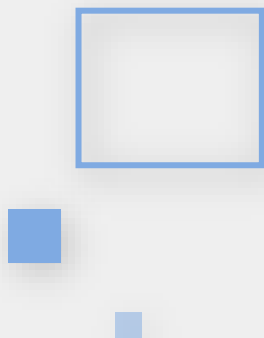


# 基于Web内核的微信小程序框架实践



陈博学

Email: [boxuechen@tencent.com](mailto:boxuechen@tencent.com)

# | 个人介绍

- 2014年加入腾讯
- 2015年加入微信终端开发
- Android 小游戏渲染引擎
- 小程序同层渲染和Canvas组件
- 微信XWeb浏览器内核
- 研究Chromium内核排版渲染
- 目前负责PC微信小程序跨平台框架



## 基于Web内核的小程序框架

- 和微信主客户端解耦，可以独立运行小程序和小游戏的一套Runtime
- 基于Chromium内核，拥抱Web特性，支持更多平台。
- 拓宽小程序生态边界，为更多非移动平台设备提供微信生态支持。
- WMPF : Wechat Mini Program Framework

# | 目录

- 小程序和PWA
- 基于Chromium内核的小程序框架
- 基于Chromium内核的小游戏框架
- 进程架构 & 跨平台实践



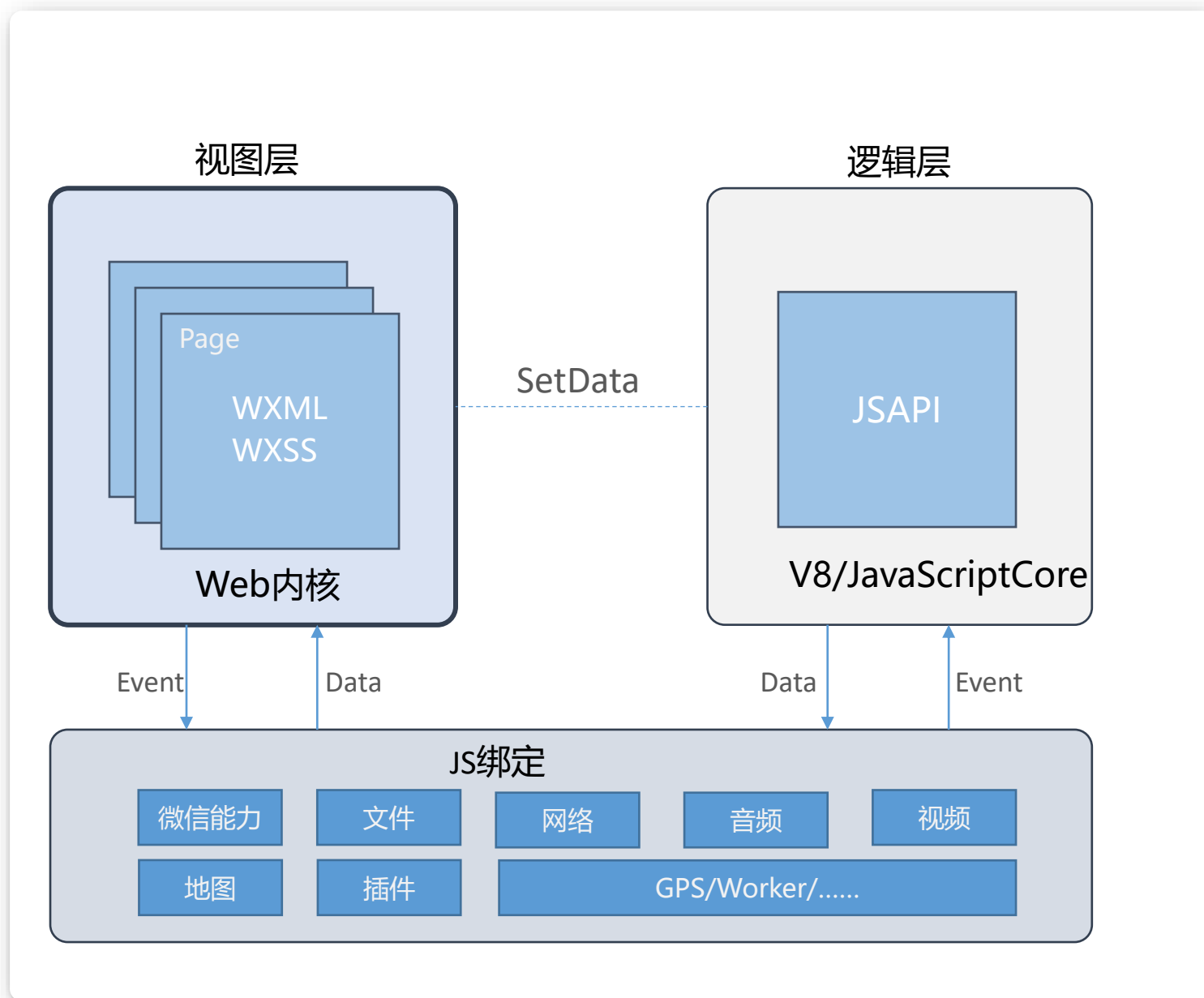
# 01

## 小程序和PWA

# 小程序和PWA

## 小程序框架简介

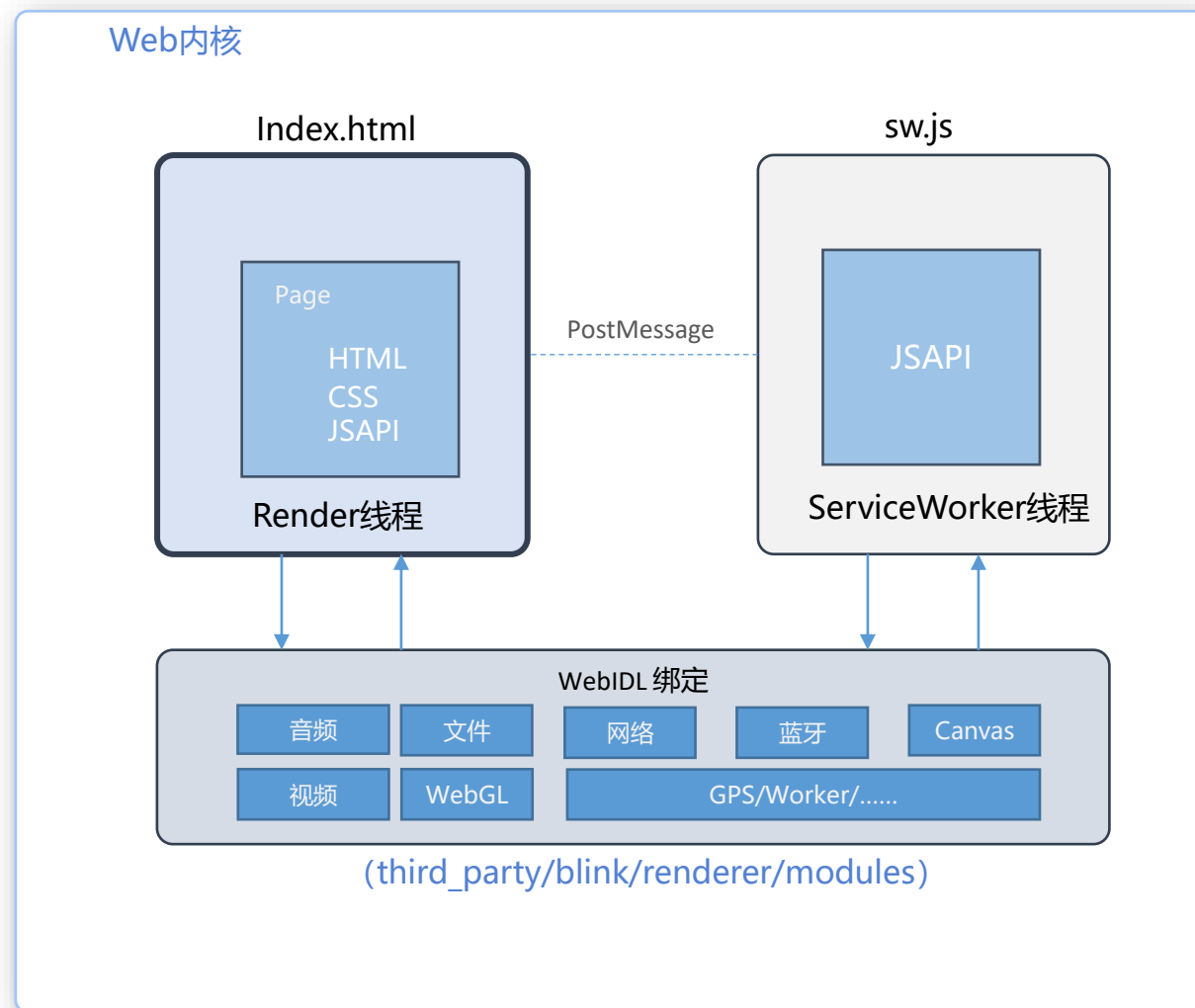
- 视图层: WebPage
- 逻辑层: 独立的 JS Runtime
- JS绑定: 为JSApi提供Native能力
- setData: 数据和事件传递



