

可 Web 编程的NativeUI 设计与实现

——张袁炜

zhangyuanwei@baidu.com

About Me

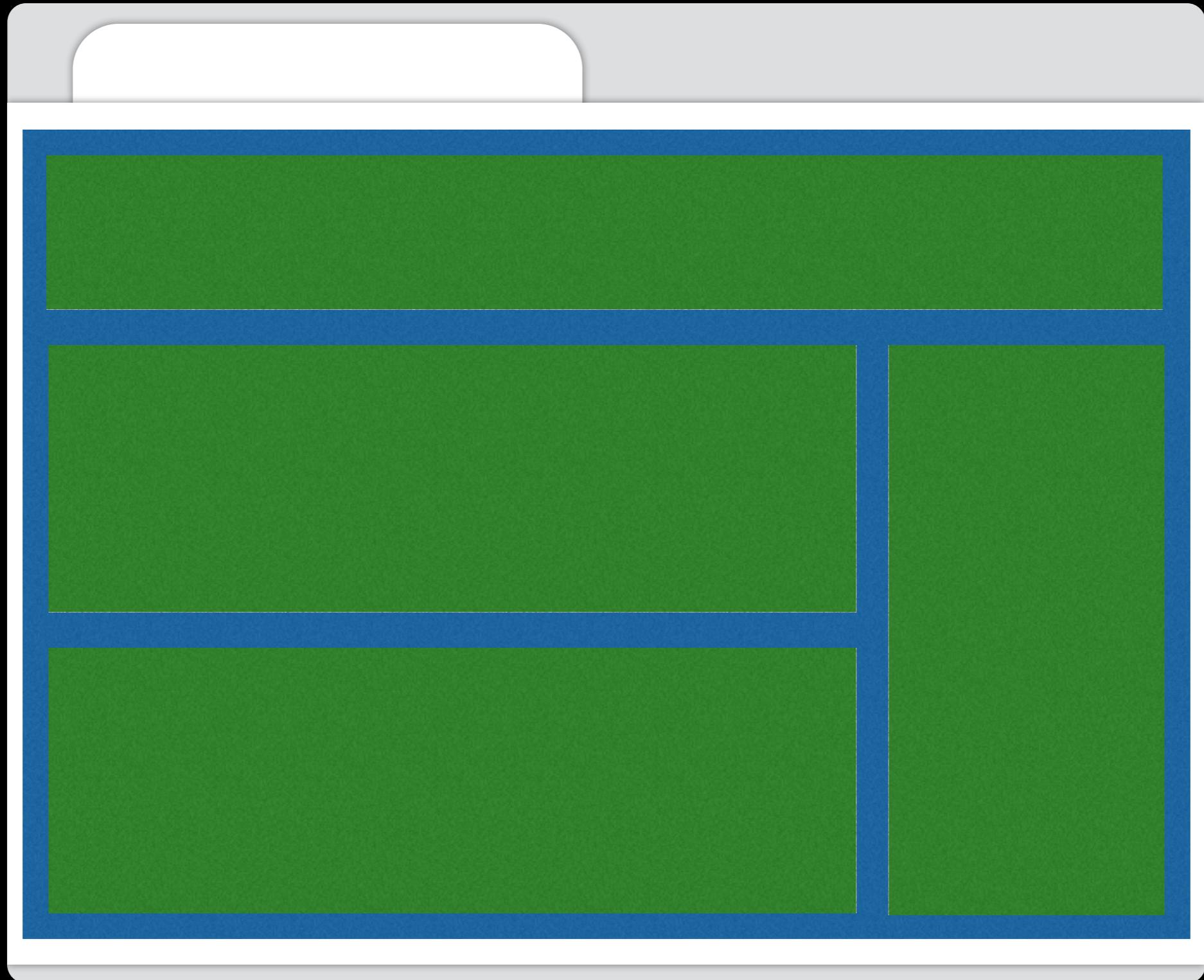
- @张袁炜
- 直达号、网址导航、百度音乐
- 前端、Node.js、Android
- Arduino、RaspberryPI
- <https://github.com/zhangyuanwei>



Outline

- 一个 Hybrid App 的进化史
- Native UI 实现原理
- 架构设计
 - UI构建
 - 布局系统
 - Web 样式支持
 - 事件机制
- 性能优化
- 前端框架设计

一个App的进化



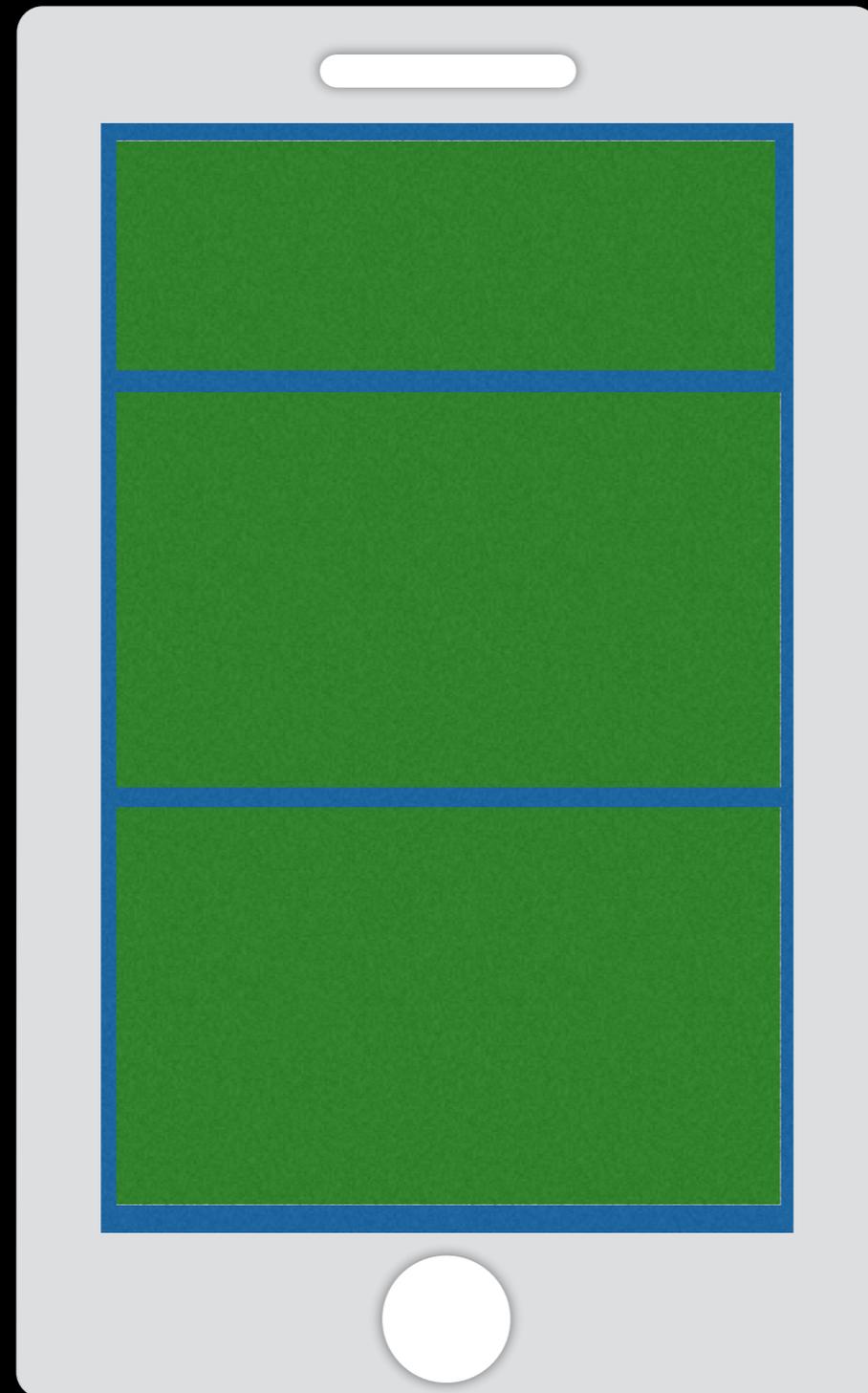
一个App的进化

地理位置

分享

摄像头

支付



一个App的进化

地理位置

分享

体验

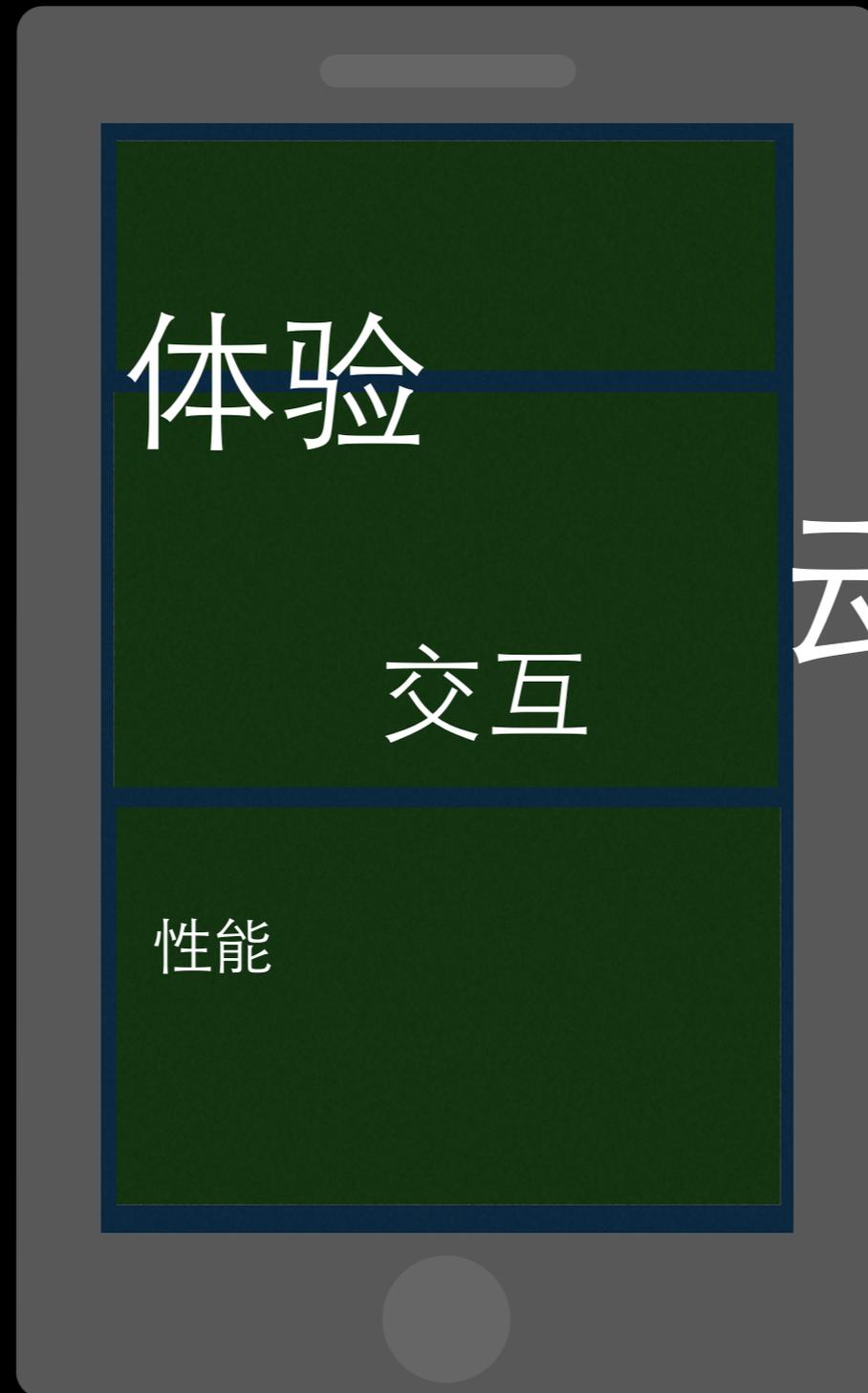
动画

交互

性能

摄像头

支付



一个App的进化



一个App的进化



一个App的进化

CSS3 硬件加速?

改 WebKit?

单页面应用?

手势跟随?



