

第十六届 D2 前端技术论坛

使用跨端解决方案 Rax 编写鸿蒙应用

大果



大果 (刘雄)



淘系技术部 - 前端架构 - 前端工程师



跨端 / 框架 / 中后台



飞冰 ICE



Rax

Contents

目录

- 01 关于跨端解决方案 Rax
- 02 当 Rax 遇到鸿蒙系统
- 03 原理：Rax 是如何对接鸿蒙的

01

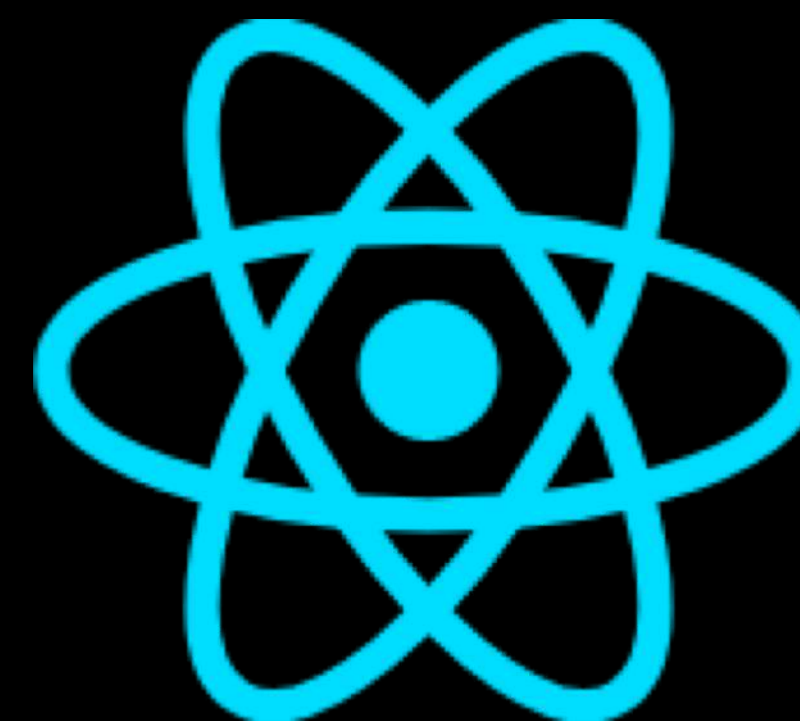
关于跨端解决方案 Rax

为什么会有跨端？



D2 前端技术论坛
D2 FRONTEND TECHNOLOGY FORUM

精心



- 设备&网络差异大带来的性能问题
- App 容器层给了可能性
- 大厂商的超级 App 生态



跨端背景下需要什么样的技术方案？

一码多端

各写一套

研发成本



适配成本



功能一致



招聘成本



社区中一码多端的方案



D2 前端技术论坛
D2 FRONTEND TECHNOLOGY FORUM

精心



Rax



Remix



Rax 是一个怎样的跨端方案？



- 类 React 语法，以及配套的工程框架体系
- 支持 Web、各厂商小程序、Weex 等不同端
- 服务手淘内 80% 以上的业务



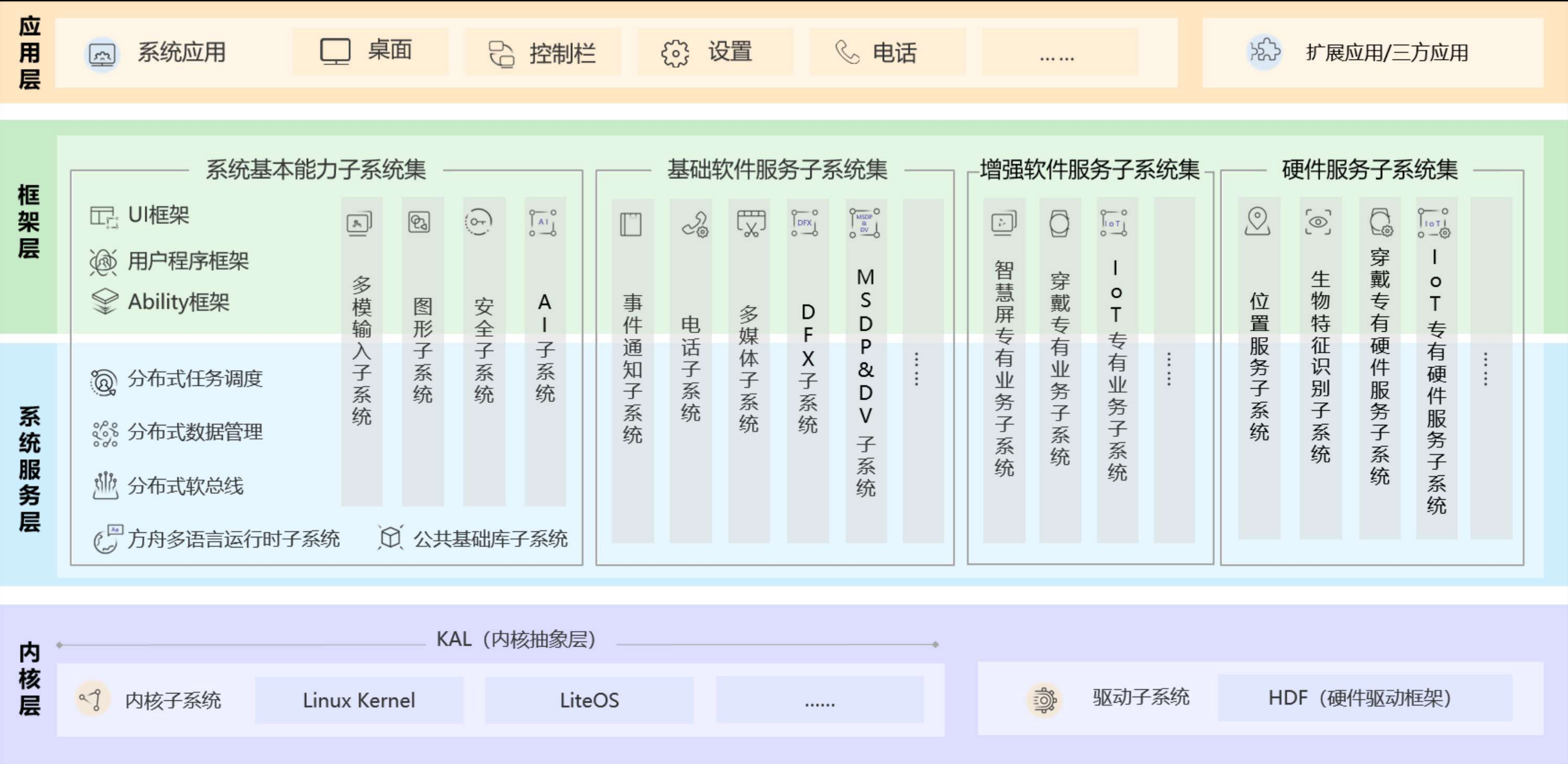


D2 前端技术论坛
D2 FRONTEND TECHNOLOGY FORUM

精心

02

当 Rax 遇到鸿蒙系统



