

# 营销页面渲染

by 文坚



## 一、开课须知

# 开课须知

- 课程相关资料都在Github，建议大家直接Watch：<https://github.com/FE-star/Plan-2022>
- Release 版本有介绍版本更替情况：<https://github.com/FE-star/Plan-2022/releases>
- 建议通过issue提问题，或者讨论。
- 标明 level3 的属于非本课程必须立刻掌握内容，可以滞后思考🤔

# 目录

- 一、课题背景
- 二、该课题要解决的核心问题
- 三、性能优化问题

## 二、课题背景

# 低逻辑运营页面

 猜你喜欢 



**Anmum™ 安满 PregnaPro™ 孕期奶粉**

3期免息

活动价: ¥ 223  
分期价: ¥ 55.01 X 3期

[立即购买](#)



**Perrier 巴黎水青柠味含气天然矿泉水**

3期免息

活动价: ¥ 149  
分期价: ¥ 46.33 X 3期

[立即购买](#)



**CITIZEN 西铁城手表男光动能**

6期免息

活动价: ¥ 1099  
分期价: ¥ 158.17 X 6期

[立即购买](#)



**FURLA 芙拉女包小方包**

6期免息

活动价: ¥ 949  
分期价: ¥ 143.17 X 6期

[立即购买](#)



**雅培 Abbott 港版金装加**

6期免息



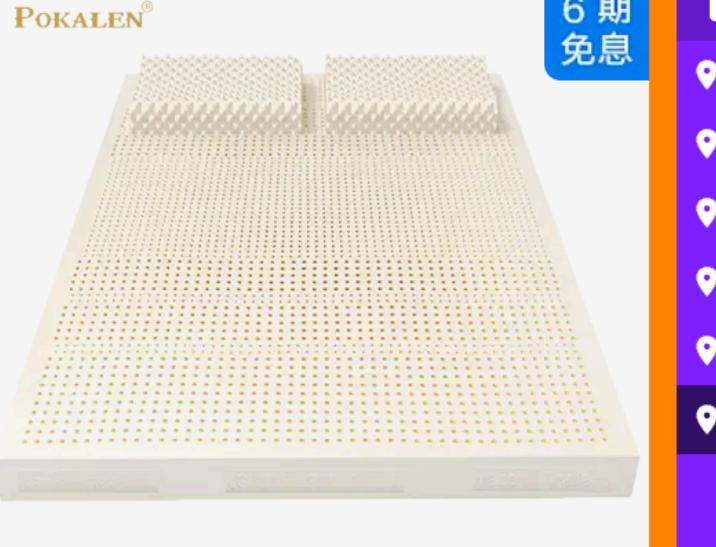
**北京同仁堂灵芝孢子粉**

3期免息



**日本花王妙而舒Merries**

3期免息



**POKALEN 乳胶床垫泰国**

6期免息

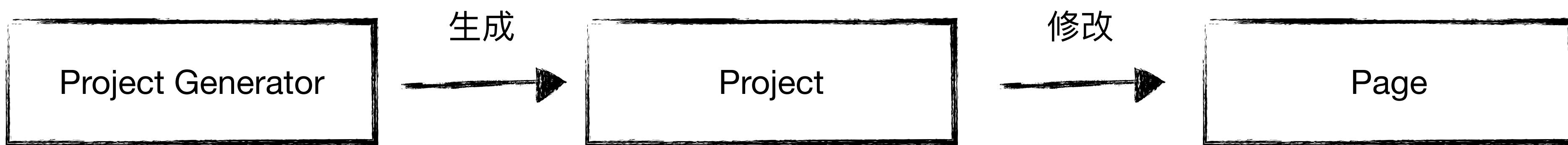


天猫

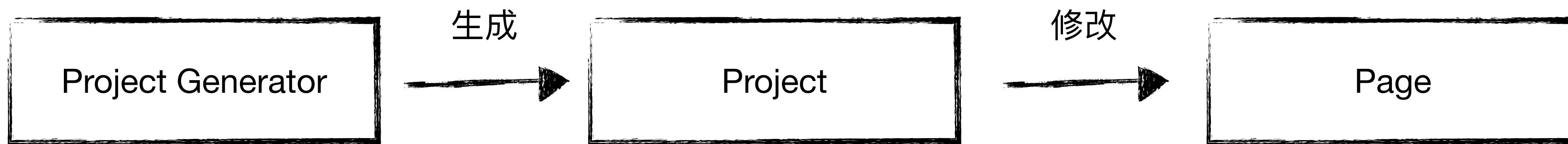
- 12期免息
- 送女神
- 送男神
- 送自己
- 送家人
- 猜你喜欢

[回顶部](#)

一周10张以下.....



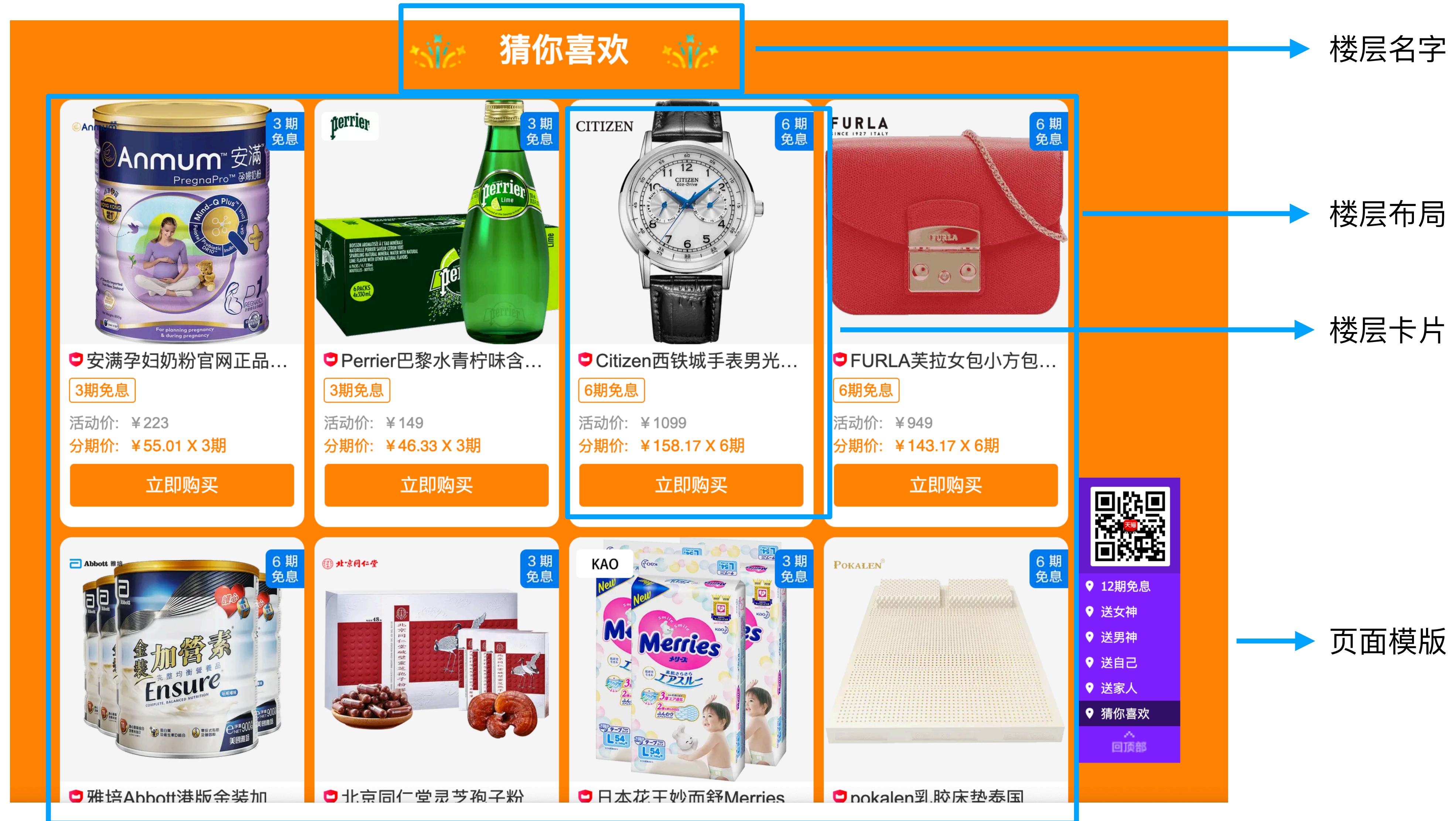
# 一周10张以上，亦或100张以上.....



能不能做得更快些?

# 每次主要在修改什么？

**猜你喜欢**



楼层名字 → 楼层名字

楼层布局 → 楼层布局

楼层卡片 → 楼层卡片

页面模版 → 页面模版

二维码

- 12期免息
- 送女神
- 送男神
- 送自己
- 送家人
- 猜你喜欢
- 回顶部

所以.....

Page = fn(data)



Page = fn(页面模版, 楼层布局, 楼层名, 楼层卡片, data, ...others)



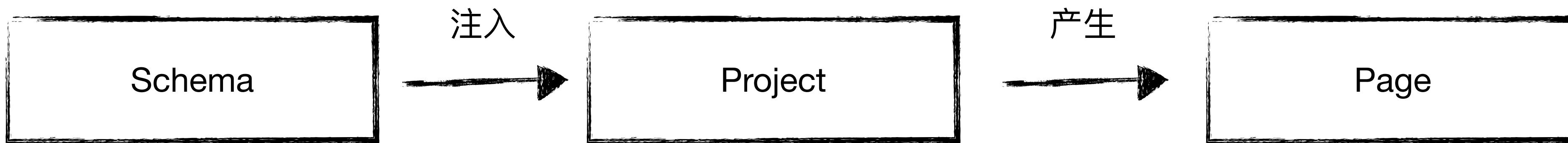
Page = fn(page schema, data)

# Page Schema是什么鬼？

```
1  {
2      "name": "6月大促页",
3      "pageType": "normal", → 页面模版
4      "floors": [
5          {
6              "title": "12期免息", → 楼层标题
7              "layout": {
8                  "type": "flex",
9                  "col": 4
10             },
11             "card": "tianmao-1",
12             "actId": 18754
13         },
14         {
15             "title": "送女神",
16             "layout": {
17                 "type": "flex", → 楼层布局
18                 "col": 4
19             },
20             "card": "tianmao-1", → 楼层卡片
21             "actId": 18755
22         },
23         {
24             "title": "送男神",
25             "layout": {
26                 "type": "flex",
27                 "col": 4
28             },
29             "card": "tianmao-1", → 数据来源
30             "actId": 18756
31         }
32     ]
33 }
```

其实就是个JSON罢了！！

所以本次课题背景就很简单



简单演示下~

## 二、该课题要解决的核心问题

## 二、该课题要解决的核心问题

1. 布局系统
2. Vue动态组件选择
3. ~~性能优化~~ (合到第三部分讲)