一个App的进化



WebView



WebView



WebView

Native

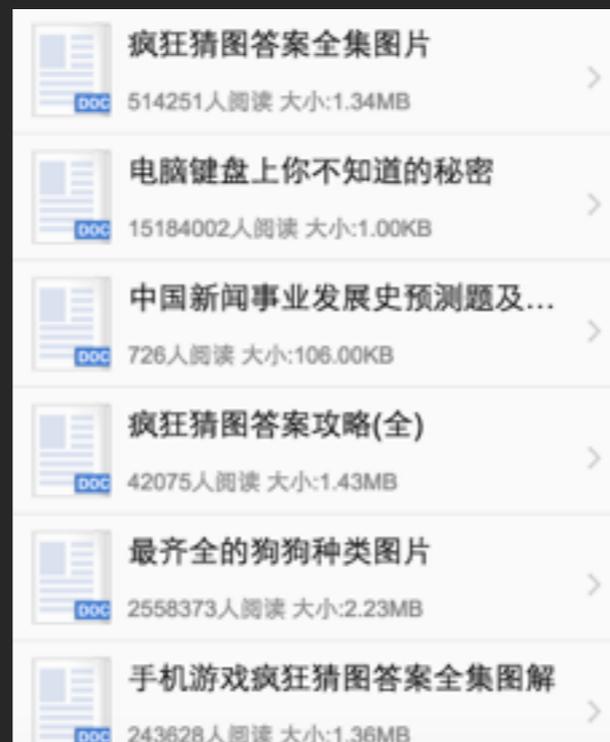
一个App的进化



WebView



WebView



WebView

background page

```
// 系统原生事件,  
// 新activity创建的时候触发webview_regist  
// activity销毁的时候触发webview_unregist  
// var _events = ['webview_register',  
var bridge = window.lc_bridge;  
var webViews = {};  
var status = {  
  ACTIVE: 1,  
  PAUSED: 2,  
  STOPPED: 3,  
  KILLED: 0  
};  
// 监听activity webView创建事件  
document.addEventListener('webview_reg  
  if (!(event.origin in webViews)) {  
    webViews[event.origin] = {  
      origin: event.origin,  
      status: status.ACTIVE,  
      data: event.data  
    };  
  }  
});
```

代码重构

内存泄露

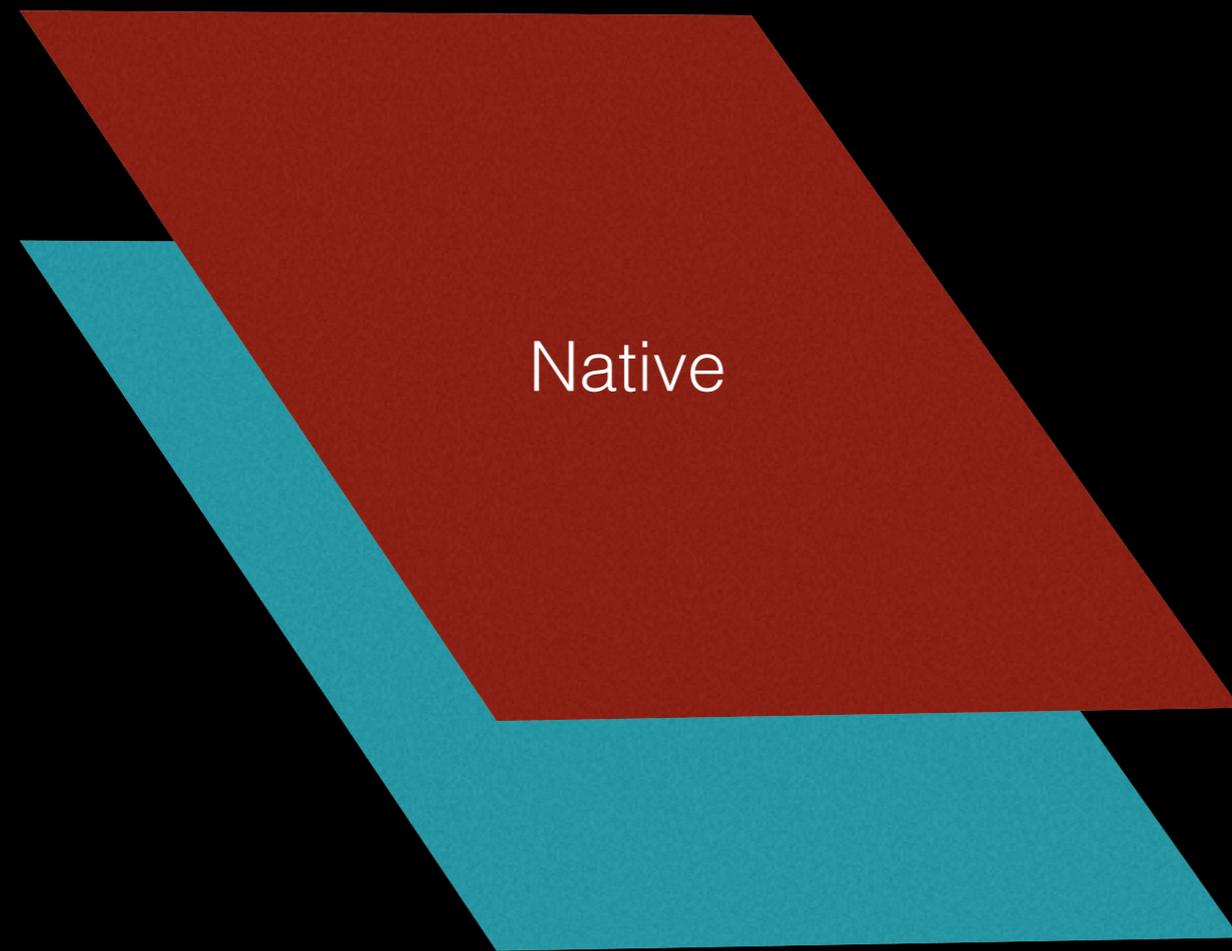
多页面通讯



理想架构?