- 华为自研开发的国产操作系统
- 未来有机会跟安卓、苹果三足鼎立
- 装机量突破 1.5 亿
- 支持使用 JavaScript 开发系统应用

比较项	Java UI框架	方舟开发框架	
语言生态	Java	JS	eTS
接口方式	命令式	类Web范式	声明式
执行方式	开发者处理，基于API驱动的UI变更	框架层处理，基于数据驱动的UI自动变更	框架层处理，基于数据驱动的UI自动变更
相对优势	UI元素更丰富，开发更灵活	轻量化，开发更简便	极简开发，内存占用更少、运行性能更高

鸿蒙系统将 JavaScript 作为应用开发的一等公民

Rax X 鸿蒙具备理论可能性

Why：业务有诉求吗？

🤔 单独开发一个鸿蒙应用

😺 开发一个有流量入口的服务卡片



Why：技术有方案吗？

比较项	Java UI框架	方舟开发框架	
语言生态	Java	JS	eTS
接口方式	命令式	类Web范式	声明式
执行方式	开发者处理，基于API驱动的UI变更	框架层处理，基于数据驱动的UI自动变更	框架层处理，基于数据驱动的UI自动变更
相对优势	UI元素更丰富，开发更灵活	轻量化，开发更简便	极简开发，内存占用更少、运行性能更高

1. 单独开发一套成本太高了？

2. 学不动了怎么办？

所以：支持用 Rax 开发鸿蒙应用吧



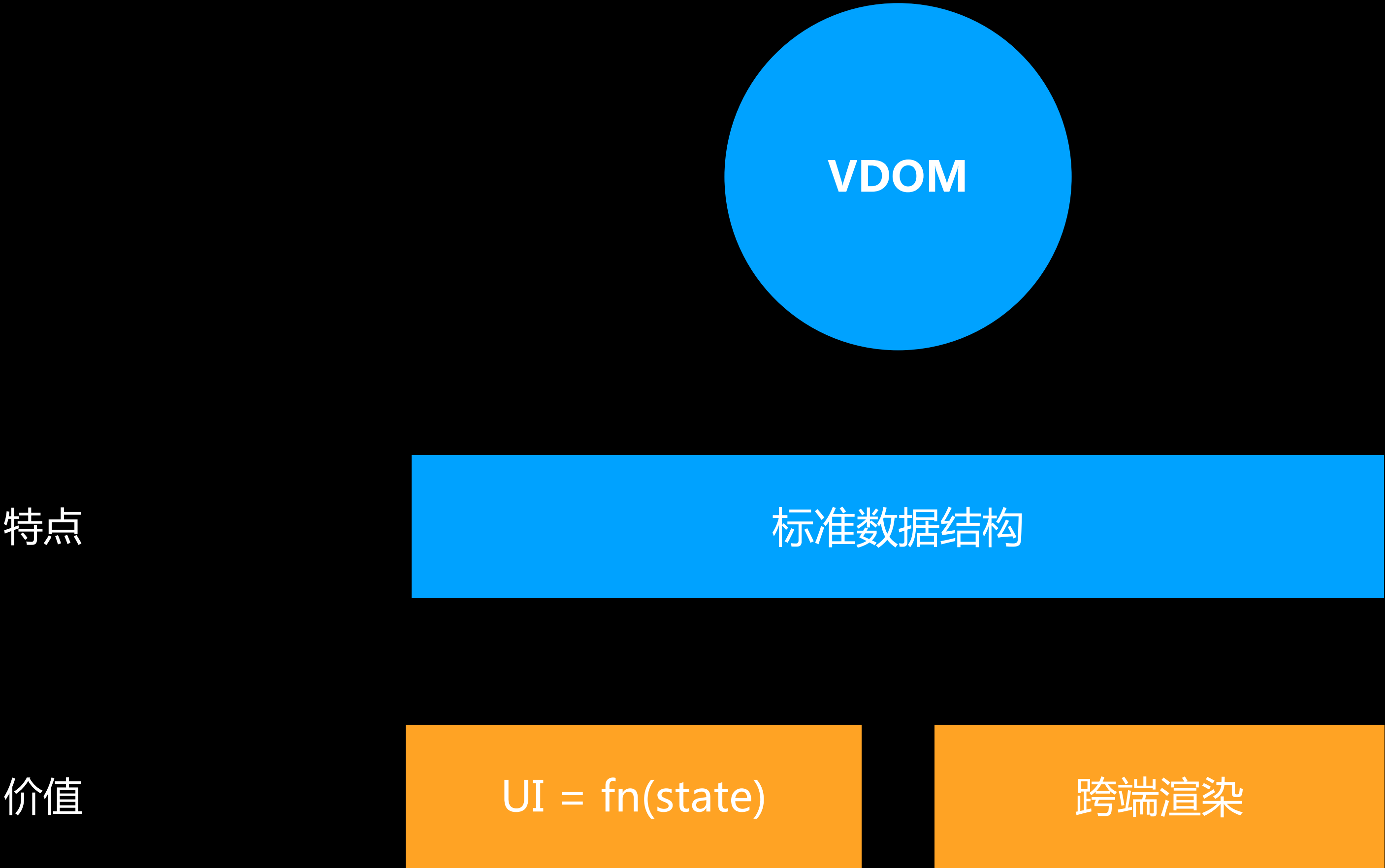
- 使用类 React 语法开发，不需要学习新的语法
- 一套代码多个端，降低开发成本
- 开发链路上可以更好的结合 Web 生态