# 小程序和PWA

## PWA简介

- 运行在Web内核中
- index.html: 视图层, 可执行JSAPI
- sw.js: service线程, 管理PWA生命周期, 可执行JSAPI
- WebIDL绑定: fugu project, H5 API
- PostMessage: 数据和事件传递



# | 小程序和PWA

## 小程序和PWA对照

微信小程序		PWA
视图端	<ol style="list-style-type: none"><li>1. Web内核渲染</li><li>2. WXML和WXSS</li></ol>	<ol style="list-style-type: none"><li>1. Web内核渲染</li><li>2. HTML和CSS</li><li>3. <b>JSAPI</b></li></ol>
逻辑端	<ol style="list-style-type: none"><li>1. 独立JSRuntime线程</li><li>2. JS绑定扩展JSAPI</li><li>3. setData通信</li></ol>	<ol style="list-style-type: none"><li>1. Web内核提供的JS线程</li><li>2. WebIDL绑定扩展JSAPI</li><li>3. PostMessage通信</li><li>4. 生命周期管理</li></ol>
其他	<ol style="list-style-type: none"><li>1. 开发者代码运行在沙箱环境</li><li>2. 微信私有接口</li></ol>	<ol style="list-style-type: none"><li>1. 开发者对代码有完全控制</li><li>2. H5 API标准接口</li></ol>



# 小程序和PWA

## 向PWA学习

---

- 轻量的JS Worker线程，不需要额外的 V8/JSCore环境
- PostMessage 高性能通信
- 丰富的H5 API（Canvas/ WebSocket等）

深入Web内核，定制小程序框架





# 02

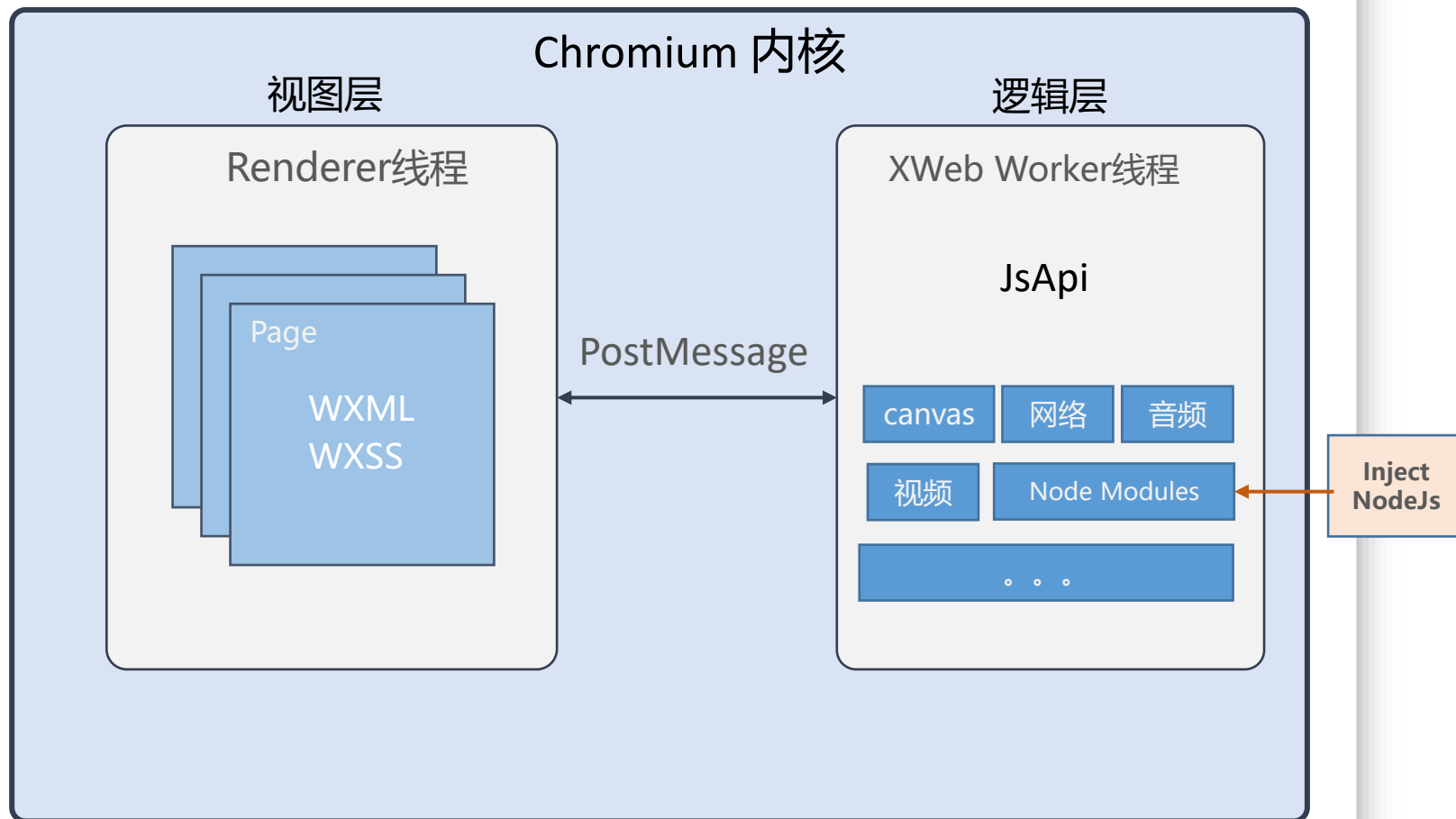
**基于Chromium内核的小程序框架**

# 基于Chromium内核的小程序框架

## 基于Chromium内核的小程序框架

运行环境：Chromium内核

- 视图层：Renderer线程的Page
- 逻辑层：XWeb Worker线程
- JS绑定：集成Node
- JSAPI：H5 API + Node API
- PostMessage：数据和事件传递

