# 小整理

* 比如H5端的浏览器有跨域限制：1.前端代码和后端接口部署在同域的web服务器上；2.由后台服务器配置策略，设为允许跨域访问H5端调试是跨域：1.使用HBuilderX内置浏览器，内置浏览器经过官方处理，不存在跨域问题，简单易用，推荐使用。（需HBuilderX 2.6以上）；2.配置webpack-dev-server代理;3.给浏览器安装跨域插件，禁止浏览器报跨域，Chrome插件名称：Allow-Control-Allow-Origin:
* 比如微信小程序会强制要求https链接，并且所有要联网的服务器域名都要配到微信的白名单中；
* 应用生命周期仅可在App.vue中监听，在其它页面监听无效
* 如果想在App端实现点击某个tabitem不跳转页面，不能使用onTabItemTap，可以使用[plus.nativeObj.view](http://www.html5plus.org/doc/zh_cn/nativeobj.html" \t "https://uniapp.dcloud.net.cn/_blank)放一个区块盖住原先的tabitem，并拦截点击事件
* 避免在 onShow 里使用需要权限的 API（比如 setScreenBrightness() 等需要手机权限）, 可能会再次触发onShow造成死循环
* App端，在 pages.json 里的 titleNView 或页面里写的 plus api 中涉及的单位，只支持 px。****注意此时不支持 rpx****
* rpx不支持动态横竖屏切换计算，使用rpx建议锁定屏幕方向
* 早期 uni-app 提供了 upx ，目前已经推荐统一改为 rpx 了
* 使用@import语句可以导入外联样式表，@import后跟需要导入的外联样式表的相对路径，用;表示语句结束
* 尽量避免将静态的样式写进 style 中，以免影响渲染速度；在 uni-app 中不能使用 \* 选择器；page 相当于 body 节点；微信小程序自定义组件中仅支持 class 选择器
* ::after ::before在 view 组件后边插入内容，****仅微信小程序和App生效****
* 定义在 App.vue 中的样式为全局样式，作用于每一个页面。在 pages 目录下 的 vue 文件中定义的样式为局部样式，只作用在对应的页面，并会覆盖 App.vue 中相同的选择器；App.vue 中通过 @import 语句可以导入外联样式，一样作用于每一个页面；nvue页面暂不支持全局样式
* var(--status-bar-height) 此变量在微信小程序环境为固定 25px，在 App 里为手机实际状态栏高度 ；由于在H5端，不存在原生导航栏和tabbar，也是前端div模拟。如果设置了一个固定位置的居底view，在小程序和App端是在tabbar上方，但在H5端会与tabbar重叠。此时可使用--window-bottom，不管在哪个端，都是固定在tabbar上方

.status\_bar { height: var(--status-bar-height); width: 100%; }

bottom: calc(var(--window-bottom) + 10px)

* 小程序不支持在css中使用本地文件，包括本地的背景图和字体文件。需以base64方式方可使用。App端在v3模式以前，也有相同限制。v3编译模式起支持直接使用本地背景图和字体
* App端Android支持不依赖Android版本号，即便是Android4.4也是上表数据。因为uni-app的js代码运行在自带的独立jscore中，没有js的浏览器兼容性问题。uni-app的vue页面在Android低端机上只有css浏览器兼容性问题，因为vue页面仍然渲染在webview中，受Android版本影响，太新的css语法在低版本不支持
* 为多端兼容考虑，建议优先从 [uni-app插件市场](https://ext.dcloud.net.cn/" \t "https://uniapp.dcloud.net.cn/_blank) 获取插件。直接从 npm 下载库很容易只兼容H5端
* 非 H5 端不支持使用含有 dom、window 等操作的 vue 组件和 js 模块，安装的模块及其依赖的模块使用的 API 必须是 uni-app 已有的 [API](https://uniapp.dcloud.net.cn/api/README)（兼容小程序 API），比如：支持[高德地图微信小程序 SDK](https://www.npmjs.com/package/amap-wx" \t "https://uniapp.dcloud.net.cn/_blank)。类似[jQuery](https://www.npmjs.com/package/jquery" \t "https://uniapp.dcloud.net.cn/_blank) 等库只能用于H5端
* 声明 lang="ts" 后，该 vue 文件 import 进来的所有 vue 组件，均需要使用 ts 编写
* 2.5.0+版本支持在pages.json内引入组件：[easycom](https://uniapp.dcloud.net.cn/collocation/pages?id=easycom)（传统vue组件，需要安装、引用、注册，三个步骤后才能使用组件。easycom将其精简为一步。 只要组件安装在项目的components目录下，并符合components/组件名称/组件名称.vue目录结构。就可以不用引用、注册，直接在页面中使用）
* 小程序组件需要放在项目特殊文件夹 wxcomponents（或 mycomponents、swancomponents）；小程序组件的性能，不如vue组件。使用小程序组件，需要自己手动setData，很难自动管理差量数据更新。而使用vue组件会自动diff更新差量数据。所以如无明显必要，建议使用vue组件而不是小程序组件。比如某些小程序ui组件，完全可以用更高性能的uni ui替代；当需要在 vue 组件中使用小程序组件时，注意在 pages.json 的 globalStyle 中配置 usingComponents，而不是页面级配置；在 pages.json 对应页面的 style -> usingComponents 引入组件