03

原理：Rax 是如何对接鸿蒙的



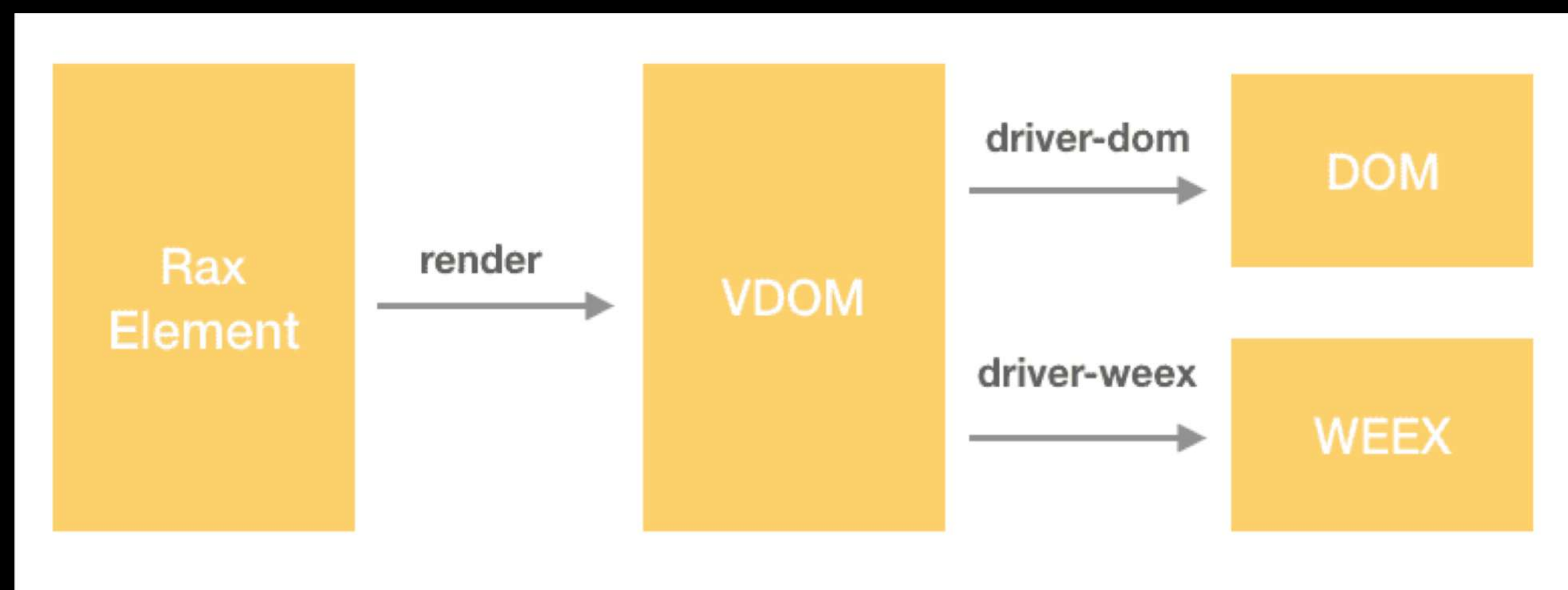
多端渲染: VDOM 的跨端能力



多端渲染: Rax Driver 适配器



D2 前端技术论坛
D2 FRONTEND TECHNOLOGY FORUM



```
const Driver = {
  getElementById(id) { ...
  },
  createBody() { ...
  },
  createElement(type, props) { ...
  },
  addEventListener(node, eventName, eventHandler) { ...
  },
  removeEventListener(node, eventName, eventHandler) { ...
  },
  removeAttribute(node, propKey, propValue) { ...
  },
  setAttribute(node, propKey, propValue) { ...
  },
  setStyle(node, style) { ...
  }
};

export default Driver;
```

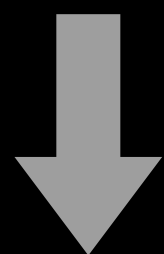

多端渲染: Rax Driver 生态



D2 前端技术论坛
D2 FRONTEND TECHNOLOGY FORUM

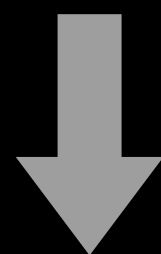


driver-web



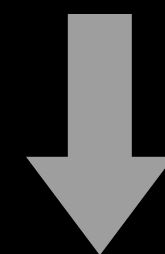
WebView

driver-server



Node.js

driver-miniapp



小程序



```
// driver-server
const Driver = {
  nodeMaps: {},

  getElementById(id) {
    return this.nodeMaps[id];
  },

  createBody() {
    return {
      nodeType: ELEMENT_NODE,
      tagName: 'BODY',
      attributes: {},
      style: {},
      eventListeners: {},
      childNodes: [],
      parentNode: null
    };
  },
};

// ...
}
```


多端渲染: 对接方案



D2 前端技术论坛
D2 FRONTEND TECHNOLOGY FORUM

精心

index.html

```
<!-- index.html -->
<div class="container">
  <text class="text">
    Hello World, Count: {{count}}
  </text>
  <button
    class="button"
    value="+"
    onclick="onClick"
  ></button>
</div>
```

index.css

```
// index.css
.text {
  font-size: 42px;
}

.button {
  width: 240px;
  height: 60px;
  background-color: #007dff;
}
```

