Адаптивная вёрстка: что это и как использовать



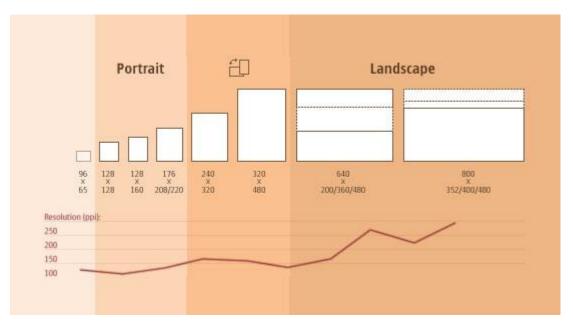
В наши дни практически каждый заказчик хочет получить мобильную версию для своего сайта. И это естественно: нужен один дизайн для iPhone, другой для iPad — и он также должен подходить под все другие размеры экранов.

В теме веб-дизайна и разработки мы быстро приближаемся к точке, когда мы будем не способны справляться с бесконечными новыми разрешениями и экранами. Для многих вебсайтов создание отдельной версии для каждого нового разрешения будет невозможным или, по крайней мере, не практичным. Должны ли мы смириться с потерей посетителей, для которых у нас нет верстки? Или можно что-то сделать?

Адаптивная верстка — подход, предполагающий изменение дизайна в зависимости от поведения пользователя, размера экрана, платформы и ориентации девайса. Другими словами, страница должна автоматически подстраиваться под разрешение, изменять размер картинок и т.д. Это позволит устранить нужду в разработке дизайна для каждого нового устройства, появляющегося в продаже.

Регулировка разрешения экрана

Можно разбить устройства на разные категории и верстать для каждой из них отдельно, но это займет слишком много времени, и кто знает, какие стандарты будут через пять лет? Тем более, согласно статистике, основанной на 400 проданных с 2005 по 2008 год устройствах (а сейчас 2016), мы имеем целых спектр разнообразных устройств:



Очевидно, что мы не сможем продолжать верстать для каждого устройства отдельно. Но что тогда делать?

Частичное решение: делаем всё гибким

Конечно, это не идеальное решение, но оно решает большую часть проблем.

В своей статье Итан Маркотт (Ethan Marcotte) создал простой шаблон, демонстрирующий использование гибкой вёрстки:



Весь дизайн — микс адаптивных слоев, картинок и в некоторых местах умной разметки. Создание адаптивных слоев — частая практика, что нельзя сказать об адаптивных картинках. Поэтому ниже представлены техники для реализации гибких картинок:

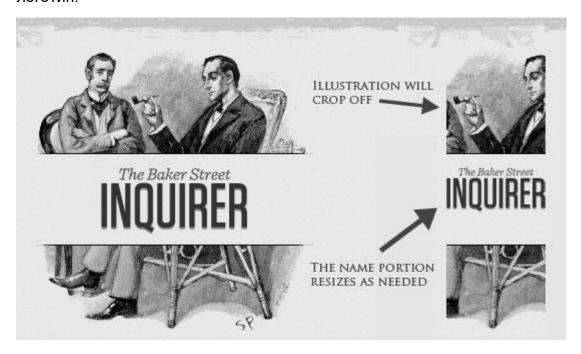
Hiding and Revealing Portions of Images;

Creating Sliding Composite Images;

Foreground Images That Scale With the Layout.

Для более детальной информации рекомендуем ознакомиться с книгой Зои Микли Гилленуотер (Zoe Mickley Gillenwater) «Flexible Web Design: Creating Liquid Layouts with CSS» и загрузить главу «Creating Flexible Images».

С первого взгляда может показаться, что все легко, но это не так. Взгляните на логотип:



Если сделать разрешение слишком маленьким, может показаться, что четкость картинки упала, но это необходимо для того, чтобы сохранить пропорции названия. Поэтому картинка поделена на две части: первая часть (иллюстрация) используется как фон, вторая (логотип) изменяет свои размеры пропорционально.

Элемент h1 содержит изображение в качестве фона, и картинка выровнена согласно фону контейнера (заголовка).