- 性能好（媲美Native）
- 定制性强（媲美Web）
- 学习成本低（最好为零）

页面增强

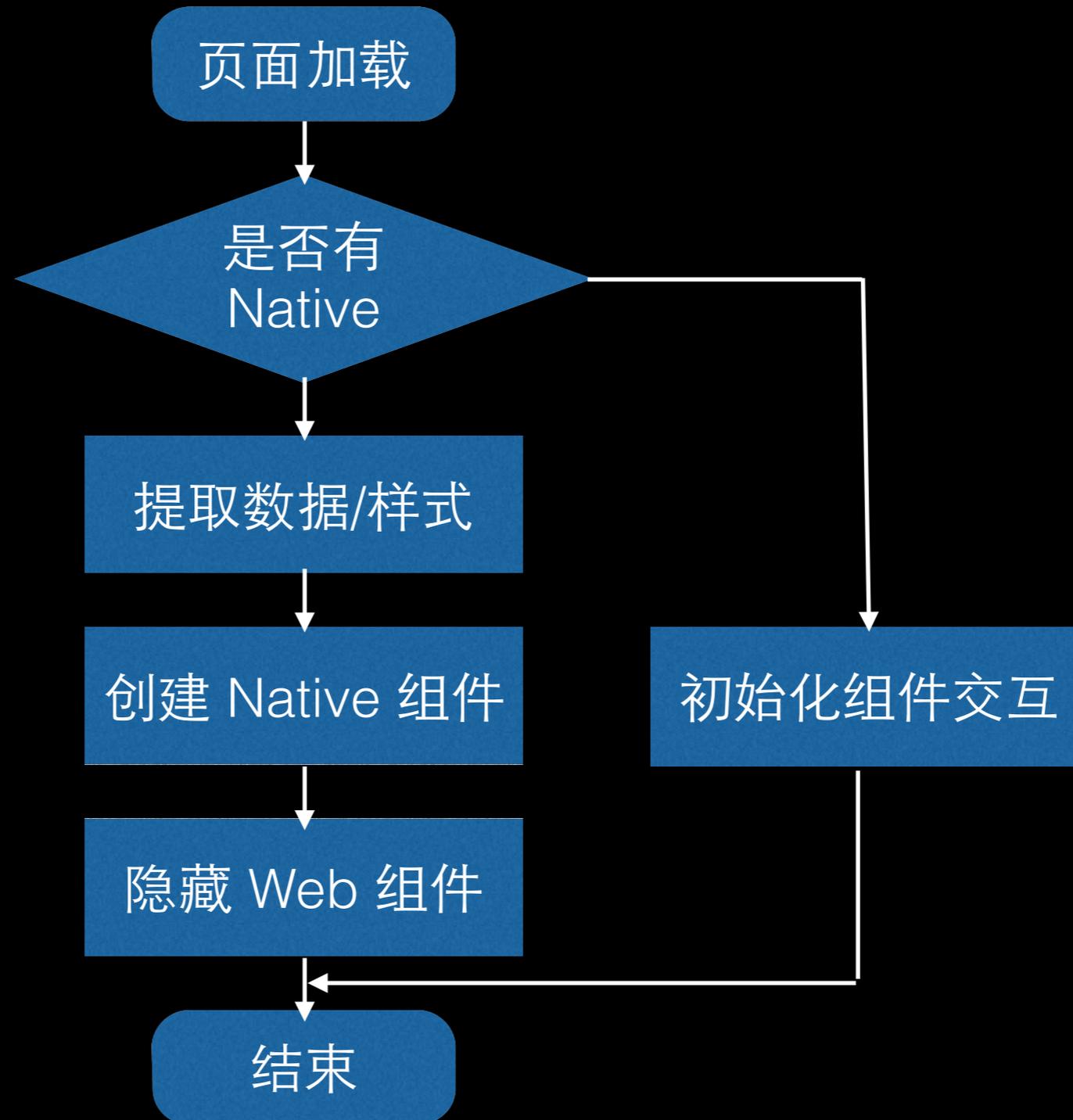


- 底层为 WebView
- 覆盖透明 Native 组件层
- Native 组件替换 Web 组件

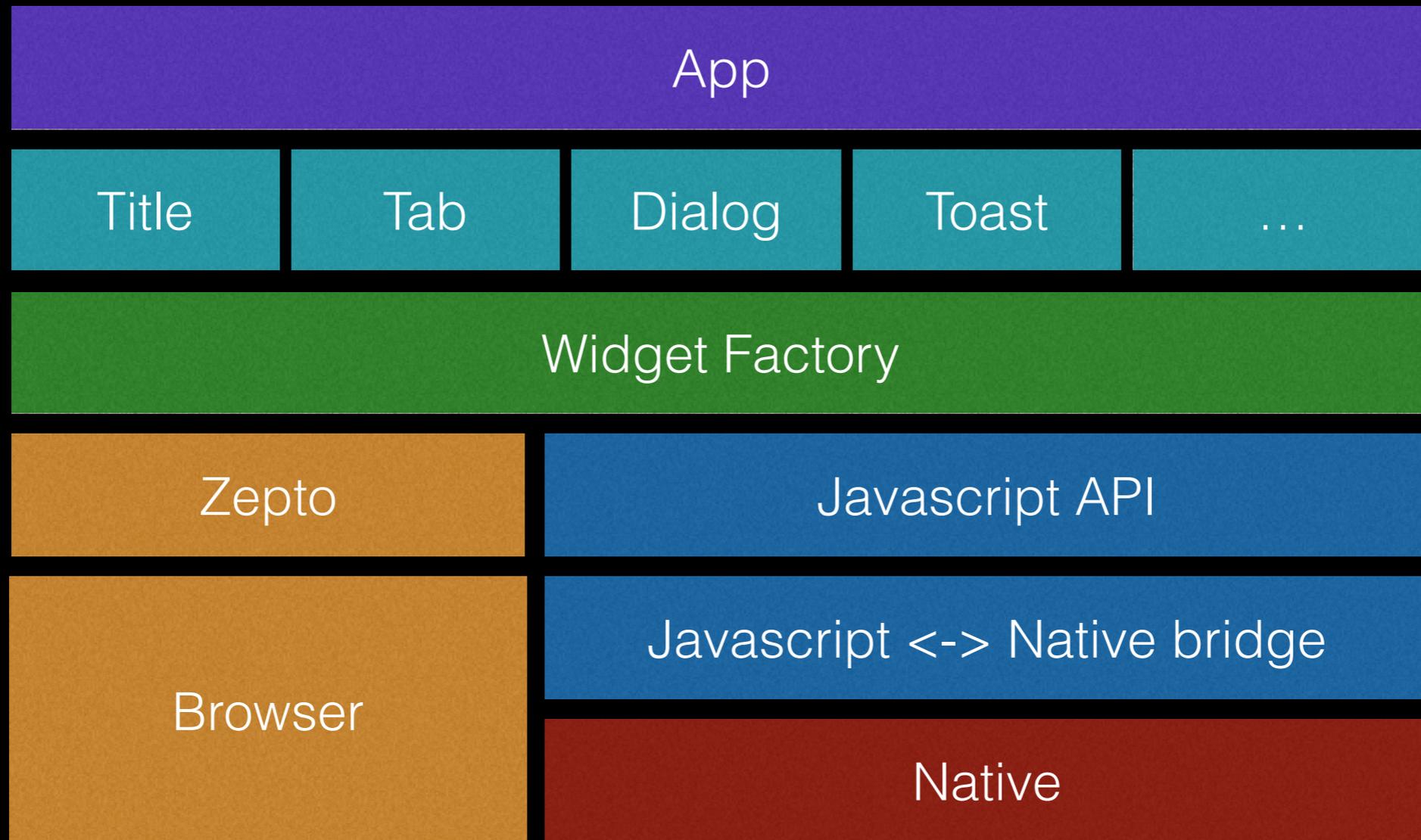
页面增强



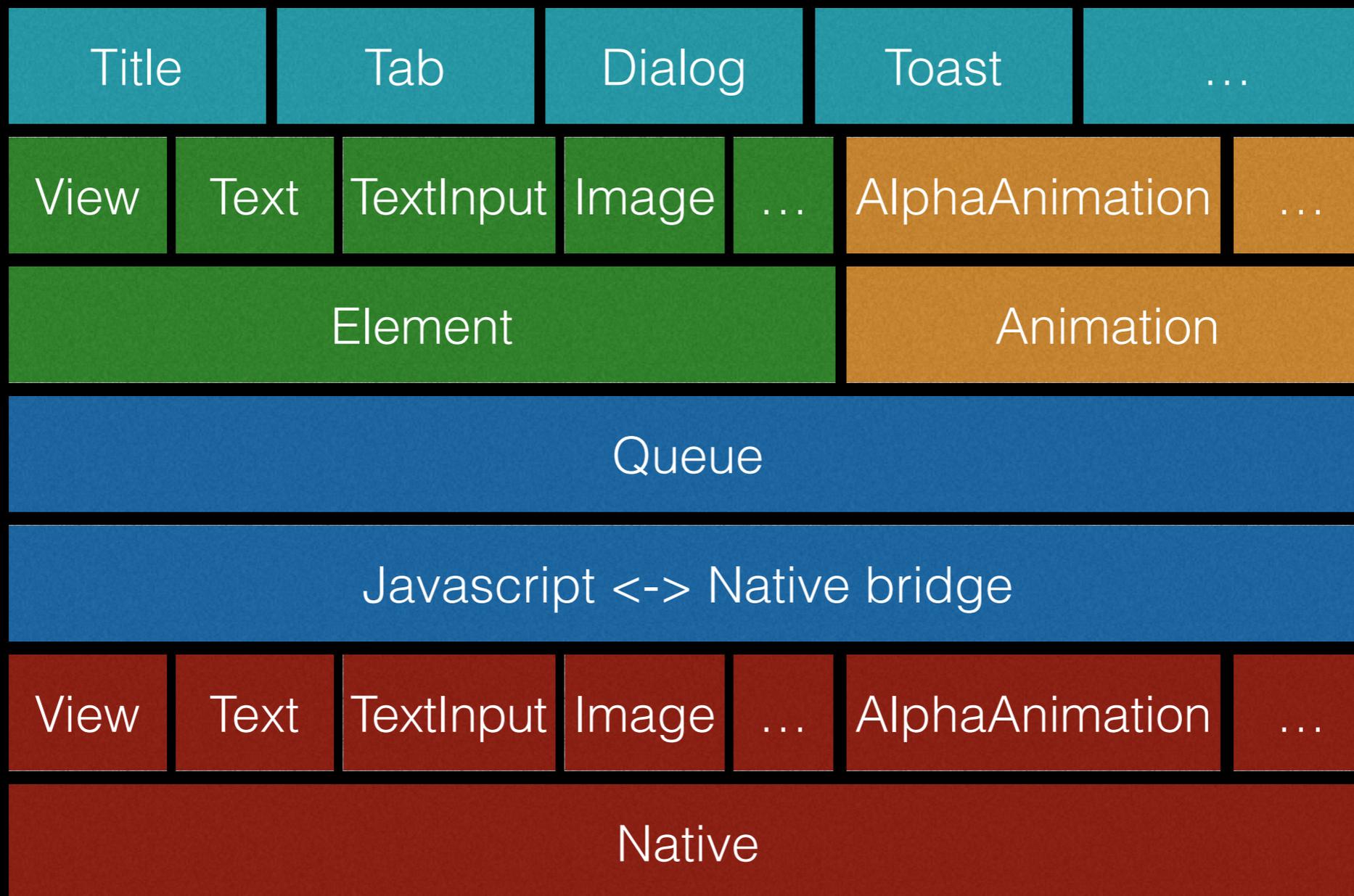
页面增强



Blend UI 系统架构



Blend UI 系统架构



UI 构建

The diagram illustrates a standard dialog box layout. It is divided into three horizontal sections. The top section is a light gray header bar containing the text '提示信息' (Information). The middle section is a large white area containing the text '提示内容' (Content). The bottom section is a light gray footer bar containing two buttons: '确定' (OK) on the left and '取消' (Cancel) on the right. The entire dialog box is enclosed in a double-line border.

提示信息

提示内容

确定 取消

UI 构建

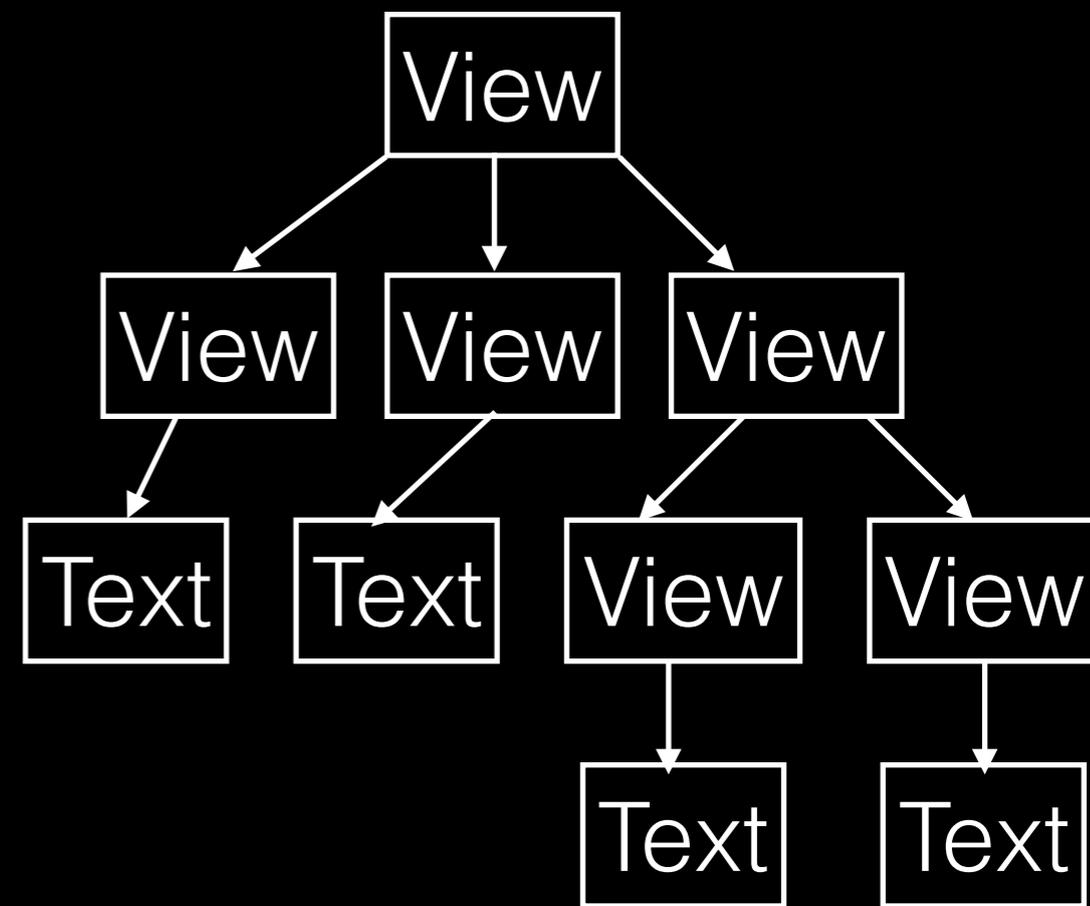
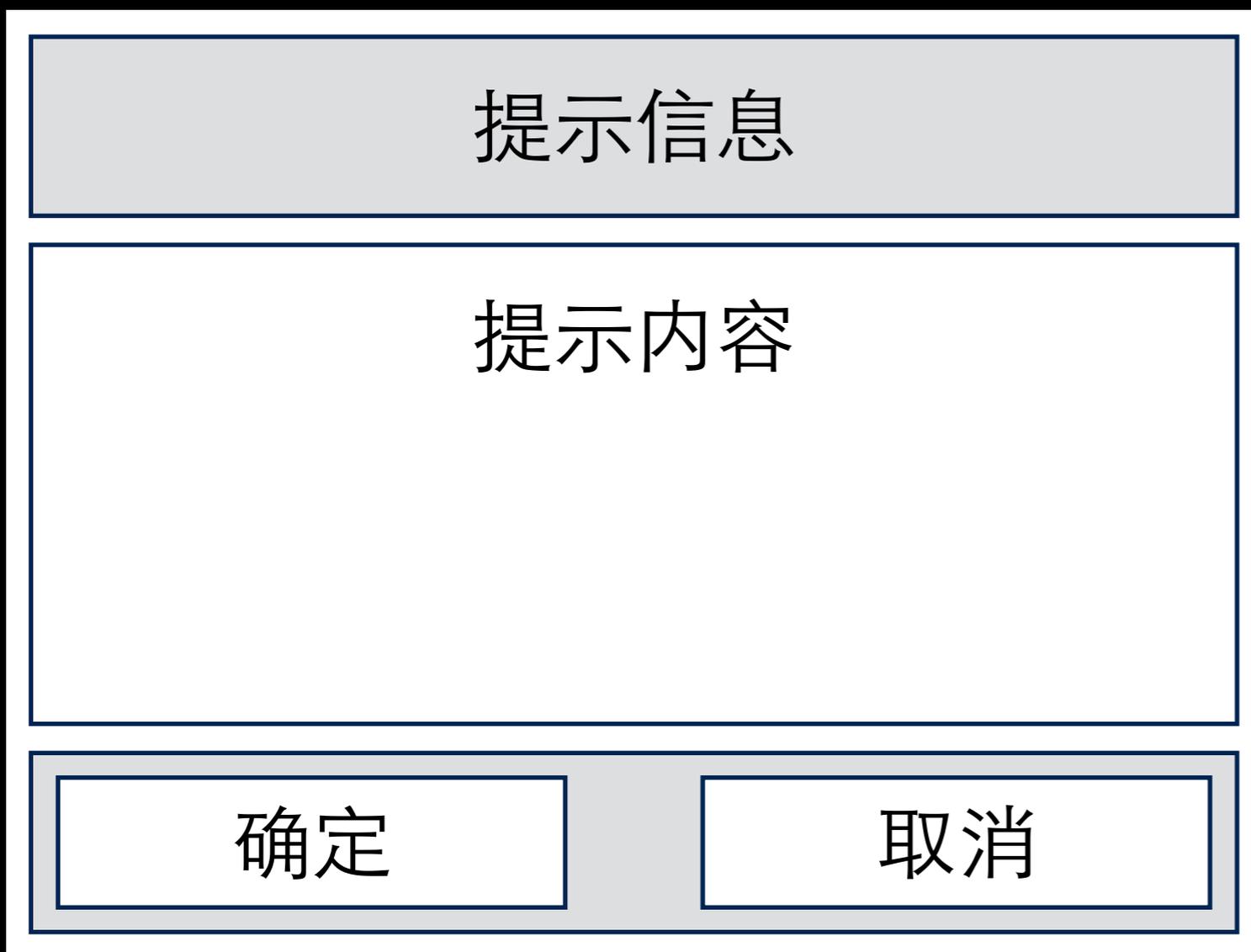
提示信息

提示内容

确定

取消

UI 构建



UI 构建

`createView("tag":string, "type":string, "config":object)`

根据给定的 config 配置，创建一个类型为 type 的基础 GUI 组件，并指定 tag。

`addView("parentTag":string, "childTag":string, "index":number)`

添加 childTag 所指定的 View 到 parentTag 所指定 View 的 index 位置。

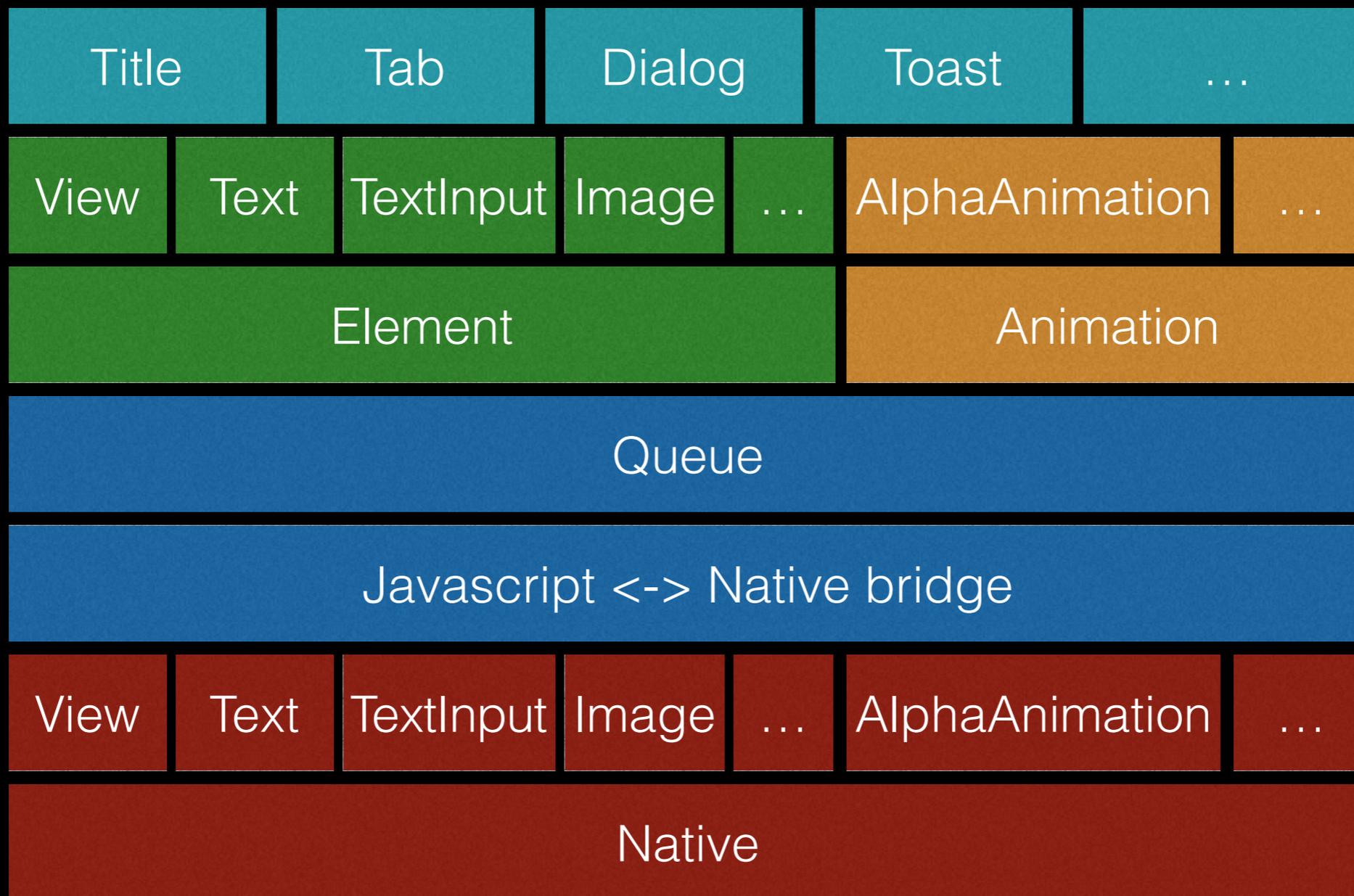
`removeView("parentTag":string, "index": number)`