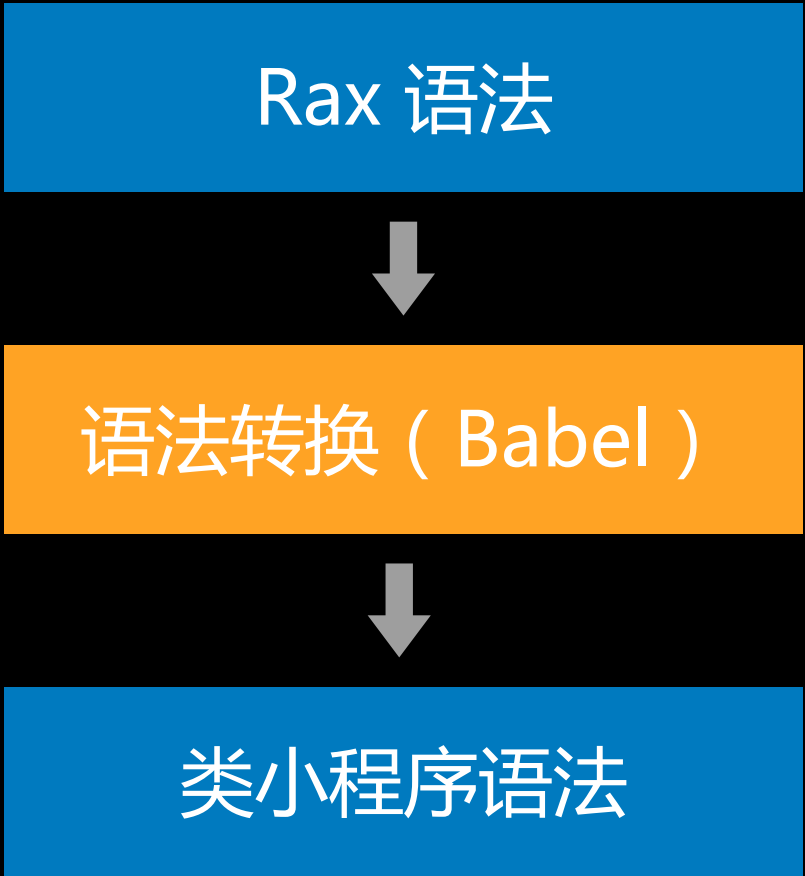
index.js

```
// index.js
export default {
  data: {
    count: 0,
  },
  onClick() {
    this.count += 1;
  },
}
```