Гибкие изображения

Одна из самых главных проблем, которая нуждается в решении при работе с адаптивным дизайном — это работа с картинками. Существует много способов изменять размер изображений, и большинство из них реализуется очень просто. Одним из таких примеров является использование max-width в CSS:

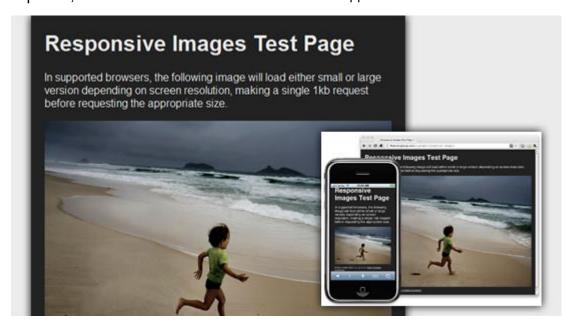
```
img {max-width: 100%;}
```

Максимальная ширина изображения равняется 100% от ширины экрана или окна браузера, и, когда ширина уменьшается, то же делает и картинка. Заметьте, что max-width не поддерживается в IE, поэтому используйте width: 100%.

Представленный способ является отличным началом в создании адаптивных изображений, но разрешение и время загрузки должны оставаться приоритетными.

Ещё один способ: отзывчивые изображения

Техника, представленная Filament Group, не только изменяет размер изображений в соответствии с пропорциями, но и сжимает разрешение картинок на маленьких экранах, чтобы на больших они не занимали дополнительного места.



Для использования данной техники требуется несколько файлов, все они доступны на Github. Сначала берём JavaScript-файл (*rwd-images.js*), файл .*htaccess* и *rwd.gif* (файл изображения). Потом используем немного HTML, чтобы связать большие и маленькие разрешения: сначала маленькое изображение с префиксом .*r* (для обозначения того, что картинка должна быть адаптивной), потом отсылка к большему изображению, используя data-fullsrc:

Для любого экрана шире 480 пикселей загрузится изображение с большим разрешением (*largeRes.jpg*); маленьким экранам не понадобится загружать большое изображение, загрузится маленькое (*smallRes.jpg*).

Интересная фича для iPhone

Есть одна интересная вещь, касающаяся iPhone и iPod. Дизайн, созданный для больших экранов, просто сожмется для маленького браузера, без необходимости добавления скролла или дополнительной мобильной верстки. Однако изображений и текста не будет видно:



Для решения данной проблемы воспользуемся тегом meta:

```
<meta name="viewport" content="width=device-width; initial-scale=1.0">
```

Если <u>initial-scale</u> равно единице, ширина картинок становится равной ширине экрана.

Настраиваемая структура макета страницы

Для чрезвычайных изменений размеров мы можем захотеть изменить расположение элементов в целом через отдельный файл со стилями или, что более эффективно, через CSS-медиазапрос. Это не должно вызвать особо много проблем, т.к. большинство стилей останутся прежними, но некоторые изменятся.

Например, у нас есть главный файл со стилями, который задает #wrapper, #content, #sidebar, #nav вместе с цветами, фоном и шрифтами. Если наши главные стили делают макет слишком узким, коротким, широким или высоким, мы можем это определить и подключить новые стили.

style.css (основной):

```
/* Default styles that will carry to the child style sheet */
html, body {
background... font... color...
}
```

```
h1,
h2,
h3{}
p,
blockquote,
pre,
code,
ol,
ul {}
/* Structural elements */
#wrapper {
width: 80%;
margin: 0 auto;
background: #fff;
padding: 20px;
#content {
width: 54%;
float: left;
margin-right: 3%;
#sidebar-left {
width: 20%;
float: left;
margin-right: 3%; }
#sidebar-right {
width: 20%;
float: left;
mobile.css (дочерний):
#wrapper {
width: 90%;
#content {
width: 100%;
#sidebar-left {
width: 100%;
```

```
clear: both;
/* Additional styling for our new layout */
border-top:
1px solid #ccc;
margin-top:
20px; }
#sidebar-right {
width: 100%;
clear: both;
/* Additional styling for our new layout */
border-top: 1px solid #ccc;
margin-top: 20px;
```

