



做真实的自己，用良心做教育

H5教学部

媒体查询

Responsive Web Design 响应式网页 设计

1、常见的布局方案

固定布局：以像素作为页面的基本单位，不管设备屏幕及浏览器宽度，只设计一套尺寸；

可切换的固定布局：同样以像素作为页面单位，参考主流设备尺寸，设计几套不同宽度的布局。通过识别的屏幕尺寸或浏览器宽度，选择最合适的那套宽度布局；

响应布局：对页面进行响应式的设计实现，需要对相同内容进行不同宽度的布局设计，有两种方式：**pc优先**（从pc端开始向下设计）；**移动优先**（从移动端向上设计）；
无论基于那种模式的设计，要兼容所有设备，布局响应时不可避免地需要对模块布局做一些变化（发生布局改变的临界点称之为**断点**），

弹性布局：以百分比作为页面的基本单位，可以适应一定范围内所有尺寸的设备屏幕及浏览器宽度，并能完美利用有效空间展现最佳效果；

混合布局：同弹性布局类似，可以适应一定范围内所有尺寸的设备屏幕及浏览器宽度，并能完美利用有效空间展现最佳效果；只是混合像素、和百分比两种单位作为页面单位。

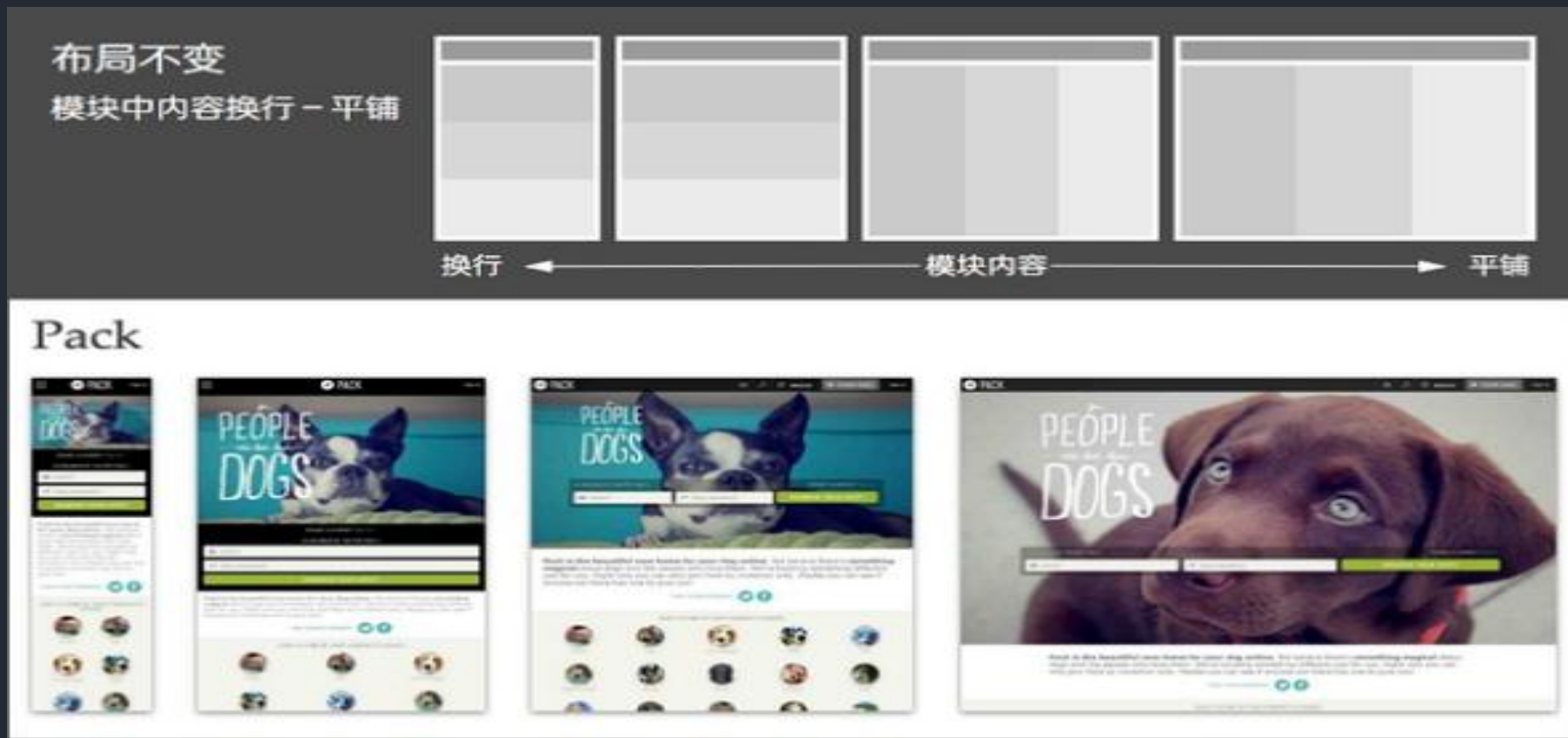
2、响应式布局方案

(1) 模块中内容：挤压一拉（布局不变）



2、响应式布局方案

(2) 模块中内容：换行—平铺（布局不变）



2、响应式布局方案

(3) 模块中内容：删减—增加（布局不变）



2、响应式布局方案

(4) 模块位置变换（布局改变）



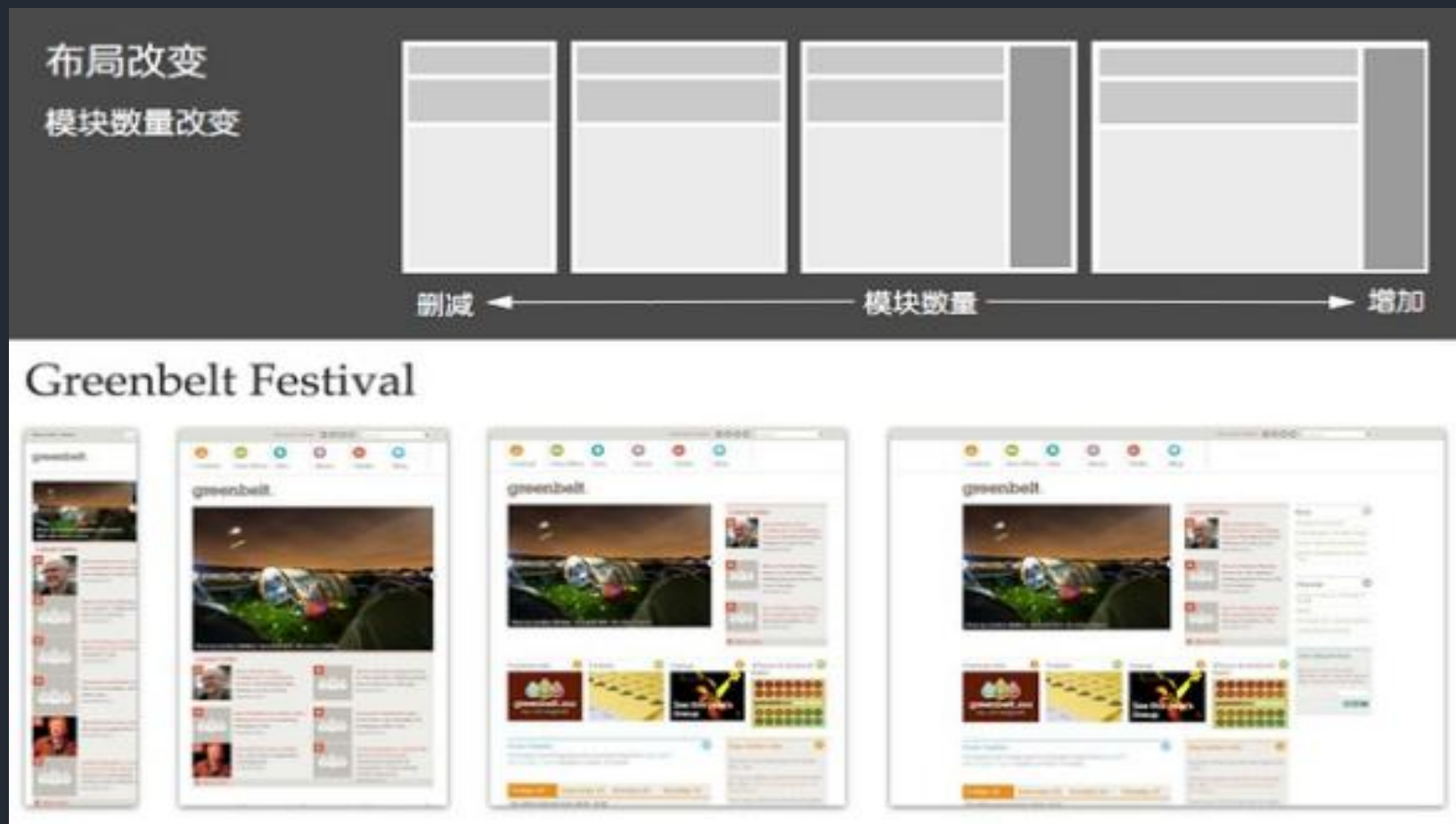
2、响应式布局方案

(5) 模块展示方式改变：隐藏—展开（布局改变）



2、响应式布局方案

(6) 模块数量改变：删减—增加（布局改变）



3、响应式布局特点

设计特点：

1. 面对不同分辨率设备灵活性强
2. 能够快捷解决多设备显示适应问题

缺点：

1. 兼容各种设备工作量大，效率低下
2. 代码累赘，会出现隐藏无用的元素，加载时间加长
3. 其实这是一种折中性质的设计解决方案，多方面因素影响而达不到最佳效果
4. 一定程度上改变了网站原有的布局结构，会出现用户混淆的情况