鸿蒙官方的类 Web 开发范式，写法跟小程序/Vue 非常类似

多端渲染: 对接方案

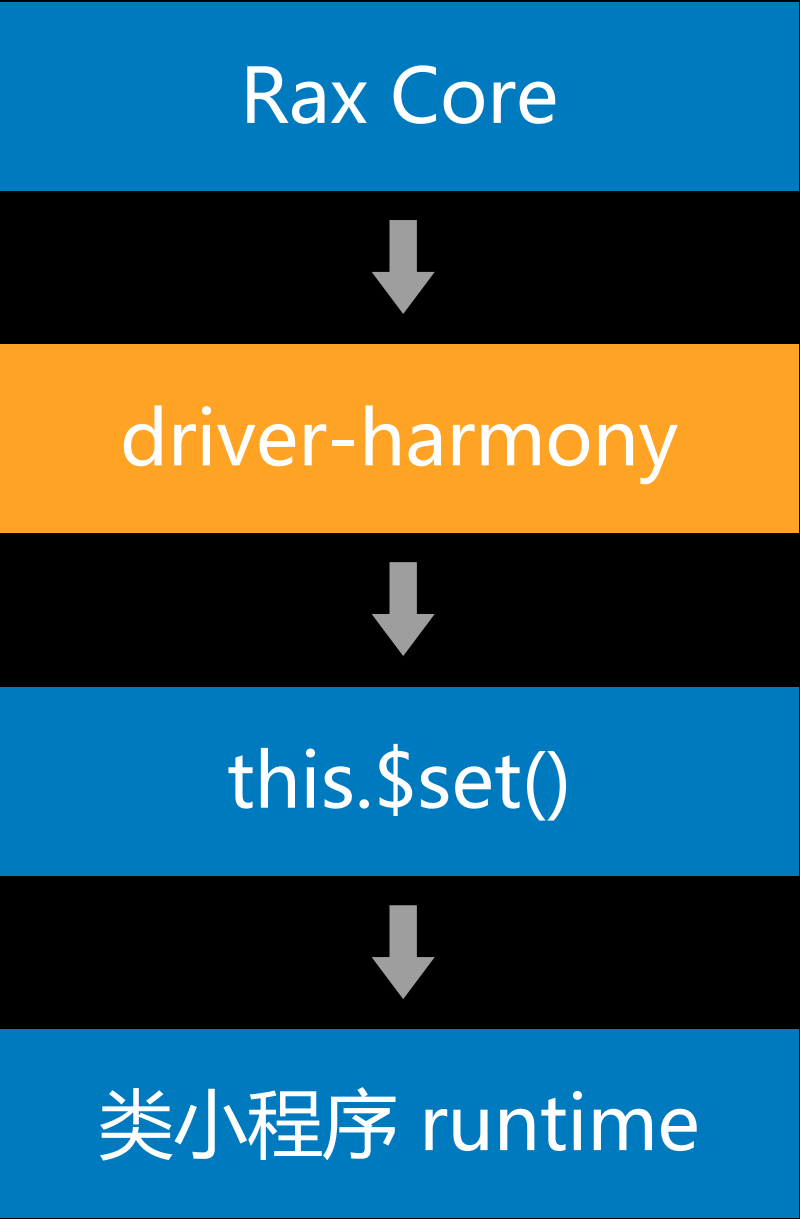
1. 语法编译



- 实现**成本高**
- 语法**约束多**



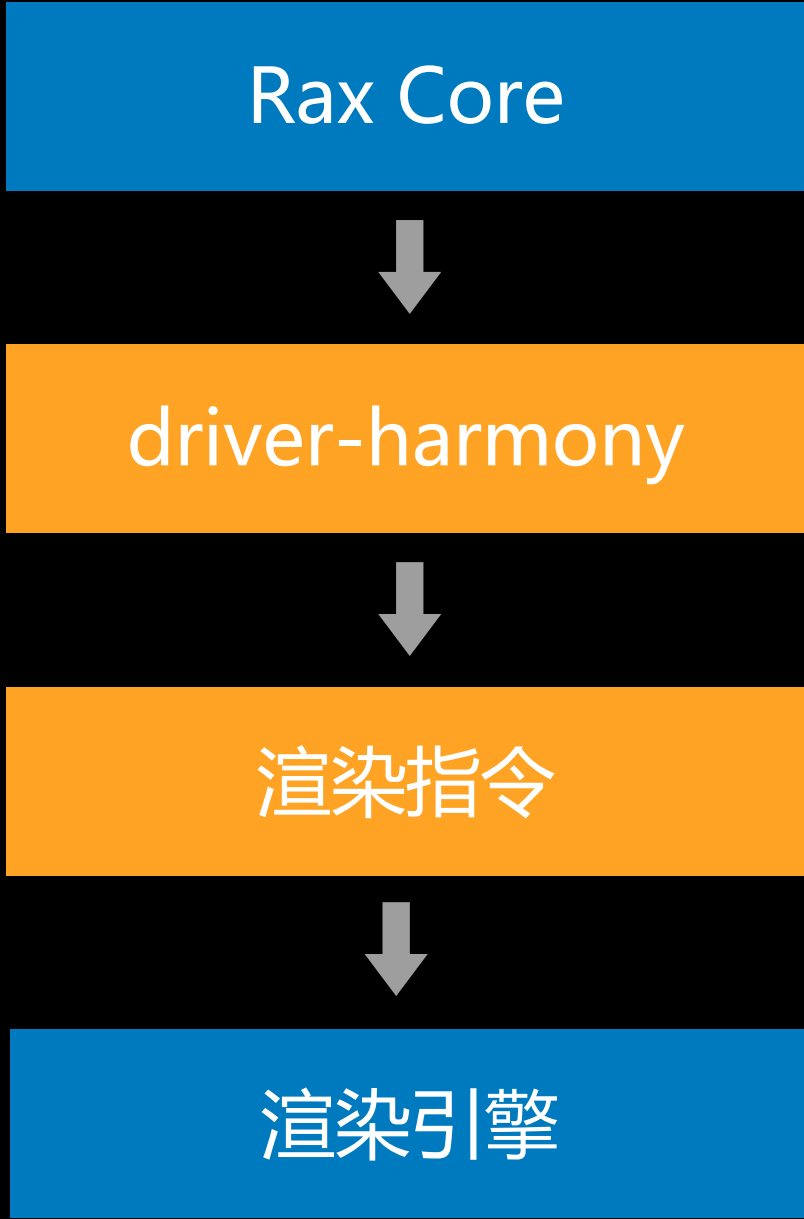
2. 类小程序运行时



- 运行时**复杂度高**
- 性能**劣化**
- 实现**周期短**



3. 对接底层渲染指令



- 运行时**轻量、高性能**
- 实现**周期较长**



多端渲染: driver-harmony



D2 前端技术论坛
D2 FRONTEND TECHNOLOGY FORUM



业务代码

Rax Core

driver-harmony

JS Framework

Render Engine

```
const Driver = {
  // Internal state
  nodeMaps: {},

  getElementById(id) {
    return this.nodeMaps[id];
  },

  createBody() {
    return document.body;
  },

  createElement(type, props) {
    const node = document.createElement(type, {
      style: props.style,
    });

    this.setNativeProps(node, props, true);
    return node;
  },

  addEventListener(node, eventName, eventHandler) {
    node.addEventListener(eventName, eventHandler);
  },

  removeEventListener(node, eventName, eventHandler) {
    node.removeEventListener(eventName, eventHandler);
  },

  removeAttribute(node, propKey, propValue) {
    if (propKey === ID) {
      this.nodeMaps[propValue] = null;
    }
    return node.setAttribute(propKey, undefined, false);
  },
};

export default Driver;
```

