

**前端生态：演化与思考**

# **THE FRONTEND ECO**



**Stylus**

**TypeScript**

**HAML**

**ECMAScript201**

**Script2015**

**SASS**

**LESS**

**BootStrap**

**WebS**

**npm**

**jQuery**

**Ajax**

**CoffeeScript**

**Angular**

**ularjs**

**React**

**Vue**

**Router**

**JEST**

**Jasn**

**t**

**Gulp**

**Webpack**

**Backbonejs**

**Browserify**

**Browserify**

**CoffeeScript**

**babel**

**TypeScript**



# STAGE 1

## “FORMATTED PAGES”



```
<h1>Hello!</h1>  
<p>Welcome to my homepage.</p>  
  
```

```
<h1>Hello!</h1>  
<p>Welcome to my homepage.</p>  

```

```
# Hello!  
Welcome to my homepage.  

```

# **STAGE 2**

## **“BEAUTIFUL STYLISH PAGES”**



```
<h1>Hello!</h1>
```

```
<p>Welcome to my homepage.</p>
```

```

```

```
<h1 font="Comic Sans">Hello!</h1>
```

```
<p background="#f3fac0">
```

Welcome to my homepage.

```
</p>
```

```

```



```
<h1 style='font:"Comic Sans"'>
```

```
Hello!</h1>
```

```
<p style='background:"#f3fac0"'>
```

```
  Welcome to my homepage.
```

```
</p>
```

```

```

# HTML

```
<h1>Hello!</h1>  
<p>Welcome to my homepage.</p>  

```

# CSS

```
h1 {font: "Comic Sans";}   
p {background: #f3fac0;}   
img {height:20px; width: 20px;}
```

# STAGE 3A

## “INTERACTIVE PAGES”



# HTML

```
<h1 id="title">Hello!</h1>  
<p>Welcome to my homepage.</p>  

```

# JS

```
document.getElementById("title")  
  .onclick=function(){  
    document.getElementsByTagName("img")[0]  
      .style.width="30px"};
```

# JS

```
document.getElementById("title")
  .onclick=function(){
    document.getElementsByTagName("img")[0]
      .style.width="30px";
    document.getElementsByTagName("img")[0]
      .style.height="30px";}
```

# jQuery

```
$('#title').click(
  $('img')[0].css("width", "30px");
  .css("height", "30px");
)
```

# JS

```
var request = new XMLHttpRequest();
request.open('GET', '/my/url', true);

request.onload = function() {
  if (request.status >= 200 && request.status < 400) {

var resp = request.responseText;
  } else {
    // We reached our target server, but it returned an error
  }
};

request.onerror = function() {
  // There was a connection error of some sort
};

request.send();
```

# jQuery

```
$.ajax({
  type: 'GET',
  url: '/my/url',
  success: function(resp) {},
  error: function() {}
});
```

# JS

```
function fadeIn(el) {  
  el.style.opacity = 0;  
  
  var last = +new Date();  
  var tick = function() {  
    el.style.opacity = +el.style.opacity + (new Date() - last) / 400;  
    last = +new Date();  
  
    if (+el.style.opacity < 1) {  
      (window.requestAnimationFrame && requestAnimationFrame(tick)) || setTimeout(tick, 16);  
    }  
  };  
  
  tick();  
}  
  
fadeIn(el);
```

# jQuery

```
$(el).fadeIn();
```

# STAGE 3B

## “DYNAMIC PAGES”





# HTML

```
<h1>Hello!</h1>
```

```
<p>
```

Welcome to my homepage.

You are the 20th visitor.

```
</p>
```

```

```

# PHP

```
// with fetched  
// $count value from database
```

```
<h1>Hello!</h1>
```

```
<p>
```

```
    Welcome to my homepage.
```

```
    You are the <?=$count?> visitor.
```

```
</p>
```

```

```

# 2004 Gmail

## AJAX

单页内数据部分地、实时地更新  
**Web** 开始具有构建桌面级应用的能力



# STAGE 4

# FRONTEND AUTOMATION



# HTML 预处理

Jade, HAML, Slim, Pug, (MarkDown)

```
%section.container  
  %h1  
  %h2  
  .content
```



```
<section  
  class='container'  
    <h1></h1>  
    <h2></h2>  
    <div class='content'>  
    </div>  
</section>
```