在页面中使用 <custom name="uni-app"></custom>

* uni-app 在发布到H5时支持所有vue的语法；发布到App和小程序时，由于平台限制，无法实现全部vue语法：非H5端不支持 [Vue官方文档：Class 与 Style 绑定](https://cn.vuejs.org/v2/guide/class-and-style.html" \t "https://uniapp.dcloud.net.cn/_blank) 中的 classObject 和 styleObject 语法
* 在H5平台 使用 v-for 循环整数时和其他平台存在差异，如 v-for="(item, index) in 10" 中，在H5平台 item 从 1 开始，其他平台 item 从 0 开始，可使用第二个参数 index 来保持一致
* 在非H5平台 循环对象时不支持第三个参数，如 v-for="(value, name, index) in object" 中，index 参数是不支持的
* 小程序端数据为差量更新方式，由于小程序不支持删除对象属性，使用的设置值为 null 的方式替代，导致遍历时可能出现不符合预期的情况，需要自行过滤一下值为 null 的数据
* 为兼容各端，事件需使用 v-on 或 @ 的方式绑定，请勿使用小程序端的bind 和 catch 进行事件绑定
* 若需要禁止蒙版下的页面滚动，可使用 @touchmove.stop.prevent="moveHandle"，moveHandle 可以用来处理 touchmove 的事件，也可以是一个空函数。

.stop：各平台均支持， 使用时会阻止事件冒泡，在非 H5 端同时也会阻止事件的默认行为

* H5 的select 标签用 picker 组件进行代替
* App端（vue页面[V3编译模式](https://ask.dcloud.net.cn/article/36599" \t "https://uniapp.dcloud.net.cn/_blank)）和H5端支持v-html，其他端不支持v-html
* 在 onLoad 里得到，onLoad 的参数是其他页面打开当前页面所传递的数据
* ****设置全局的数据和全局的方法：****uni-app 内置了 [vuex](https://vuex.vuejs.org/zh/guide/" \t "https://uniapp.dcloud.net.cn/_blank) ，在app里的使用
* ****组件属性设置不生效解决办法：****当重复设置某些属性为相同的值时，不会同步到view层。 例如：每次将scroll-view组件的scroll-top属性值设置为0，只有第一次能顺利返回顶部。 这和props的单向数据流特性有关，组件内部scroll-top的实际值改动后，其绑定的属性并不会一同变化：1.监听事件，记录组件内部变化的值，在设置新值之前先设置为记录的当前值；2.监听事件，获取组件内部变化的值，实时更新其绑定值（增加防抖处理）
* Vue.nextTick、vm.$el、v-cloak、transition仅H5支持；Vue.directive、is微信小程序不支持...
* 在App端，如果使用vue页面，则使用webview渲染；如果使用nvue页面(native vue的缩写)，则使用原生渲染
* 原生组件层级问题 H5没有原生组件概念问题，非H5端有原生组件并引发了原生组件层级高于前端组件的概念，要遮挡video、map等原生组件，请使用cover-view组件
* [H5正常但小程序异常的可能性](https://uniapp.dcloud.io/matter?id=h5%e6%ad%a3%e5%b8%b8%e4%bd%86%e5%b0%8f%e7%a8%8b%e5%ba%8f%e5%bc%82%e5%b8%b8%e7%9a%84%e5%8f%af%e8%83%bd%e6%80%a7):小程序要求连接的网址都要配白名单;小程序不支持组件比例v-html
* H5和小程序界面正常，而App界面异常:大多是因为css兼容性(从 uni-app 2.5.3 起，Android端支持引入腾讯x5浏览器内核)
* [小程序或App正常，但H5异常的可能性](https://uniapp.dcloud.io/matter?id=%e5%b0%8f%e7%a8%8b%e5%ba%8f%e6%88%96app%e6%ad%a3%e5%b8%b8%ef%bc%8c%e4%bd%86h5%e5%bc%82%e5%b8%b8%e7%9a%84%e5%8f%af%e8%83%bd%e6%80%a7): App特有的API和功能，比如plus、Native.js、subNVue、原生插件等;使用了小程序专用的功能，比如微信卡卷、小程序插件、微信小程序云开发。对于云开发，建议使用可跨端的uniCloud
* data 属性必须声明为返回一个初始数据对象的函数；否则页面关闭时，数据不会自动销毁，再次打开该页面时，会显示上次数据
* 非H5端，不能使用浏览器自带对象，比如document、window、localstorage、cookie等，更不能使用jquery等依赖这些浏览器对象的框架
* App端若要使用操作window、document的库，需要通过renderjs来实现
* H5开发组件内（页面除外）不支持 onLoad、onShow 等页面生命周期
* 微信小程序：目前小程序分包大小有以下限制整个小程序所有分包大小不超过 16M；单个分包/主包大小不能超过 2M