从 parentTag 所指定 View 删除 index 位置的子 View。

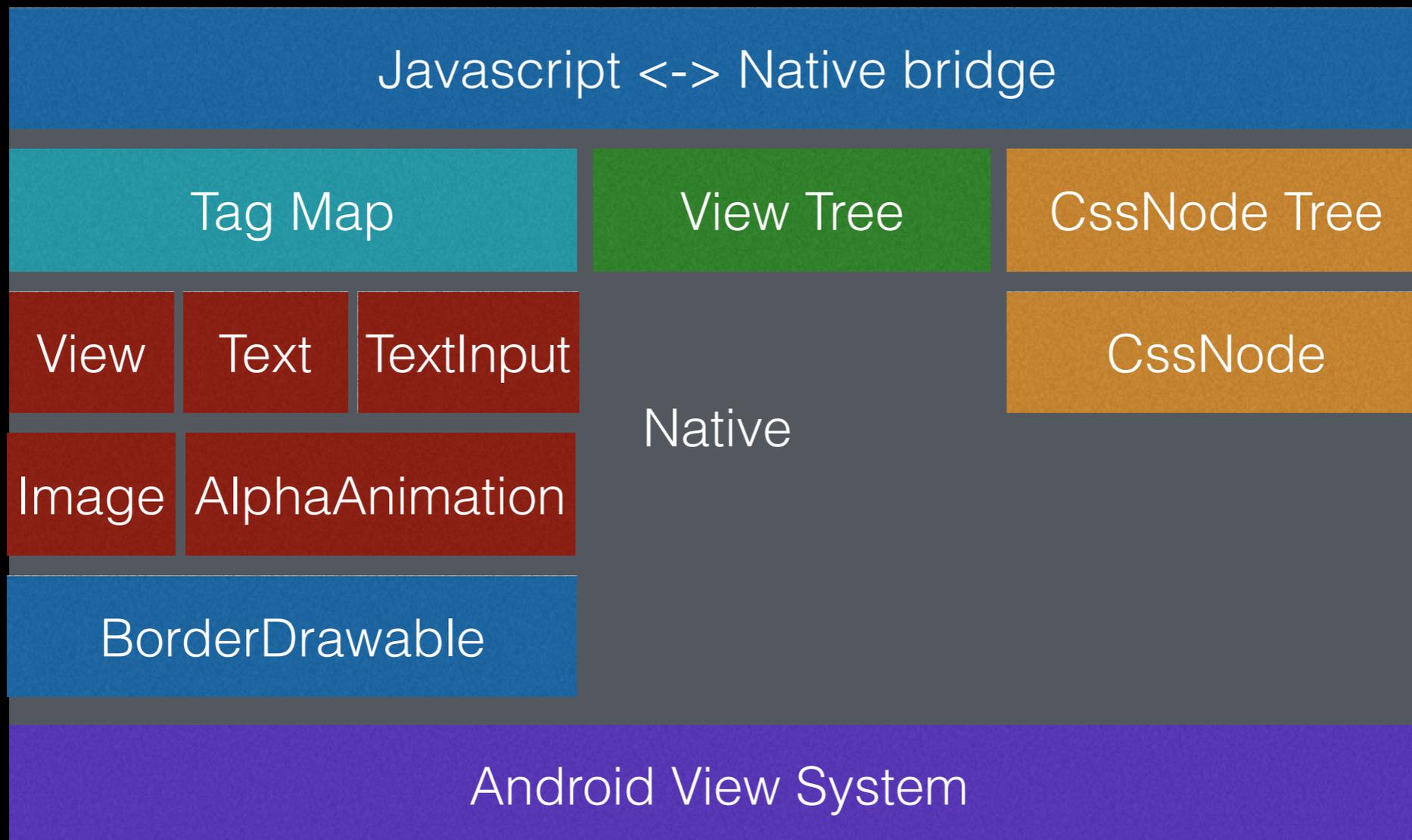
`updateView("tag":string, "type":string, "config":object)`

更新 tag 所指定的 View。

Blend UI 系统架构



Blend UI 系统架构



CSS Layout

CssNode

```
public void addChildAt(CSSNode child, int i)
public void removeChildAt(int i)

public void setFlexDirection(CSSFlexDirection direction)
public void setJustifyContent(CSSJustify justifyContent)
public void setAlignItems(CSSAlign alignItems)
public void setAlignSelf(CSSAlign alignSelf)
public void setPositionType(CSSPositionType positionType)
public void setWrap(CSSWrap flexWrap)
public void setFlex(float flex)
public void setMargin(int spacingType, float margin)
public void setPadding(int spacingType, float padding)
public void setBorder(int spacingType, float border)
public void setPositionTop(float positionTop)
public void setPositionBottom(float positionBottom)
public void setPositionLeft(float positionLeft)
public void setPositionRight(float positionRight)
public void setStyleWidth(float width)
public void setStyleHeight(float height)
```

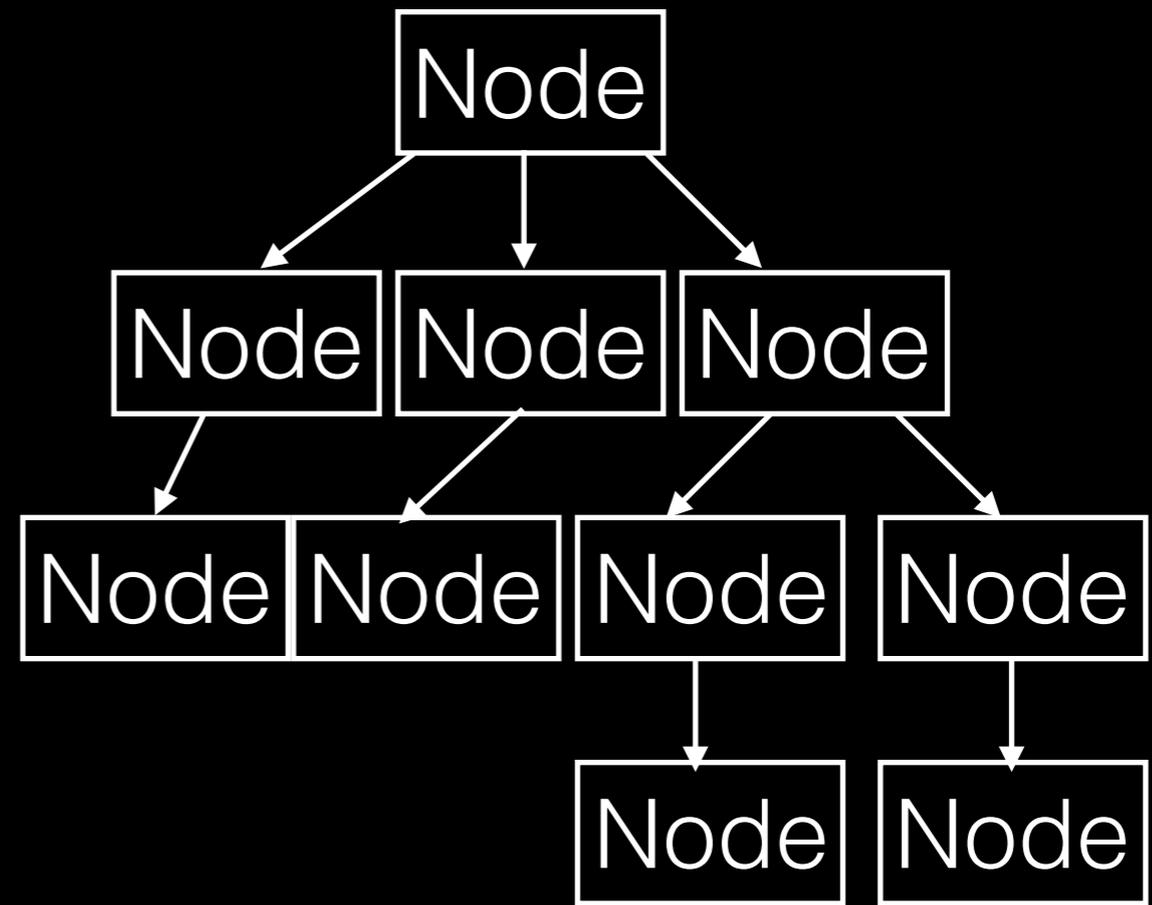
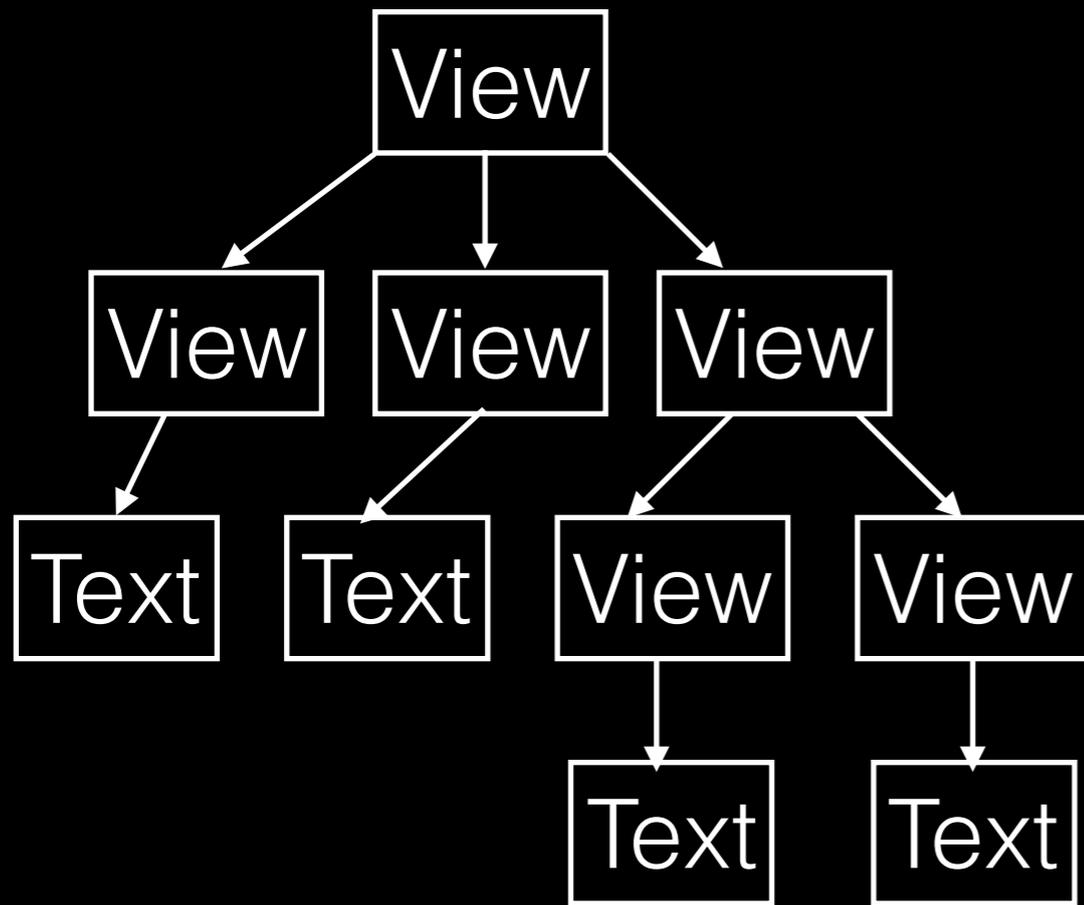
CSS Layout

CssNode

```
public void calculateLayout()  
public boolean hasNewLayout()  
public void markLayoutSeen()  
  
public float getLayoutX()  
public float getLayoutY()  
public float getLayoutWidth()  
public float getLayoutHeight()
```