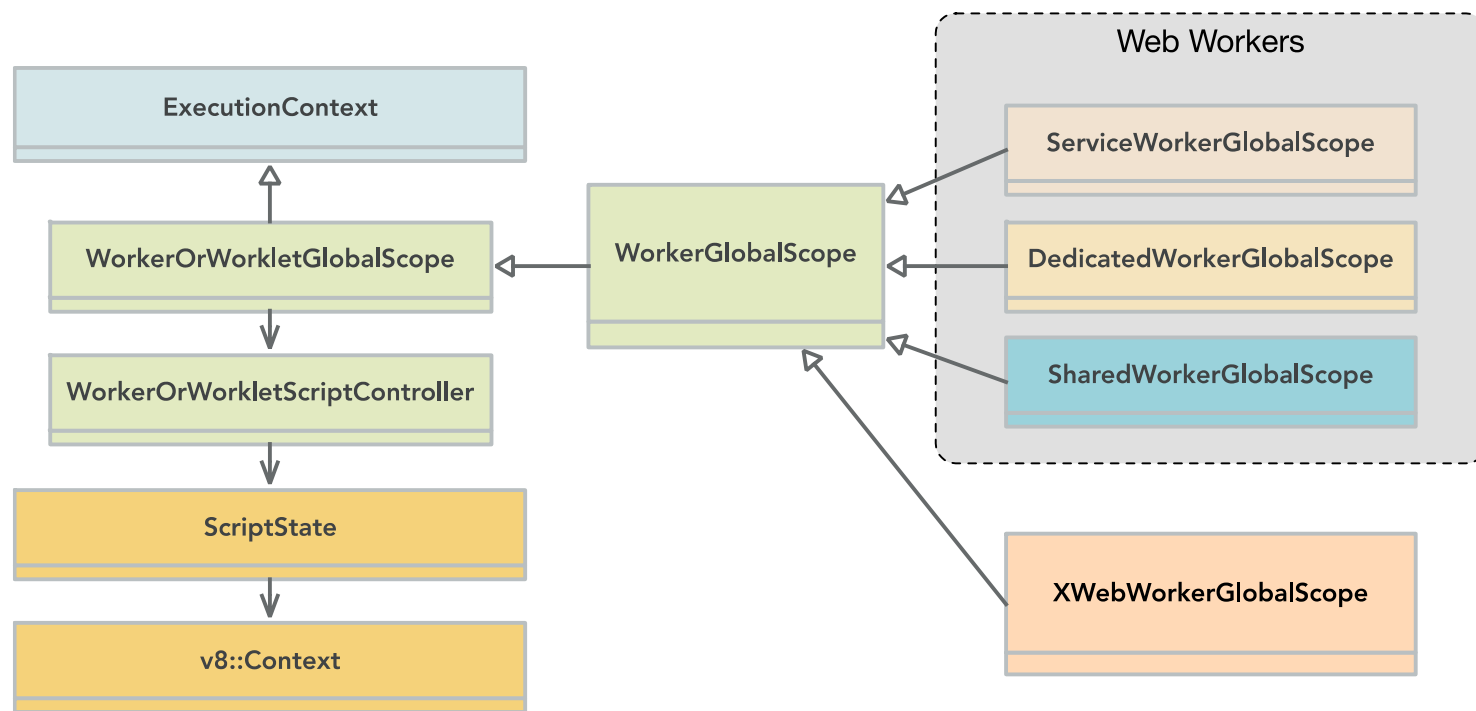


# 基于Chromium内核的小程序框架

## 设计XWeb Worker

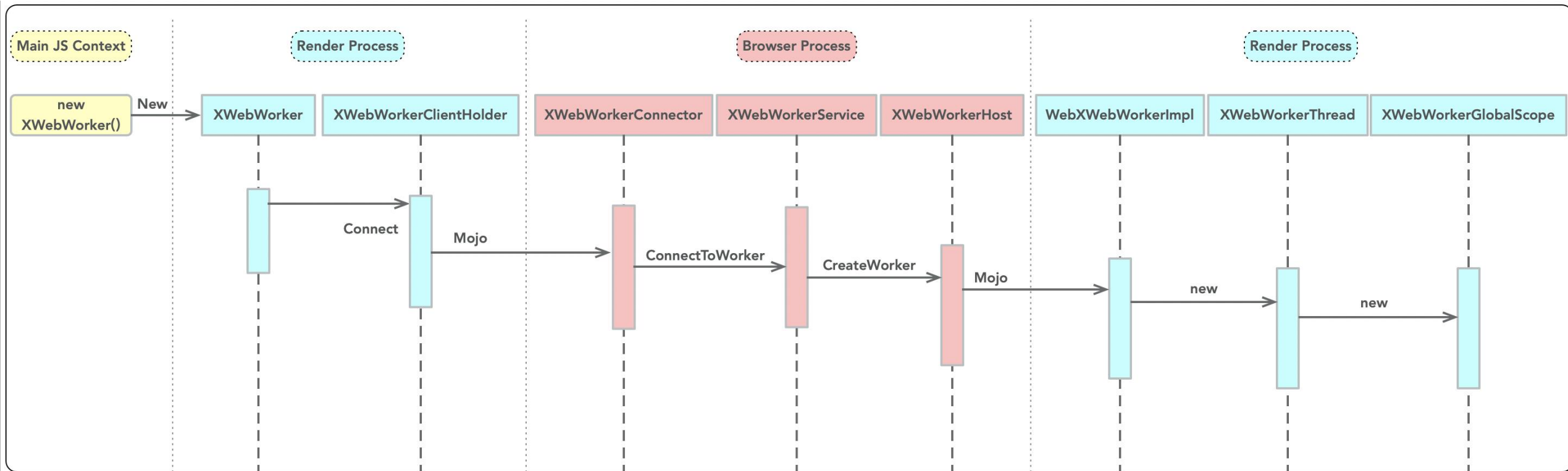
Chromium创建Web Worker

- blink/render/core/workers
- Web Worker的JS运行环境，来自WorkerGlobalScope



# 基于Chromium内核的小程序框架

## 设计XWeb Worker



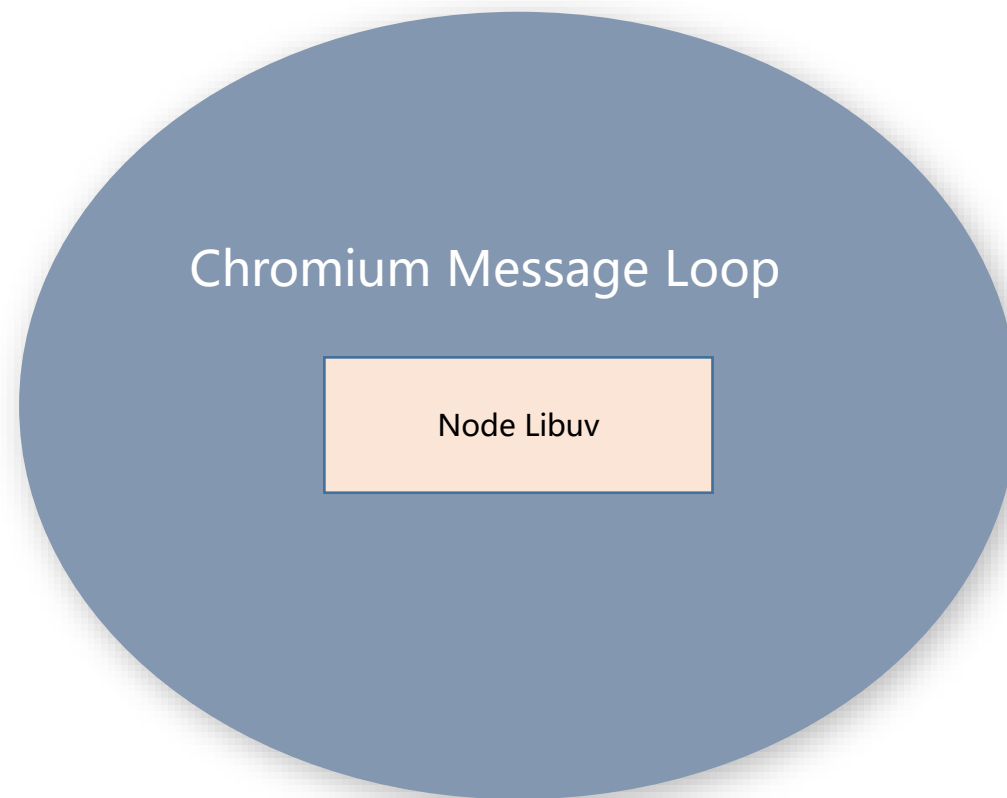
- 参考 SharedWorker 流程, 设计 XWebWorker
- 共享 V8
- H5 API 接口
- PostMessage 通信

# 基于Chromium内核的小程序框架

## JSAPI能力支持

Chromium内核集成 Node

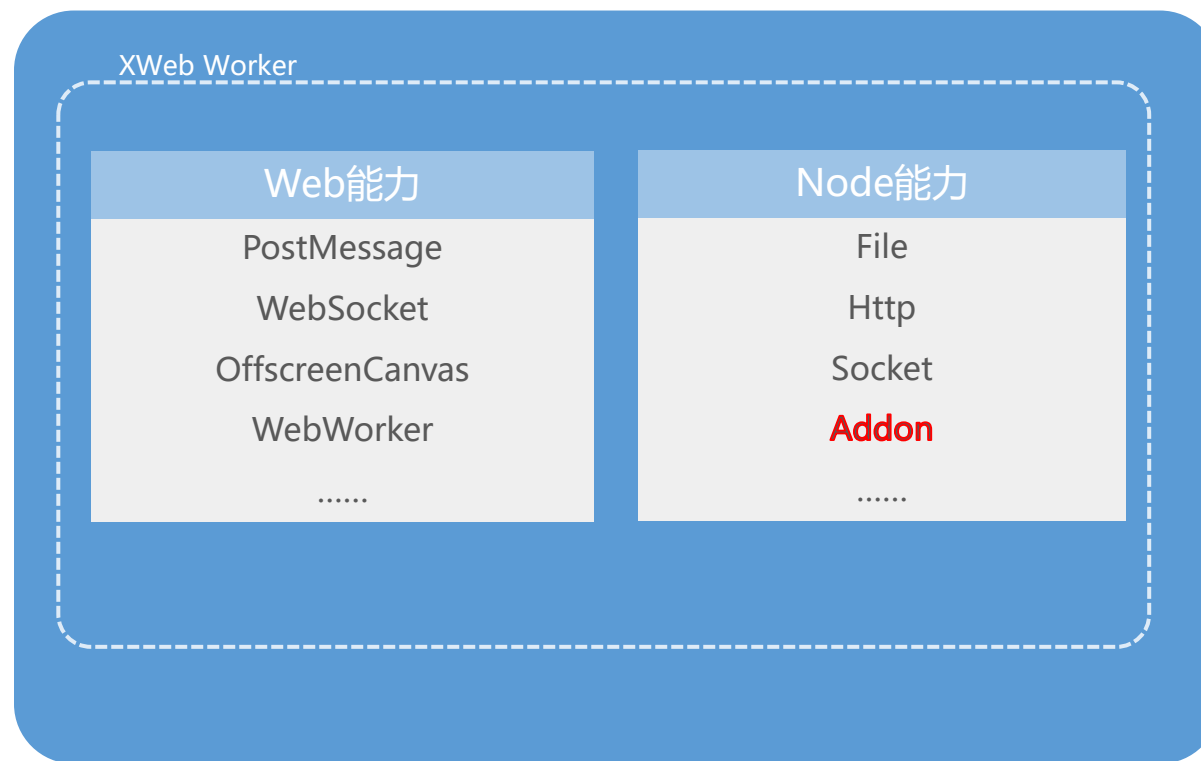
- 参考Electron思路
- 消息循环融合
- Node Libuv
- Chromium MessageLoop
- 《Electron Internals: Message Loop Integration》