## 二.2、布局系统

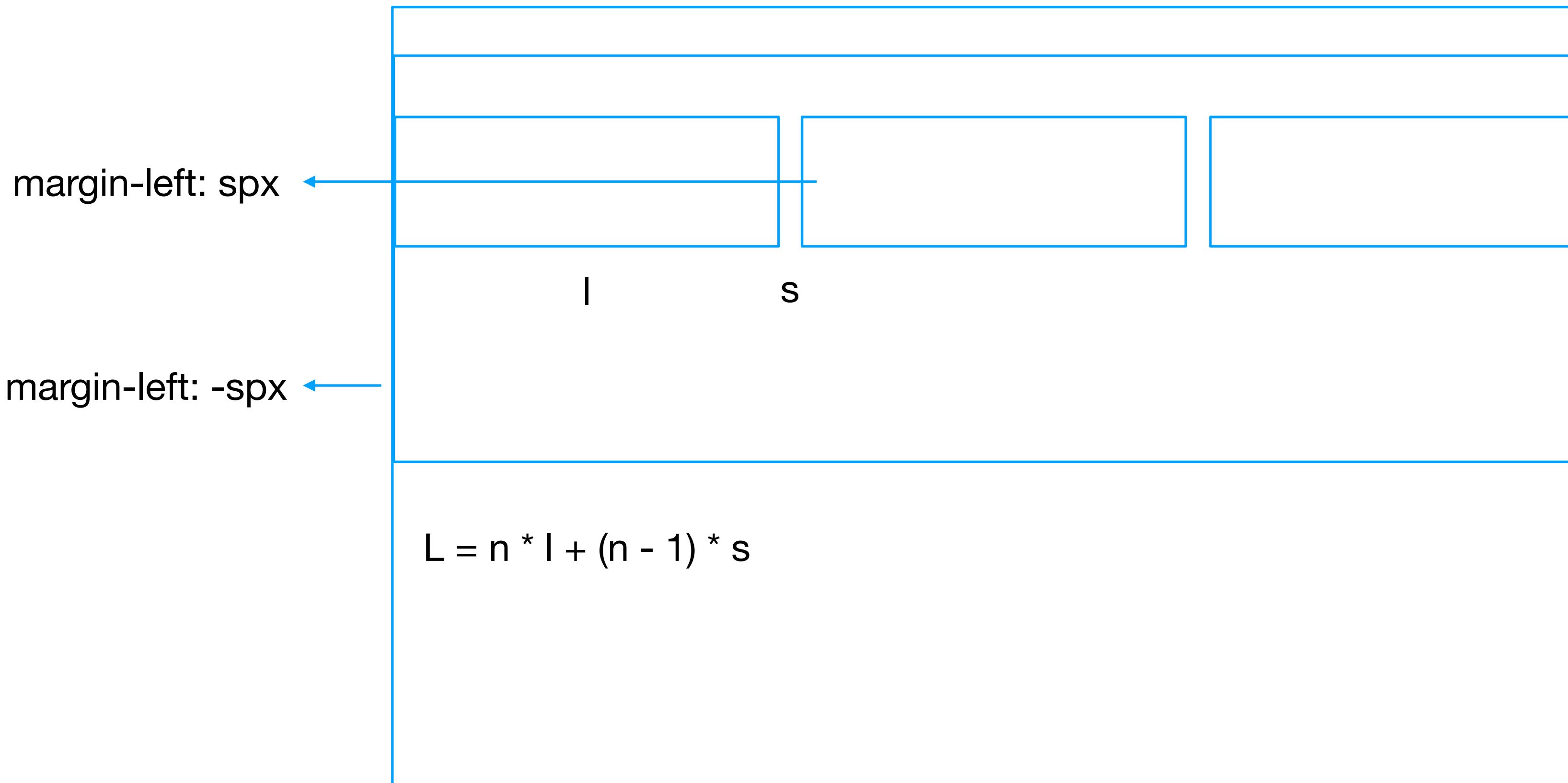
定宽栅栏

响应式栅栏

slot插槽

# 定宽栅栏

$L = 980px/1180px/1440px$  等



具体参考：<https://github.com/FE-star/Plan-2022/blob/main/page-transformer/src/components/grid.vue>

# 响应式栅栏

你以为我会跟你讲flexbox? !

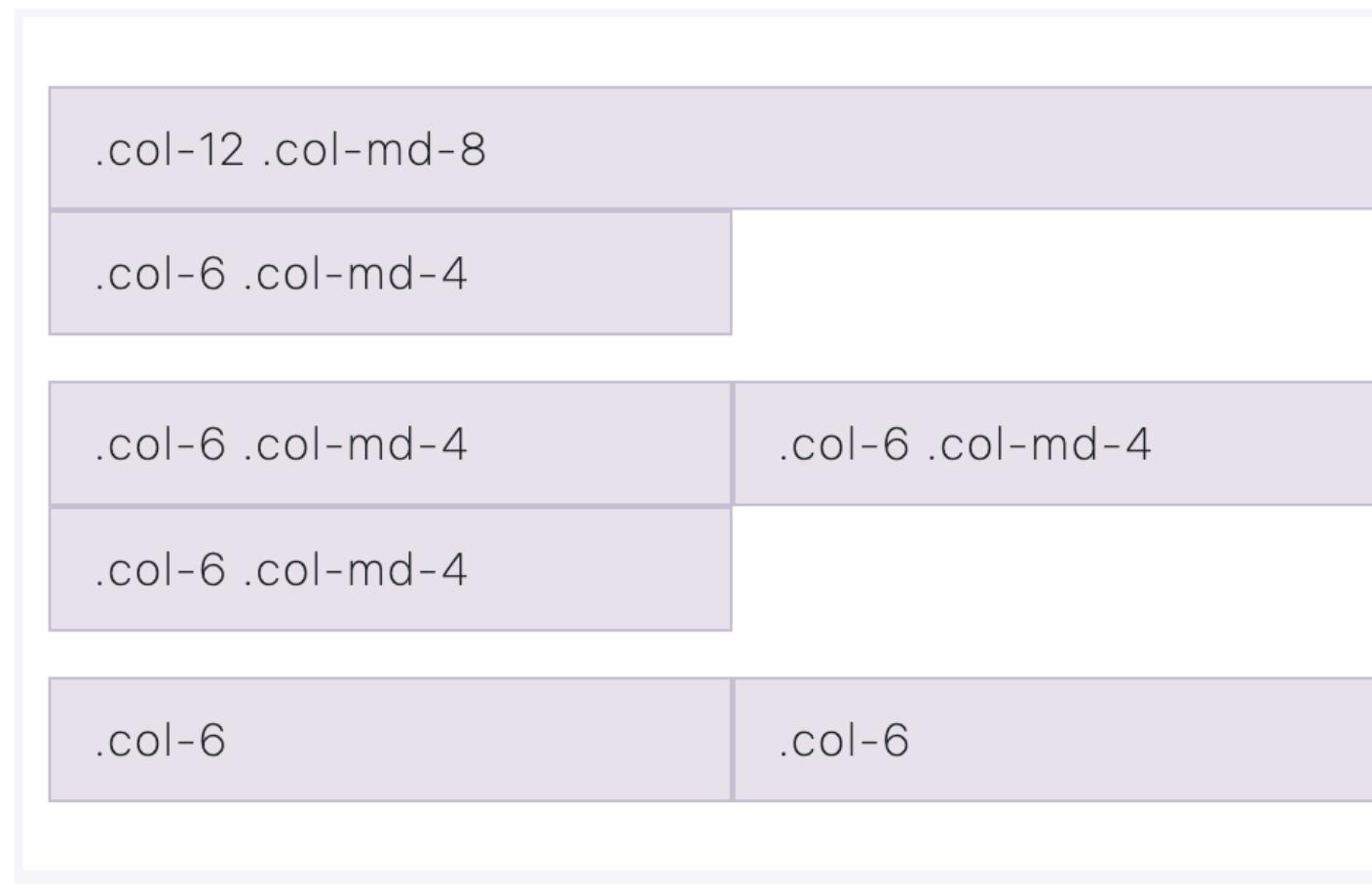
Flexbox概念太多，讲了跟白讲一样，大家都记不住😅

正经人谁讲flexbox啊，都是用到时候查一查.....

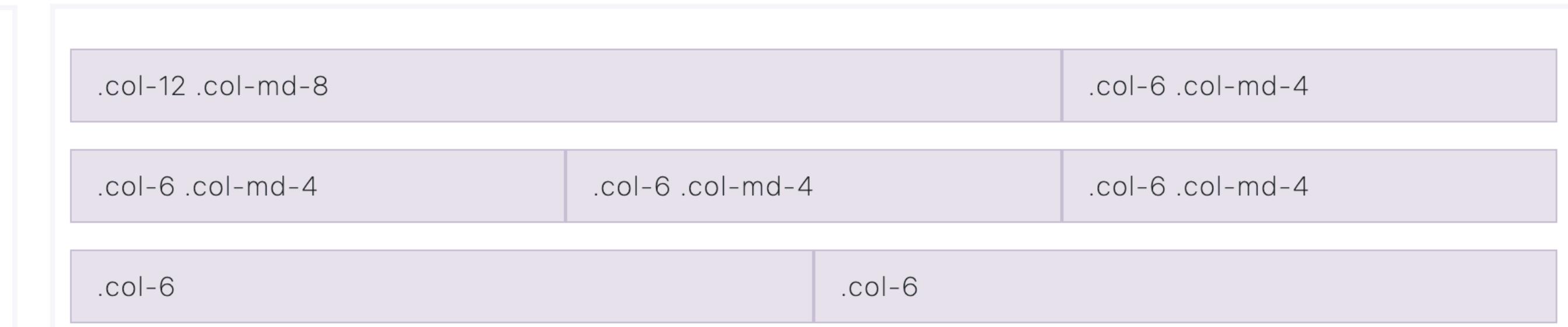
一般情况下会用Bootstrap4的响应式布局就OK了

# 响应式栅栏

<768px



>768px



```
<!-- Stack the columns on mobile by making one full-width and the other half-width -->
<div class="row">
  <div class="col-12 col-md-8">.col-12 .col-md-8</div>
  <div class="col-6 col-md-4">.col-6 .col-md-4</div>
</div>
```

```
<!-- Columns start at 50% wide on mobile and bump up to 33.3% wide on desktop -->
<div class="row">
  <div class="col-6 col-md-4">.col-6 .col-md-4</div>
  <div class="col-6 col-md-4">.col-6 .col-md-4</div>
  <div class="col-6 col-md-4">.col-6 .col-md-4</div>
</div>
```

```
<!-- Columns are always 50% wide, on mobile and desktop -->
<div class="row">
  <div class="col-6">.col-6</div>
  <div class="col-6">.col-6</div>
</div>
```

复制

常用的响应式布局一般抄自bootstrap 😅

# 响应式栅栏

	<b>Extra small</b> ≤576px	<b>Small</b> ≥576px	<b>Medium</b> ≥768px	<b>Large</b> ≥992px	<b>Extra large</b> ≥1200px
<b>Max container width</b>	None (auto)	540px	720px	960px	1140px
<b>Class prefix</b>	.col-	.col-sm-	.col-md-	.col-lg-	.col-xl-
<b># of columns</b>	12				
<b>Gutter width</b>	30px (15px on each side of a column)				

Bootstrap 4的断点

# Slot插槽——Web Component (Level3)

<slot>来自于Web Component，用于解决对Component的DOM注入。

具体参考：[https://developer.mozilla.org/zh-CN/docs/Web/Web\\_Components/Using\\_templates\\_and\\_slots](https://developer.mozilla.org/zh-CN/docs/Web/Web_Components/Using_templates_and_slots)

```

1 <!DOCTYPE html>
2 <html>
3 <head>
4   <meta charset="utf-8">
5   <title>Simple template</title>
6   <script src="main.js" defer></script>
7 </head>
8 <body>
9   <h1>Simple template</h1>
10
11  <template id="my-paragraph">
12    <style>
13      p {
14        color: white;
15        background-color: #666;
16        padding: 5px;
17      }
18    </style>
19    <p><slot name="my-text">My default text</slot></p>
20  </template>
21
22  <my-paragraph>
23    <span slot="my-text">Let's have some different text!</span>
24  </my-paragraph>
25
26  <my-paragraph>
27    <ul slot="my-text">
28      <li>Let's have some different text!</li>
29      <li>In a list!</li>
30    </ul>
31  </my-paragraph>
32
33 </body>
34 </html>
35