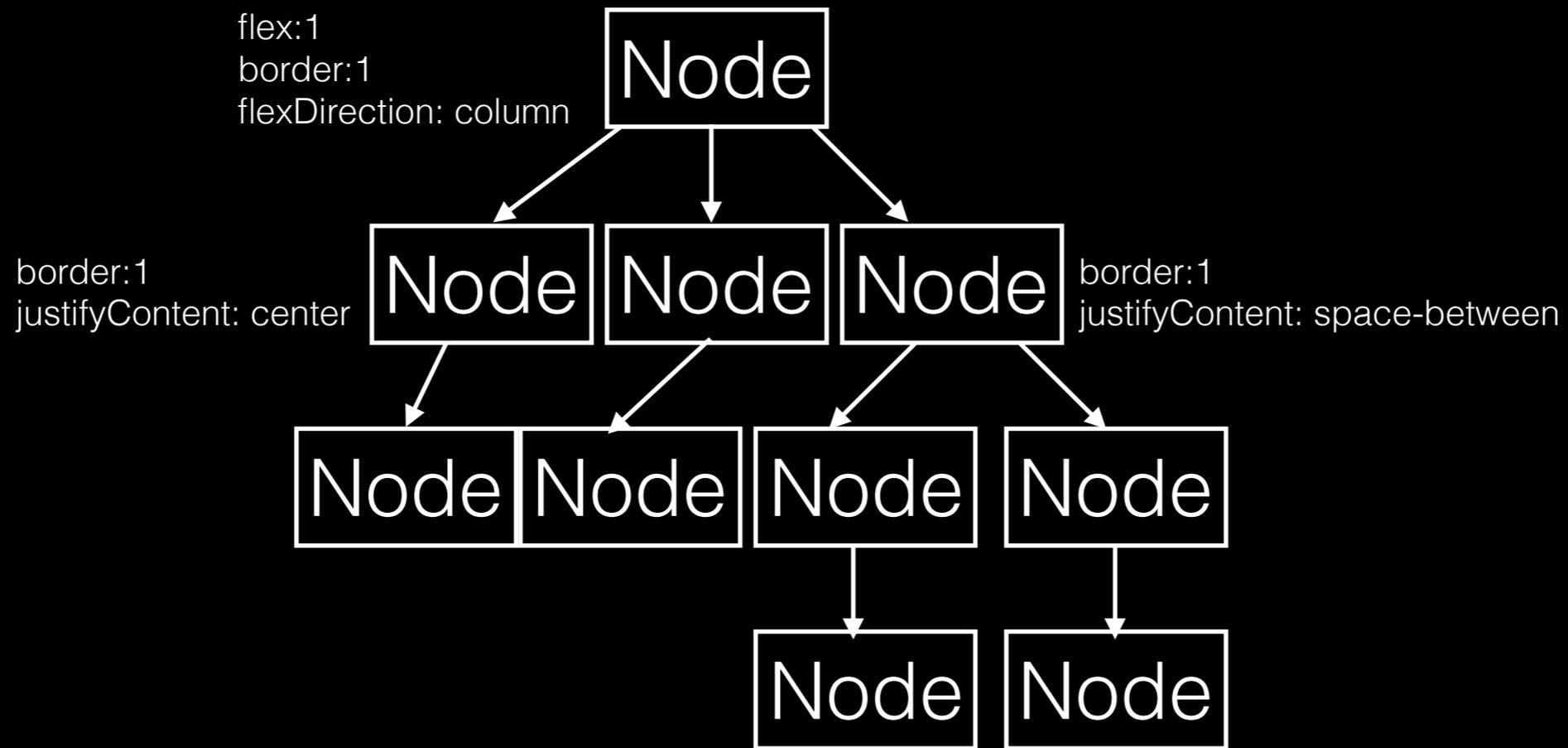
CSS Layout

CssNode



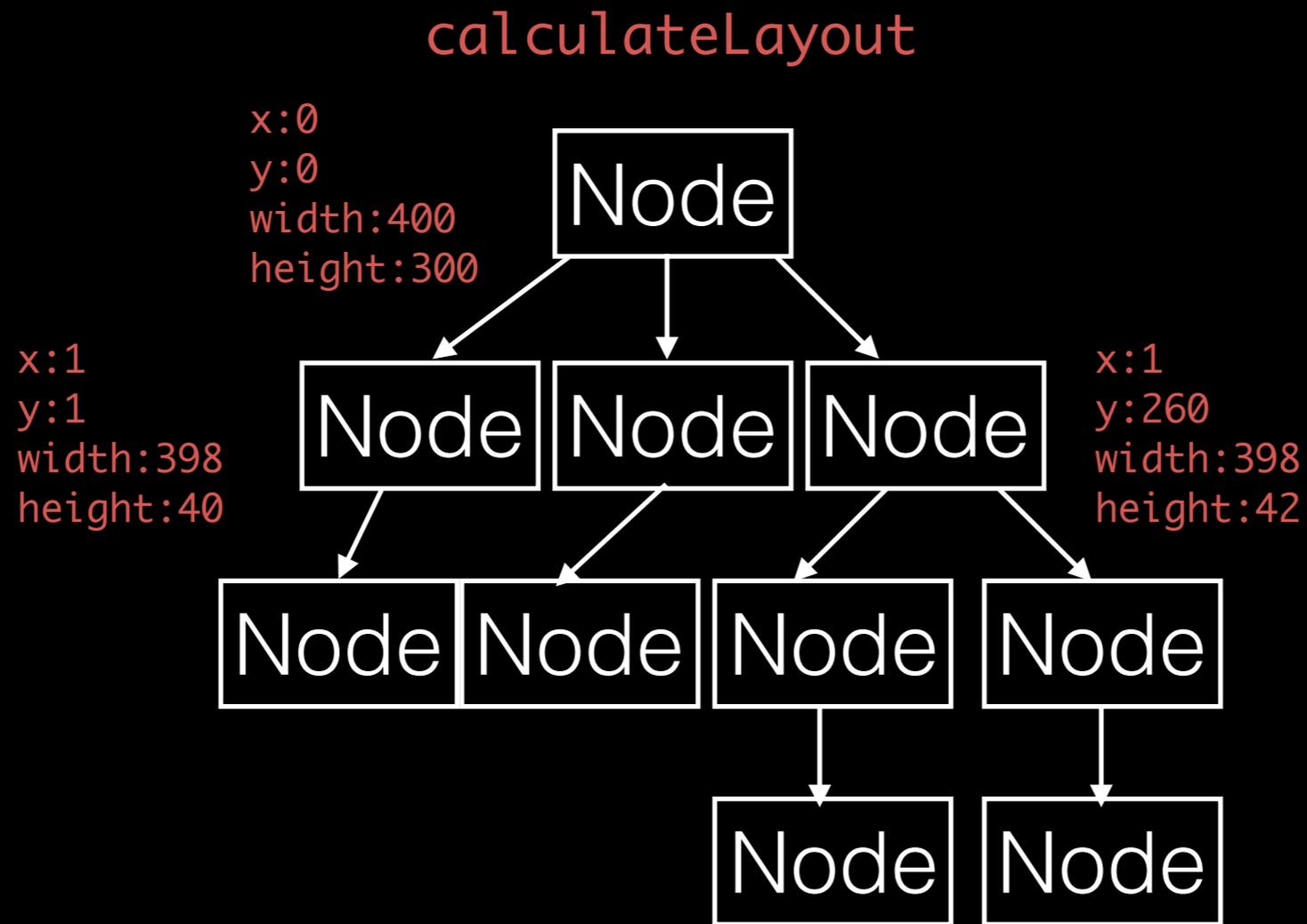
CSS Layout

CssNode

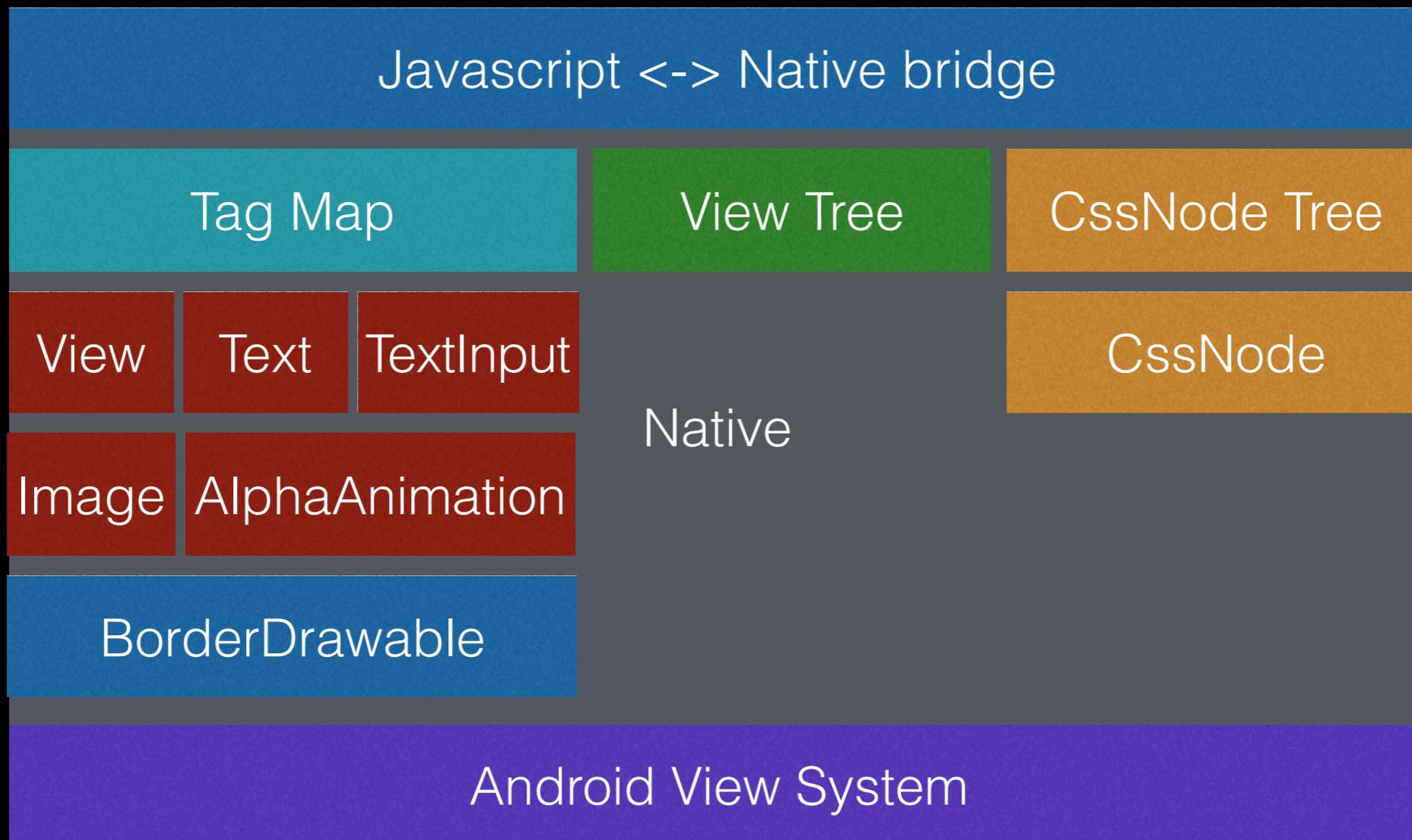


CSS Layout

CssNode



Blend UI 系统架构



Border Drawable

The image shows a mobile application interface with a white background and rounded corners. At the top, there is a promotional banner for '猫爪U盘' (Cat Paw USB Drives). The banner includes the text '储存旅行中的点点滴滴' (Store up every drop of travel) and a red button that says '0元免费抽' (Free draw for 0 yuan). To the right of the text are three 3D-rendered cat paw USB drives in white, black, and blue colors.

Below the banner is a navigation bar with four circular icons: a red icon with a building (景区门票), a yellow icon with a map (热门线路), a blue icon with a shopping cart (桂林特产), and a green icon with a person (我的订单).

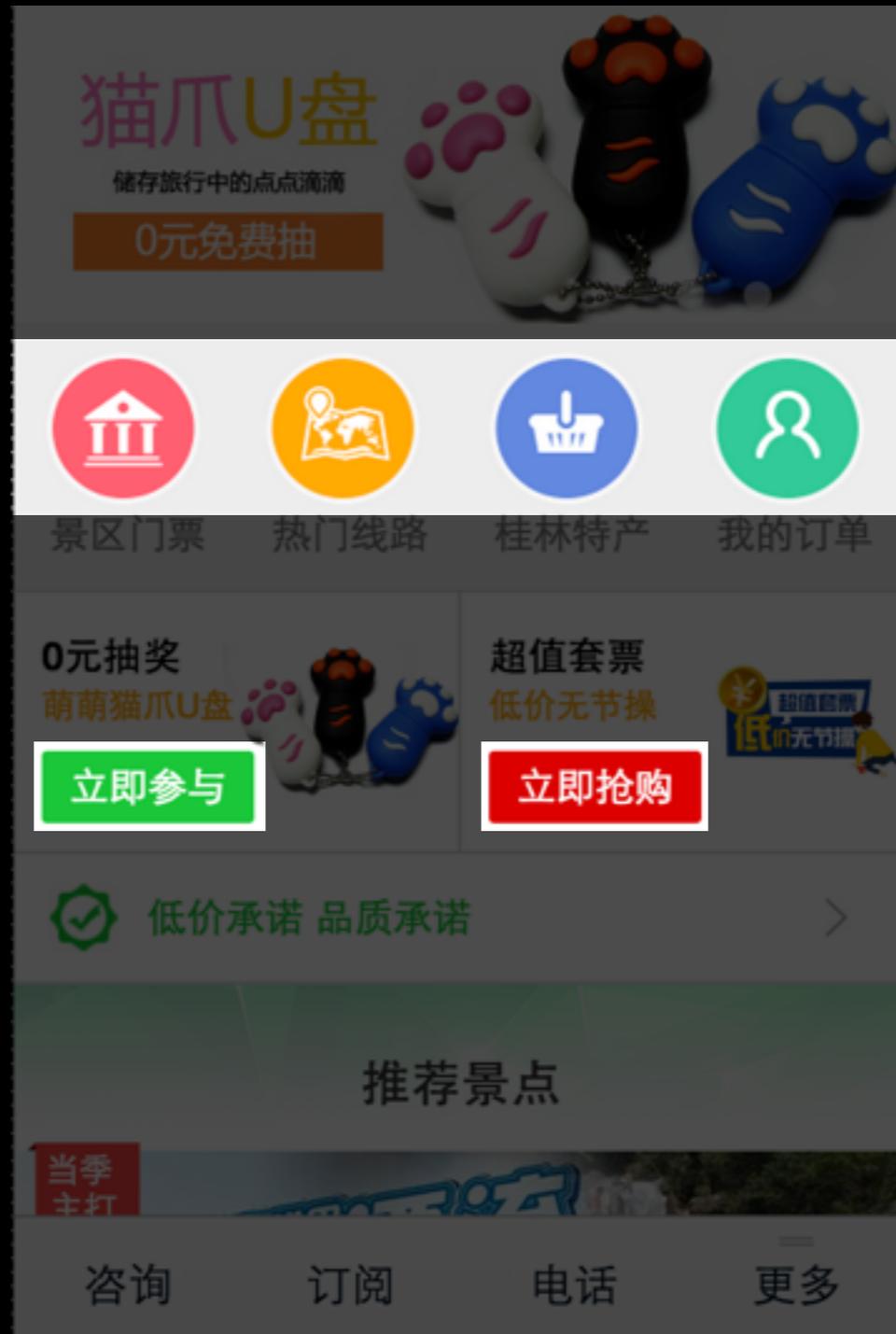
The main content area is divided into two columns. The left column features a '0元抽奖' (0 Yuan Lottery) section with the text '萌萌猫爪U盘' (Cute Cat Paw USB Drives) and a green button labeled '立即参与' (Participate Now). The right column features a '超值套票' (Super Value Package) section with the text '低价无节操' (Low Price, No Restraints) and a red button labeled '立即抢购' (Grab It Now). To the right of the '超值套票' text is a small graphic with the text '超值套票 低价无节操'.