多端渲染: driver-harmony

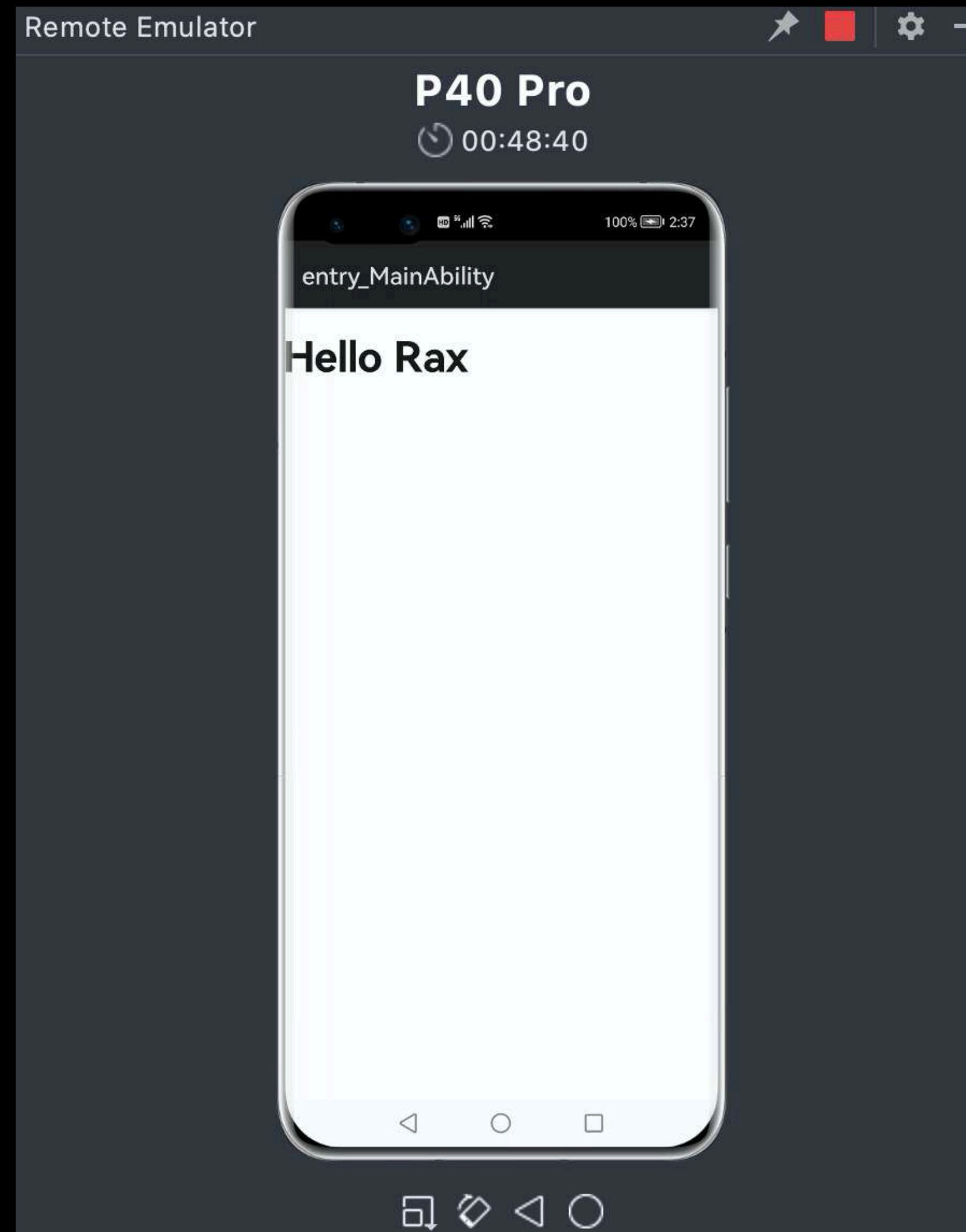


D2 前端技术论坛
D2 FRONTEND TECHNOLOGY FORUM



```
// pages/home/index.js
import { createElement, render } from 'rax';
import DriverHarmony from 'driver-harmony';

export default {
  onReady() {
    render(<text>Hello Rax</text>, {
      driver: DriverHarmony
    });
  },
  onDestroy() {
    console.info('Page onDestroy');
  }
}
```

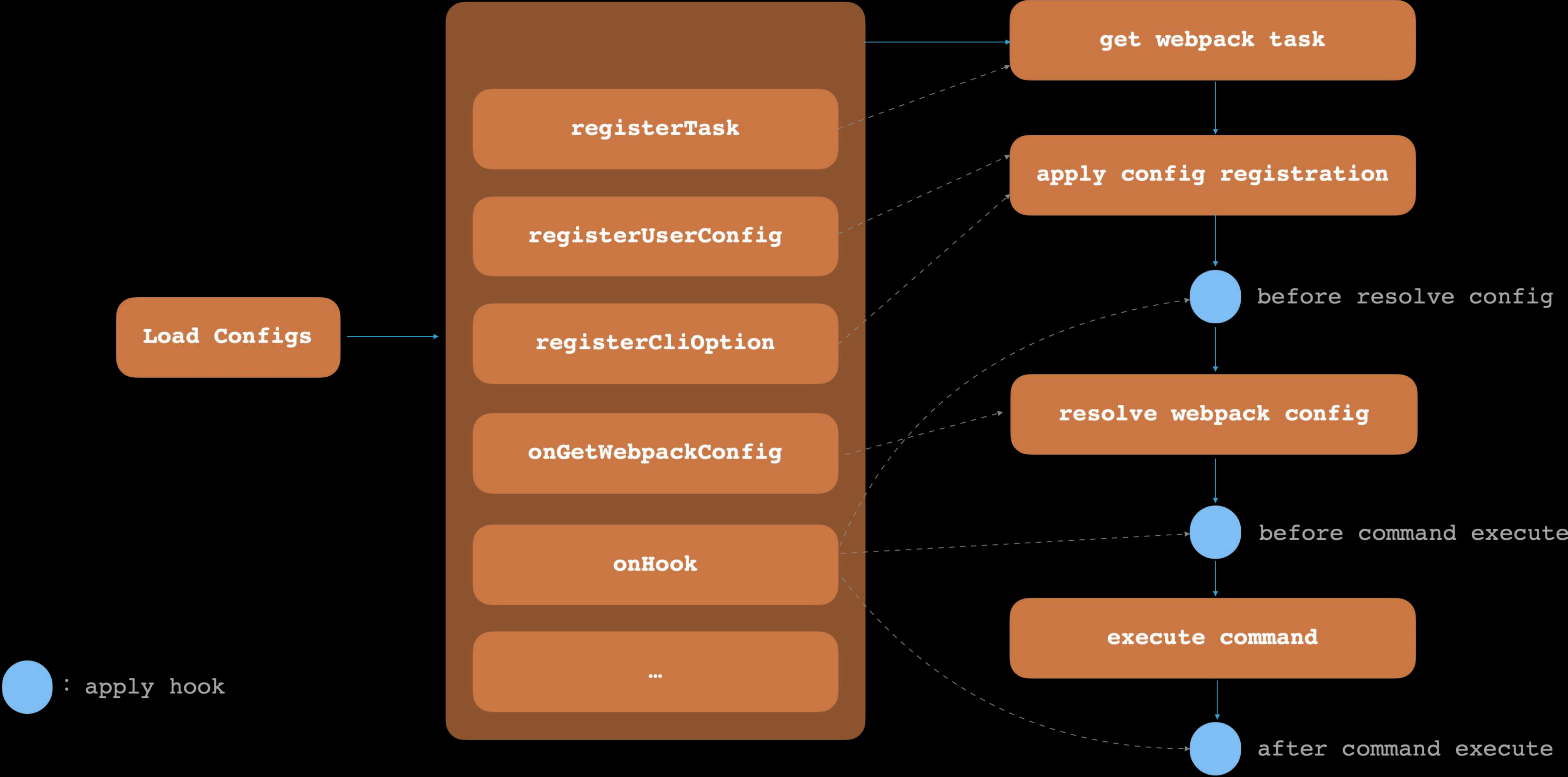


工程配置 build.json

```
{
  "targets": ["harmony", "web"],
  "harmony": {
    "mpa": true
  },
  "webpack5": true,
  "plugin": [
    ["build-plugin-rax-harmony"]
  ]
}
```

运行时信息配置 app.json

```
{
  "routes": [
    {
      "name": "home",
      "source": "pages/Home/index",
      "spm": "b456"
    },
    {
      "name": "about",
      "source": "pages/About/index"
    }
  ],
}
```