```

```

customElements.define('my-paragraph',
  class extends HTMLElement {
    constructor() {
      super();

      const template = document.getElementById('my-paragraph');
      const templateContent = template.content;

      this.attachShadow({mode: 'open'}).appendChild(
        templateContent.cloneNode(true)
      );
    }
  }
);

const slottedSpan = document.querySelector('my-paragraph span');

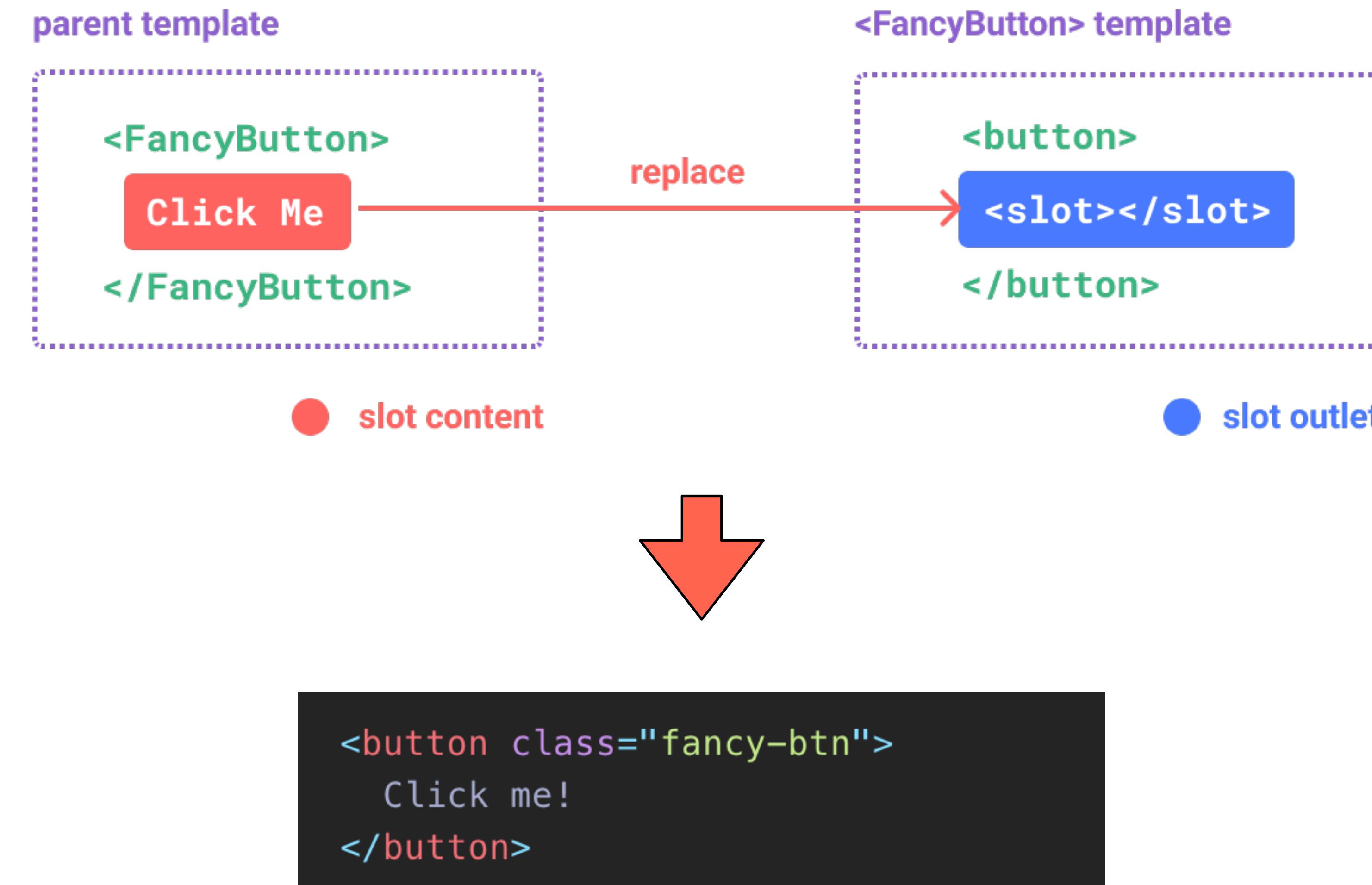
```

## Simple template

Let's have some different text!

- Let's have some different text!
- In a list!

# Slot插槽



# Slot插槽

fallback content (可以看成默认)

SubmitButton定义：

```
<button type="submit">
  <slot>
    Submit <!-- fallback content -->
  </slot>
</button>
```

直接用，不填东西：

```
<SubmitButton />
```

则：

```
<button type="submit">Submit</button>
```

named slots (就是有名字的插槽)

BaseLayout定义：

```
<div class="container">
  <header>
    <slot name="header"></slot>
  </header>
  <main>
    <slot></slot>
  </main>
  <footer>
    <slot name="footer"></slot>
  </footer>
</div>
```

那我可以这么填东西：

```
<BaseLayout>
  <template v-slot:header>
    <!-- content for the header slot -->
  </template>
</BaseLayout>
```

v-slot:header也可以简写成：#header

# Slot插槽

我们可以很轻易地利用`<slot>`将页面Layout和楼层栅栏容器解耦，使得外部可以控制什么楼层注入Layout。

```
1 <script>
2   import Layout from './components/layout.vue'
3   import Grid from './components/grid.vue'
4   import FlexGrid from './components/flex-grid.vue'
5
6   export default {
7     props: {
8       page: Array
9     },
10    components: {
11      'grid': Grid,
12      'flex-grid': FlexGrid,
13      Layout
14    },
15  }
16 </script>
17
18 <template>
19   <Layout>
20     <template #default>
21       <div v-for="floor in page">
22         <component :floor="floor" :is="floor.type"></component>
23       </div>
24     </template>
25   </Layout>
26 </template>
```

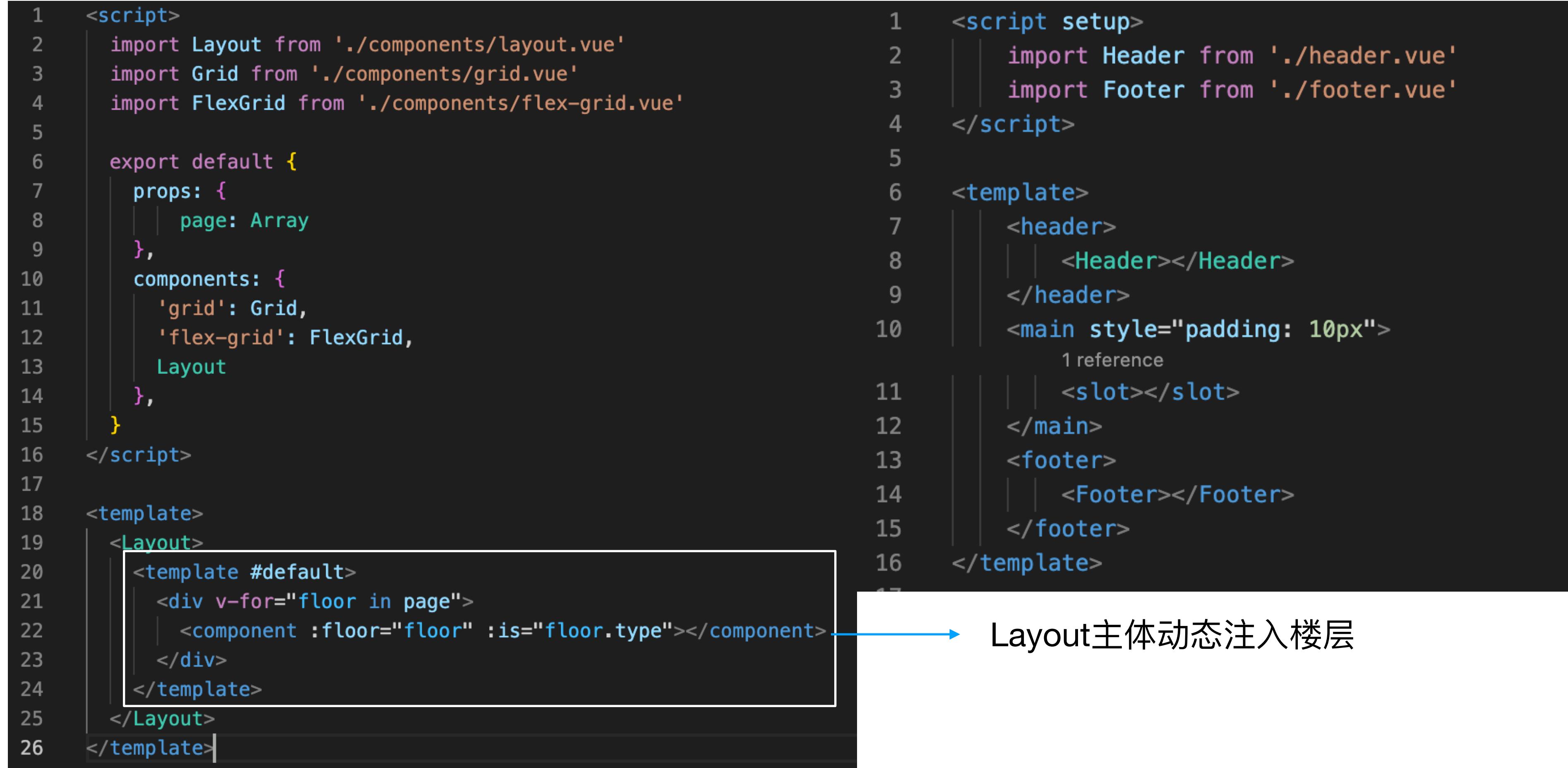
```
1 <script setup>
2   import Header from './header.vue'
3   import Footer from './footer.vue'
4 </script>
5
6 <template>
7   <header>
8     <Header></Header>
9   </header>
10  <main style="padding: 10px">
11    <slot></slot>
12  </main>
13  <footer>
14    <Footer></Footer>
15  </footer>
16 </template>
```