Below these two columns is a green bar with a checkmark icon and the text '低价承诺 品质承诺' (Low Price Promise, Quality Promise), followed by a right-pointing arrow.

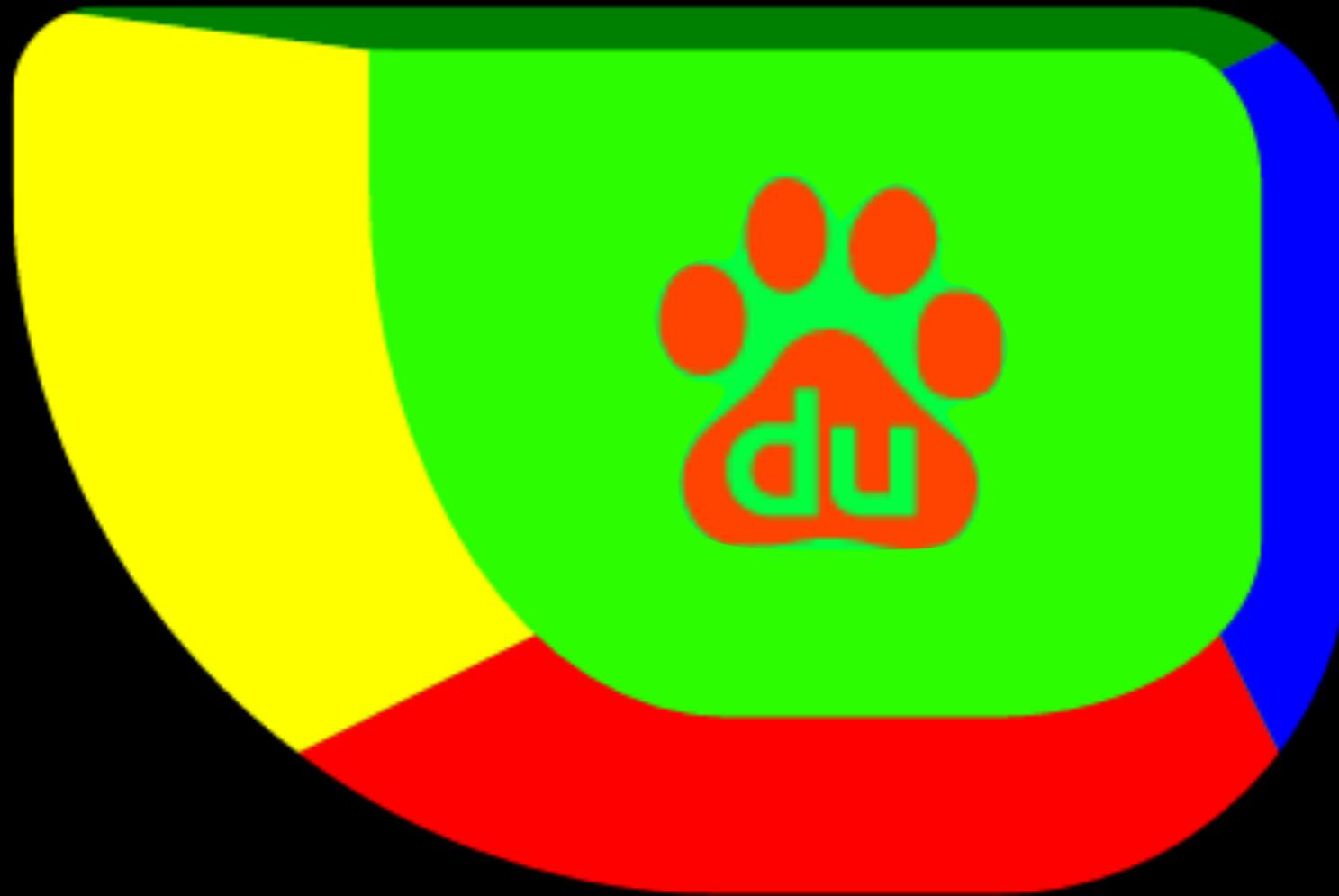
The bottom section of the app has a light green background with the text '推荐景点' (Recommended Scenic Spots). Below this is a banner for '当季主打' (Seasonal Highlight) with a blue and white graphic.

At the very bottom is a white navigation bar with four buttons: '咨询' (Consultation), '订阅' (Subscription), '电话' (Phone), and '更多' (More).