# 基于Chromium内核的小程序框架

## JSAPI能力支持

- H5 API 能力
- Node API 能力
- Node的 Addon 灵活扩展

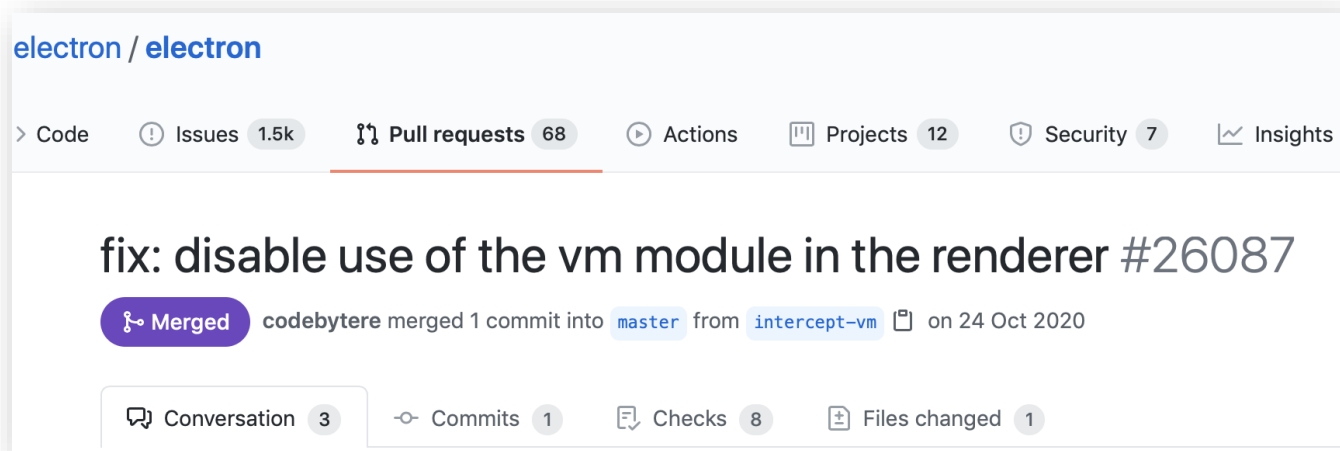


# 基于Chromium内核的小程序框架

## 沙箱能力支持

- 沙箱能力隔离JS执行上下文，安全保证
- Web Worker自身没有创建沙箱的能力
- Node VM 不可用

Node VM方案不可行



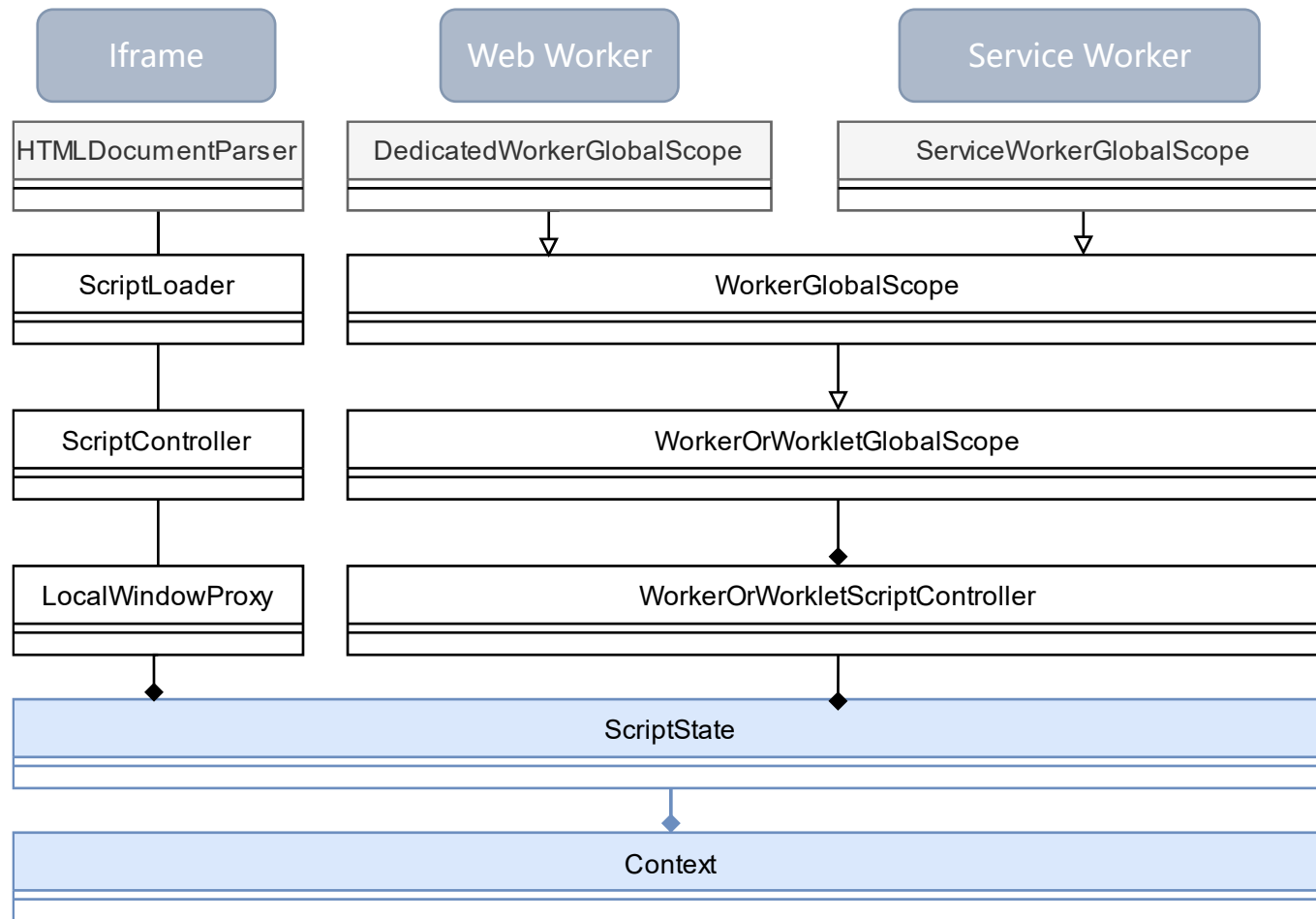


# 基于Chromium内核的小程序框架

## 沙箱能力支持

Chromium封装V8::Context的分析

- Iframe
- Web Worker

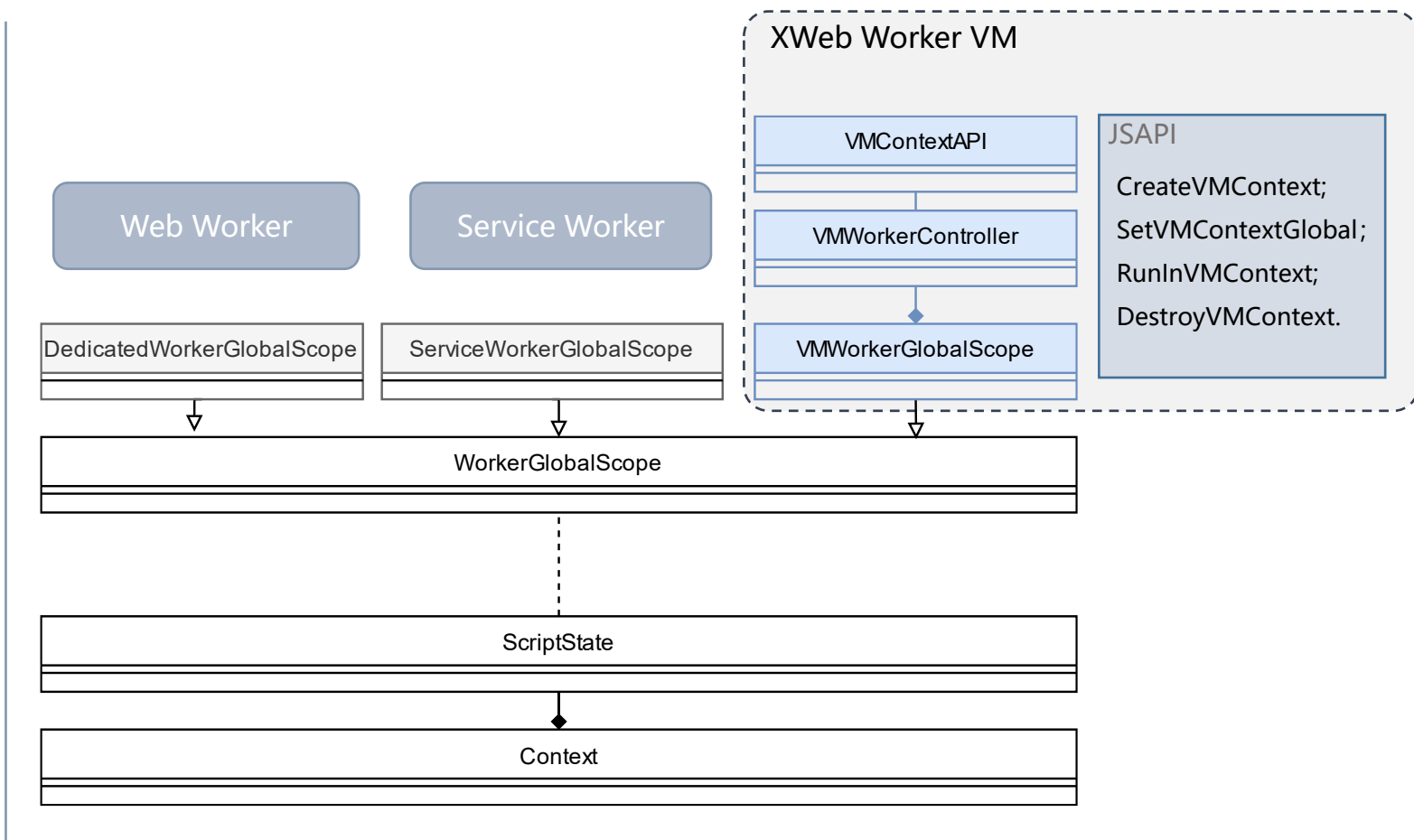


