



机械工业出版社
华章分社

8月27日 | 49届前端早早聊

无/低代码 资本热宠 搭建卷城

搭建原理
低代码实践
业务适配
提效流程
生产力工具



2022 全年行程 ▶

1/29	2/26	3/19	4/9	4/23	5/14	5/28	6/18	6/25	7/16	7/17	8/6	8/27	9/3	9/17	9/24	10月	10月	11月	11月	12月	12月	12月
前端职业前路	TS技术专场	管理专场	跳槽面试专场	组件专场	监控专场	性能专场	成长与晋升	跨端专场	海外工作专场	一站式基建	工程化专场	低代码专场	低代码微前端	音视频专场	创业专场	开发语言专场	保养健康专场	自由职业专场	框架专场	运维专场	鸿蒙专场	可视化专场

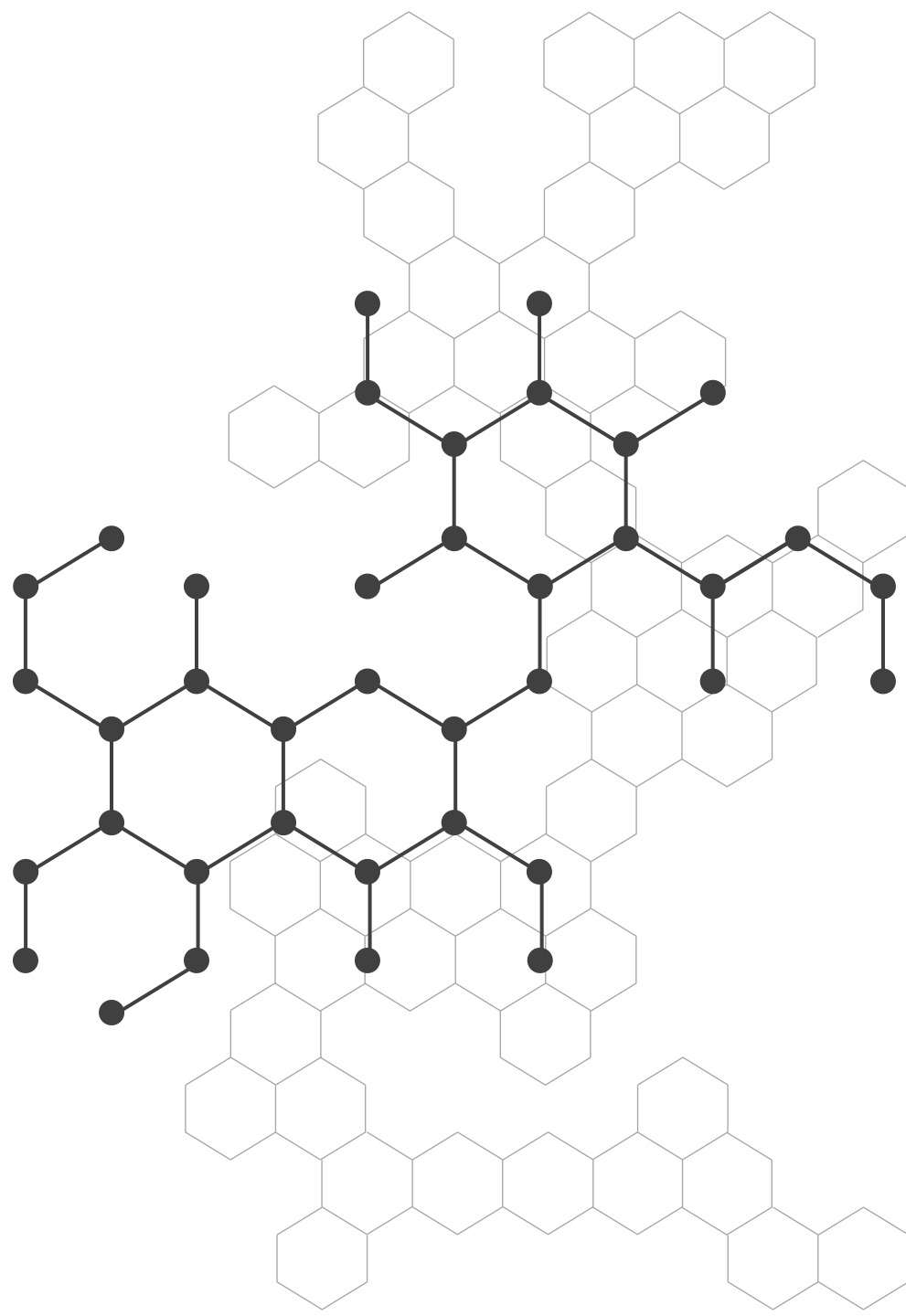
09:00	编排:《从界面到逻辑-低代码逻辑编排的思考和实践》	姜天意	腾讯 CSIG-医疗健康事业部-前端负责人
10:00	商业:《低代码体系组合拳:最低成本实现商业化》	谭景琳	销帮帮 CRM 业务架构组-前端架构师
11:00	应用:《低代码企业应用的建设与实践》	钟柳	兴盛优选 前端工程师
12:00	开发流:《营销场景下的低代码建设》	周颖 Jojo	自由开发者
13:00	业务:《复杂业务场景下的低代码实践》	鱼鱼	政采云 共享平台前端团队-低代码平台前端负责人
14:00	营销:《字节游戏营销平台低代码应用实践》	谢军令	字节跳动 游戏前端-营销平台负责人
15:00	工具:《跨境 B2B 导购搭建工具》	高凯	行云集团 体验技术团队-前端专家
16:00	智能:《智能生成: CODE.FUN 一键从设计稿到源代码》	杨帆	CODE.FUN CODE.FUN 创始人
17:00	UI:《榫卯:可扩展的低代码工具开发框架》	余彦臻	SmartX 管理平台与内部效率团队-研发经理
18:00	中后台:《如何利用低代码快速高效搭建中后台页面》	小虫	芒果TV 会员产品技术部-前端工程师
19:00	方案:《低代码:基于 React 和 Mobx 的实现》	千瑜	Akulaku 项目8线-前端专家
20:00	智能:《低码化与智能化融合》	文子穰	兴盛优选 兴盛优选体验技术部-低码引擎智能化负责人
21:00	UI:《基于现有 UI 框架快速落地低代码框架》	王浩	善诊 善诊研发部-高级前端开发
22:00	引擎:《B 端页面的低代码引擎之路》	吕洋	腾讯 腾讯视频-高级前端工程师