# 主题替换

主题主要是背景、按钮、字体颜色、边框等设置颜色，因uniapp没有link引用全局样式，如果不是自定义导航栏、tabbar，只能通过jsapi动态修改颜色故与PC端不同，增加了难度。有以下几种思路实现主题变换：

1. uni.scss修改变量值 各子页面样式直接引用uni.scss 变量 统一设置系统主题颜色。

优点：简单 <style lang=”scss”> xx { color:$uni-text-color}</style>

缺点：原生导航栏、tabbar需要设置全局变量通过jsapi动态修改；uni.scss发包后通过增量更新会改变app版本号

1. 类名动态绑定 :class=”[‘content-’+theme]”

优点：设置简单

缺点：原生导航栏、tabbar需要设置全局变量通过jsapi动态修改；css翻本添加，增加css开发量，并且设置后（打包后）不够灵活

1. 初始化请求后台获得相应的颜色，设置成全局变量或缓存或vuex保存等

优点：原生导航栏、tabbar不需重新请求接口；改变主题不需要更新包；设置颜色更加灵活性

缺点：页面绑定颜色比较麻烦，需要通过 :style 设置颜色

有些需要权限控制，渐变导航栏需选择自定义导航栏、tabbar，这种情况下则不需要jsapi修改主题颜色。

# unipush推送服务

1.准备工作：推送服务分别能用官方提供集成的个人中心开发推送服

务和个推服务自己集成，两个服务都能实现单推和群推

2.推荐选择：推送有其他业务需求一般用个推服务后台自己集成，前

端用uniapp的5+pushapi监听推送然后处理自己所需要处理的业

务等

代码示例：

监听系统通知栏消息点击事件

plus.push.addEventListener('click',function(msg){

处理点击消息的业务逻辑代码

},false);

监听接收透传消息事件

plus.push.addEventListener('receive',function(msg){

处理透传消息的业务逻辑代码

},false);

（PS：爬坑，这里安卓和ios两个一样的返回，但是返回的数据类型居然是不一样的，

安卓的返回的str型数据要用josn.parse()方法处理ios端返回的是objec型数据可以用

keyvalue的方式直接取值）

3.安卓推送服务只支持在线推送，离线推送服务需要去申请第三方厂

商的key之类的信息证书，IOS推送服务需要用苹果账号去申请专门

的推送证书后支持离线和在线推送

（ps：安卓离线推送服务每个不同型号手机都需要申请不同手机厂商的key之类的信息

证书，而且所需要的材料都不一样，ios需要用苹果账号去网站申请审核过了就能得到

我们所需要的推送打包证书）

4.处理复杂的业务都需要自己做透传消息手动创建处理后续业务

透传消息模板分为两种，一种是符合官方的数据格式透传消息，一种是不符合的，不符

的需要自定义内容

代码示例：

{title:"标题",content:"内容",payload:"自定义数据"}

这个是他们官方定义符合透传消息的数据格式

自行摸索各种查论坛loading...，发现当符合他们的自定义透传消息模板时，

安卓端会让这条消息处理为普通推送消息不会进行透传消息处理也就是我们上

一个示例代码的‘receive’方法安卓端一定不会触发但是问题来了

ios端又不一样了不管是符不符合他们的模板都是触发透传消息事件

loading....

问题又来了各种调试发现ios端当用户在线使用app时，推送消息会发送到手

机上面但是不会在通知栏显示（PS：巨坑）

Loading....

最终解决方案思路如下：

后台是用个推服务自己集成的，所以选择安卓和ios两端都不传符合模板的透

传消息内容，这样两端都能监听到透传消息，

前端透传在监听事件里监听到透传内容自行创建一条通知栏推送消息再用

这个方法处理业务

创建方法用的是5+pushapi

代码示例：

plus.push.createMessage(content,payload);（ps：注意传值顺序“cont

ent”是通知栏内容“payload”是你所需要的透传内容或者参数）