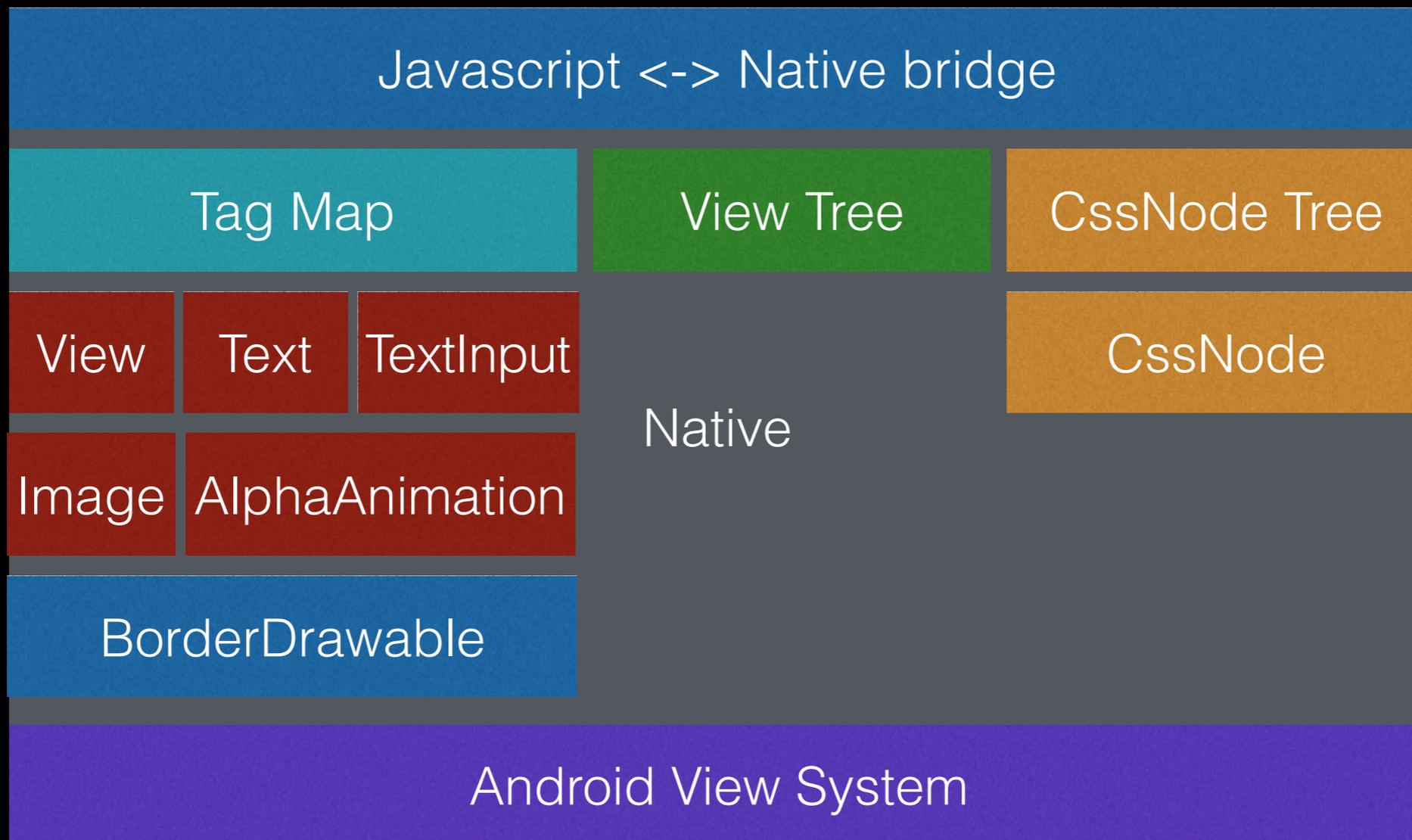
Border Drawable



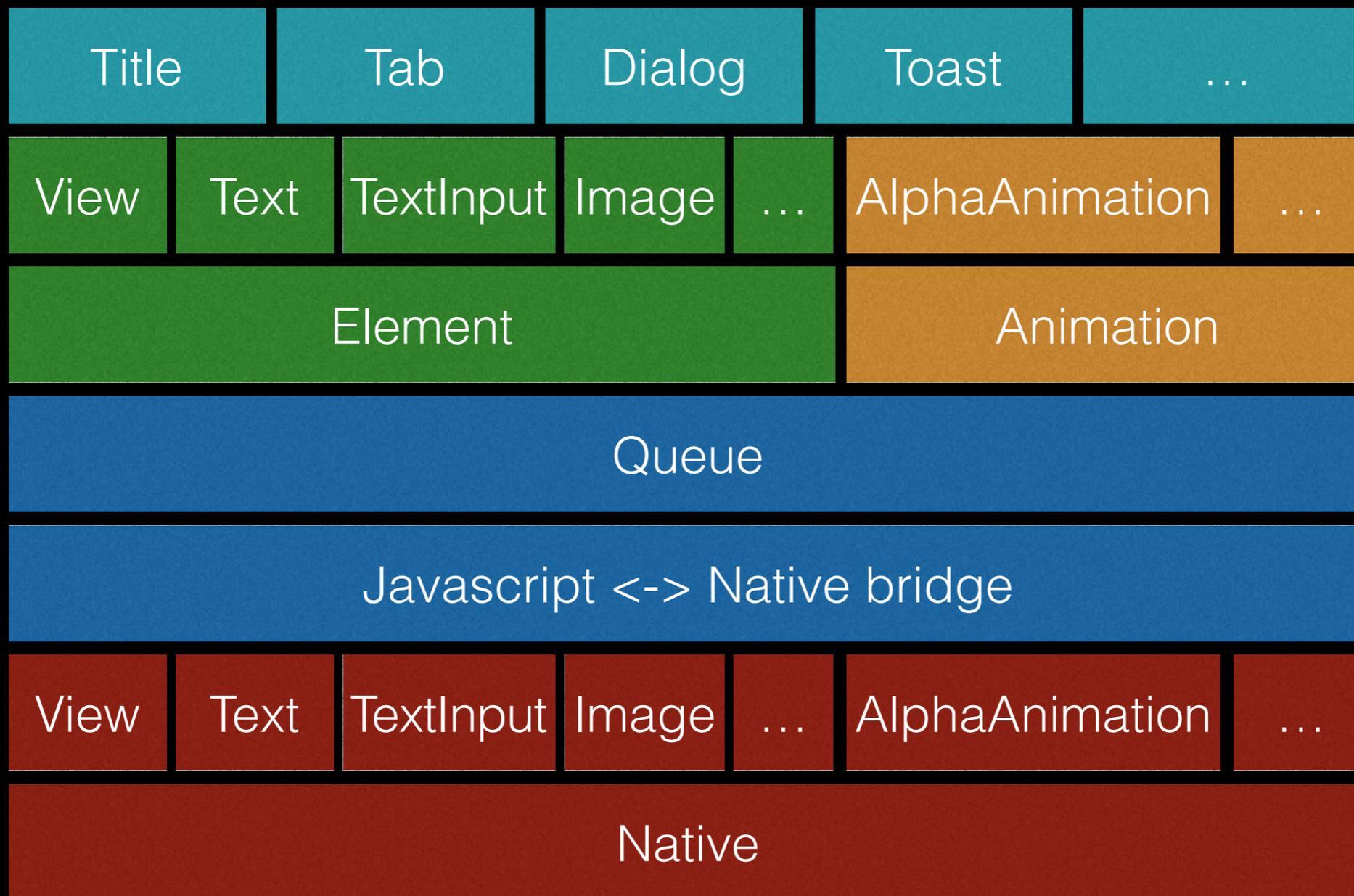
Border Drawable



Blend UI 系统架构



Blend UI 系统架构



事件机制

Javascript <-> Native bridge

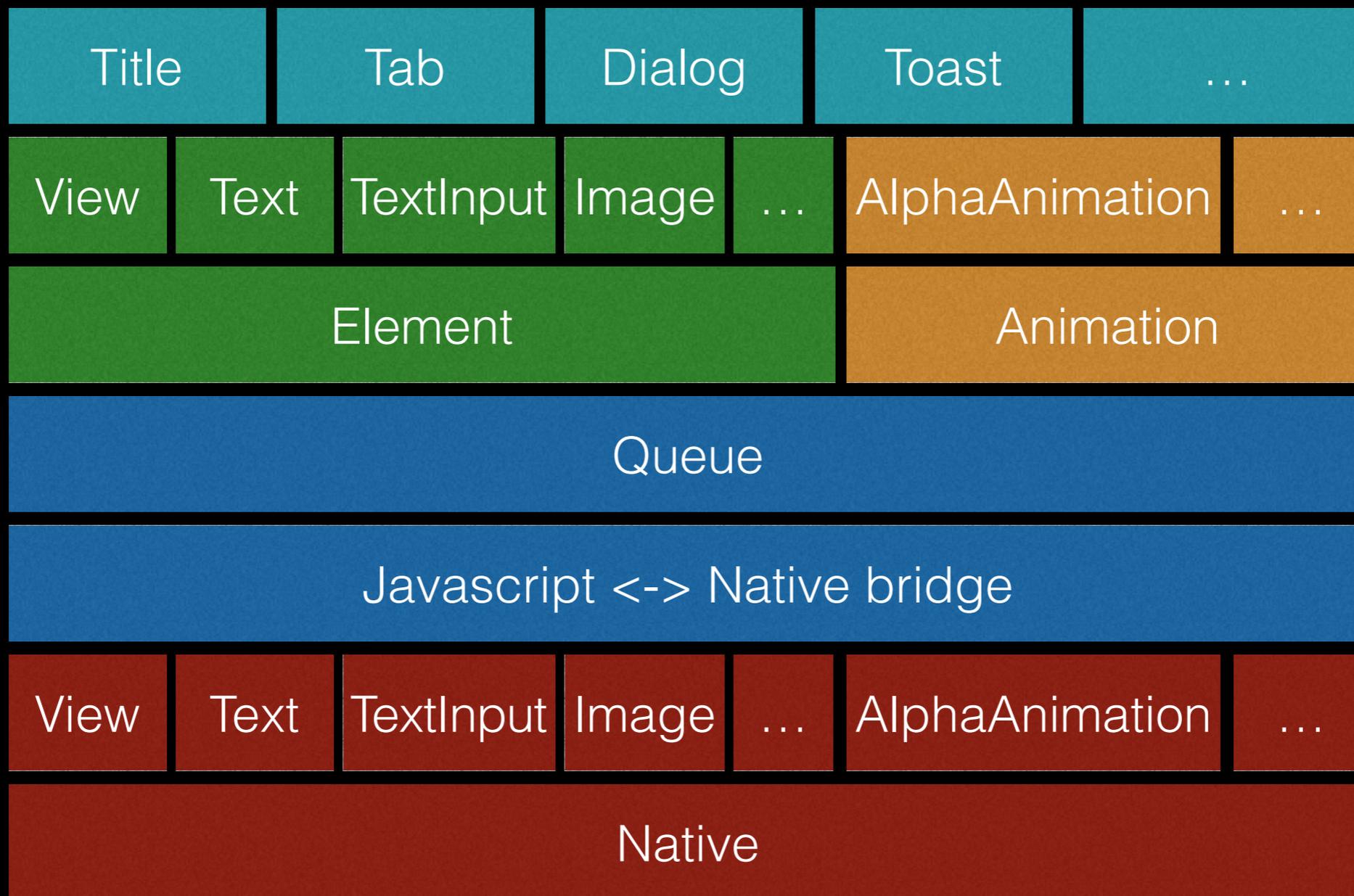
```
webView.loadURL("javascript:XXXX");  
  
(function(){  
    var event = document.createEvent("Event");  
  
    // 事件固定名称为 "boost" ,方便 Js 统一监听  
    event.initEvent("boost", false, false);  
  
    // 事件数据  
    event.data = someData;  
  
    // 事件源的 Tag  
    event.origin = "originTag";  
  
    // 具体的事件类型  
    event.boostEventType = "touchend";  
    document.dispatchEvent(event);  
})();
```

事件机制

Element

```
document.addEventListener("boost", function (e) {  
  
    var target = getElementByTag(e.origin);  
    if (!target) {  
        return;  
    }  
    assert(target instanceof Element, target);  
  
    var event = {  
        type: e.boostEventType,  
        target: target,  
        data: e.data,  
    };  
  
    target.dispatchEvent(event);  
}, false);
```

Blend UI 系统架构



性能优化

Queue

```
var view = boost.createElement("VIEW");

view.style.position = "absolute";
view.style.top = 100;
view.style.left = 100;
view.style.width = 200;
view.style.height = 200;
view.style.backgroundColor = "#ff6600";

var text = boost.createElement("TEXT");

text.style.color = "#000000";
text.style.fontSize = 20;
text.value = "hello world";

view.appendChild(text);
boost.rootElement.appendChild(view);
```

性能优化

Queue

```
var view = boost.createElement("VIEW"); // createView

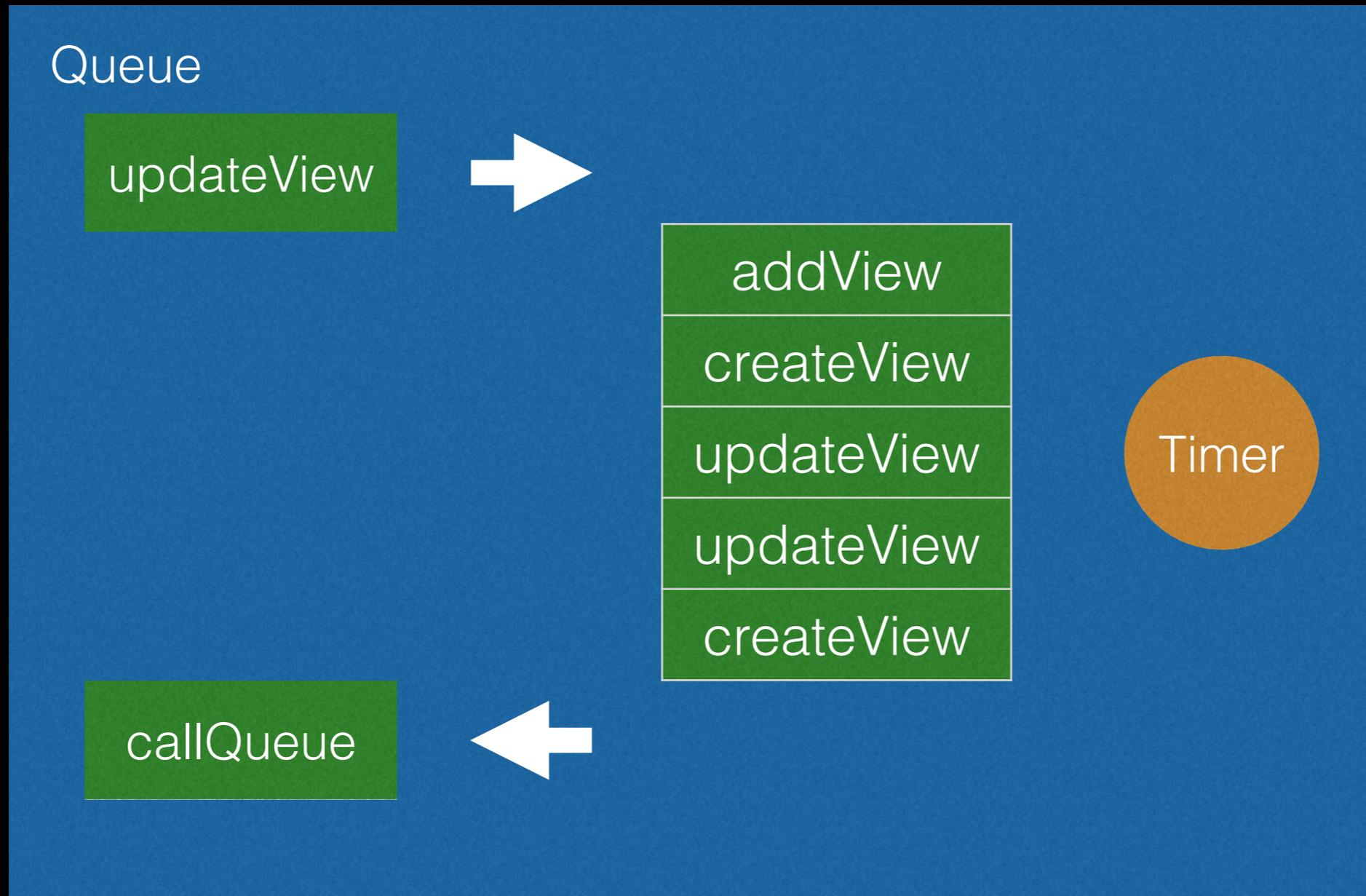
view.style.position = "absolute"; // updateView
view.style.top = 100; // updateView
view.style.left = 100; // updateView
view.style.width = 200; // updateView
view.style.height = 200; // updateView
view.style.backgroundColor = "#ff6600"; // updateView

var text = boost.createElement("TEXT"); // createView

text.style.color = "#000000"; // updateView
text.style.fontSize = 20; // updateView
text.value = "hello world"; // updateView

view.appendChild(text); // addView
boost.rootElement.appendChild(view); // addView
```

性能优化



性能优化

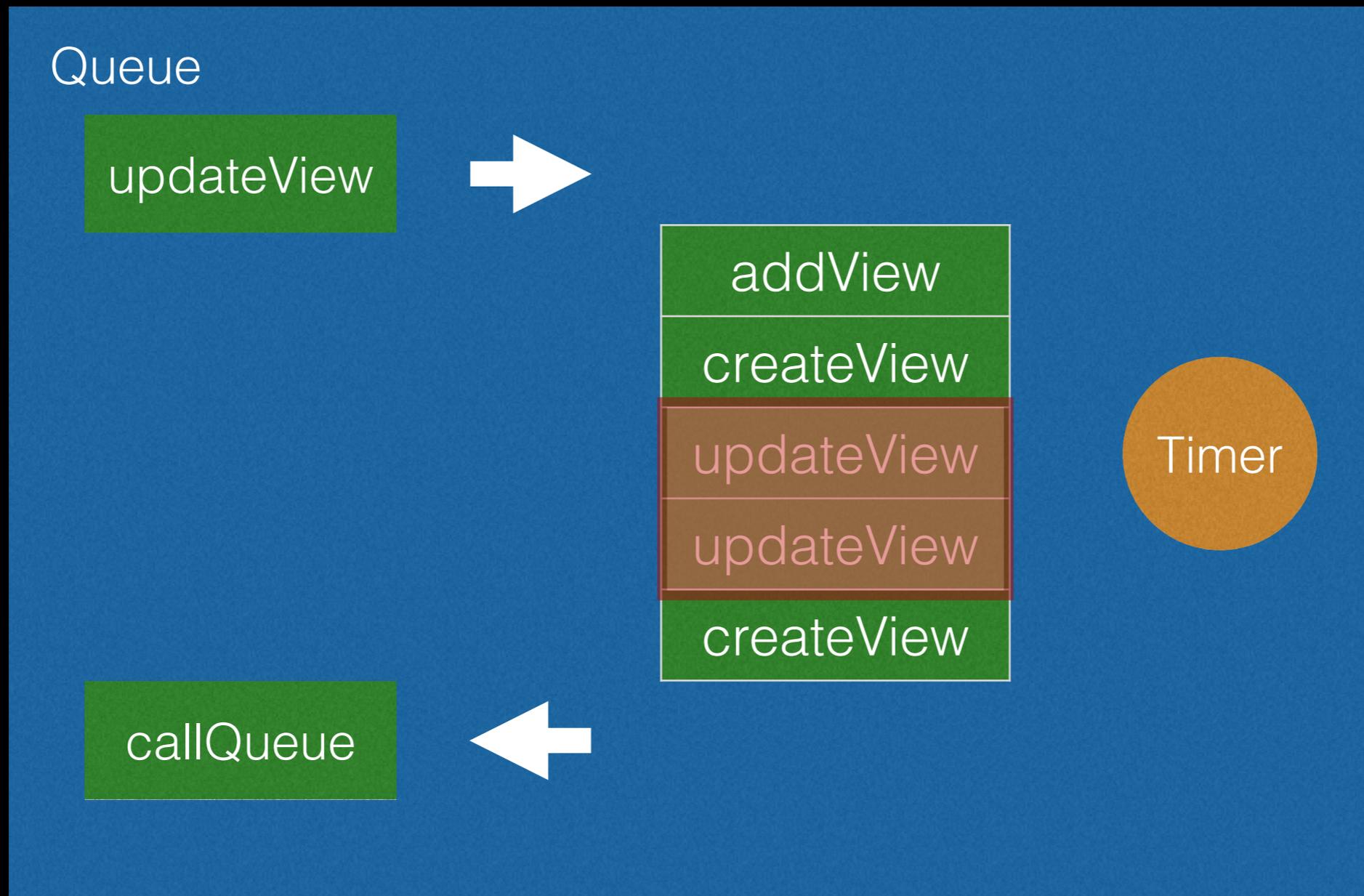
Queue

```
function queue(callback) {
  var list = [];
  var running = false;

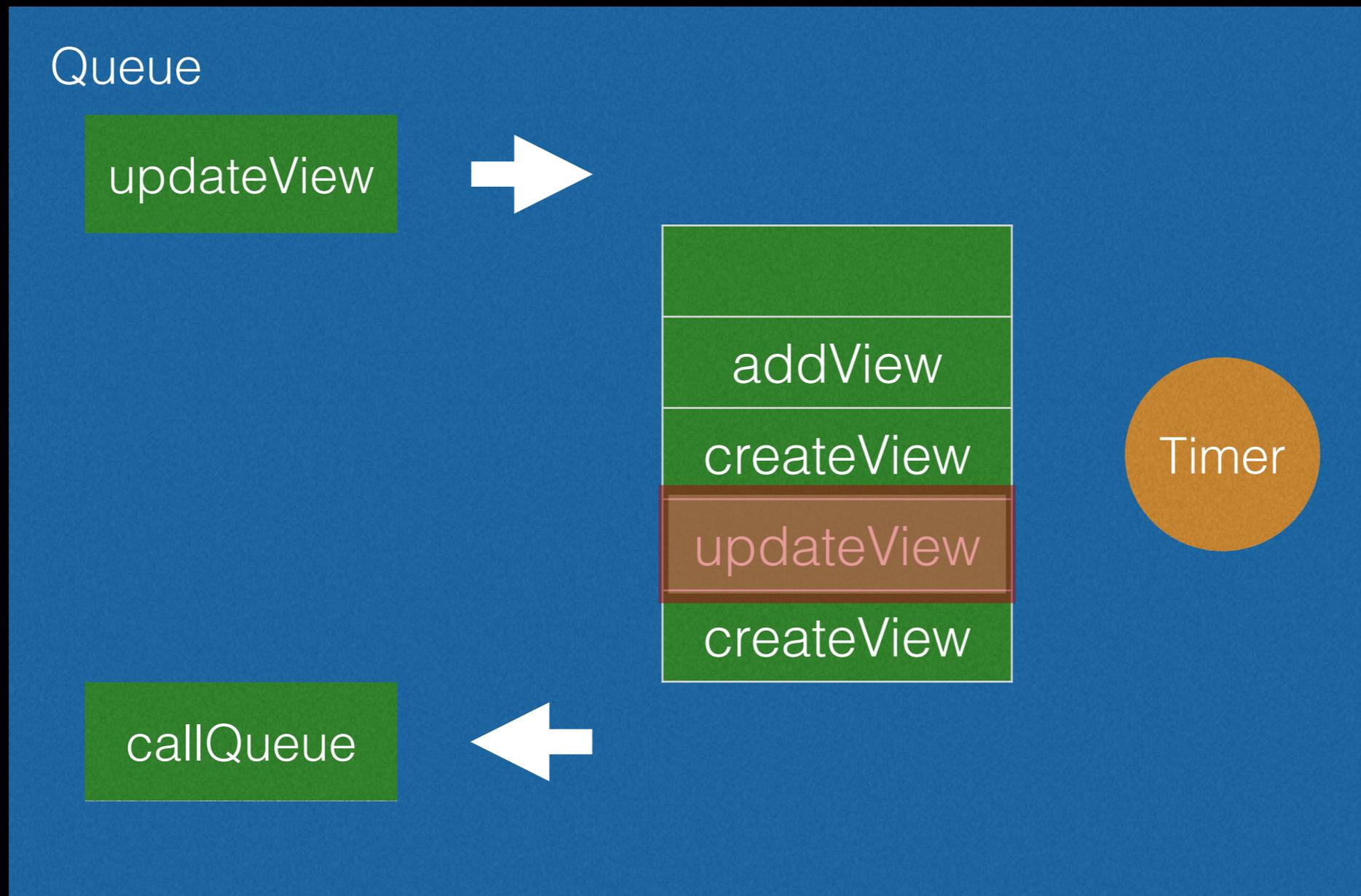
  function flush() {
    running = false;
    callback(list);
    list = [];
  }

  return function (cmd) {
    list.push(cmd);
    if (!running) {
      setTimeout(flush, 0);
      running = true;
    }
  };
}
```