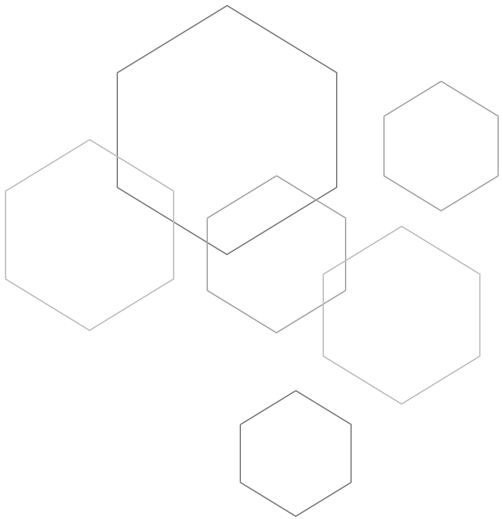
低代码企业应用的建设与实践

Construction and practice of low code enterprise applications

兴盛优选研发信息中心/UXT体验技术部

主讲人：钟柳





目录

Catalog

产品简介

Product introduction

设计模式

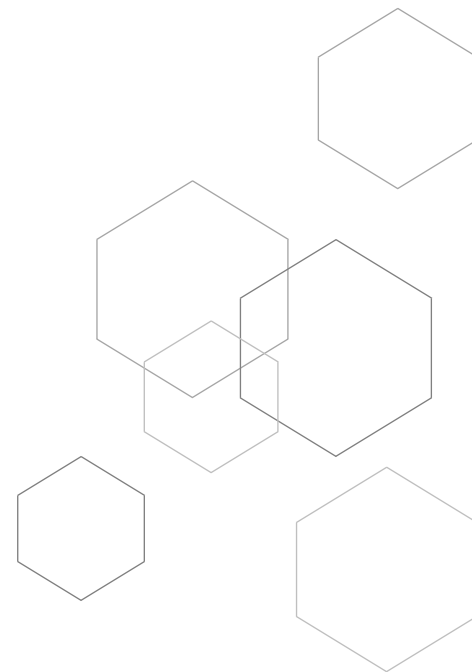
Design patterns

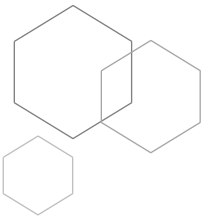
业务支持

Business support

数据卡片

Data Card





产品简介

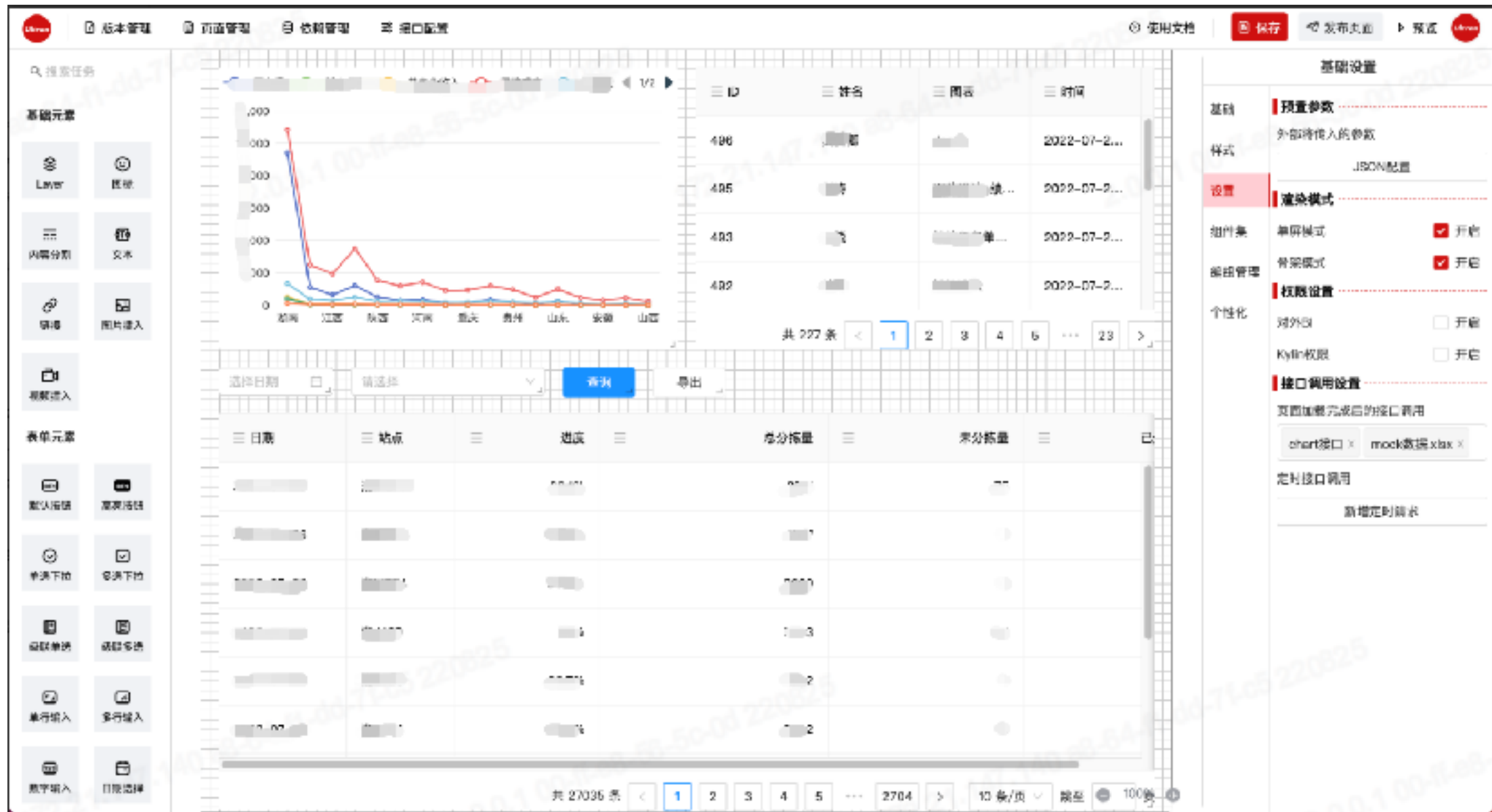


兴盛优选——奥创低代码搭建平台

奥创是兴盛优选研发信息中心UXT体验技术部数据中台团队自主研发出品的一款低代码搭建平台。是集PC端、移动端可视化搭建为一体的服务化系统，提供了方便，快捷的多类型、强兼容性的图表、表单、布局等组件，能高效、高频、稳定的满足企业内部各业务线中各业务的开发，维护与展示工作，同时能提供面向开发，测试，生产的多环境配置。通过微前端技术实现的数据卡片SDK也能极大程度的提效业务开发，友好的拥抱开发人员。



网页应用



网页应用



ID	姓名	图表	时间
496	张超	柱状图	2022-07-27T04:4...
495	李强	折线图-绩效统计	2022-07-27T03:3...
493	王明	饼图-销售分布	2022-07-26T06:1...
492	赵刚	仪表盘	2022-07-26T02:3...
491	陈超	热力图	2022-07-25T13:17...
490	孙强	散点图-用户行为	2022-07-23T17:12...