# 基于Chromium内核的小程序框架

## 沙箱能力支持

### 实现XWeb Worker VM

- 核心类 VMWorkerGlobalScope
- 导出 VMContextAPI 给JS使用



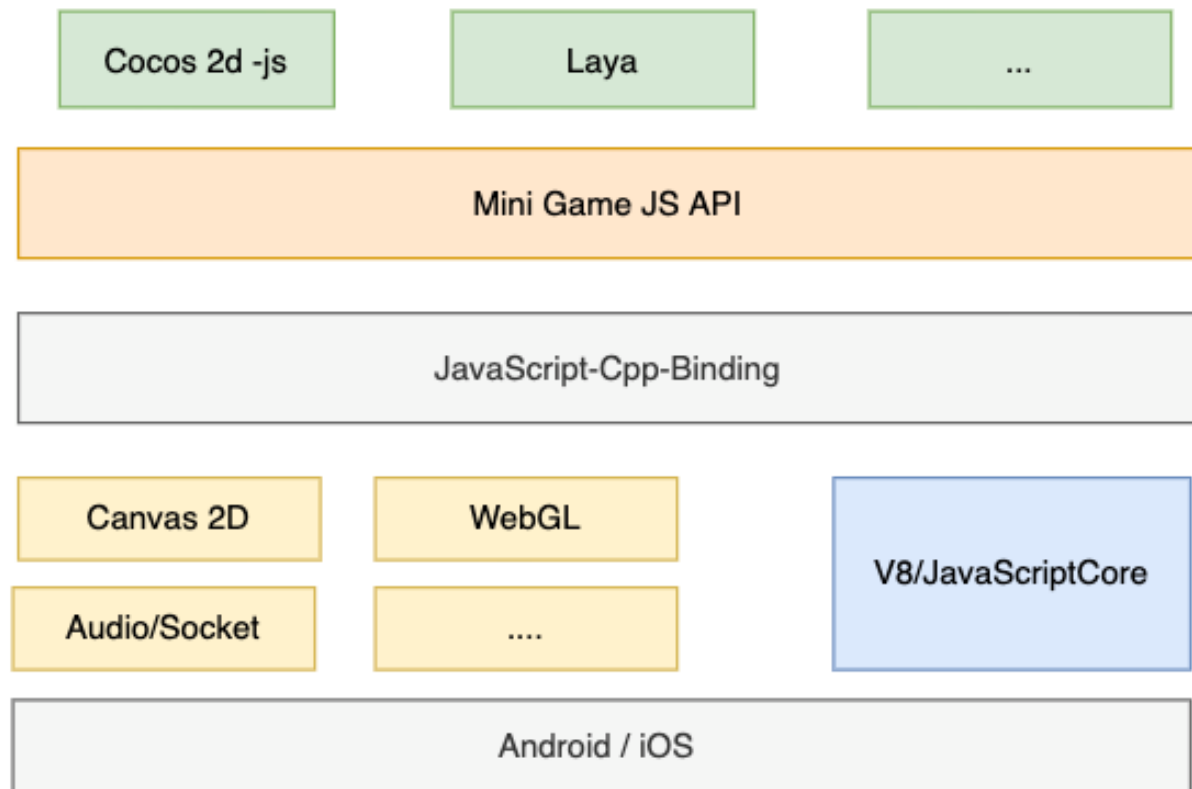


# 03

**基于Chromium内核的小游戏框架**

## 移动端小游戏框架介绍

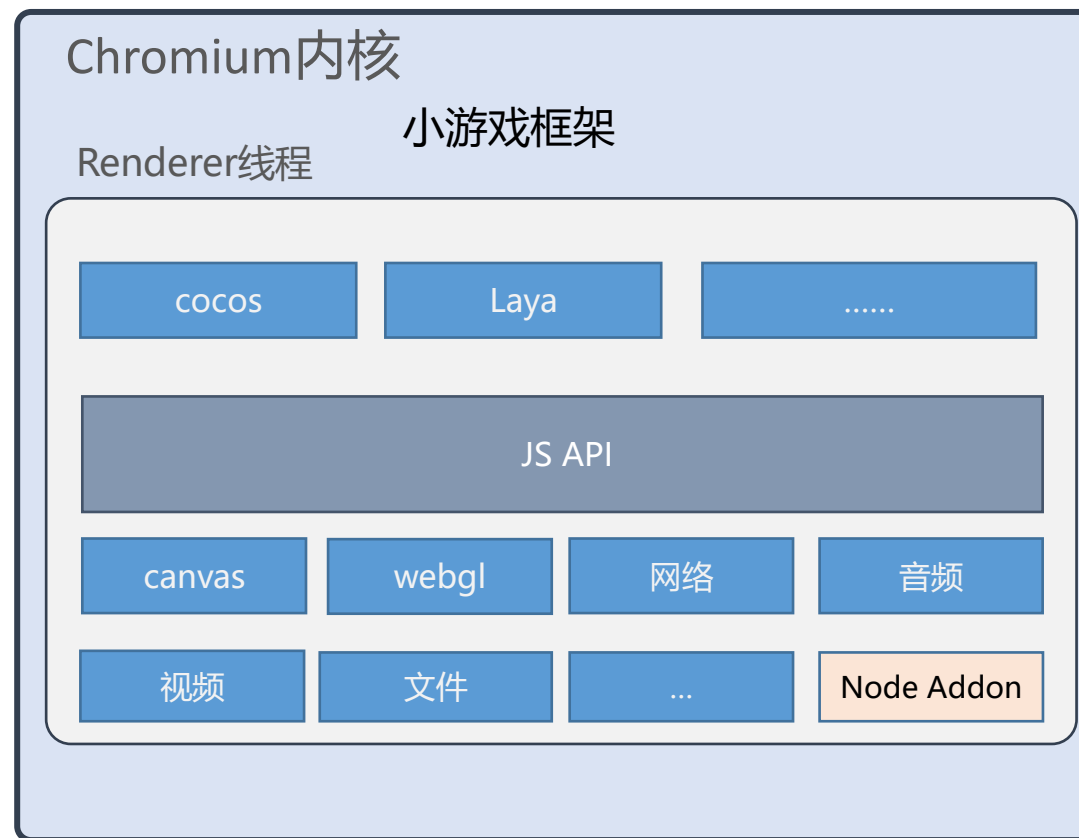
- 与Web内核无关
- JS运行环境：V8、JavaScriptCore
- OpenGL ES 实现 Canvas 2D 和 WebGL
- JS绑定，提供JSAPI



# | 基于Chromium内核的小游戏框架

## 基于Chromium内核的小游戏框架

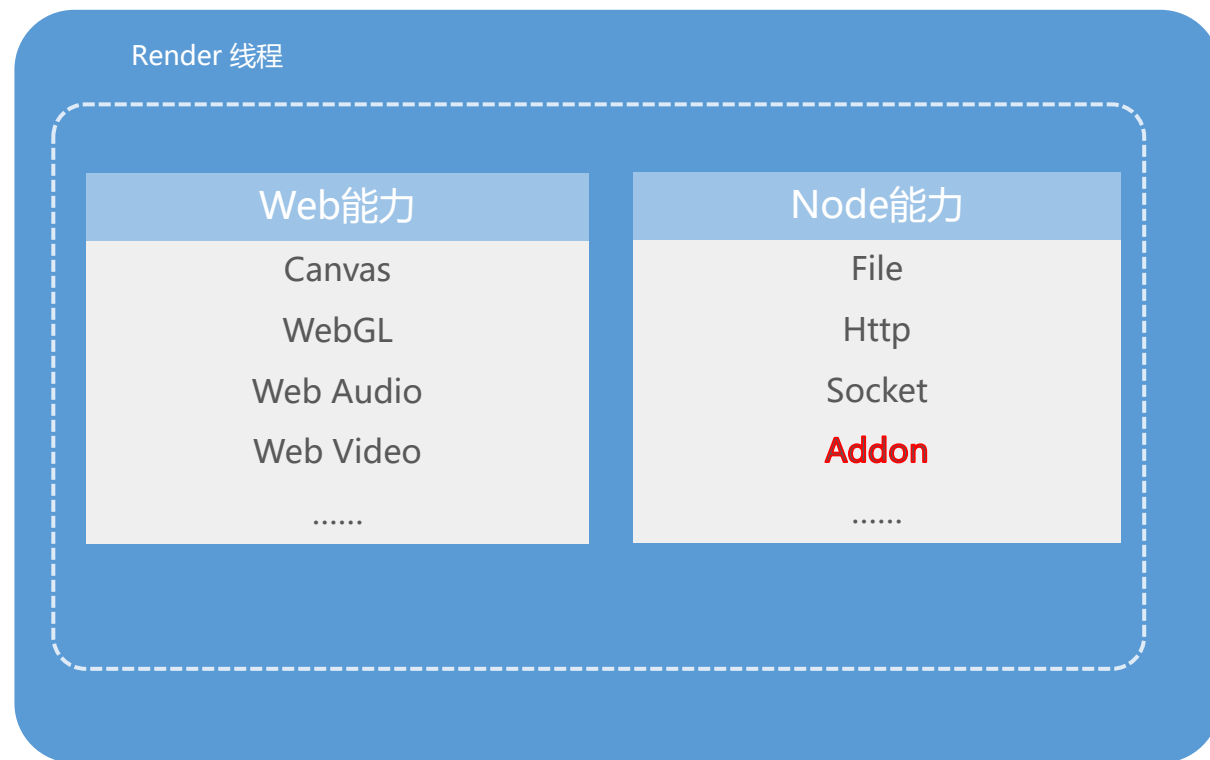
- 运行环境：Chromium内核
- Render线程，作为JS线程
- JSAPI: H5 API + Node