性能优化



性能优化



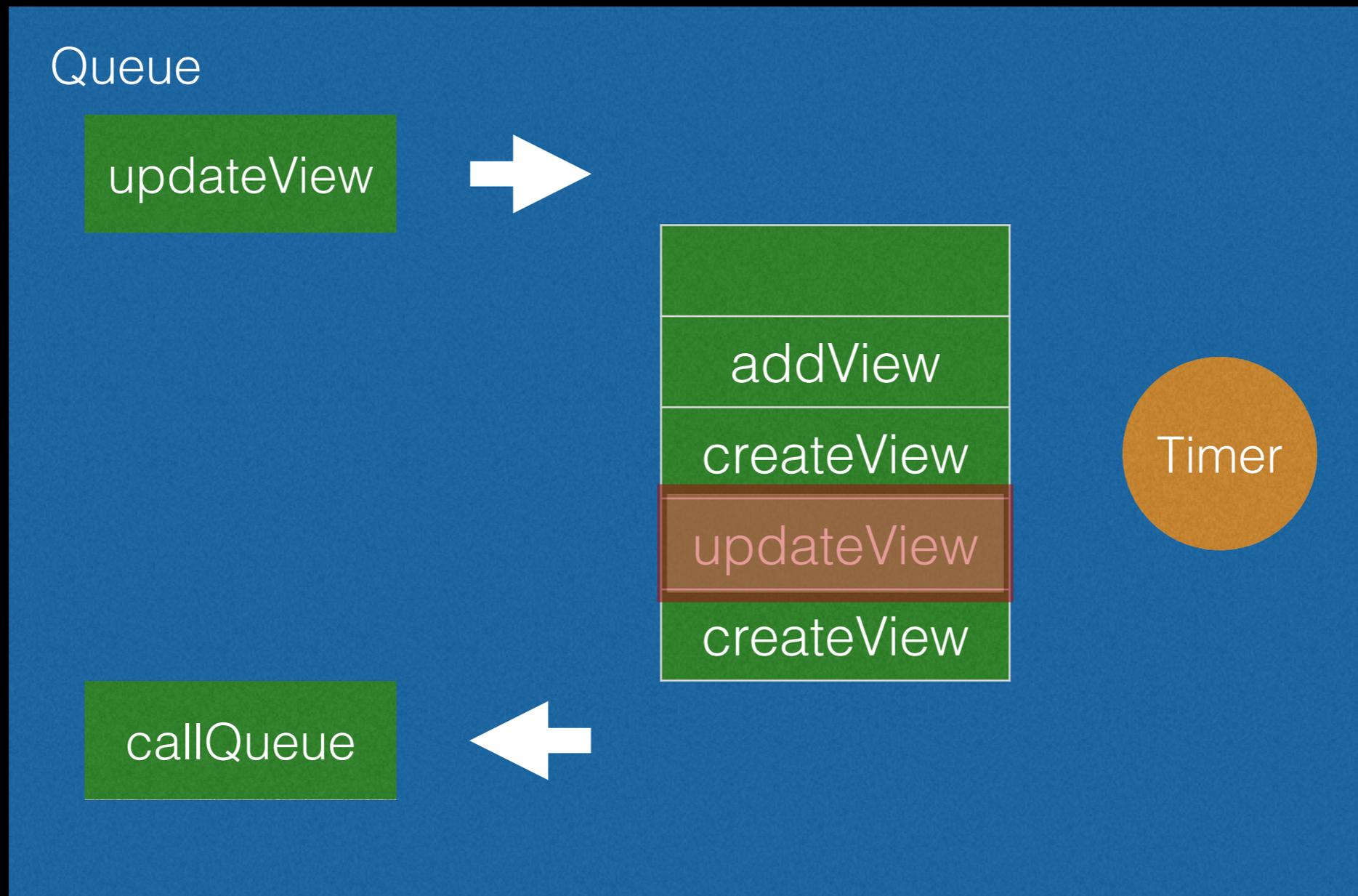
性能优化

Queue

```
window.lc_bridge.callQueue("commands": jsonString);
```

```
[  
  // 第一个命令  
  {  
    "tag": string, // 需要调用函数的目标 Tag  
    "method": string, // 函数名  
    "args": [ // 参数列表  
      object, // 参数 1  
      object, // 参数 2  
      ...  
    ]  
  },  
  
  // 第二个命令  
  { ... },  
  
  // ...  
]
```

性能优化

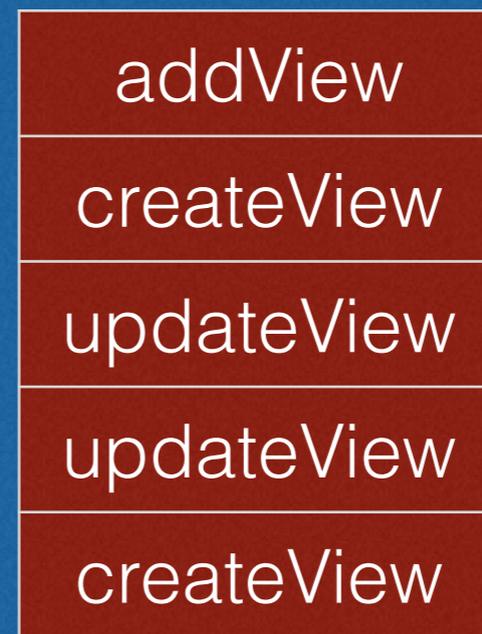


性能优化



Js

反射
→



Native

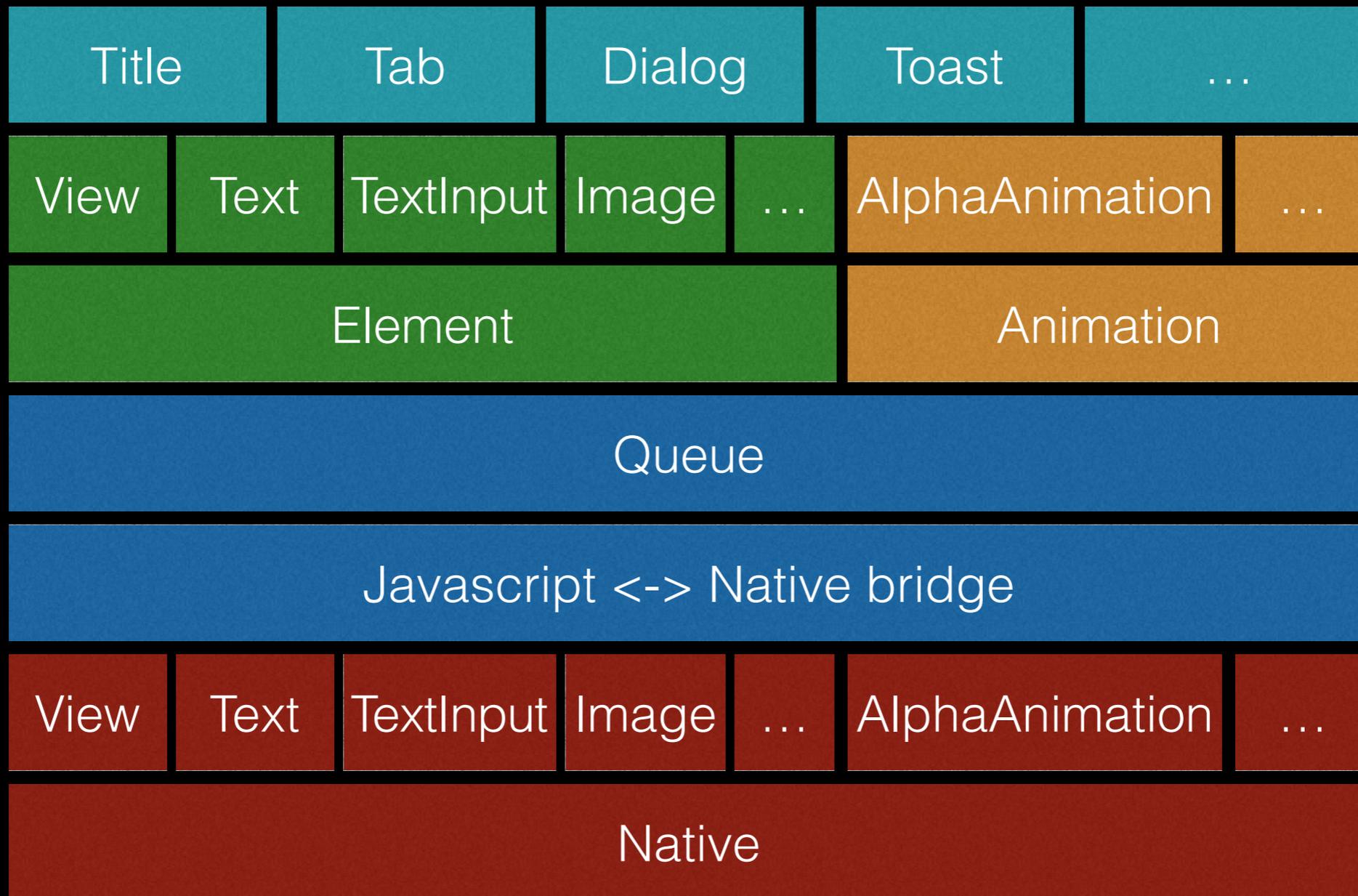
性能优化

addView
createView
updateView
updateView
createView

提取
→

```
switch (methodId) {  
  case 80:  
    addView();  
    break;  
  case 81:  
    createView();  
    break;  
  case 82:  
    updateView();  
    break;  
  default:  
    break;  
}
```

Blend UI 系统架构



Dom API 模拟

```
var view = boost.createElement("VIEW");  
  
view.style.position = "absolute";  
view.style.top = 100;  
view.style.left = 100;  
view.style.width = 200;  
view.style.height = 200;  
view.style.backgroundColor = "#ff6600";  
  
boost.rootElement.appendChild(view);  
  
var text = boost.createElement("TEXT");  
  
text.style.color = "#000000";  
text.style.fontSize = 20;  
text.value = "hello world";  
  
view.appendChild(text);
```

Dom API 模拟

Style 属性

事件模拟

```
view.addEventListener("touchend",  
  function (event) {  
    text.value = "doun't touch me!";  
    view.style.top = Math.random() * 400;  
    view.style.left = Math.random() * 400;  
  },  
  false);
```

← Dom事件

HTML & CSS 解析

```
.multi-wrapper {padding-bottom: 0;}
```

```
.empty {background-color: #f8f8f8; flex: 1;}
```

```
.empty .cart-img-view {height: 105; justify-content: center;}
```

```
.empty-tip {height: 24; line-height: 24; align-items: center;}
```

```
.empty-tip-text {color: #666;}
```

```
.shopping-btn {margin-top: 15; width: 114; height: 28; background-color: #fff; border: 1px solid #ccc; border-radius: 4px;}
```

```
.shopping-btn-text {color: #fff; font-size: 14;}
```

```
.goods-disable Text {color: #939393;}
```

```
</style>
```

```
▼ <View class="wrapper ">
```

```
  ▼ <ScrollView class="scrollview">
```

```
    ▼ <View class="content">
```

```
      ▼ <View class="login-view">
```

```
        <Text class="login-tip">登录后可同步购物车中商品</Text>
```

```
        ▶ <View class="login-btn">...</View>
```

```
      </View>
```

```
      ▶ <View class="outer-view">...</View>
```

```
    </View>
```

```
  </ScrollView>
```

类 HTML 支持
CSS 样式支持

CSS 选择器实现

```
require("widget/checkbox");
```

jQuery/Zepto支持

```
require("widget/counter");
```

```
require("widget/cartItem");
```

```
function init() {  
    $(".check-btn, .checkall-btn").checkbox();  
    $(".item-counter").counter();  
    $(".item-container").cartItem();  
}
```

```
module.exports = {  
    init: init  
};
```

前端框架



Thanks