Content

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Morbi feugiat, nisl in lacinia egestas, leo est ultricies ipsum, nec posuere felis quam nec liguta. Aliquam erat volutpat. Vestibulum a dolor at nisi tristique pharetra. Maecenas convallis aliquam massa, et portitor nisi hendrerit vel. Proin justo massa, facilisis nec suscipit in, aliquam vel neque. Pettertes que orci enim, convaltis sed faucibus sit amet, semper vitae diam. Mauris laçus mauris, semper id mollis vitae, bibendum et metus. Proin tempor velit eu mi ultricles sed pulvin turpis ultricies. Integer egestas sodales portitor

Aenean quis metus arcu. Nam adipiscing suscipit nulla ut dapibus. Nul metus massa. Suspendisse id lectus vitae nisi conque hendreril vitae dolor. Aliquam erat volutpat. Aenean elementum ante at nich omare volutpat massa aliquet. Nam gravida fringilla elementum. Nulla fa tellus quam, tincidunt in aliquet sed, lobortis a risus. Vestibulu Suspendisse potenti. Donec sit amet condimentum leo

ON A WIDER SCREEN, THE LEFT AND RIGHT SIDEBAR CONTENT FIT NICELY TO THE SIDE. FOR THINNER SCREENS, WE MOVE THIS CONTENT BELOW FOR BETTER USABILITY.

Left Sidebar

Nulla in nist eros. Donec fermentum uma ut nibh rutrum non aliquet purus euismod. Praesent tempor gravida massa eget vulputate. Morbi lobortis volutpat elit. Vivamus. molestie dictum nist, in

Right Sidebar

Nulta in nisl eros. Donec fermentum uma ut nibh rutrum non aliquet purus euismod. Praesent tempor gravida massa eget vulputate. Morbi lobortis vulputate. Morbi lobortis
tincidunt ante sit amet
placerat. Fusce leo nisi
puthinar suscipit mulestie
sed, lobortis qui purus.
Quisque porta enim eu ante
consepsal varius. Curabitur
sit met arcu ante. Esam
usi stermentum risus. Sed id volutpat elit. Vivamus molestie dictum nist, in wenenatis nulla malesuada venenatis nulla malesuada

Content

Aprem graum goore at armet, consendent adoptioning ells florio Proposit, mais in somme operation, for and stribinion injurier, men producer feste graum non floriosis. Alloquim entit volvigosit Viriatiouhum a dolor at mai fittaboue pharentes. Maeconina comiestis adquium massia, et portitor rissis hendrard visi. Primi putiti massia, factoria inno sucception, sitipusmini ele modor. Pathendrous de crisi entri, portivolorialis and factorios stat armet, semiger vitare diam. Maurisi facus mauris, semiger lid mobile vitas, sitiendum et metus. Prom tumpor velit eu mi-utilicies sed pulvinar turpis utilicies. Meger egiistas sodales

Annalin suits metus arou. Nam adorsong suo-colt nulta ut stapitus. Nultam at metu masia. Saspendiase et lectur viter misi conque hecchert vite sed dote: Abquan erali volutat. Aereian elementum ante africito norser et amel volutat massa altouet. Nam praeda filogital elementum. Italia facilia: Pries tellas quant. Socioto et altiquat sed, labodis a risus. Vestibulum in mi augue. Suspendiase potenti. Dones atlames

Left Sidebar

Those art fumpor grants massa egit suputatis fuma to rain return ten august pour a finase art fumpor grants massa egit suputatis funtsi loboriti tinosunt atte et anust platerat. Pusse les mai, pudma susopil mineste eed, loboriti quis guvus. Guistuse porta emm su ante consequal vanus. Curabitar sit amet accu ante. Ellam quis fermentum risse. Sed id volutgad elit. Vivamus malestie dictum nisi, in venenatis nulla

Right Sidebar

Nuts in not eros. Donec tementum ums ut not nutrum non aliquet purus euromot. Procesel tempor granda maissa eget valgutate. Morta loborita finodarit ante et amel placeral. Fusce les resi pulserar suscipit molestie sed, lobortis quis purus. Quinque one consequal varios. Curabilius ell'amet anu arte. Eli. ormentum rinus. Seti of votoficial set, Vounus molestre didum nigi in relevuada vitae. porta enim eu ante consequal varius. Curatifur sit amet ansu ante. Eliam quis

Давайте посмотрим, как можно использовать CSS3-медиазапросы для создания адаптивного дизайна.

| min-width | задает минимальную ширину окна браузера или экрана,к которой будут применены определенные стили. Если какое-нибудь значение будет ниже | min-width |, то стили будут проигнорированы; | max-width | делает противоположное.