Layout主体动态注入楼层

# Slot插槽——课后思考题 🤔 (Level3)

💡 当前楼层卡片类型其实是跟楼层有绑定关系，这样不利于楼层扩展，应该怎么做更好？

<https://github.com/FE-star/Plan-2022/issues/2>

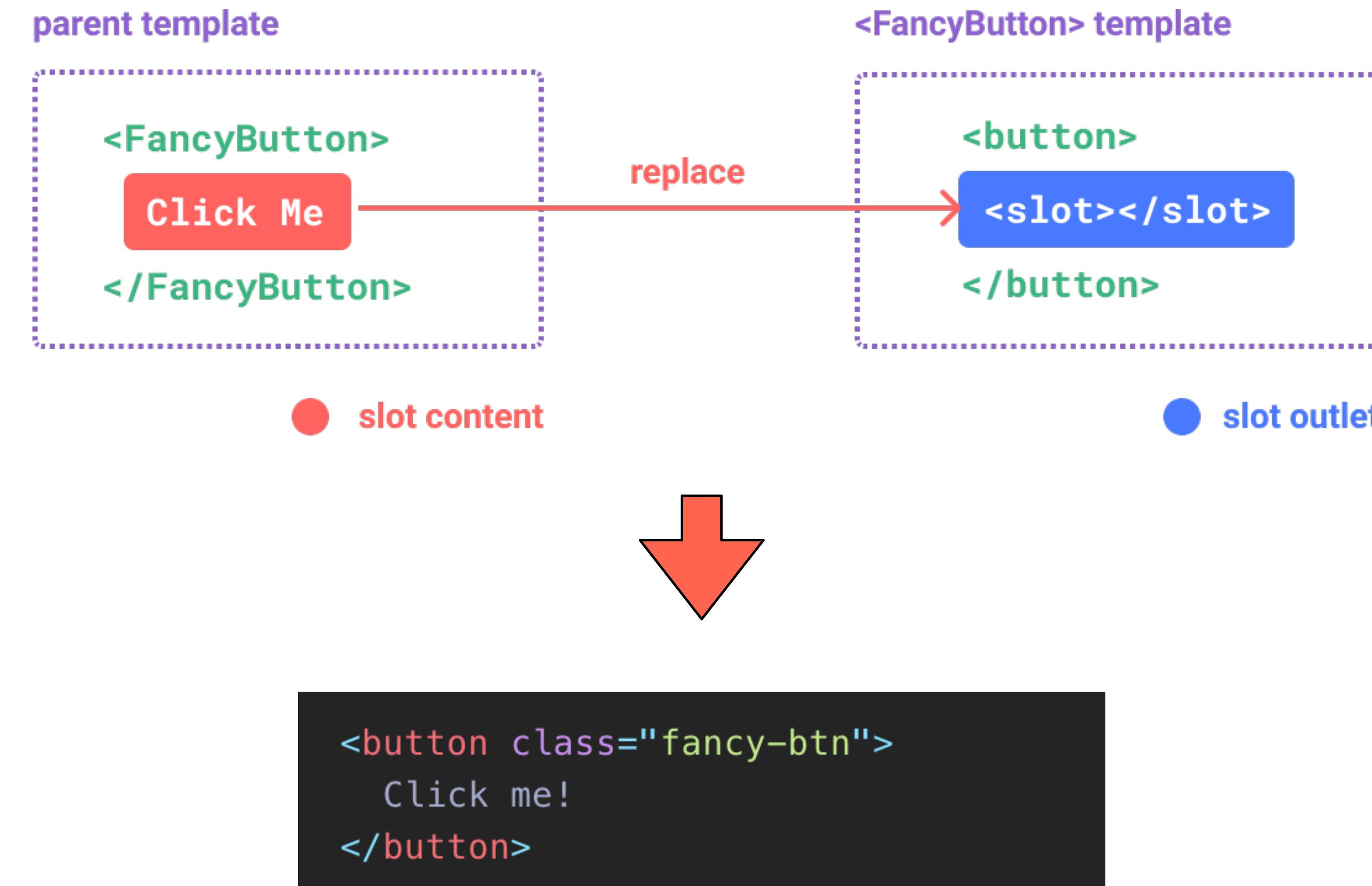


```
1 <script>
2   import Layout from './components/layout.vue'
3   import Grid from './components/grid.vue'
4   import FlexGrid from './components/flex-grid.vue'
5
6   export default {
7     props: {
8       page: Array
9     },
10    components: {
11      'grid': Grid,
12      'flex-grid': FlexGrid,
13      Layout
14    },
15  }
16 </script>
17
18 <template>
19   <Layout>
20     <template #default>
21       <div v-for="floor in page">
22         <component :floor="floor" :is="floor.type"></component>
23       </div>
24     </template>
25   </Layout>
26 </template>
```

```
1 <script setup>
2   import Header from './header.vue'
3   import Footer from './footer.vue'
4 </script>
5
6 <template>
7   <header>
8     <Header></Header>
9   </header>
10  <main style="padding: 10px">
11    <slot></slot>
12  </main>
13  <footer>
14    <Footer></Footer>
15  </footer>
16 </template>
```

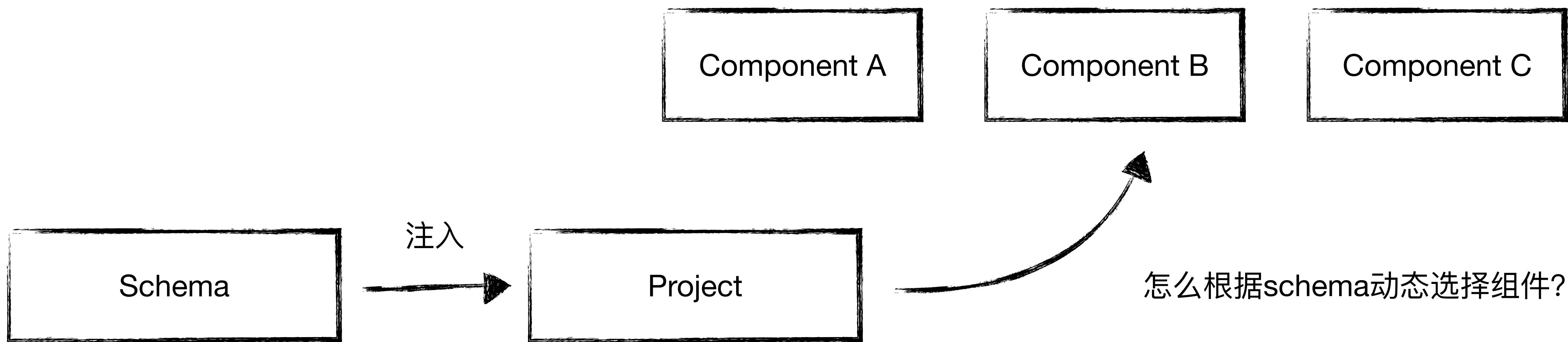
Layout主体动态注入楼层

# Slot插槽



## 二.2、Vue动态组件选择

# 组件动态选择



# <component :is>

```
<script>
import Foo from './Foo.vue'
import Bar from './Bar.vue'

export default {
  components: { Foo, Bar },
  data() {
    return {
      view: 'Foo'
    }
  }
}
</script>

<template>
  <component :is="view" />
</template>
```

# <component :is>

```
<script setup>
import Foo from './Foo.vue'
import Bar from './Bar.vue'
</script>

<template>
  <component :is="Math.random() > 0.5 ? Foo : Bar" />
</template>
```

# Vue动态组件选择

```
1 <script>
2   import Layout from './components/layout.vue'
3   import Grid from './components/grid.vue'
4   import FlexGrid from './components/flex-grid.vue'
5
6   export default {
7     props: {
8       page: Array
9     },
10    components: {
11      'grid': Grid,
12      'flex-grid': FlexGrid,
13      Layout
14    },
15  }
16 </script>
17
18 <template>
19   <Layout>
20     <template #default>
21       <div v-for="floor in page">
22         <component :floor="floor" :is="floor.type"></component>// 高亮显示动态组件选择语句
23       </div>
24     </template>
25   </Layout>
26 </template>
```

```
1 <script setup>
2   import Header from './header.vue'
3   import Footer from './footer.vue'
4 </script>
5
6 <template>
7   <header>
8     <Header></Header>
9   </header>
10  <main style="padding: 10px">
11    1 reference
12    <slot></slot>
13  </main>
14  <footer>
15    <Footer></Footer>
16  </footer>
17 </template>
```

指向第22行代码的箭头注释：动态选择component

# 课间测试

Vue Slot设计灵感来自于?

A、Polymer

B、Google

C、Svelte

D、Web Component

# 课间测试

<component is="span">hello world</component> 是啥意思？

休息一下，等下讲性能～～

## 三、性能优化

1. 缓存优化
2. 请求优化
3. 运行效率优化

### 三.1、缓存优化

HTTP缓存

本地存储

客户端技术

### 三.1、缓存优化

HTTP缓存

# HTTP缓存——Expires

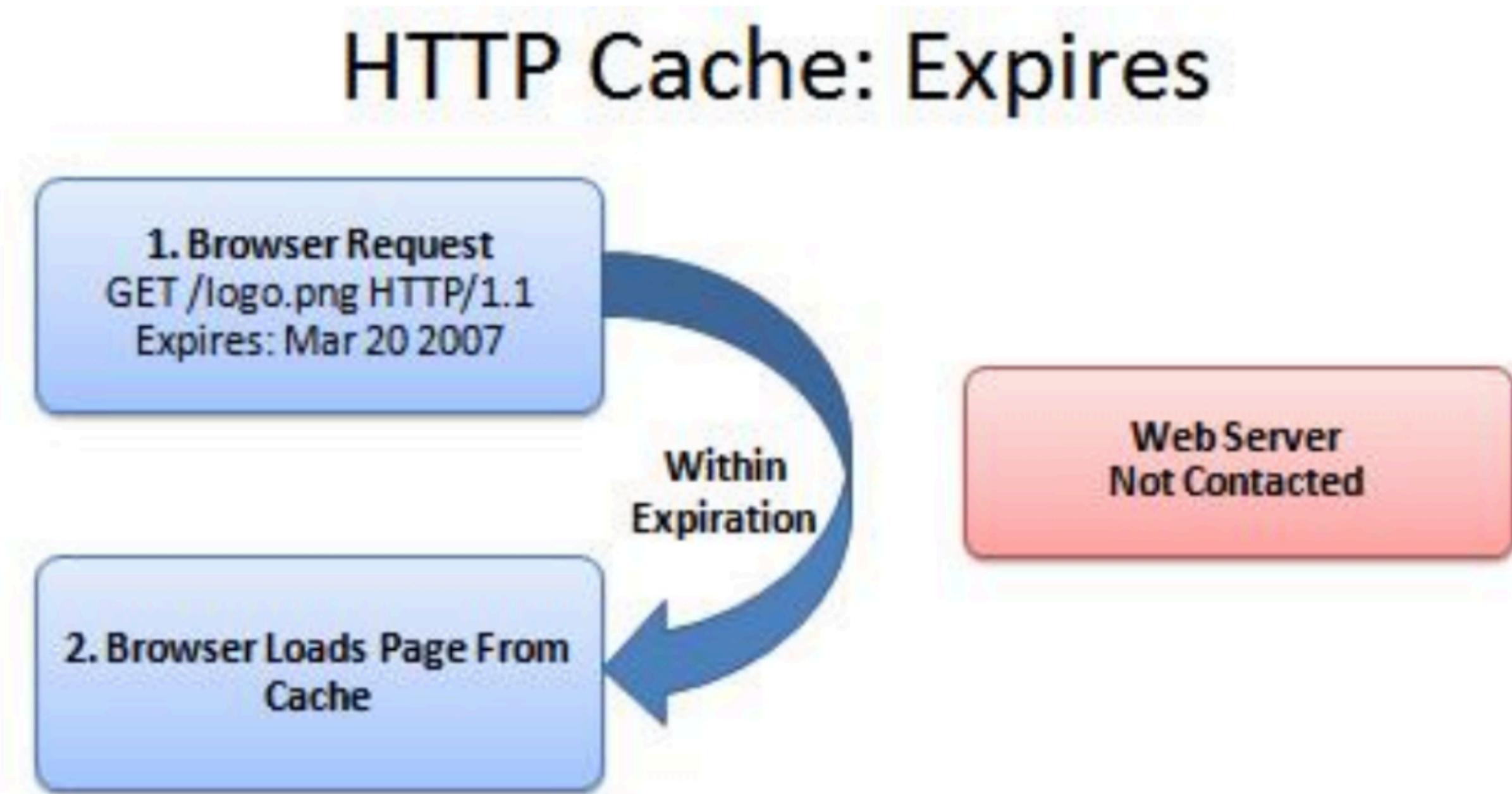
设置对象有效期

## ▼ 响应标头

```
access-control-allow-origin: *
age: 288869
ali-swift-global-savetime: 1656725115
cache-control: max-age=31536000
content-length: 150090
content-type: image/png
date: Sat, 02 Jul 2022 01:25:15 GMT
eagleid: 7499481f16570139849097568e
expires: Sun, 02 Jul 2023 01:25:15 GMT
last-modified: Wed, 29 Jun 2022 12:55:19 GMT
picasso-ret-code: SUCCESS
request-time: 0.090
s-rt: 62
```