工程构建: plugin-rax-harmony



D2 前端技术论坛
D2 FRONTEND TECHNOLOGY FORUM



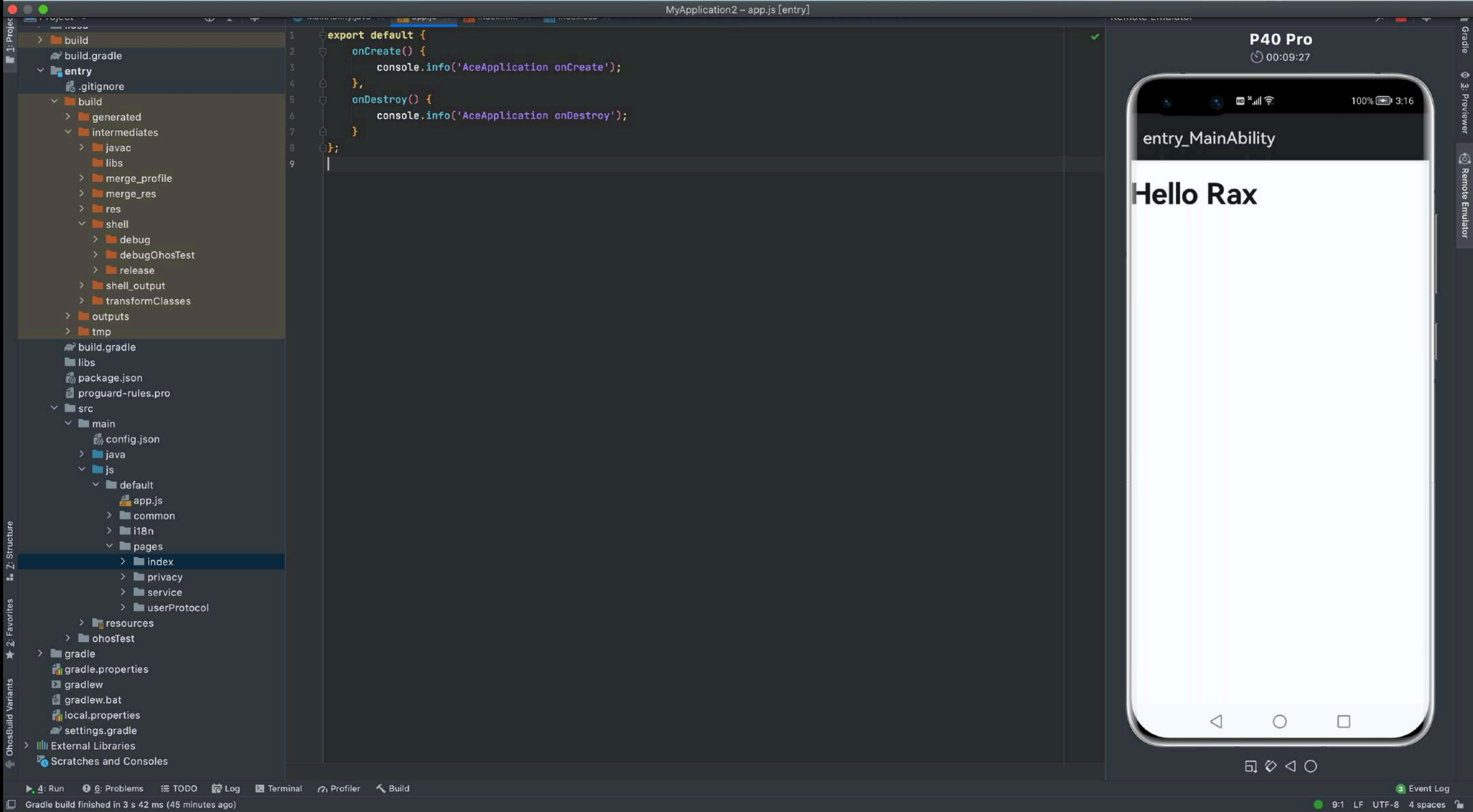
build-plugin-rax-harmony

注册独立的 Webpack 任务

定制 Webpack 配置

生成鸿蒙产物

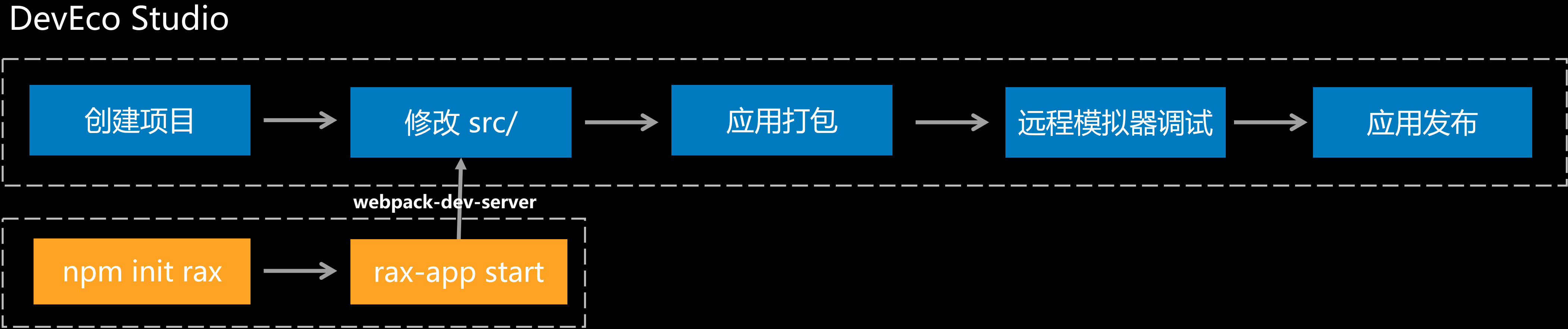
研发链路：方案设计



鸿蒙官方提供了 DevEco Studio IDE 支持应用的全链路开发调试与打包。

Rax 的工程链路如何对接？

短期方案

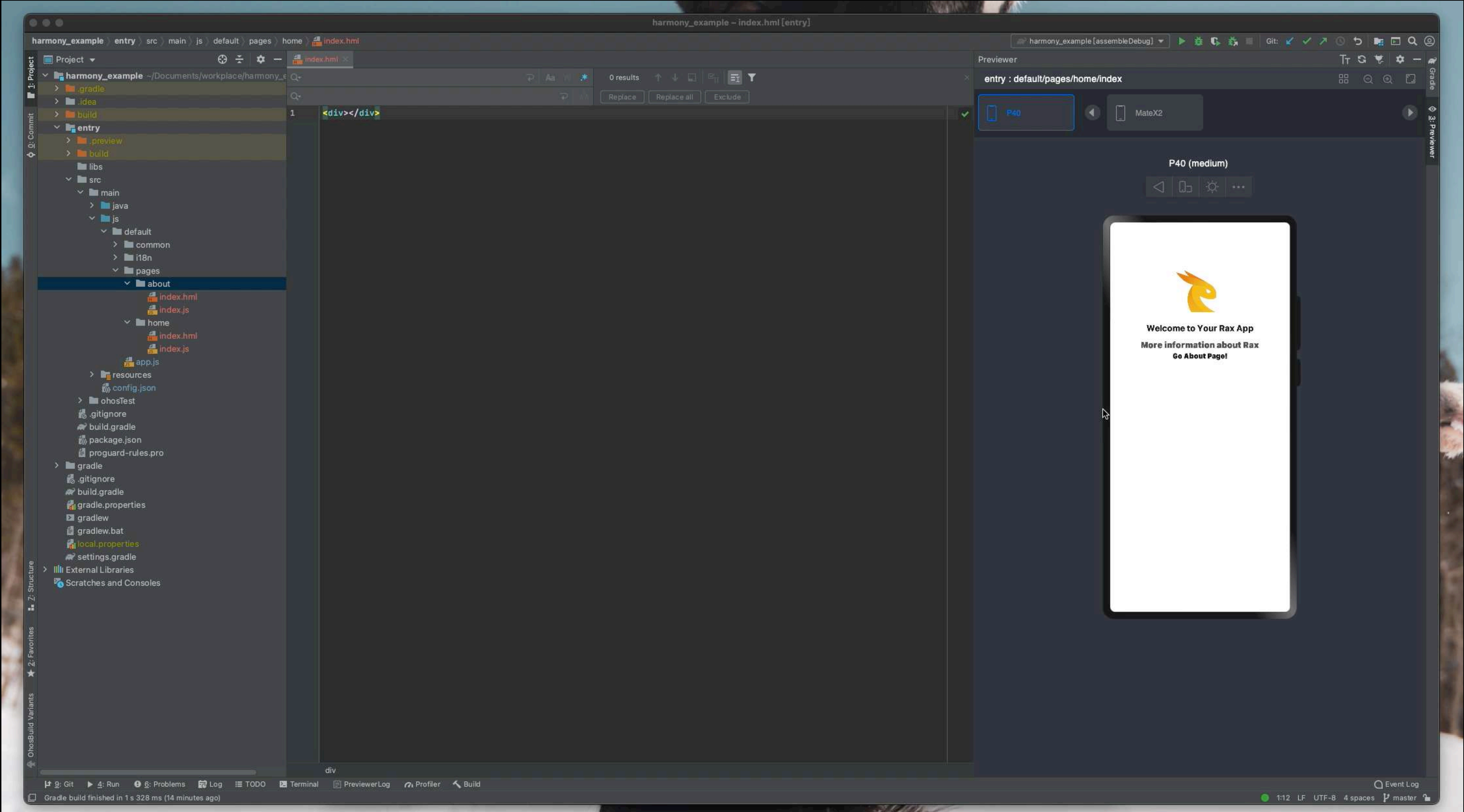


长期方案

