

HTML5 移动端适配

十七、移动端适配

我们先来了解一下什么是像素

接着来说一下什么是视口

布局视口

视觉窗口

理想视口

完美视口

设置像素比

为什么要适配

适配需要关注的要素

适配方案

固定高度，宽度自适应

固定宽度，viewport缩放

rem做宽度，viewport缩放

十七、移动端适配

我们先来了解一下什么是像素

像素分为两种，一种是设备像素，一种是CSS像素

1. 设备像素（物理像素）

屏幕的物理像素，任何设备屏幕的物理像素的数量都是固定不变的，单位是pt。（屏幕的分辨率）

2. CSS像素

在CSS、JS中使用的一个抽象的概念，单位是 px。

CSS像素也可以称为设备独立像素(device-independent pixels)，简称为dips，单位是dp。

1个CSS元素相当于多少个设备像素取决于两个条件：

- 页面是否缩放
- 屏幕是否为高密度

这两方面后面再解释，先梳理一下手机硬件之间的关系，注意这里使用的都是物理像素。

以 iPhone5 为例，我们已知的是：

1. 分辨率：1136pt x 640pt

指屏幕上垂直有 1136 个物理像素，水平有 640 个物理像素

2. 屏幕尺寸：4英寸

注意英寸是长度单位，不是面积单位。4英寸指的是屏幕对角线的长度。

3. 屏幕像素密度: 326dpi

屏幕像素密度 (Pixel Per Inch) 简称 ppi, 单位是 dpi (dot per inch)。这里指屏幕水平或垂直每英寸有326个物理像素。原则上来说, ppi越高越好, 因为图像会更加细腻清晰。ppi 是可以通过 分辨率 和 屏幕尺寸 计算得到的:

$$326ppi = \frac{\sqrt{1136^2 + 640^2}}{4}$$

[这个网站](#)列出了很多设备的分辨率和屏幕尺寸, 并且计算了ppi。

接着来说一下什么是视口

桌面浏览器中, 浏览器窗口就是约束你的CSS布局视口 (又称初始包含块)。它是所有CSS百分比宽度推算的根源, 它的作用是给**CSS布局限制了一个最大宽度**, 视口的宽度和浏览器窗口宽度一致。

但是在移动端, 情况就很复杂了。

布局视口

浏览器厂商定的视口被称为布局视口。

可以这样设置布局视口的宽度:

```
<meta name="viewport" content="width=640">
```

视觉窗口

视觉视口是用户正在看到的网页的区域, 大小是屏幕中CSS像素的数量。

- window.innerWidth/Height返回视觉视口的尺寸

理想视口

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

完美视口

```
<meta name="viewport" content="width=device-width,initial-scale=1">
```

设置像素比

在谈到像素的时候，讲到除了缩放，**屏幕是否为高密度**也会影响设备像素和CSS像素的关系。

在**缩放程度为100%**（这个条件很重要，因为缩放也会影响他们）时，他们的比例叫做设备像素比(device pixel ratio)：

$$dpr = \frac{\text{设备像素}}{\text{CSS像素}}$$

可以通过js获取：`window.devicePixelRatio`

为什么要适配

- 为了适应各种移动设备，完美呈现应有的布局效果。
- 各种移动设备，分辨率大小不一致，网页想铺满整个屏幕，并在各种分辨率下等比缩放

适配需要关注的要素

一般来说，我们需要关注的是：字体、高宽间距、图像（图标、图片）。

适配方案

总的分为3种方案：

- 固定高度，宽度自适应
- 固定宽度，viewport缩放
- rem做宽度，viewport缩放

这三种方法的核心都是**视口的确定**，现在以实现这个设计图为例说明。

固定高度，宽度自适应

目前使用最多的吧，垂直方向用定值，水平方向用百分比、定值、flex都行。[腾讯](#)、[京东](#)、[百度](#)、[天猫](#)、[亚马逊](#)的首页都是使用的这种方法。

原理：

这种方法使用了完美视口：

```
<meta name="viewport" content="width=device-width,initial-scale=1">
```

这样设置之后，我们就可以不用管手机屏幕的尺寸进行开发了。

固定宽度，viewport缩放

设计图、页面宽度、viewport width使用一个宽度，浏览器帮我们完成缩放。单位使用px即可。

目前已知[荔枝FM](#)、[网易新闻](#)在使用这种方法。有兴趣的同学可以看看是怎么做的。

原理：

这种方法需要根据屏幕宽度来动态生成viewport，生成的 viewport 基本是这样：

```
<meta name="viewport" content="width=640,initial-scale=0.5,maximum-scale=0.5,minimum-scale=0.5,user-scalable=no">
```

rem做宽度，viewport缩放

这也是[淘宝](#)使用的方案，根据屏幕宽度设定 rem 值，需要适配的元素都使用 rem 为单位，不需要适配的元素还是使用 px 为单位。

原理：

实际上做了这几件事情：

- 动态生成 viewport
- 屏幕宽度设置 rem的大小，即给 `<html>` 设置font-size
- 根据设备像素比（window.devicePixelRatio）给 `<html>` 设置data-dpr

设置rem

设置rem的意义在于得到一个与屏幕宽度相关的单位，本来 `vw` 是最合适的，但是因为兼容性的问题，只能使用 `rem` 来做。这样，让**不同设备的rem显示一样的长度**。

`vw`是CSS3引入的单位，`1vw = 1%窗口宽度`

设置 viewport 缩放 和 data-dpr

这两个设置其实是干的一件事，就是适配高密度屏幕手机的px单位。

```
1. .a{
2.   font-size:12px;
3. }
4. [data-dpr="2"] .a{
5.   font-size: 24px;
6. }
7. [data-dpr="3"] .a{
8.   font-size: 36px;
9. }
```

而缩放是动态的，这样，**不同设备下的px显示一样的长度**。

之前说过CSS像素和物理像素与缩放、dpr都有关系，这里说明：

在普通手机上，.a字体设置为12px；

在dpr是2的手机上，[data-dpr="2"] .a字体为24px，又因为页面缩放50%，字体为还是12px