4.Meta标签的设置

准备工作：设置Meta标签

```
<meta name="viewport" content="width=device-width, initial-scale=1.0, maximum-scale=1.0,minimum-scale=1.0, user-scalable=no">
```

这段代码的几个参数解释：

width = device-width：宽度等于当前设备的宽度

initial-scale：初始的缩放比例（默认设置为1.0）

minimum-scale：允许用户缩放到的最小比例（默认设置为1.0）

maximum-scale：允许用户缩放到的最大比例（默认设置为1.0）

user-scalable：用户是否可以手动缩放（默认设置为no，因为我们不希望用户放大缩小页面）

4.Meta标签的设置

- ◆ H5页面窗口自动调整到设备宽度，并禁止用户缩放页面

```
<meta name="viewport" content="width=device-width,initial-scale=1.0,minimum-scale=1.0,maximum-scale=1.0,user-scalable=no" />
```

- ◆ 忽略将页面中的数字识别为电话号码

```
<meta name="format-detection" content="telephone=no" />
```

- ◆ 忽略Android平台中对邮箱地址的识别

```
<meta name="format-detection" content="email=no" />
```

- ◆ 当网站添加到主屏幕快速启动方式，可隐藏地址栏，仅针对ios的safari

```
<meta name="apple-mobile-web-app-capable" content="yes" />
```

```
<!-- ios7.0版本以后，safari上已看不到效果 -->
```

- ◆ 将网站添加到主屏幕快速启动方式，仅针对ios的safari顶端状态条的样式

```
<meta name="apple-mobile-web-app-status-bar-style" content="black" />
```

```
<!-- 可选default、black、black-translucent -->
```

什么是媒体查询

媒体查询可以让我们根据设备显示器的特性（如视口宽度、屏幕比例、设备方向：横向或纵向）为其设定**CSS**样式，媒体查询由**媒体类型**和一个或多个检测媒体特性的**条件表达式**组成。媒体查询中可用于检测的媒体特性有 **width**、**height** 和 **color**（等）。使用媒体查询，可以在不改变页面内容的情况下，为特定的一些输出设备定制显示效果。

1、媒体查询操作方式

实际操作为：对设备提出询问（称作表达式）开始，如果表达式结果为真，媒体查询中的**CSS**被应用，如果表达式结果为假，媒体查询内的**CSS**将被忽略。

2、媒体查询结构

```
@media screen and (min-width:320px) and (max-width:750px){  
    body { background-color:blue;}  
}
```

设备类型 (默认为all)

media_type	设备类型说明
all	所有设备
aural	听觉设备
braille	点字触觉设备
handed	便携设备，如手机、平板电脑
print	打印预览图等
projection	投影设备
screen	显示器、笔记本、移动端等设备
tty	如打字机或终端等设备
tv	电视机等设备类型
embossed	盲文打印机

- ◆ **screen** 是媒体类型里的一种，CSS2.1定义了10种媒体类型
- ◆ **and** 被称为关键字，其他关键字还包括 **not**(排除某种设备)，**only**(限定某种设备)
- ◆ **(min-width: 400px)** 就是媒体特性，其被放置在一对圆括号中。

默认样式 注意：默认样式要写在最前面

/* 打印样式 */

@media print {}

/* 手机等小屏幕手持设备 */

@media screen and (min-width: 320px) and (max-width: 480px) {}

/* 平板之类的宽度 1024 以下设备 */

@media only screen and (min-width: 321px) and (max-width: 1024px) {}

/* PC客户端或大屏幕设备: 1028px 至更大 */

@media only screen and (min-width: 1029px) {}

/* 竖屏 */

@media screen and (orientation:portrait) {对应样式}

/* 横屏 */

@media screen and (orientation:landscape){对应样式}

主断点



目前常用断点值是：320px、480px、768px和1024px

3、CSS2 Media用法

其实并不是只有CSS3才支持Media的用法，早在CSS2开始就已经支持Media，具体用法，就是在HTML页面的head标签中插入如下的一段代码

```
<link rel="stylesheet" type="text/css" media="screen" href="style.css">;
```

想知道现在的移动设备是不是纵向放置的显示屏，可以这样写：

```
<link rel="stylesheet" type="text/css" media="screen and (orientation:portrait)"  
      href="style.css">
```

orientation:portrait：指定输出设备中的页面可见区域高度大于或等于宽度

landscape：除portrait值情况外，都是landscape

第一段的代码也用CSS2来实现，让它一样可以让页面宽度小于960的执行指定的样式文件：

```
<link rel="stylesheet" type="text/css" media="screen and (max-width:960px)" href="style.css">
```


4、常涉及到的css属性

Display;
Width;
Float;
Text-align;
Font

THANK YOU



做真实的自己，用良心做教育