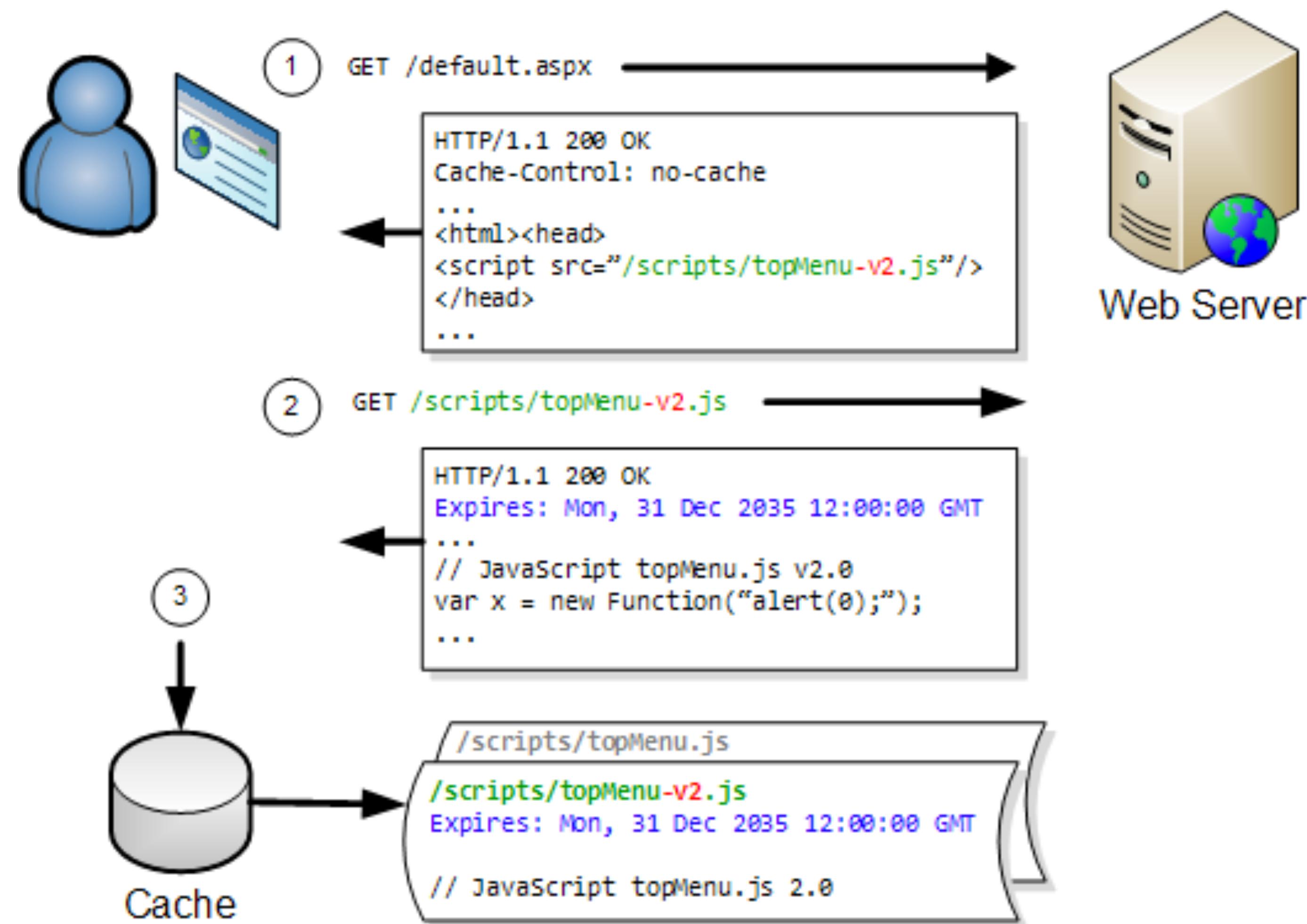
# HTTP缓存——Expires

设置对象有效期



# HTTP缓存——Expires

设置对象有效期



# HTTP缓存——Cache-Control

本地时间可能跟服务器时间不一致

- 1 max-age=num(s), 设置最大缓存时间
- 2 public, 缓存能被多用户共享
- 3 private, 缓存不能在用户间共享
- 4 no-cache, 不会被缓存
- 5 no-store, 不能被存储

# HTTP缓存——last-modified

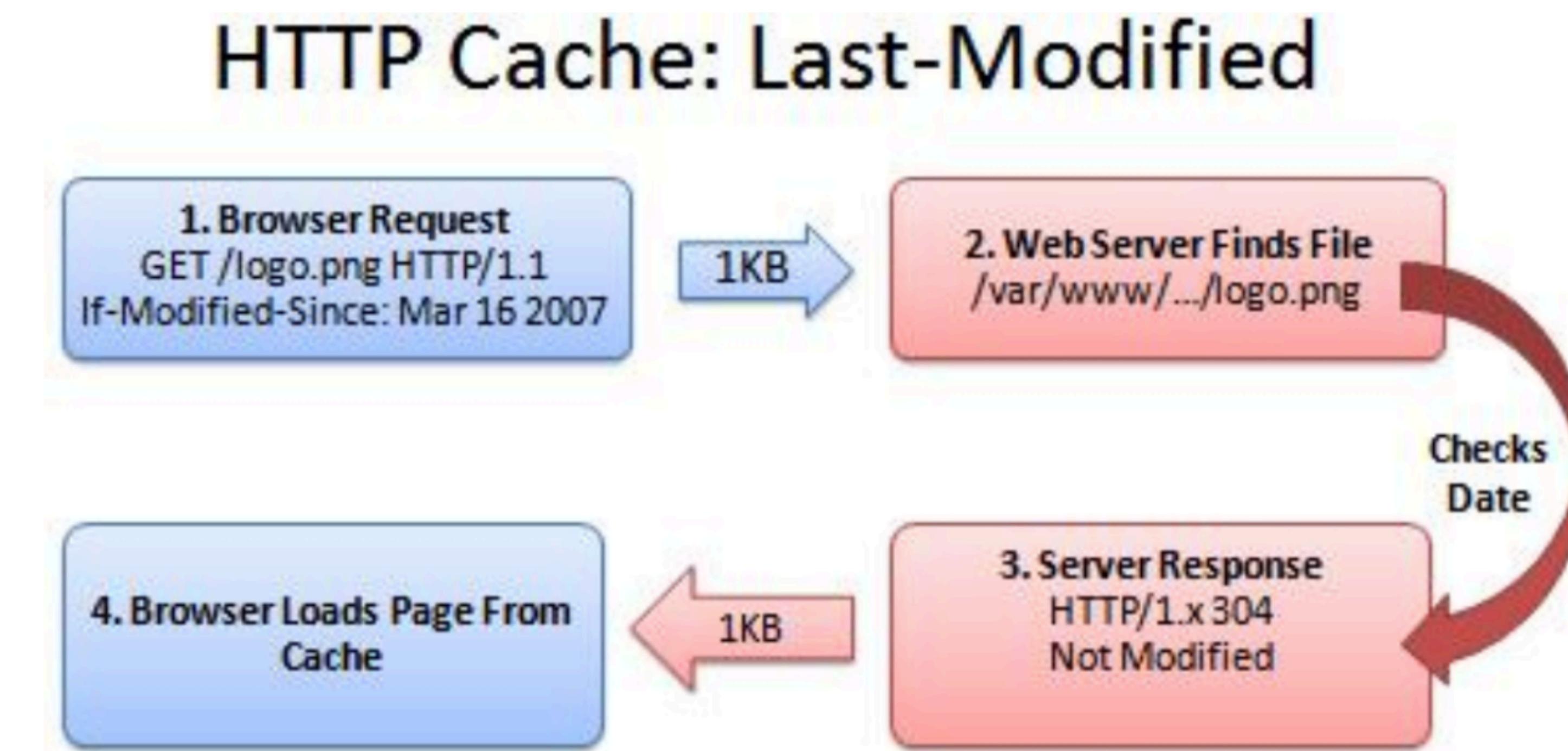
设置对象最后修改时间

## ▼ 响应标头

```
access-control-allow-origin: *
age: 288869
ali-swift-global-savetime: 1656725115
cache-control: max-age=31536000
content-length: 150090
content-type: image/png
date: Sat, 02 Jul 2022 01:25:15 GMT
eagleid: 7499481f16570139849097568e
expires: Sun, 02 Jul 2023 01:25:15 GMT
last-modified: Wed, 29 Jun 2022 12:55:19 GMT
picasso-ret-code: SUCCESS
request-time: 0.090
s-rt: 62
```