Пример:

```
@media screen and (min-width: 600px) {
.hereIsMyClass {
width: 30%;
float:
right; }
```

Медиазапрос заработает только в том случае, когда min-width будет больше или равна 600 рх.

```
@media screen and (
max-width: 600px
)
{
   .aClassforSmallScreens
{
   clear:
   both;
   font-size: 1.3em;
}
```

В этом случае класс (aClassforSmallscreens) сработает, когда ширина экрана будет меньше или равна 600 рх.

В то время как min-width и max-width могут быть применимы к экранам и окнам браузеров, нам может понадобиться работать только с шириной устройства. Для этого можно использовать min-device-width и max-device-width:

```
@media screen and (
max-device-width: 480px
)
{
   .classForiPhoneDisplay {
   font-size: 1.2em;
}
.
```

```
@media screen and (min-device-width: 768px
)
{
.minimumiPadWidth
{ clear: both;
margin-bottom: 2px solid #ccc; } }
```

Специально для iPad у медиазапросов есть свойство *orientation*, значениями которого могут быть либо *landscape* (горизонтальный), либо *potrait* (вертикальный):

```
@media screen and (orientation: landscape)
{
    .iPadLandscape
{
    width: 30%;
    float: right; } }
@media screen and
    (orientation: portrait)
{
    .iPadPortrait {
    clear: both; }
}
```

Также значения медиазапросов можно комбинировать, например:

```
@media screen and (
min-width: 800px)
and (max-width: 1200px) {
   .classForaMediumScreen {
   background: #cc0000;
   width: 30%;
   float: right; }
}
```

Этот код будет выполнен только для экранов или окон браузеров шириной между 800 и 1200 рх.

Кто-то может захотеть загрузить определенный лист со стилями для разных значений медиазапросов, это можно сделать вот так:

```
<link rel="stylesheet" media="screen and (max-width: 600px)" href="small.css"/>
<link rel="stylesheet" media="screen and (min-width: 600px)" href="large.css"/>
<link rel="stylesheet" media="print" href="print.css"/>
```

JavaScript

Если ваш браузер не поддерживает CSS3-медиазапросы, то замену стилей можно организовать с помощью ¡Query следующим образом:

```
<script type="text/javascript"
src="http://ajax.googleapis.com/ajax/libs/jquery/1.4.4/jquery.min.js">
</script>
<script type="text/javascript">
$(document).ready(function() {
$(window).bind("resize", resizeWindow);
function resizeWindow(e) {
var newWindowWidth = $(window).width();
// If width width is below 600px, switch to the mobile stylesheet
if(newWindowWidth < 600) {
$("link[rel=stylesheet]").attr({href: "mobile.css"});
}
else if(newWindowWidth > 600) {
// Else if width is above 600px, switch to the large stylesheet
$("link[rel=stylesheet]").attr({href: "style.css"}); }
}); </script>
```

Опциональное отображение контента

Возможность сжимать и менять местами элементы, так ,чтобы они все уместились на маленьких экранах, это замечательно. Но это не лучший вариант. Для мобильных устройств обычно используются: упрощенная навигация, более сфокусированный контент, списки или строки вместо колонок.



К счастью, CSS дает нам возможность показывать и прятать контент с неимоверной легкостью!

display: none;

display: none используется для объектов, которые нужно спрятать.

Пример:

Main Content

Lorem insum dolor sit amet, consedetur adipiscing eitt, Morbi feugiat, nist in lacinia egestas, leo est ultricies ipsum, nec posuere fetis quam nec ligula. Aliquam erat volutpat. Vestbulum a dolor at nist histique pharetra. Maecenas corivatiis aliquam massa, et portitor nist hendrent vel. Proin justo massa, facilistis nec suscipit in, aliquam vel neque. Pellentesque ond erim, convalita sed fauctious at amet, semper vitae diam. Mauris lacus mauris, semper id mollis vitae, bibendum et metus. Proin tempor velit eu mi ultricies sed guivinat turpis ultricies, integer egestas sodales portitor. Cras varius elementum rissus veli dignipsim. In monicus todor eget eros auctor cursus.