# 基于Chromium内核的小游戏框架

## JSAPI能力支持

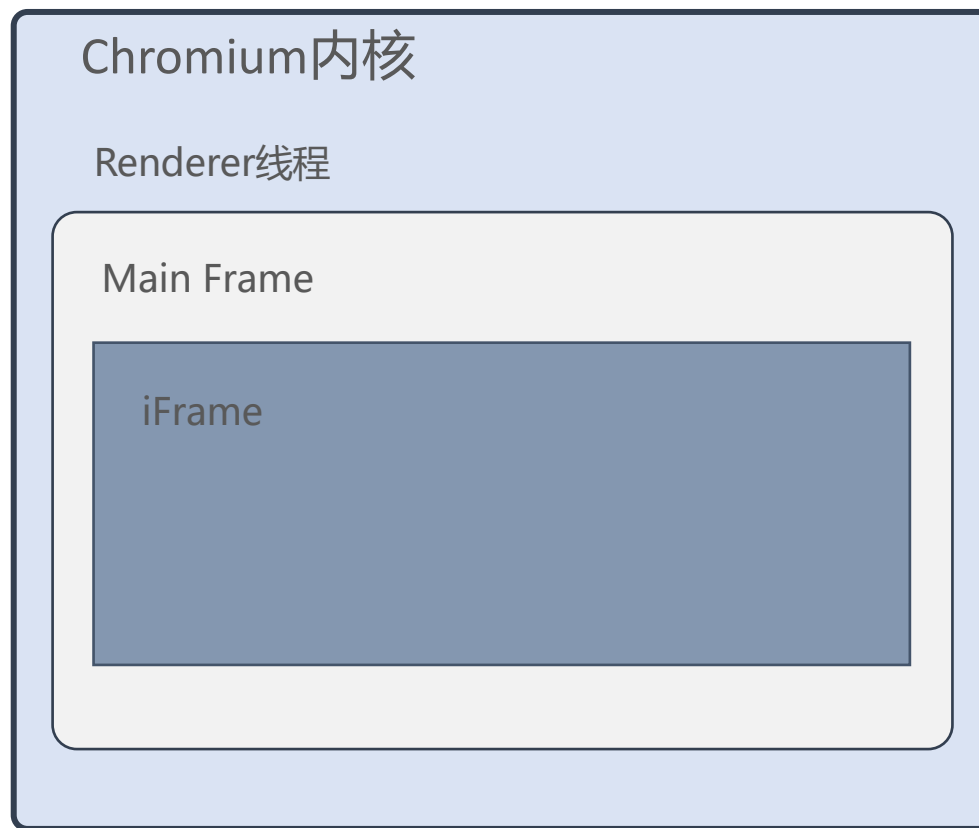
- H5 API 能力
- Node 融入到 Render 线程
- Node Api
- Node的 Addon 能力扩展



