



Dokit For 小程序

王川

目录

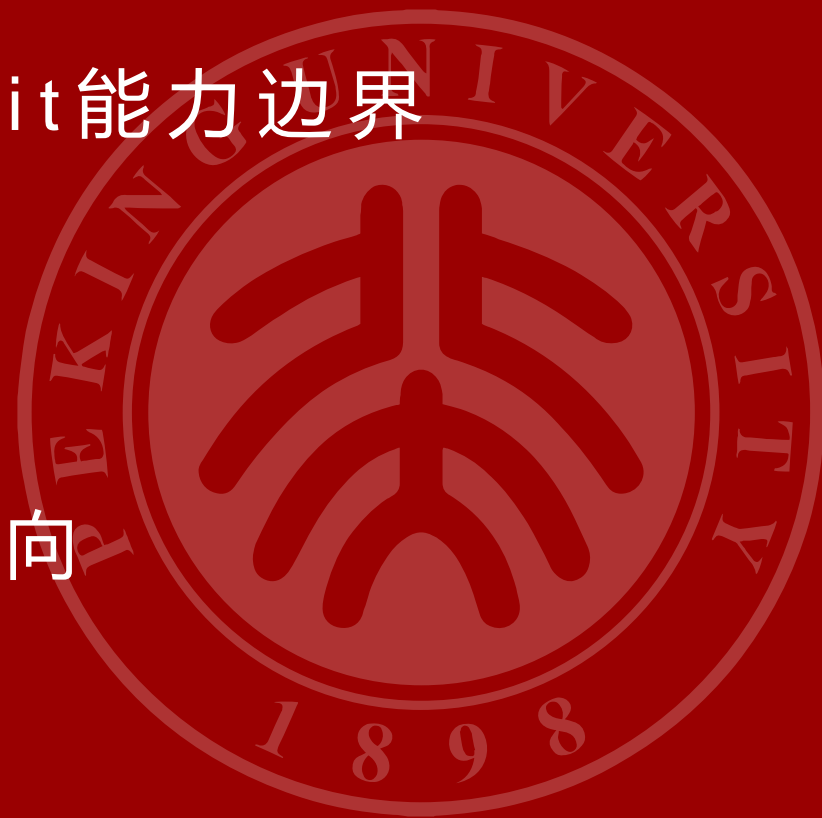
CONTENTS

01 小程序dokit能力边界

02 源码和思路

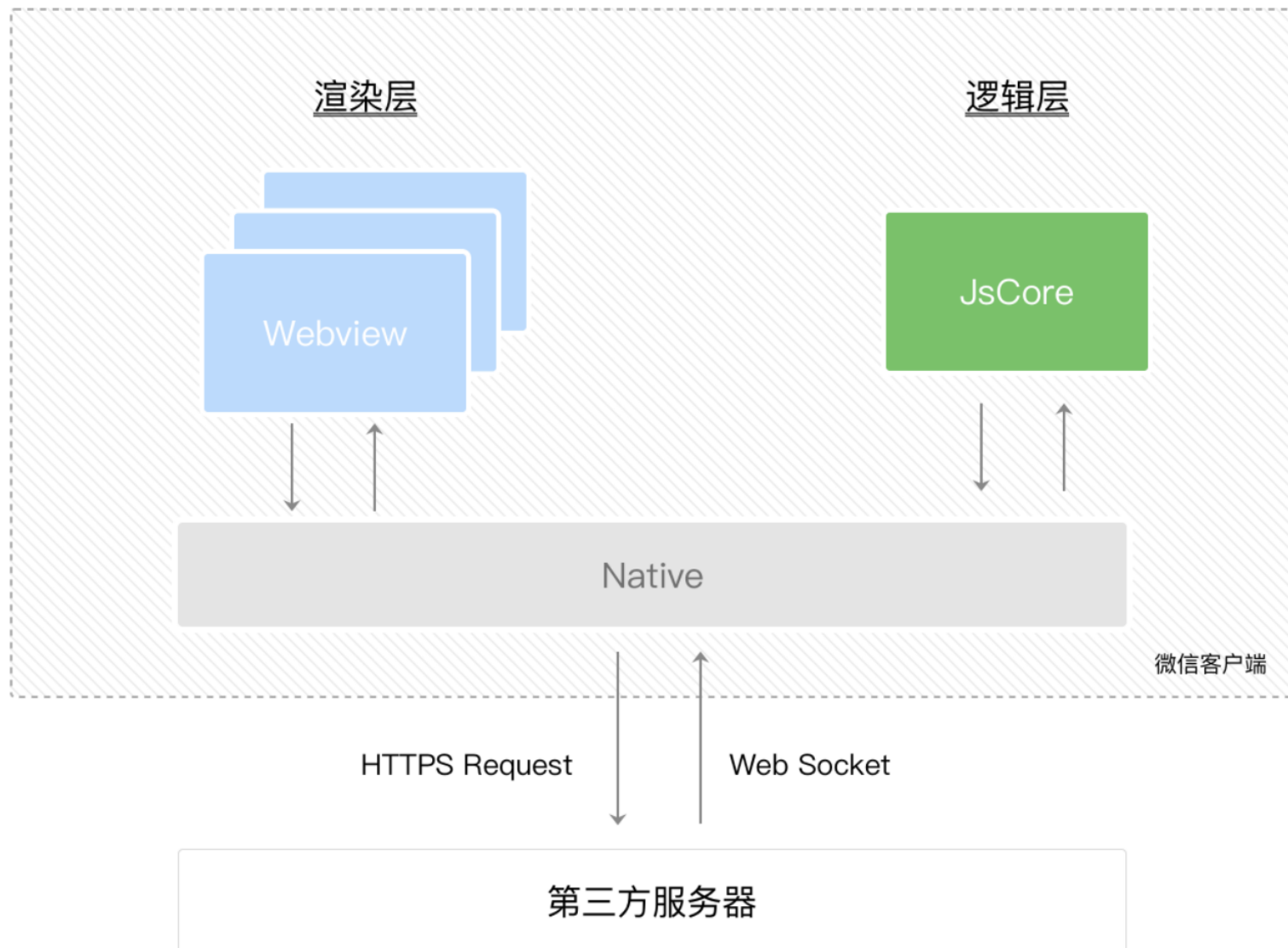
03 后续迭代方向

04 课后作业



Part 01 能力边界

小程序系统架构 —— 双线程模型



回顾双线程模型由来

使用web技术
完成小程序渲
染工作



web技术本身
非常开放和
灵活，进而
带来了风险



提供单独的线程用于
执行JS，并且屏蔽了
部分浏览器接口

微信App

小程序基础库

Dokit-miniapp



Part 02 源 码 及 思 路

1. 理解Object.defineProperty()

- 参考链接: https://developer.mozilla.org/zh-CN/docs/Web/JavaScript/Reference/Global_Objects/Object/defineProperty