# CSS 预处理

SASS, LESS, Stylus

```
$var: 0;

nav {
  ul {
    margin: $var;
    padding: $var;
    list-style: none;
  }

  &-item { display:
inline-block; }
}
```



```
nav ul {
  margin: 0;
  padding: 0;
  list-style: none;
}

nav-item {
  display: inline-block;
}
```



# JavaScript 预处理

TypeScript, CoffeeScript, BabelJS

```
[1, 2, 3].map(n => n ** 2);
```



```
[1, 2, 3]  
.map(function(n) {  
  return Math.pow(n, 2);  
});
```



**nodejs!**





前端社区推进标准

**The community pushes  
the standard forward**



## 移动端设备兴起 -> Minify Code

```
.foo { color: #ff0000; } /*comment*/  
.bar { color: rgba(255, 0, 0, 1); }
```



```
.bar,.foo{color:red}
```



# Gulp/Grunt

```
> gulp build
```



**src**

**source.js**  
**style.sass**  
**index.html**



**dist**

**source.min.js**  
**style.min.css**  
**index.html**



# Webpack

```
> webpack src/entry.js public/bundle.js
```



src

entry.js  
dependency1.js  
module2.js  
...

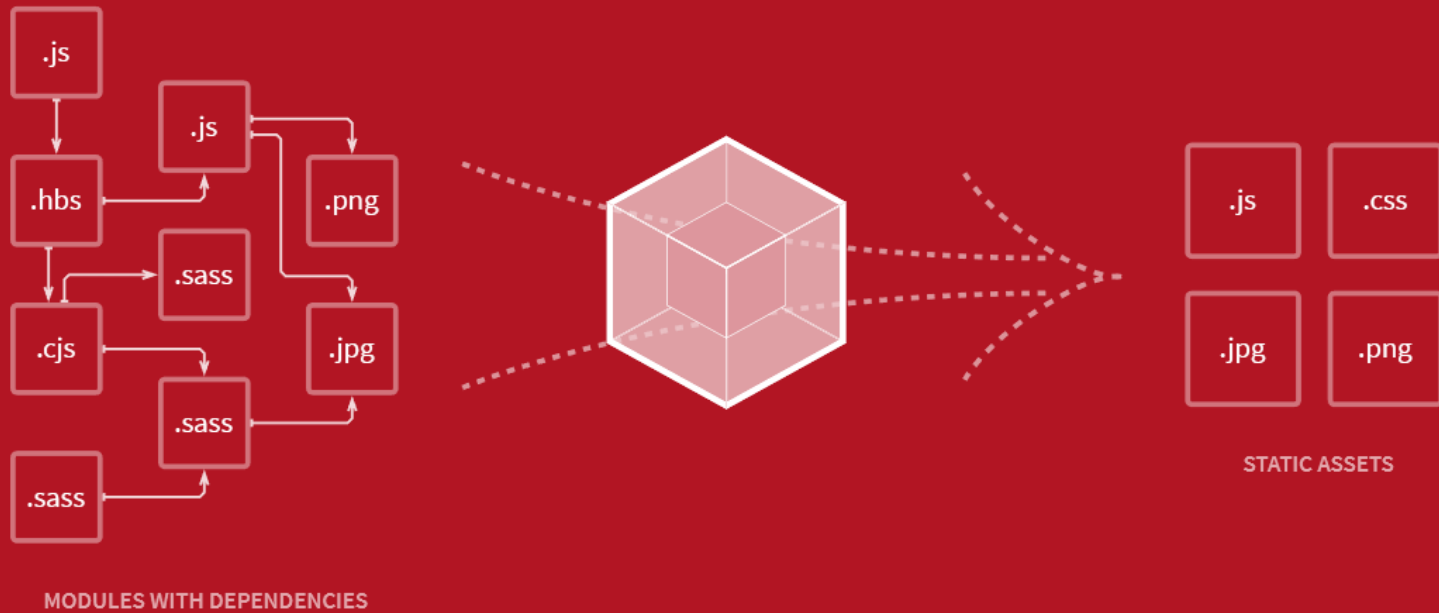


public

bundle.js



# Webpack



**STAGE 5**  
**THE GREAT SEPARATION**



**Gee, this is not the fed I knew 🖱️.**

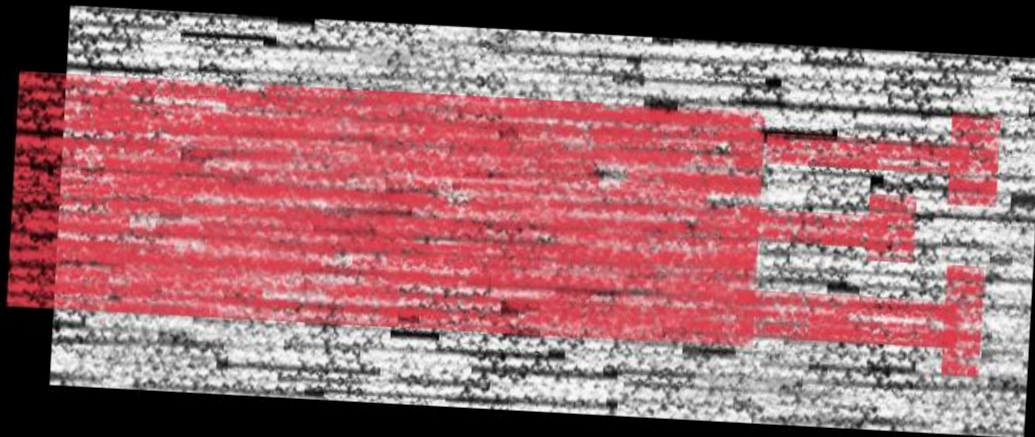
```
<h1>Hello!</h1>  
<p>Welcome to my homepage.</p>  

```

**I need some fed folks on this...**



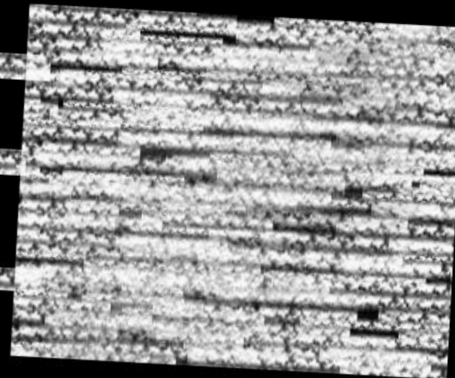
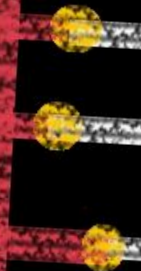
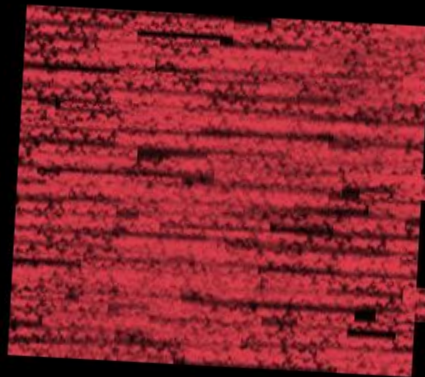
**fed  
folks**



**php  
coders**

**api**

**fed  
folks**



**php  
coders**





# 模板渲染

组件重用和视图重用 ✓

列表渲染 ✓

MVx 架构实现 ✓

渲染耗费资源 😞

数据接口多平台复用 😞

前后端耦合式开发 😞



# 前后分离

组件重用和视图重用 😞

渲染耗费资源 😞

列表渲染 😞

数据接口多平台复用 😞

MVx 架构实现 😞

前后端耦合式开发 😞



**Backbone**

**Angular**

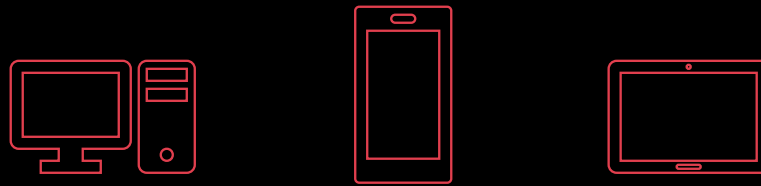
**Vue**

**React**



**STAGE 6**  
**“WRITE ONCE,  
RUN EVERYWHERE”**





**Node Webkit**  
**React + React Native**  
**Vue + Weex**



关于“最佳实践”



[www.wappalyzer.com/technologies](http://www.wappalyzer.com/technologies)

**THAT'S IT!**