(建设中)

支持在 DevEco Studio 从 Rax 应用创建 -> 开发 -> 调试 -> 发布的全链路研发能力

研发链路：方案演示

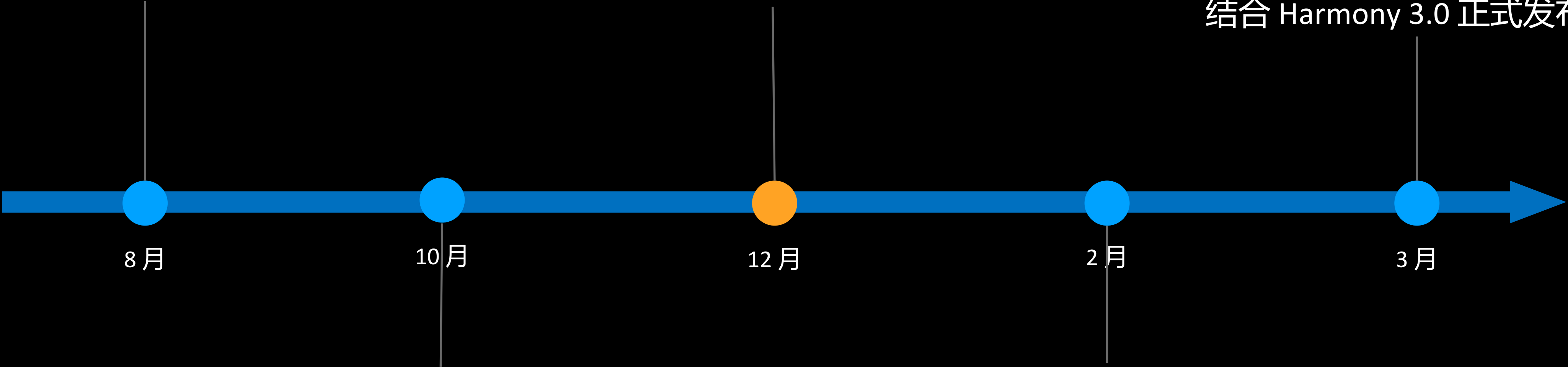


Roadmap

Rax 团队和鸿蒙团队第一次会议交流

- 完成渲染链路的完整对接
- 与鸿蒙 IDE 团队沟通全链路方案

结合 Harmony 3.0 正式发布



结合鸿蒙提供的渲染指令跑通 demo

- IDE 全链路流程开发完成
- Rax 相关组件&API 适配完成

1. 为什么需要一码多端的跨端解决方案
2. 鸿蒙系统以及对应的应用开发方案
3. 我们是如何实现使用 Rax 编写鸿蒙应用的
4. 后续的一些规划

Thanks