2. 浏览微信小程序API

- 参考链接: <https://developers.weixin.qq.com/miniprogram/dev/api/>

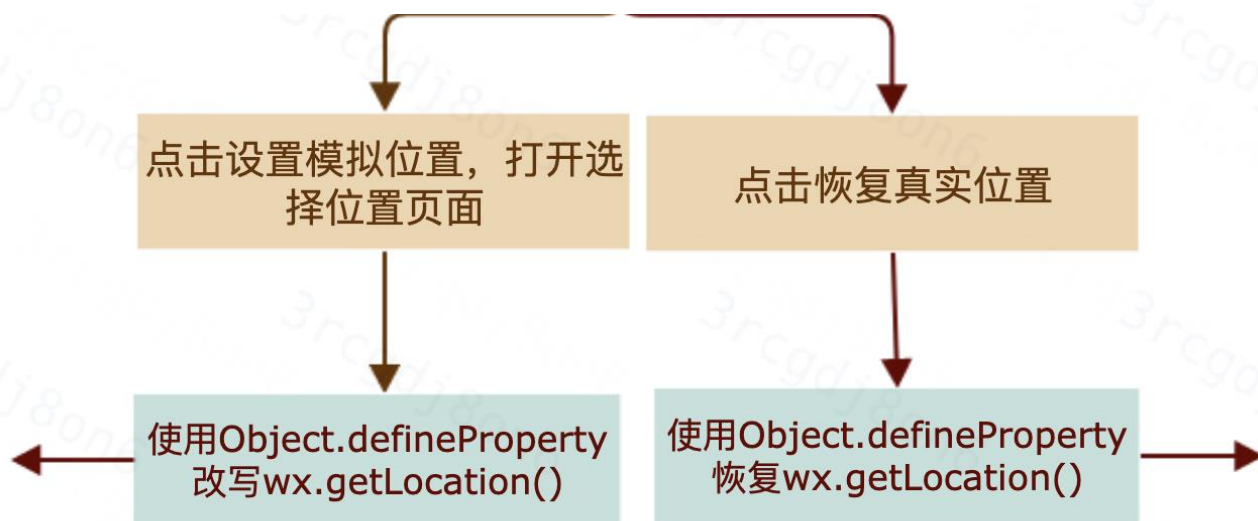
3. 了解组件的生命周期

- 参考链接:
<https://developers.weixin.qq.com/miniprogram/dev/reference/api/Component.html>

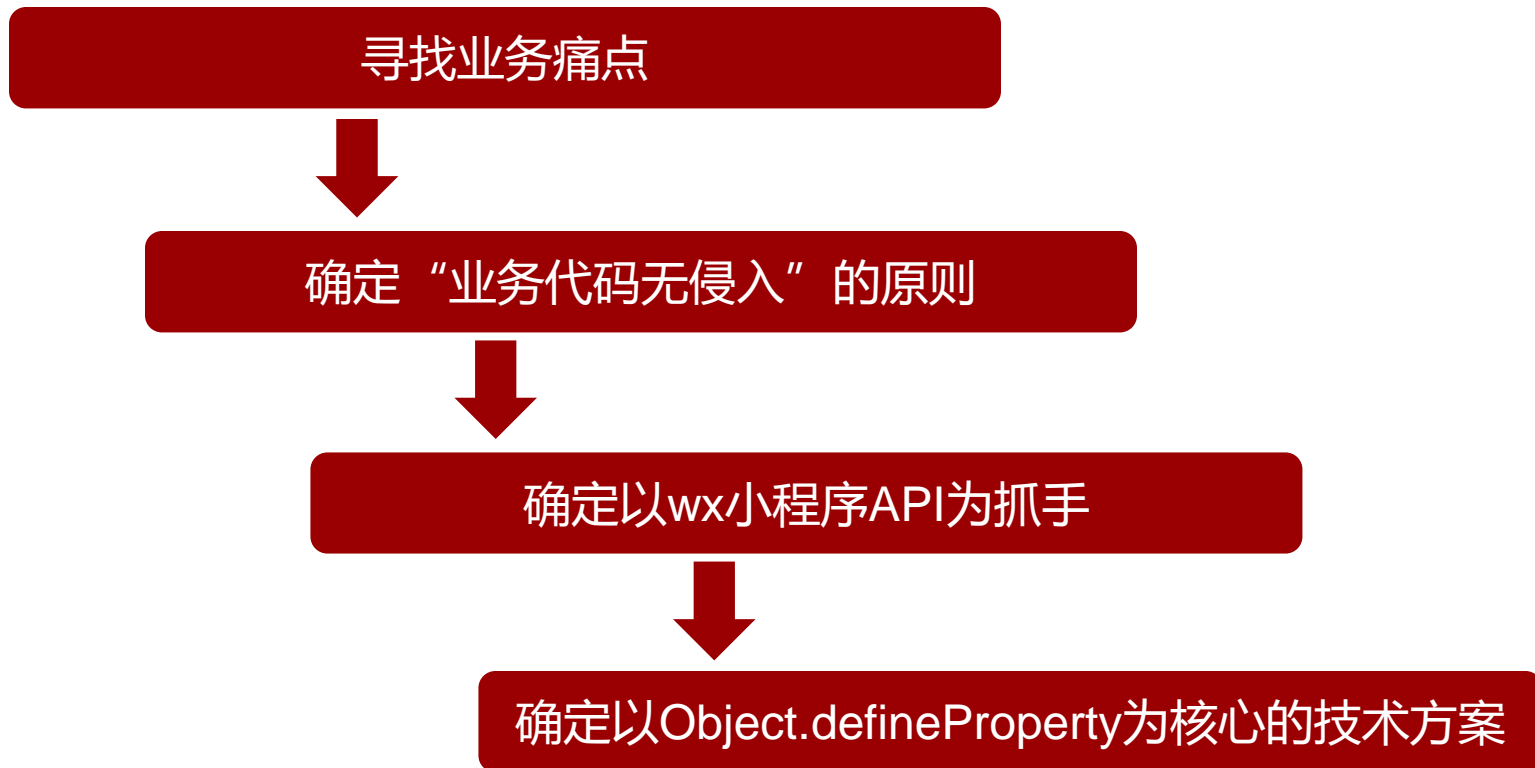
demo-位置模拟

```
wx.chooseLocation({
  success: res => {
    this.setData({ currentLatitude: res.latitude });
    this.setData({ currentLongitude: res.longitude });
    Object.defineProperty(wx, 'getLocation', {
      get(val) {
        return function (obj) {
          obj.success({latitude: res.latitude, longitude: res.longitude})
        }
      }
    })
  }
})
})
```

```
const app = getApp()
app.originGetLocation = wx.getLocation
wx.getLocation({
  type: 'gcj02',
  success: res=> {
    this.setData({currentLatitude:res.latitude}),
    this.setData({currentLongitude:res.longitude})
  }
})
```