# HTTP缓存——last-modified

设置对象最后修改时间



# HTTP缓存——Etag

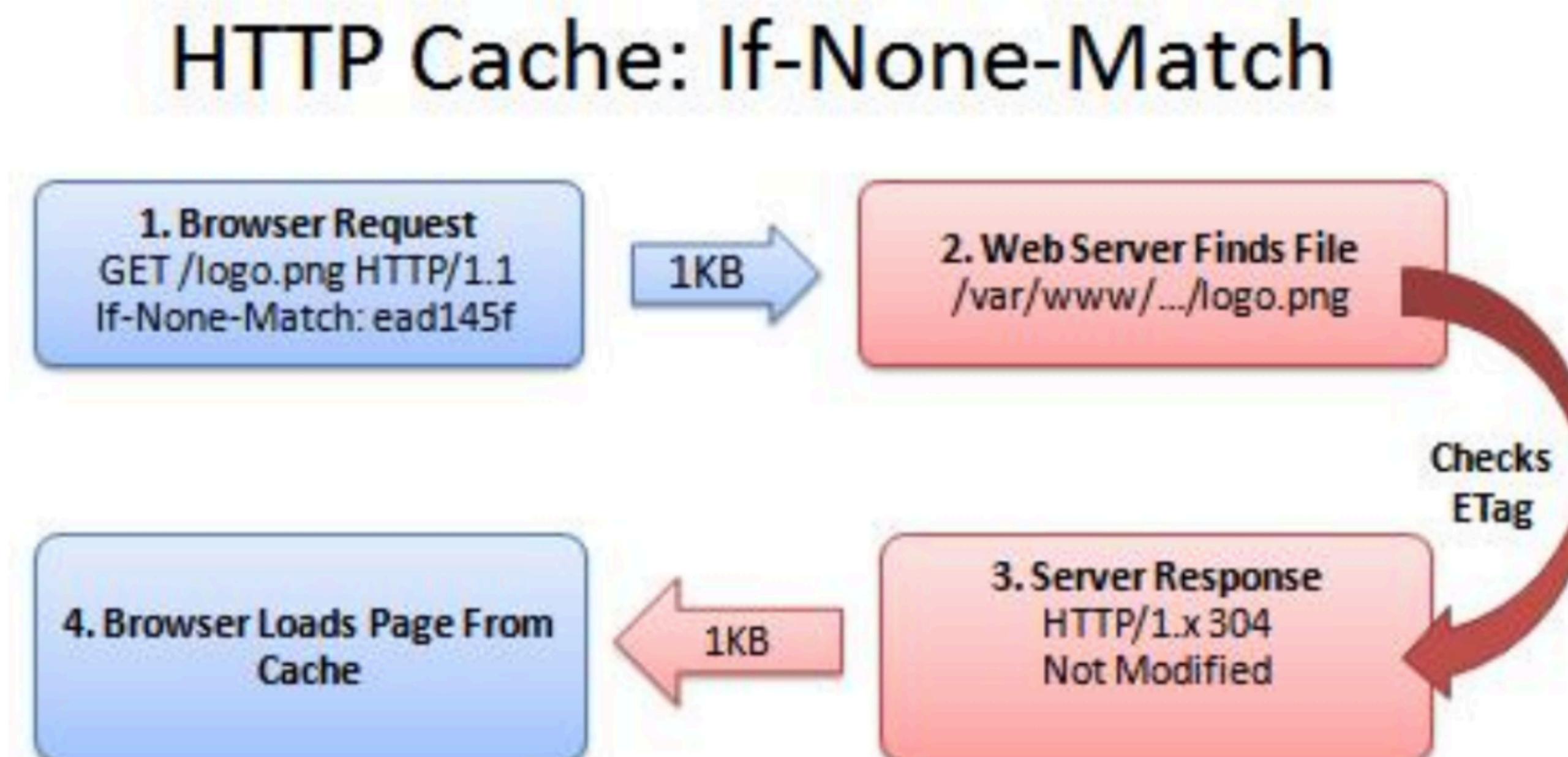
可能在同一秒保存两次，也可能服务器上文件修改时间不一致

## ▼ Response Headers

```
age: 39
cache-control: max-age=0, s-maxage=60
content-encoding: gzip
content-type: text/html; charset=utf-8
date: Wed, 11 Oct 2017 12:06:19 GMT
eagleeye-traceid: 0a67920815077235792645080e758e
etag: W/"380f8-CYSgfB0i7zAQT6V8SYCwr/r6GRs"
server: Tengine/Aserver
status: 200
strict-transport-security: max-age=0
timing-allow-origin: *
vary: Origin, Ali-Detector-Type
vary: Accept-Encoding
via: db568081[0,200-0,H]
x-cache: HIT TCP_MEM_HIT dirn:-2:-2
x-readtime: 118
x-response-time: 116
x-server-id: wormholesource010178132229.test.et2
```

# HTTP缓存——Etag

可能在同一秒保存两次，也可能服务器上文件修改时间不一致



# HTTP缓存——Etag

备课时候发现淘宝和腾讯的CDN都没有Etag~查了下发现：

```
ngx_int_t
ngx_http_set_etag(ngx_http_request_t *r)
{
    ngx_table_elt_t          *etag;
    ngx_http_core_loc_conf_t *clcf;

    clcf = ngx_http_get_module_loc_conf(r, ngx_http_core_module);

    if (!clcf->etag) {
        return NGX_OK;
    }

    etag = ngx_list_push(&r->headers_out.headers);
    if (etag == NULL) {
        return NGX_ERROR;
    }

    etag->hash = 1;
    ngx_str_set(&etag->key, "ETag");

    etag->value.data = ngx_pnalloc(r->pool, NGX_OFF_T_LEN + NGX_TIME_T_LEN + 3);
    if (etag->value.data == NULL) {
        etag->hash = 0;
        return NGX_ERROR;
    }

    etag->value.len = ngx_sprintf(etag->value.data, "%"OFF_T%"x0"",
                                   r->headers_out.last_modified_time,
                                   r->headers_out.content_length_n)
                     - etag->value.data;

    r->headers_out.etag = etag;

    return NGX_OK;
}
```

nginx Etag = fn(文件最后修改时间, 文件内容长度)

由于CDN每次回源都会导致文件最后修改时间变化,  
所以.....

分析下

max-age + Last Modified 会怎么样?

分析下

max-age + Last Modified + Etag 会怎么样？

那么

让我们去设计一个资源缓存方案，应当怎么做？

# 资源缓存方案

1 HTML

2 Javascript、CSS

3 Image

# 资源缓存方案

1 HTML (no cache)

2 Javascript、CSS (MD5/timestamp  
出来)

3 Image (随机名 + 长缓存)

```
1 import { fileURLToPath, URL } from 'url'
2
3 import { defineConfig } from 'vite'
4 import vue from '@vitejs/plugin-vue'
5 import { viteExternalsPlugin } from 'vite-plugin-externals'
6
7 // https://vitejs.dev/config/
8 export default defineConfig({
9   plugins: [
10     vue(),
11     viteExternalsPlugin({
12       vue: 'Vue'
13     })
14   ],
15   resolve: {
16     alias: {
17       '@': fileURLToPath(new URL('./src', import.meta.url))
18     }
19   }
20 })
21 |
```

# 将库拆出来，利用缓存

```
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4  |  <meta charset="UTF-8" />
5  |  <link rel="icon" href="/favicon.ico" />
6  |  <meta name="viewport" content="width=device-width, initial-scale=1.0" />
7  |  <script src="https://cdnjs.cloudflare.com/ajax/libs/vue/3.2.37/vue.global.prod.min.js"></script>
8  |  <title>Vite App</title>
9  </head>
10 <body>
11 |  <div id="app"><!--app-html--></div>
12 |  <script type="module" src="/src/entry-client.js"></script>
13 </body>
14 </html>
```

```
1  import { fileURLToPath, URL } from 'url'
2
3  import { defineConfig } from 'vite'
4  import vue from '@vitejs/plugin-vue'
5  import { viteExternalsPlugin } from 'vite-plugin-externals'
6
7  // https://vitejs.dev/config/
8  export default defineConfig({
9    plugins: [
10      vue(),
11      viteExternalsPlugin({
12        vue: 'Vue'
13      })
14    ],
15    resolve: {
16      alias: {
17        '@': fileURLToPath(new URL('./src', import.meta.url))
18      }
19    }
20  })
```

<https://cdnjs.cloudflare.com/ajax/libs/vue/3.2.37/vue.global.prod.min.js>

# 将库拆出来，利用缓存

中国大陆 亲，请登录 免费注册 手机逛淘宝 网页无障碍

**猜你喜欢**



置物架不锈钢色储物架厨房五层落地可调节收纳架

送运费险

¥ 125



北美红雀冬青浆果欧式陶瓷餐具汤锅汤碗汤盆手绘

送运费险

¥ 125



吉普尔士胸内搭女士打

元素 控制台 源代码 网络 性能 内存 应用 安全 Lighthouse Recorder »

过滤  反转  隐藏数据网址  全部  Fetch/XHR  JS  CSS  图片  媒体  字体  文档  WS  Wasm  清单  其他

有已拦截的 Cookie  被屏蔽的请求  第三方请求

瀑布图 (240 毫秒)

20 毫秒	40 毫秒	60 毫秒	80 毫秒	100 毫秒	120 毫秒	140 毫秒	160 毫秒	180 毫秒	200 毫秒	220 毫秒	240 毫秒
—	—	—	—	—	—	—	—	—	—	—	—

名称	状态	类型	启动器	大小	时间	瀑布
localhost	200	document	其他	72.8 kB	14 毫秒	█
vue.global.prod.min.js	200	script	(索引)	(内存缓...)	0 毫秒	█
index.706ab148.js	304	script	(索引)	266 B	5 毫秒	█
index.52f6829e.css	304	stylesheet	(索引)	266 B	7 毫秒	█
data.5b2b33b7.js	304	script	index.706ab148.js:1	267 B	3 毫秒	█
font_1404888168_2057645.woff	200	font	index.52f6829e.css	(内存缓...)	0 毫秒	█
data:image/gif;base64...	200	gif	index.706ab148.js:1	(内存缓...)	0 毫秒	█
favicon.ico	304	x-icon	其他	266 B	2 毫秒	█
TB2CWKUD79WBuNjSspeXXaz5VXa...	200	jpeg	index.706ab148.js:1	(内存缓...)	0 毫秒	█
O1CN01F9w7VR1Y0fgB6hZMD_!!227...	200	jpeg	index.706ab148.js:1	(内存缓...)	0 毫秒	█
O1CN01YYjm0f1K0IEATUOkc_!!0-sat...	200	jpeg	index.706ab148.js:1	(内存缓...)	0 毫秒	█
TB18YHBKFXXXXaVXpXXXXXXXXXX_...	200	jpeg	index.706ab148.js:1	(内存缓...)	0 毫秒	█