# 基于Chromium内核的小游戏框架

## 沙箱能力支持

- 使用Iframe提供独立的 js context;
- 修改Chromium内核, 禁用iFrame中的DOM和BOM接口





# 04

## 进程模型&跨平台实践

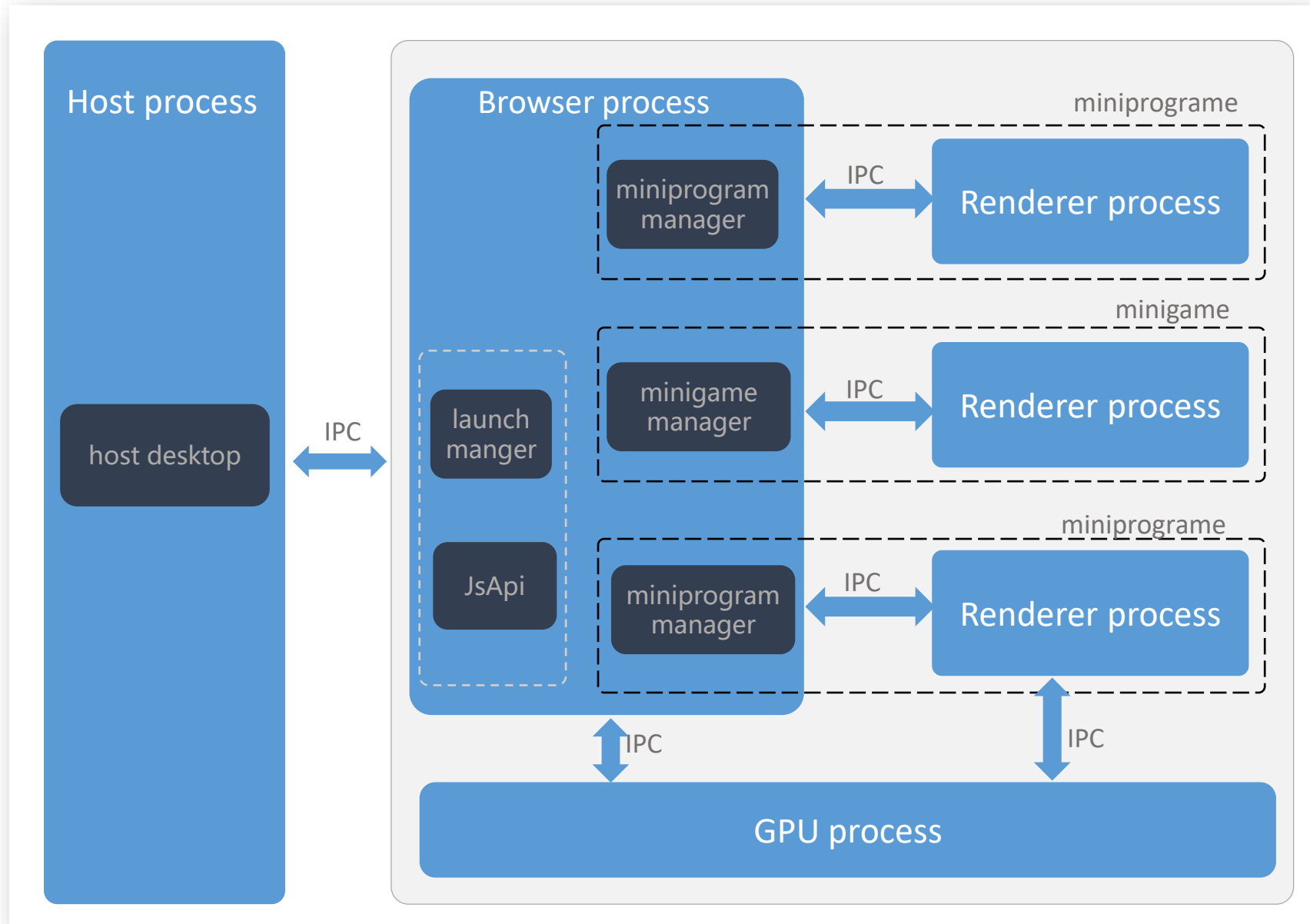


# 进程模型&跨平台实践

## 进程模型

- Host进程, PC微信
- Chromium 多进程模型
- Browser 进程
- Render进程: 小程序和小游戏
- IPC 通信

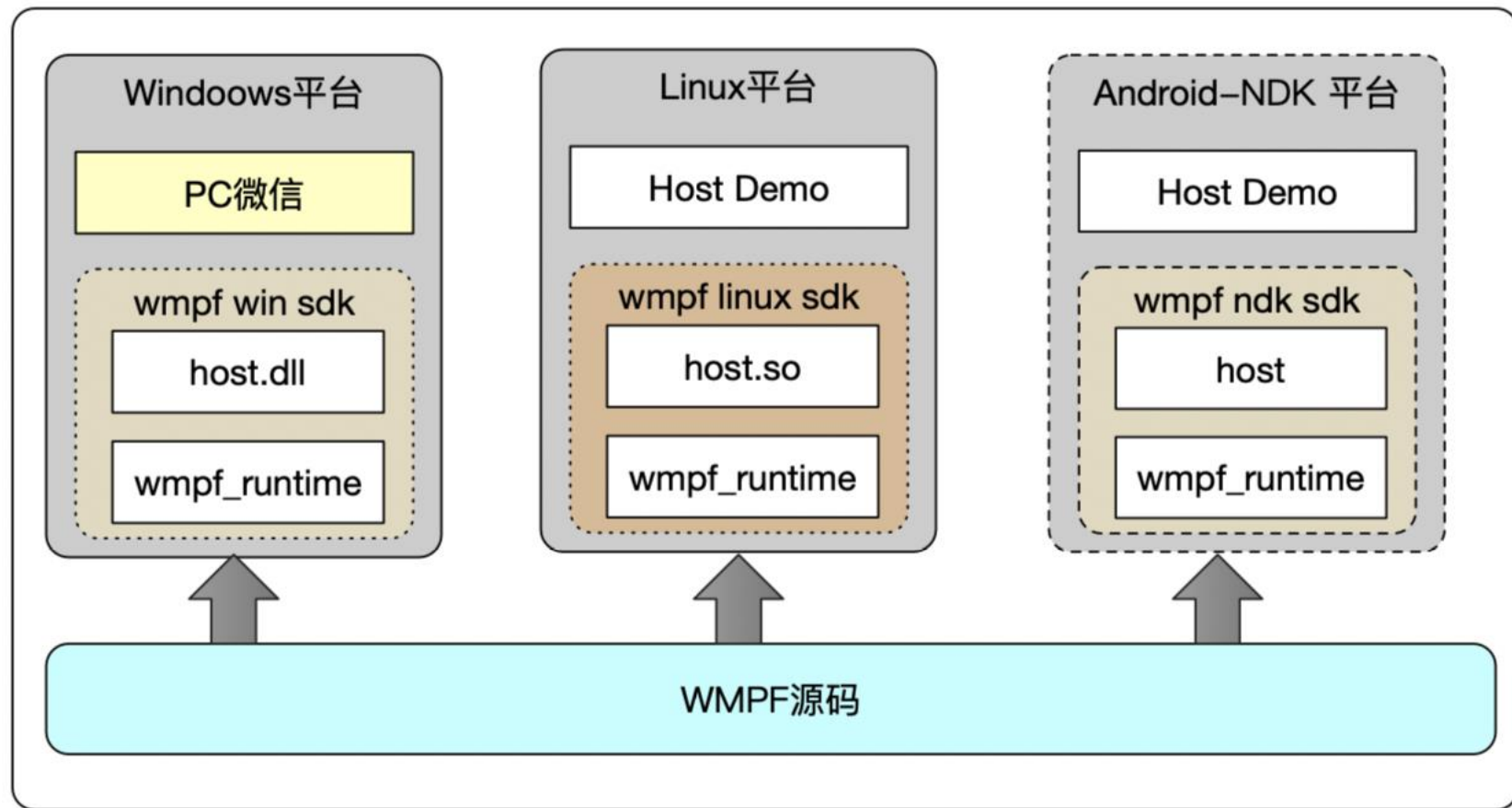
低耦合、高容错



# 进程模型&跨平台实践

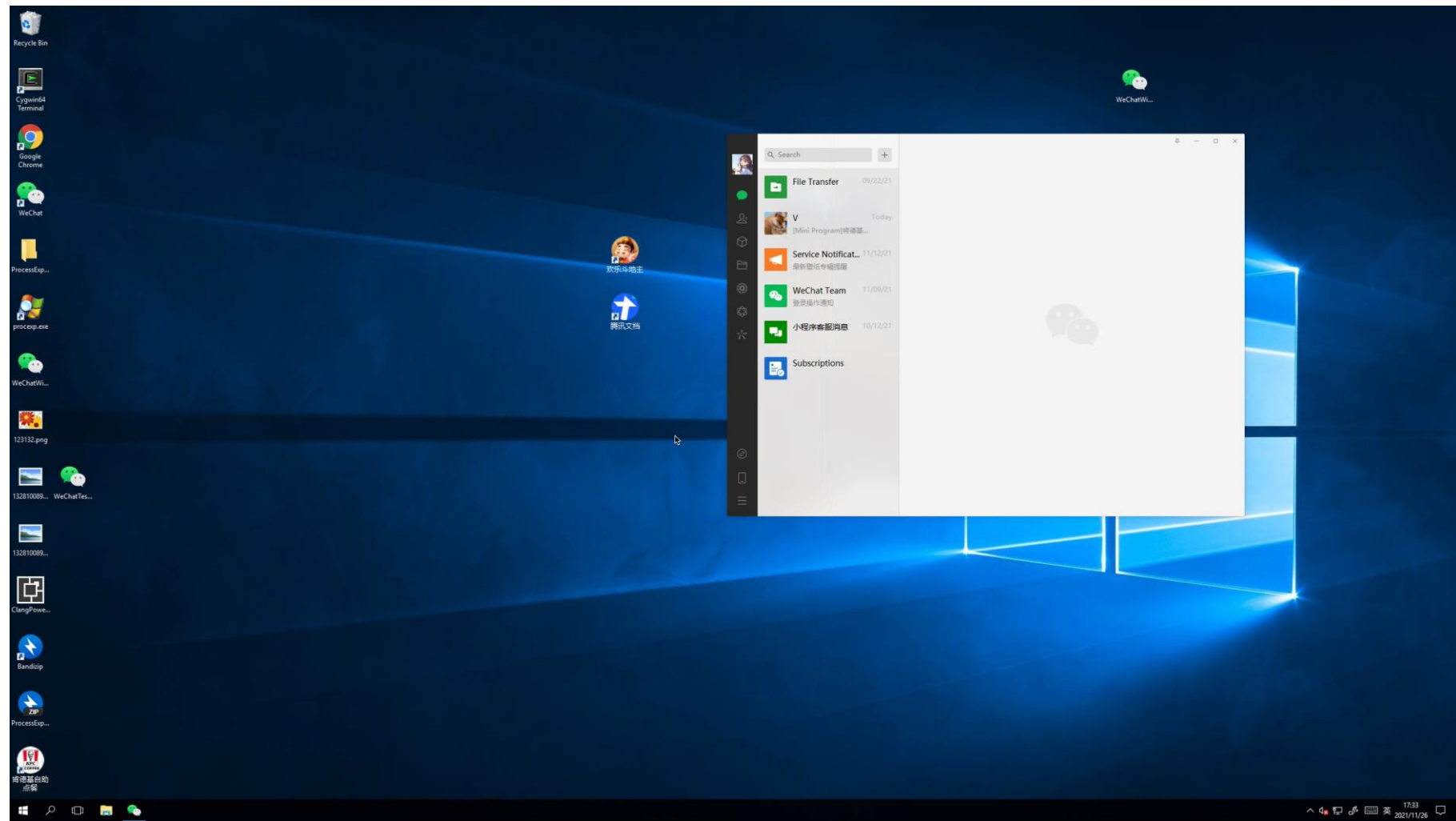
## 跨平台SDK

- Chromium跨平台特性
- 多个平台，使用同一份源码
- SDK 可独立使用
- PC 微信集成