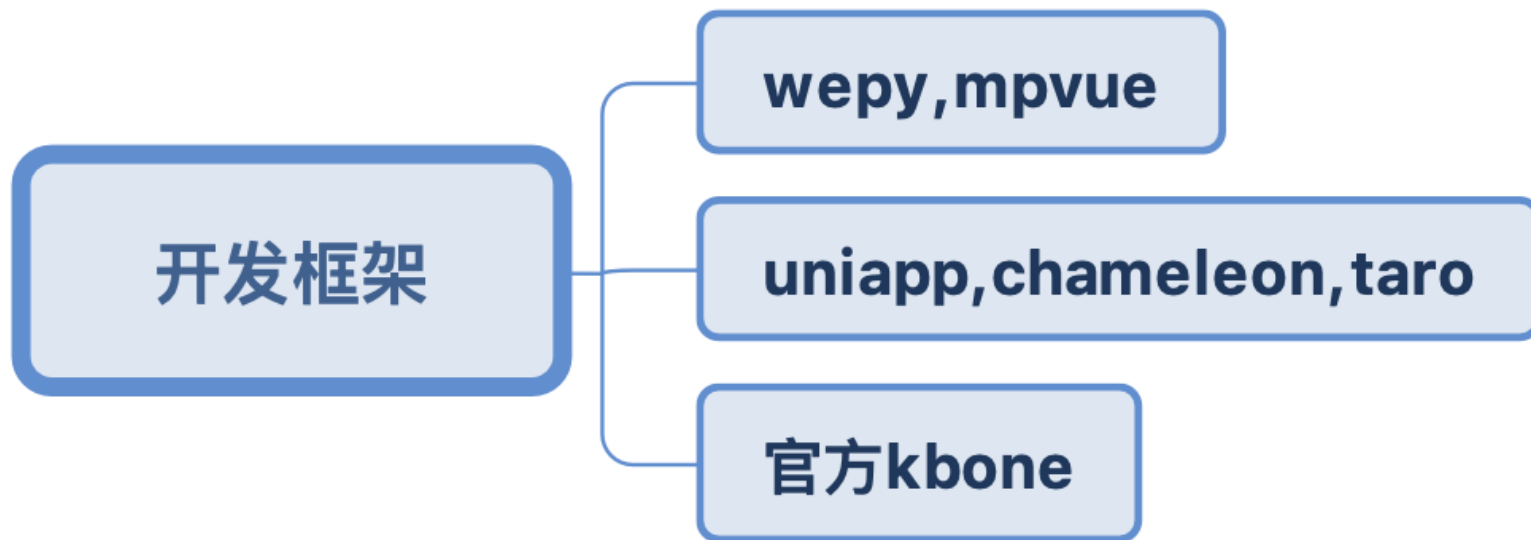


```
Object.defineProperty(wx, 'getLocation',
{
  get(val) {
    return app.originGetLocation
  }
});
```