### 三.1、缓存优化

本地存储

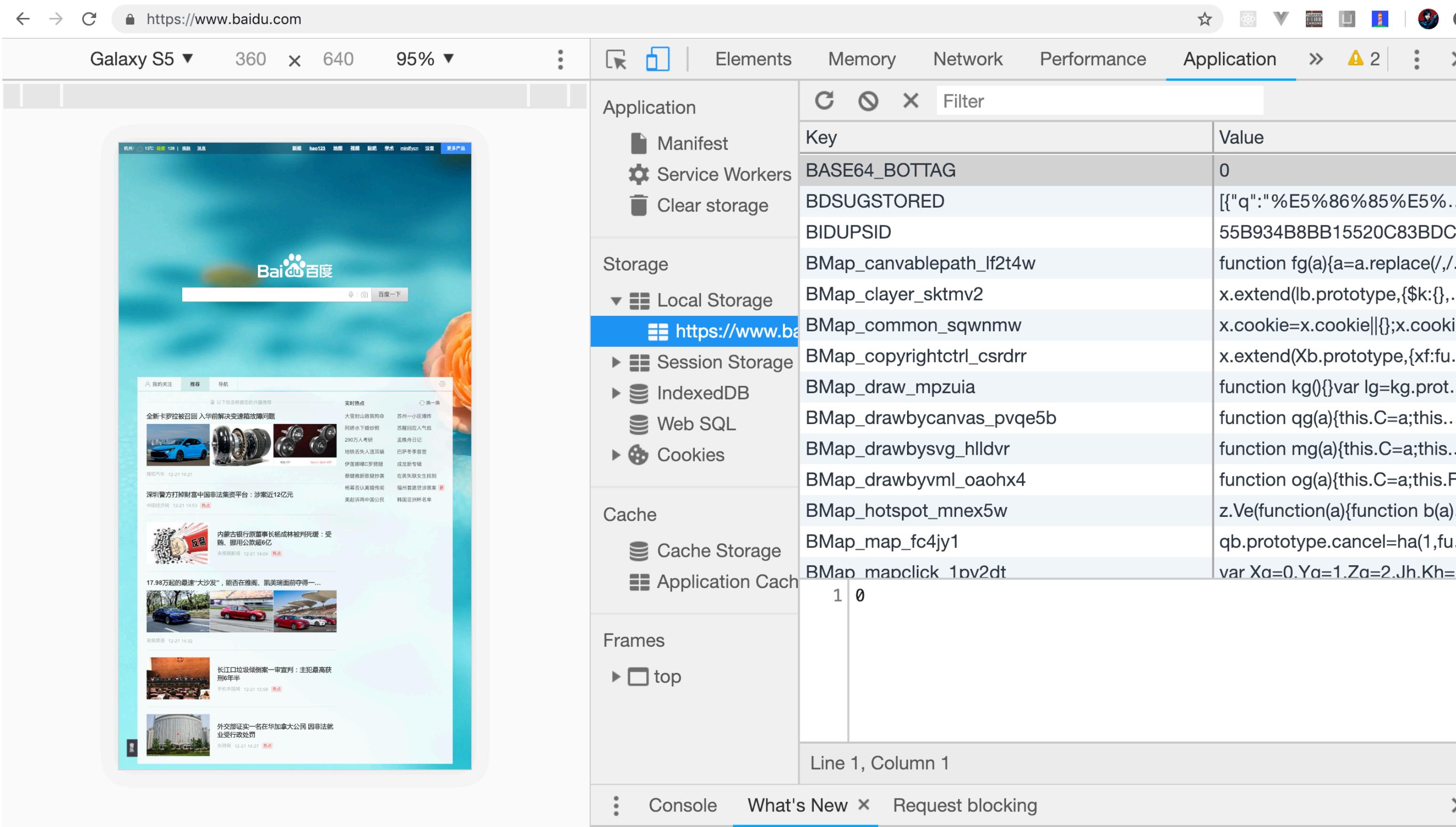
# Cookies

存储数据量有限，并且每次都会带到服务器，占用宽带

```
accept-encoding: gzip, deflate, br
accept-language: zh-CN,zh;q=0.8,en;q=0.6
cache-control: max-age=0
cookie: pgv_pvi=251840512; tvfe_boss_uuid=57bf934d0521fd9e; RK=jS/SftE/YX; pac_
uid=1_734205675; pgv_si=s7860182016; LW_uid=u1i5J0e3C2h3g8j5J0V601L9F2; eas_si
d=a115k0a3g2i3Q8T540s6L1k9C8; p_uin=o0734205675; p_skey=*SJK*GeS80n-0ByxIrGGjY
*5iL9GiTJ4NPQJja7Mmm4_; pt4_token=Fzvc-gorkgW8EHnhPZIASGlvcQbYQGgUZoa0J1ICKoQ
_; rv2=80F26E9BAA13A780259E322EB5E1033AB0A30B510A007D79FE; property20=E74A988E
425AC91058E1252BABADD6BBAD7DEDB3A57F5D3994A215FAA167F715C2700E199938D129; qqmu
sic_uin=; qqmusic_key=; qqmusic_fromtag=; LW_sid=31P500Z441Q7X0M7S5b8J5w4X7; i
swebp=1; qv_als=g8mYxN0uSB5tyUWVA11505999777AVuZrw==; o_cookie=734205675; qm_a
uthimgs_id=1; qm_verifyimagesession=h01f483194679207034df0ee9d2832f7306a2870d5
c2380bd3990ea9ae685ee04309bc5a6ce1b7f714a; ptisp=cm; ptcz=2a1f195ae54693b61e4b
72e8760527e92c335a411bb72bd87a403c81f98823fe; pt2ggui=o0734205675; uin=o07342
05675; skey=@WzCo5RVb4; enc_uin=Mi9HgYvTs_9d2X0Mlk7gg; course_origin=[{"cid":163603,"ext":{"pagelocation":"newagencyhome,1.10"}}]; pgv_info=ssid=s557227165
1&pgvReferrer=http://tiem-cdn.qq.com/html/mms/nativeads/game_rank.html; ts_ref
er=imweb.ke.qq.com/; pgv_pvid=6707401337; ts_uid=9213423499; webwx_data_ticket
=gSdp0BacJCa+NjvlWdjGmz6u; _pathcode=0.9115907389153888; tdw_data={"ver4":"4
1","ver5":"","ver6":"","refer":"www.baidu.com","from_channel":"41","path":"a-
0.9115907389153888","uin":734205675}; ke_login_type=1; tdw_first_visited=1
```

# LocalStorage

有大小限制，数据不安全，不应当把重要数据存在这



The screenshot shows a browser window with the URL <https://www.baidu.com>. The developer tools are open, specifically the Application tab under the Storage section. The Local Storage table lists numerous items, many of which contain sensitive or executable code. For example, the item `BMap_canvablepath_lf2t4w` contains the function `fg(a){a=a.replace(/,/..`. Another item `BMap_map_fc4jy1` contains `qb.prototype.cancel=ha(1,fu..`. This demonstrates that sensitive data, including functions and scripts, can be stored in LocalStorage, which is a security risk.

Key	Value
BASE64_BOTTAG	0
BDSUGSTORED	[{"q": "%E5%86%85%E5%...
BIDUPSID	55B934B8BB15520C83BDC.
BMap_canvablepath_lf2t4w	function fg(a){a=a.replace(/,/..
BMap_clayer_sktmv2	x.extend(lb.prototype,{\\$k:{}..
BMap_common_sqwnmw	x.cookie=x.cookie  {};x.cooki.
BMap_copyrightctrl_csrdr	x.extend(Xb.prototype,{xf:fu..
BMap_draw_mpzuia	function kg(){var lg=kg.prot...
BMap_drawbycanvas_pvqe5b	function qg(a){this.C=a;this....
BMap_drawbysvg_hlldvr	function mg(a){this.C=a;this...
BMap_drawbyvml_oaohx4	function og(a){this.C=a;this.F.
BMap_hotspot_mnex5w	z.Ve(function(a){function b(a)..
BMap_map_fc4jy1	qb.prototype.cancel=ha(1,fu..
BMap_mapclick_1pv2dt	var Xa=0.Ya=1.Za=2.Jh.Kh=.

古老的百度首页会把代码存在localStorage

# LocalStorage——课后思考题 (Level3)

localStorage能干嘛？或者你知道localStorage有哪些地方应用？

由于营销型页面一般不会反复打开，  
缓存用途不大，故在这里不具体讲应用

# SessionStorage

很少使用的东西，大家知道跟localStorage的差别是什么吗？

# IndexedDB

持久化存储，可在 Web Worker 中使用。

为什么很少见使用 IndexedDB 来做缓存？

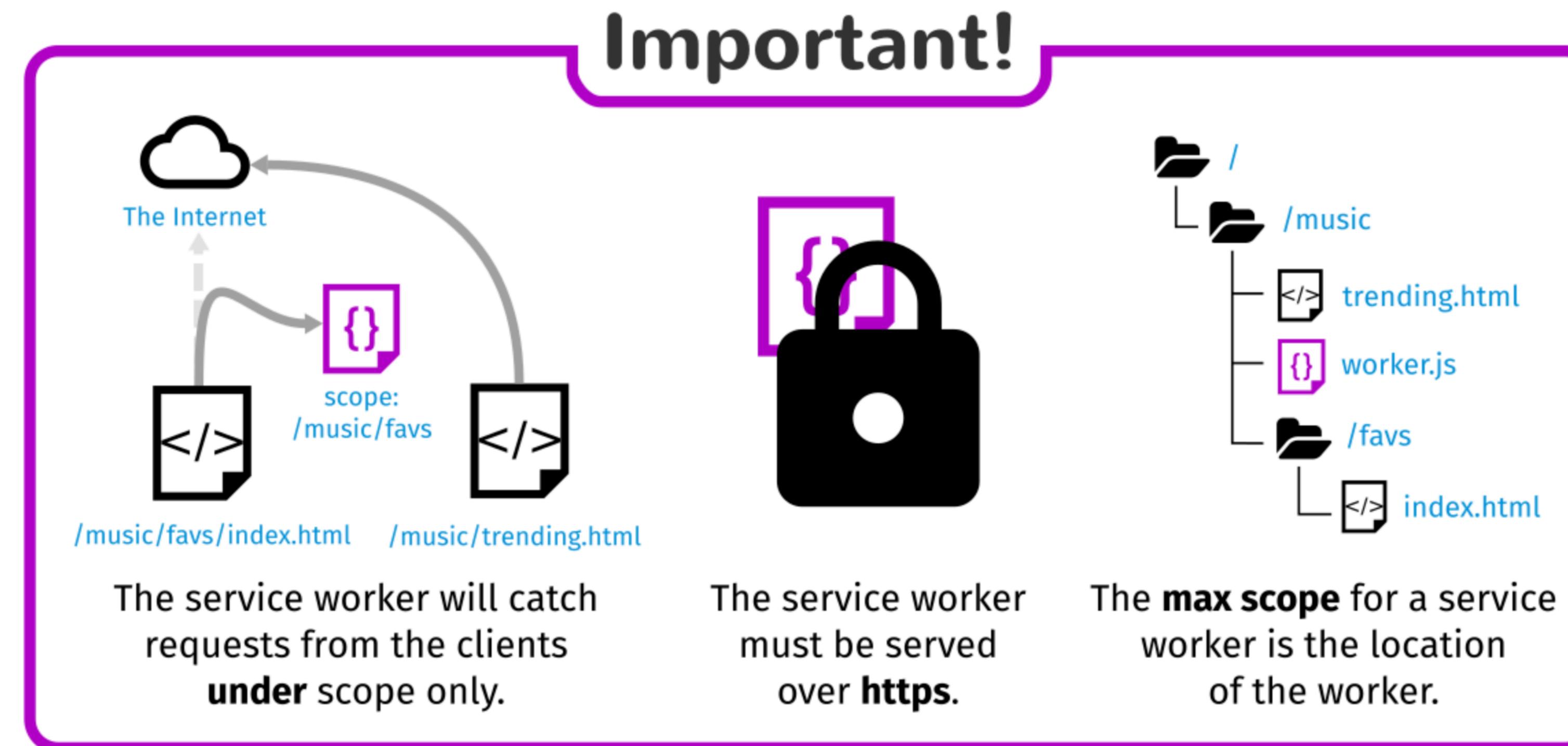
# App Cache

反正是大坑货，不要用就是了

- 1 可能会缓存错误页面，甚至缓存运营商的劫持广告
- 2 更新中有一个文件失败了，就会全部退回上一个版本
- 3 由于是 Lazy 更新，如果后台接口有 break change 将会是大麻烦
- 4 manifest 文件也有可能不小心被缓存了

