does not address how the content will look on screens with critically wide or narrow width, such as a smartphone.

Adaptive Layout

Toolset: px + media-queries

Adaptive layouts and responsive layouts (see below) are very similar, in that they both use media queries to adjust the content to different screen sizes.

The main difference from the responsive layout is that in adaptive layout a page "jumps" to breakpoints, selecting the closest one and shifting and adapting the content to display appropriately. This means that media queries describe fixed content positioning for each breakpoint. As a result, the "page" actually consists of several fixed layouts for different screen resolutions.

The disadvantage of this approach to page layout is obvious: since the spacing between breakpoints can be quite large, we can't predict how content will look on all devices.

This approach is most relevant if the critical breakpoints do not cover the width of the most widespread devices, but instead are determined by the design of the page itself.

Responsive Layout

Toolset: % + media-queries

Responsive page layout, in contrast to adaptive layout, is based on the fluid principle, but it also uses media queries to adapt the content to

screen width.

As a result, a responsive page doesn't jump to breakpoints, but adjusts smoothly between them.

Mixed Layout

Toolset: px + % + media-queries

Of course, there are other solutions for creating layouts by combining the styles outlined above.

For example, adaptation to different devices is sometimes achieved by leveraging aspects of both adaptive and responsive page layout. Such an approach is often taken when a developer is asked to optimize an existing website for non-PC devices (sometimes known as "mobile last" because optimization for mobile wasn't included in the original development plan).

Approaches to Creating Websites and Web Applications

Now, let's consider approaches and strategies for creating websites. This is something bigger than simple page layout, as it will often include solving user interaction problems, page restructuring or even adaptation of the content provided to the user.

Responsive Web Design

Toolset: % + media-queries + %-media

Ethan Marcotte coined the term "responsive web design" (RWD) in a May 2010 article in A List Apart. This approach takes its name from the discipline of responsive architecture. Responsive architecture of a building means that the structure itself will adapt its form, color or other factors depending on environmental conditions – including the people inhabiting it.

In 2011, Marcotte published a book on responsive web design, in which he assembled some long-known content adaptation patterns into a single technique. Rather than creating something entirely new, what he did was systemize the knowledge and recommended solutions for problems on old devices and browsers. This type of thinking was incredibly useful to address the era's multitude of operating systems, devices and browsers.

Responsive web design uses only the means of HTML and CSS to solve the task of adapting content to different devices. The approach combines fluid layouts (usually utilizing a grid system), media queries and flexible media elements (images, videos). And in certain cases, a developer can also leverage the power of server-side as well.

Ultimately, responsive web design is responsive page layout plus responsive images and video.

Adaptive Web Design

Toolset: % + media-queries + %-media + JavaScript + magic

Before we discuss adaptive web design, it may be helpful to review the other strategies, concepts and techniques that this approach is based on: namely, "mobile first" and "progressive enhancement".

Progressive enhancement is a strategy in web design that is based on content availability and semantics at the lowest level, while treating techniques such as stylization (CSS) and additional scripts (JavaScript) simply as add-ins that

make life easier. Stephen Champeon introduced this strategy in 2003 at SXSW. In contrast to the concept of graceful degradation, in which developers focus only on modern browsers and "trim" possibilities for older browsers, progressive enhancement starts from the reverse side to make content available for most devices and browsers. As this problem is quite prevalent nowadays, this strategy was even named Top #1 trend in web design in 2012.

In already distant 2011, Luke Wroblewski – with help from "A Book Apart" – published Mobile First, which soon became a bestseller. The "mobile first" strategy assumes that all development, from planning of business goals, UX and web design to the last line of code, starts from the most compact (e.g. mobile) version and ends with desktops, game consoles or TV-sets.

Aaron Gustafson describes adaptive web design in his book of the same name. This approach takes the best aspects of "mobile first" and "progressive enhancement" and complements them. To put it simply, this strategy assumes the use of pure HTML as the basis and further improvements and features by using CSS and JavaScript layers. If, for a specific browser or device, a new feature becomes available, then it can only be used after availability tests and maintaining backward compatibility with unsupported devices (for example, touch events).

> Pure HTML works everywhere.
>
> Someone at some conference somewhere

Some in the development community have argued recently that content does not need to be universally adapted for "all" devices and instead we can make assumptions about the content sought based on the type of device being used. It may be true that, in most cases, a person who accesses a website from an ancient phone is seeking basic information like an address or phone number. Whereas, a person who enters a site from a Smart TV is more likely to want to read the news or watch videos. Such flexibility in the content delivered to the user can be regarded as a strategy of adaptive web design, but it would be foolish to think that we can accurately judge every user's needs based simply on his or her device, its characteristics set and supported features, or even connection speed.

Finally, it should be noted that adaptive web design is the strategy for creating websites and web applications, while responsive web design is the technique for adapting the layout to different devices – and is often itself a part of adaptive web design.

Conclusions

Progressive enhancement, as an approach, has existed since 2003. Fluid layouts and media queries are even older (first mention was published as far back as April 2001). But the problem of optimization to various devices and platforms has become critical with the proliferation of smartphones, tablets and other non-PC devices, such as Smart TV and web-enabled game consoles. The acceleration of creation and development of web standards, and also systemization of approaches in web design, has allowed our community to address this challenge to some extent. But reaching a common language and common understanding of what is meant when we say “adaptive” or “responsive” is an important part of that systemization process.

<epam> Excellence in Software Engineering

Established in 1993, EPAM Systems (NYSE: EPAM) provides complex software engineering solutions through its award-winning Central and Eastern European service delivery platform. Headquartered in the United States, EPAM employs over 8,700 IT engineers and serves clients worldwide from its locations in the United States, Canada, UK, Switzerland, Germany, Sweden, Belarus, Hungary, Russia, Ukraine, Kazakhstan, and Poland.

EPAM is recognized among the top companies in IAOP's "The 2012 Global Outsourcing 100", featuring EPAM in a variety of sub-lists, including "Leaders – Companies in Eastern Europe". The company is also ranked among the best global service providers on "The 2012 Global Services 100" by Global Services Magazine and Neogroup, which names EPAM "Leaders - Global Product Development" category.

For more information, please visit www.epam.com

Global

41 University Drive Suite 202,
Newtown (PA), 18940, USA
Phone: +1-267-759-9000
Fax: +1-267-759-8989

EU

Corvin Offices I. Futó street 47-53
Budapest, H-1082, Hungary
Phone: +36-1-327-7400
Fax: +36-1-577-2384

CIS

9th Radialnaya Street, bldg. 2
Moscow, 115404, Russia
Phone: +7-495-730-6360
Fax: +7-495-730-6361