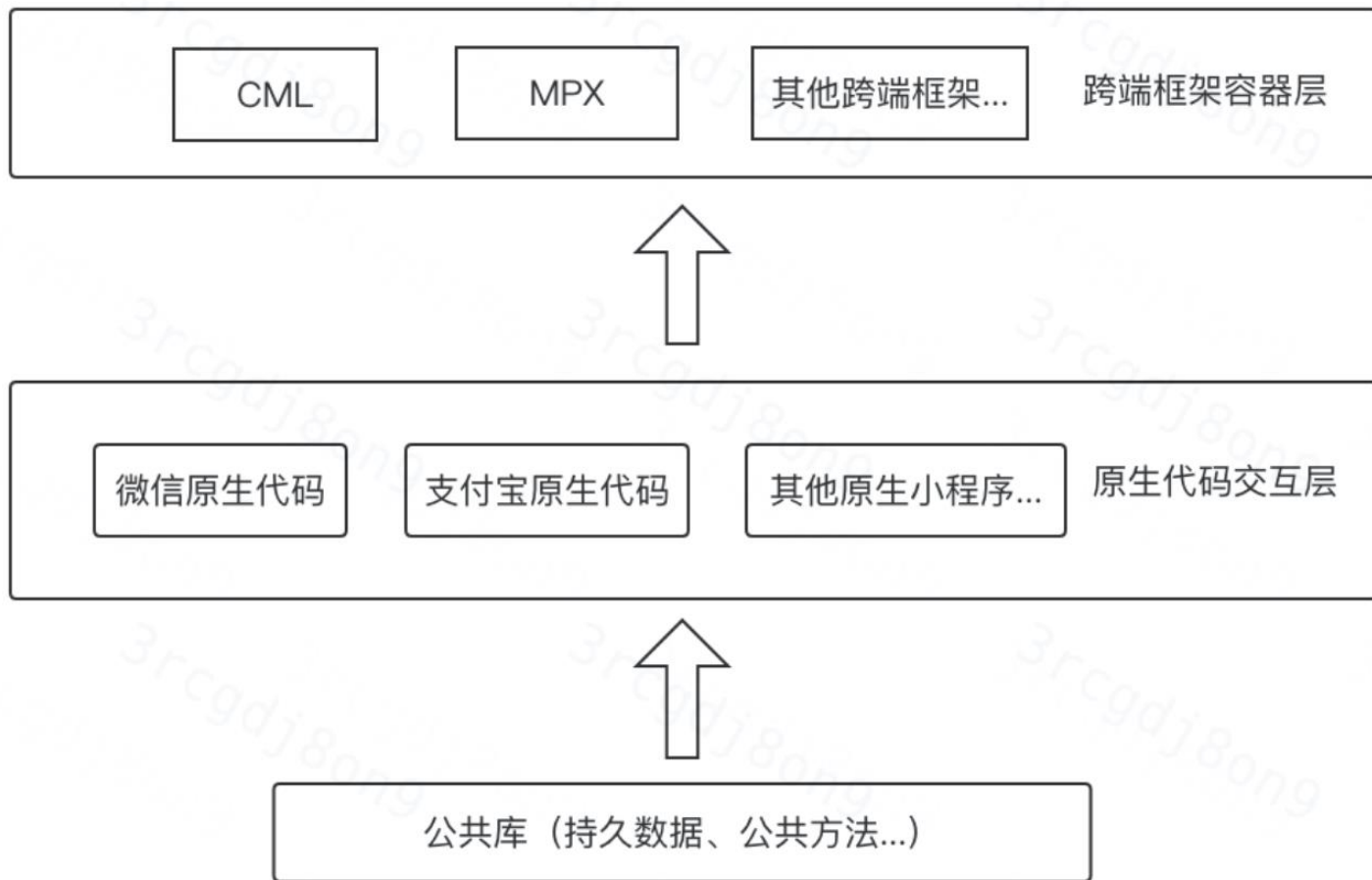


Part 03 后续迭代方向



2.0架构设计

目的：为满足更加丰富的引入方式



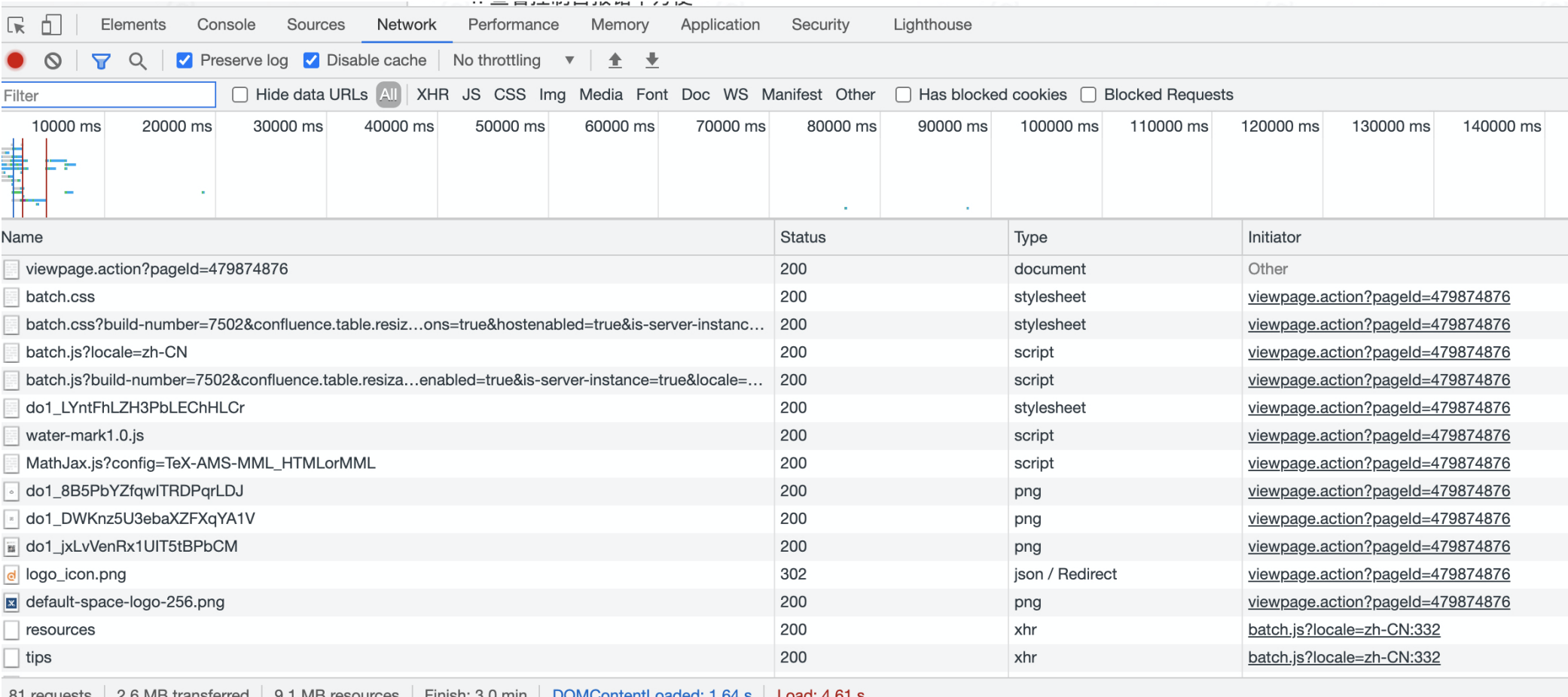
dokit-mp

- dokit-mp-cml
 - src
 - components
 - index.cml
 - package.json
- dokit-mp-mpx
 - src
 - components
 - index.mpx
 - package.json
- dokit-mp-wx
 - src
 - components
 - network(功能组件)
 - index
 - package.json
 - dokit-mp-alipay
- dokit-mp-api
 - src
 - utils
 - index
 - package.json
 - package.json



Part 04 课 后 作 业

在现有dokit源码基础上实现类似chrome devtool中的network功能，用于监测单个小程序的请求发送情况



Q&A