Aenean quis metus arcu. Nam adipiscing suscipit nulla ut appibus. Nullam at metus massa. Suspendisse id lectus vitae nis conque hendrent vitae sed dolor. Aliquam erat volutpat. Aeneagetementum ante at nith omare sit amet volutpat massa aliquet. Mam gravida tingilla elementum. Nulla fabilisi. Proin tellus quay, incidunt in aliquet sed. Iobortis a risus. Vestibulum in mi ar gue. Suspendisse potenti. Donec ati amet condimentum leo:

A Left Sidebar

Nulla in nist eros. Donec. fermentum uma ut nich rubrum non alliquet purus. eulamod Praesent tempor gravida massa. get vulputate. Morbi obortis tincidunt ante si amet clacerat. Fusce t nisi, pulvinar susci molestie sed. quis pun in eu ante supply serior abitur sit amet arou ante Etiam quis fermentum risus. Sed id

A Right Sidebar

fermentum uma ut nibh rutrum non aliquet purus. euismod. Praesent tempor gravida massa eget vulputate. Morzi lobortis fincidunt ante sit. amet stacerat. Fusice led misi, pulvinar suscipit. molestie sed, lobortis uis purus. Quisque orta enim eu ante on seducid various urabitur sit amet arcu ite. Etiam quis mentum risus. Sed id olutpat elit. Vivarnus leste didum nist, i

Left Sidebar Content | Right Sidebar Content

volutpat elit. Vivamus

nolestie dictum nist

Main Content

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Morbi feugiat, nist in lacinia egestas, leo est ultricies ipsum, nec posuere felis quam nec ligula. Aliquam erat volutpat. Vestibulum a dolor at nist tristique pharetra. Maecenas convallis aliquam massa, et portitor nisi hendrent vel. Proin justo massa, facilisis nec suscipit in, aliquam vel neque. Pellentesque orci enim, convallis sed faucibus sit amet, semper vitae diam. Mauris lacus mauris, semper id mollis vitae, bibendum et metus. Proin tempor velit eu mi ultricies sed pulvinar turpis ultricies. Integer egestas sodales portitor. Cras varius elementum risus vel dignissim. In rhoncus tortor eget eros auctor cursus.

Aenean quis metus arcu. Nam adipiscing suscipit nulla ut dapibus. Nullam at metus massa. Suspendisse id lectus vitae nisi conque hendrerit vitae sed dolor. Aliquam erat volutpat. Aenean elementum ante at nibh ornare sit amet volutpat massa aliquet. Nam gravida fringisla elementum. Nusia facilisi. Proin tellus quam, tincidunt in aliquet sed, lobortis a risus. Vestibulum in mi augue. Suspendisse potenti. Donec sit amet condimentum leo.

ON THE WIDER SCREEN
IS HIDDEN WHEN
MINIMIZED, AND THEN
REPLACED WITH LINKS.

Вот наша разметка:

В главном файле стилей мы меняем ссылки на колонки, т.к. у нас достаточно большой экран, чтобы отобразить весь контент.

style.css (основной):

```
#content {
width: 54%;
float: left;
margin-right: 3%;
}
#sidebar-left {
width: 20%;
float: left;
margin-right: 3%;
}
#sidebar-right {
width: 20%;
float: left;
}
.sidebar-nav {
display: none;
}
```

Теперь прячем колонки и показываем ссылки:

mobile.css (упрощенный):

```
#content {
width: 100%;
}
#sidebar-left {
display: none;
}
#sidebar-right {
display: none;
}
.sidebar-nav
{
display: inline;
}
```

С возможностями прятать и показывать элементы, изменять размеры картинок, элементов и многое другое, дизайн смог приспосабливаться к различным устройствам и экранам. Если размер экрана уменьшается, можно, например,

использовать скрипт или альтернативный файл со стилями, чтобы увеличить белое пространство, или заменить навигацию для большего удобства.