# 关于响应式需求的一点思考

近期做了一些“响应式”相关的网站，在借鉴相关网站后，发现不同网站的响应式并不相同。查资料后总结如下：

如今主流的三种方式以实现多终端的用户需求：

1. 使用响应式布局的响应式网页。
2. 使用服务器端组件的响应式网页。
3. 通过分开制作多套网站的响应式网页。（一般需要重定向以方便用户记忆域名）

对三种方式的优缺点，特点展开讨论：（主要从开发成本、维护成本、性能、用户体验分析）

## 使用响应式布局的响应式网页。

直接通过css媒体查询设计不同尺寸的样式即可实现（@media only screen（max-width/min-width：\*\*））。【自适应网页设计/弹性设计】

特点：

1. 开发成本随着网页复杂度的提升迅速提升。
2. 前端开发人员的维护成本极高，一单更改需求或页面结构，需要考量各尺寸下的影响。网站数据维护人员成本低，只需提供一次数据即可。
3. 性能较低，比如图片在不同终端并未得到优化。
4. 用户体验较差，不能保证各终端用户体验的最优。
5. 设计元素容易被复用，用户在不同终端上会感到熟悉（风格统一），设计成本低。

适用案例（简单的展示网）：

* 博客网站
* 小型企业站
* 媒体类（新闻、报纸）网站
* 活动专题页

总结：适合于偏重信息表达而轻交互，需求、前端不需要经常变化的网站。

常用手段：在不同终端上控制对次要信息的增减，对触摸屏进行一定优化。（手势滑动、放大点击域）、Media Query 媒体查询、百分比流式布局，网格与Typography系统。

## 使用服务器端组件的响应式网页

通过在服务器端识别UA（User-agent），即识别不同的终端设备，渲染不同版本的模板，组合相应的静态资源（数据中数据等）返回给终端即可实现。

特点

1. 开发难度取决于网站的复杂程度和需要创建的代码库数量，要进行服务器端编程。后端（服务器端）、前端人员的开发成本增加。
2. 需求、页面结构改动时维护成本较低，因为是基于模块开发的。如果用户只上传一次，则必须牺牲服务器相应速度或数据库存储量。
3. 性能较高。对相应的模块等都进行了相应的选择和优化。
4. 用户体验较好。

适用案例：

* 大型企业站
* 交互性比较强的网站

常用手段：图片静态资源相应，前端写2套逻辑用以分别兼容键鼠、触摸设备，RESS（Responsive Web Design with Server Side Components），对资源进行异步加载

【拓展阅读：百度的模块化方案： [F.I.S -前端工具框架](http://fis.baidu.com/)】

## 通过分开制作多套网站的响应式网页。

是使用服务器端组件的响应式网页的极致，在开发资源（时间和精力）充足的情况下，分别订制和维护可以得到更加好的用户体验。适用于复杂的交互性、不同终端差异巨大的情况。适合增量式开发（迭代开发）。

注：主流电商（淘宝、天猫、京东、唯品会）大都这么干。

常用手段：一般需要进行重定向以方便用户记忆域名。

附加资料：（百度站长平台官方出了一个对照表）