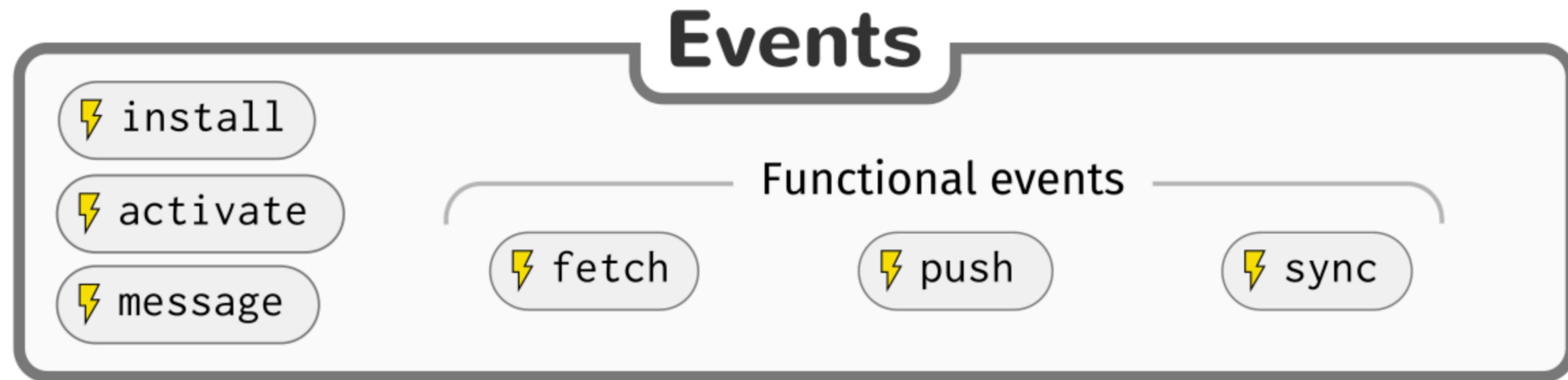
# Service Worker

App Cache 太失败，以至于 W3C 终于明白用户需要更细致的缓存控制



# Service Worker

App Cache 太失败，以至于 W3C 终于明白用户需要更细致的缓存控制



<https://serviceworke.rs/>

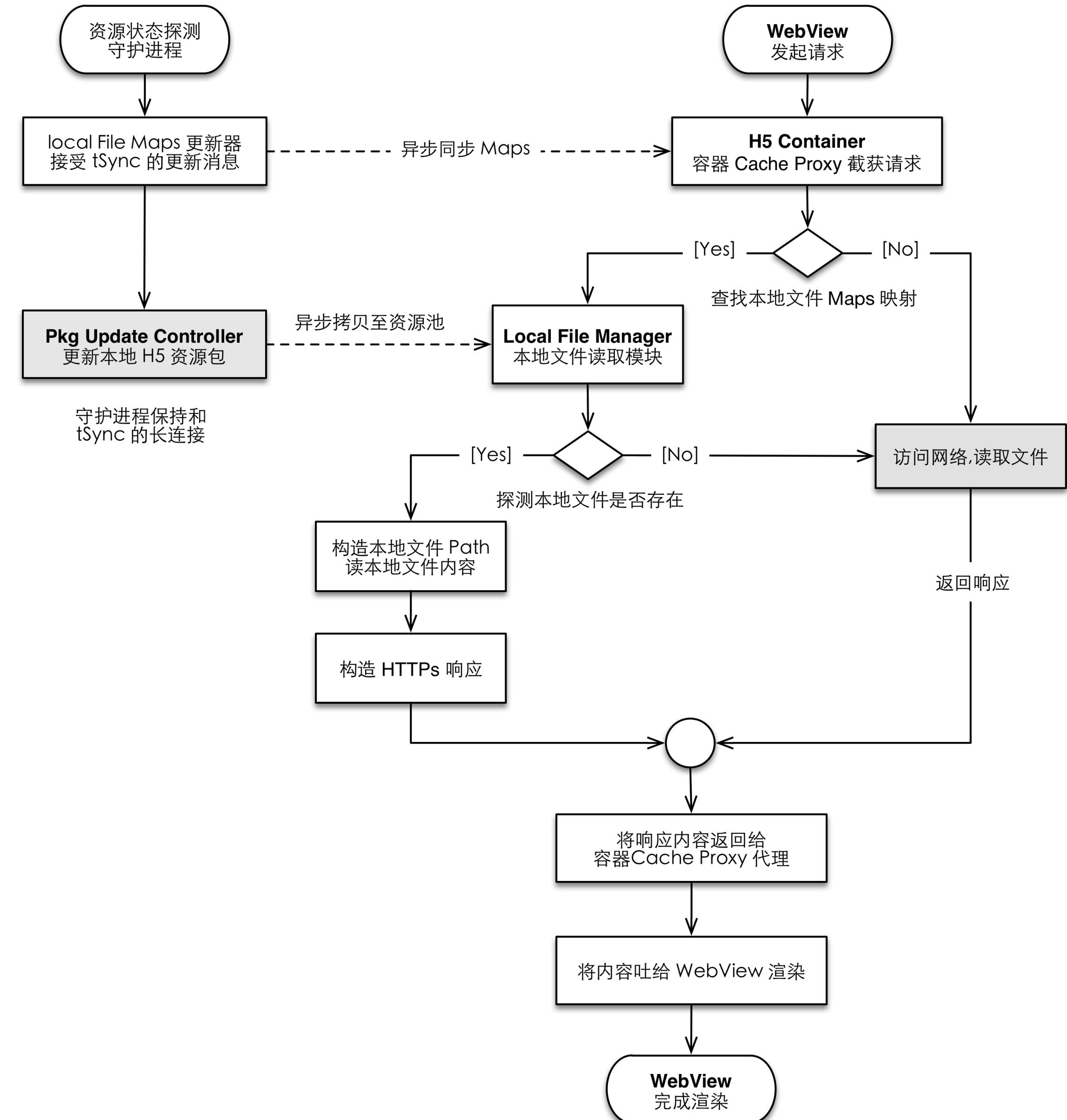
### 三.1、缓存优化

客户端技术（不仅仅缓存）

# 离线包技术

大公司用了很多年了，可以看成是私有service worker

<https://yq.aliyun.com/articles/2939>



活动离线化管理 > 页面管理

标题	URL	提交者	提交时间	状态	操作
双11环游世界	<a href="http://quamarket.m.taobao.com/markets/h5/15qa1111/hc/index-m">http://quamarket.m.taobao.com/markets/h5/15qa1111/hc/index-m</a>	弘树	2015-10-13 10:53:18	在线	
我的双11	<a href="http://quamarket.m.taobao.com/markets/h5/15qa1111/hc/myCenter">http://quamarket.m.taobao.com/markets/h5/15qa1111/hc/myCenter</a>	弘树	2015-10-15 19:49:50	在线	
双11玩转中国	<a href="http://quamarket.m.taobao.com/markets/h5/15qa1111/hc/index-ne-m?spm=181.7788770.38200.2">http://quamarket.m.taobao.com/markets/h5/15qa1111/hc/index-ne-m?spm=181.7788770.38200.2</a>	弘树	2015-10-16 17:17:06	在线	
双11酒店	<a href="https://quamarket.m.taobao.com/markets/h5/15qa1111/hc/hotel-m">https://quamarket.m.taobao.com/markets/h5/15qa1111/hc/hotel-m</a>	弘树	2015-10-22 12:11:41	在线	
双11未来景区	<a href="https://quamarket.m.taobao.com/markets/h5/15qa1111/hc/ticket-brand">https://quamarket.m.taobao.com/markets/h5/15qa1111/hc/ticket-brand</a>	弘树	2015-10-22 12:12:33	在线	
双11分期购	<a href="https://quamarket.m.taobao.com/markets/h5/15qa1111/hc/fqg-m">https://quamarket.m.taobao.com/markets/h5/15qa1111/hc/fqg-m</a>	弘树	2015-10-22 12:12:57	在线	
双11港台站	<a href="https://quamarket.m.taobao.com/markets/h5/hongkong">https://quamarket.m.taobao.com/markets/h5/hongkong</a>	弘树	2015-10-22 14:08:53	在线	
双11旅行红包	<a href="http://quamarket.m.taobao.com/markets/h5/15qa1111/hd/lottery-tianmao">http://quamarket.m.taobao.com/markets/h5/15qa1111/hd/lottery-tianmao</a>	弘树	2015-10-26 12:59:43	在线	
双11分会场-门票品牌红包	<a href="http://quamarket.m.taobao.com/markets/h5/15qa1111/hc/ticket-redpack">http://quamarket.m.taobao.com/markets/h5/15qa1111/hc/ticket-redpack</a>	弘树	2015-10-29 16:39:02	在线	
落跑去台湾	<a href="http://h5.m.taobao.com/market/trip/act/luopaobaaqing.html">http://h5.m.taobao.com/market/trip/act/luopaobaaqing.html</a>	弘树	2015-08-28 16:06:12	已下线	

提交新页面 立即同步 监控中 上一页 1 2 3 4 5 6 下一页

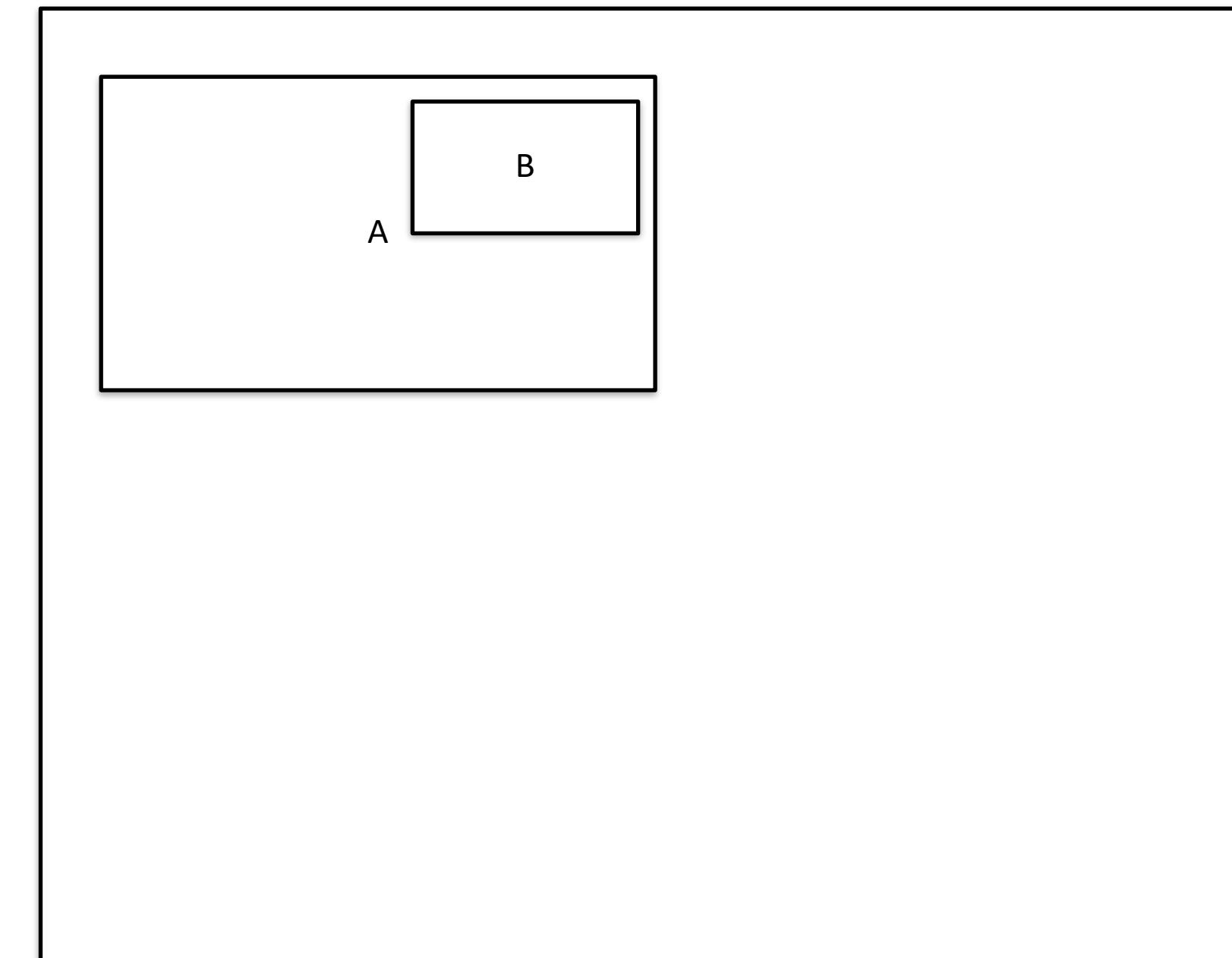
© 2015 阿里旅行 · 去啊前端团队  
技术支持: @弘树

# 离线包技术

同样的我们也可以跟客户端一起开发模拟service worker的其他技术，比如有一个js进程可以偷偷的装载.....

# ReactNative (Weex死了没事烧香)

不要用ReactNative写组件.....由于其盒模型是通过计算转换来的，嵌套越多计算难度会越大，结果.....



图层绝对定位  
vs  
栅栏相对定位

# Q&A

# THANKS