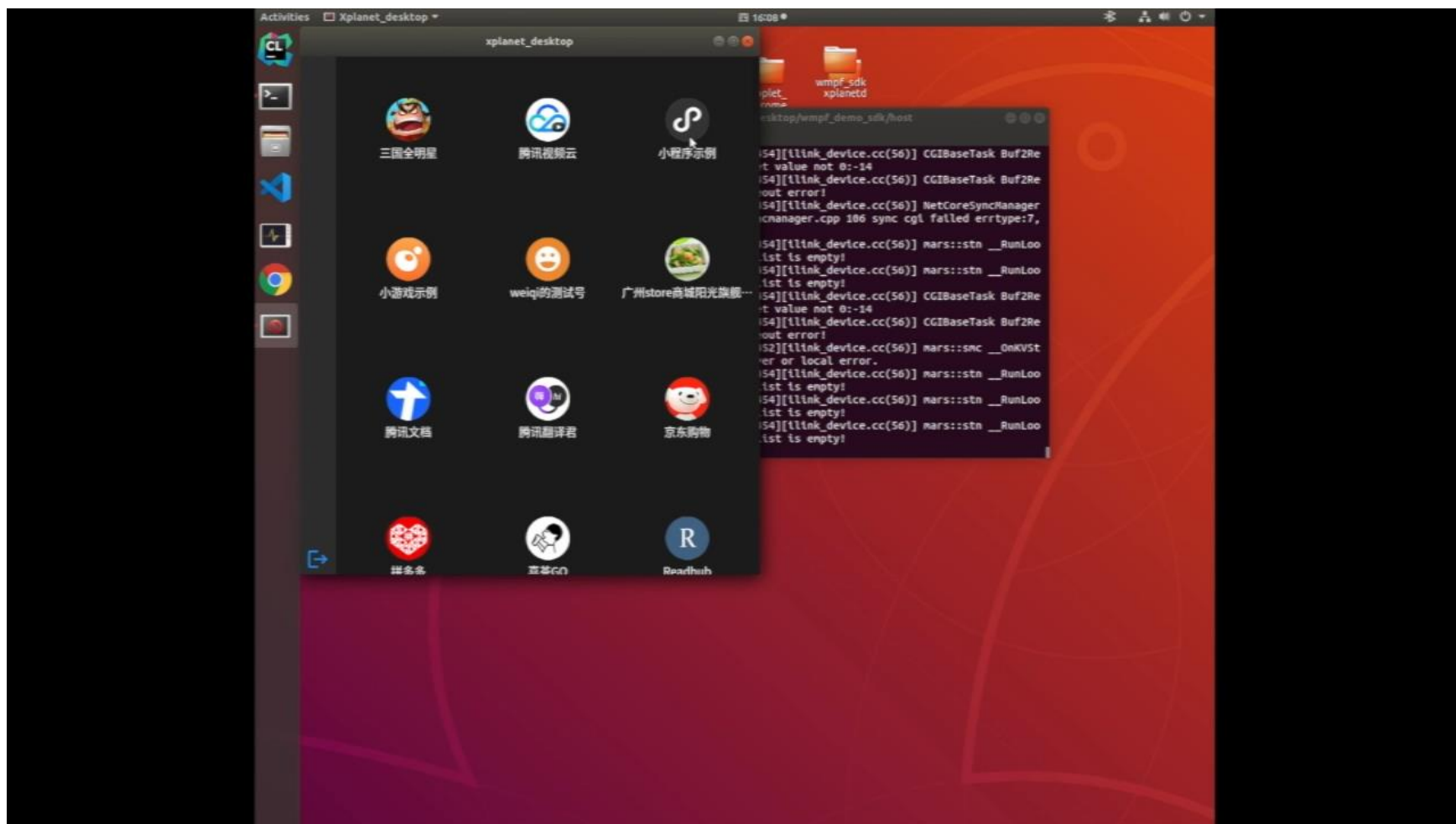
# 进程模型&跨平台实践

## Windows平台演示



# 进程模型&跨平台实践

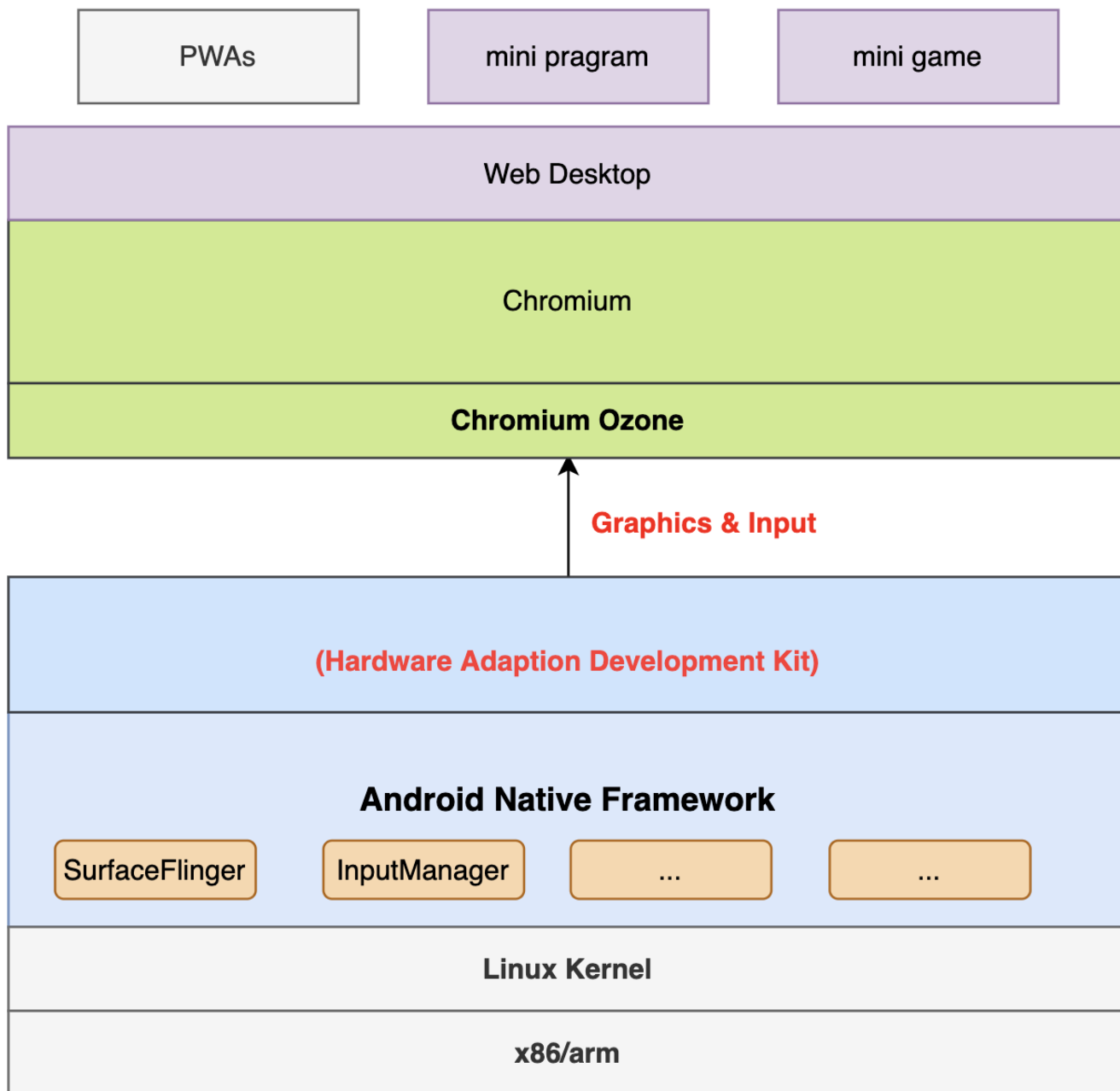
## Linux平台演示



# 进程模型&跨平台实践

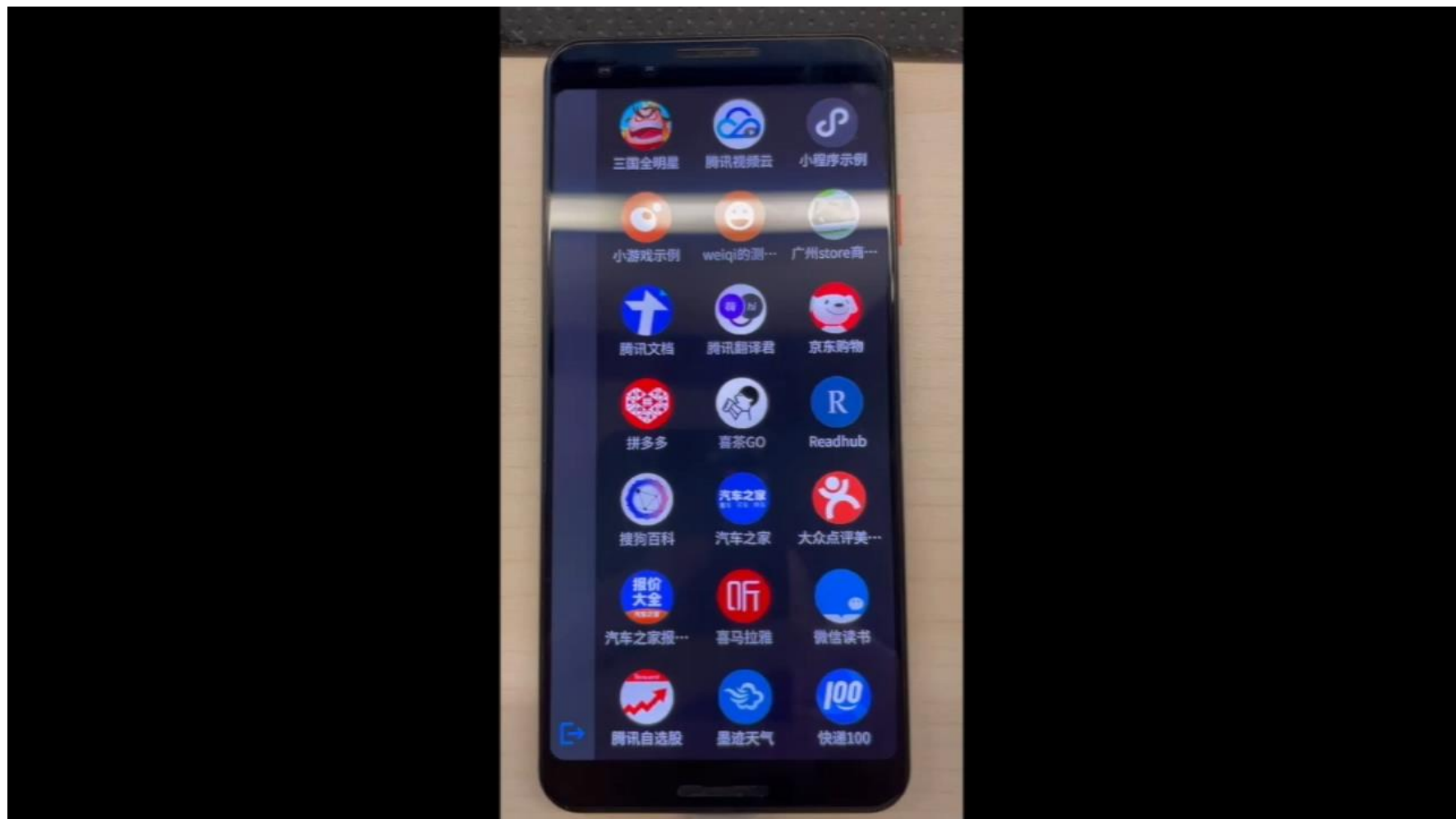
## Android NDK 平台

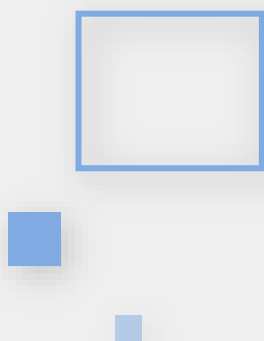
- 不依赖Android Java代码
- NDK工具链编译Chromium
- Android Native Framework
- SurfaceFlinger 和 Input Manager
- 实现 Chromium Ozone



# 进程模型&跨平台实践

## Android-NDK平台演示





Thank You