共 227 条 < 1 2 3 4 5 ... 23 > 10 条/页 页 页

选择日期

情这样

毒物

导出

日期	站名	进度	总分数量	未分数量	已分数量
2023-07-26	100009	100%	100	0	100
2023-07-26	100012	100%	100	0	100
2023-07-26	100064	100%	100	0	100
2023-07-26	10039	100%	100	0	100
2023-07-26	10008	100%	100	0	100
2023-07-26	100077	100%	100	0	100
2023-07-26	100051	100%	100	0	100
2023-07-26	100010	100%	100	0	100

共 27035 条 < 1 2 3 4 5 ... 2704 > 10 条/页 排序 页

移动应用



4:15

大 > 刷新

兴盛优选高薪招聘
快来加入我们吧!

个人姓名* 请输入个人姓名

手机号* 请输入手机号码

身份证号* 请输入身份证号

年龄* 请输入真实年龄

性别* ☐ 男 ☐ 女

入职时间* 请选择入职时间

提交

< > 分享 收藏 打印

移动应用



移动应用



4:15

大

兴盛优选高薪招聘
快来加入我们吧!

个人姓名* 请输入个人姓名

手机号* 请输入手机号码

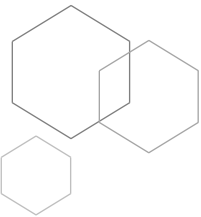
身份证号* 请输入身份证号

年龄* 请输入真实年龄

性别* ☐ 男 ☐ 女

入职时间* 请选择入职时间

提交

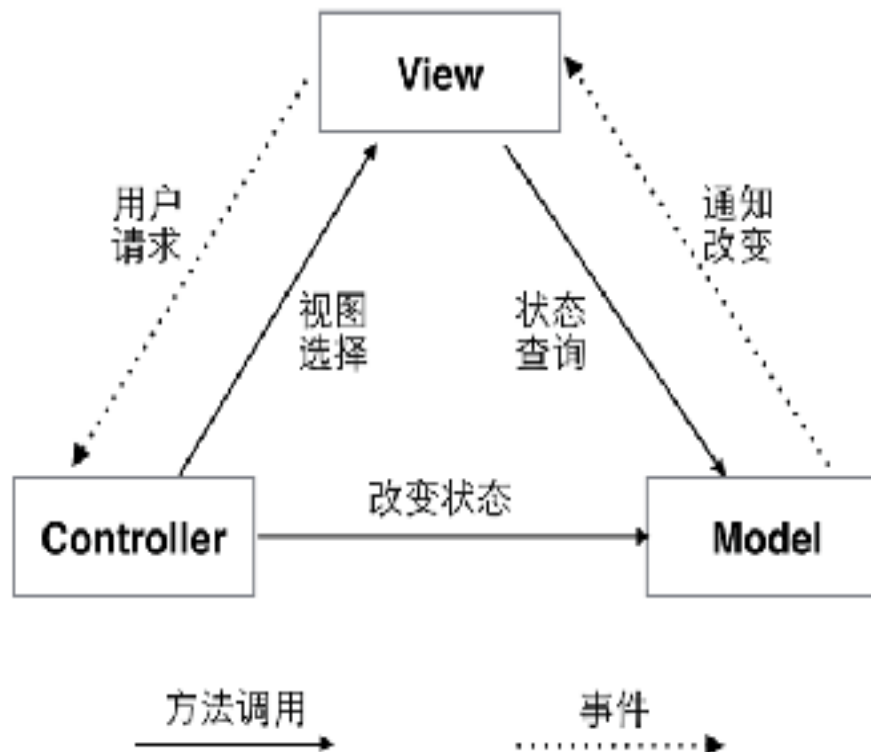


奥创低代码平台的设计模式——MVC

MVC的全称是Model View Controller，是模型(Model)－视图(view)－控制器(controller)的缩写。它是用一种业务逻辑、数据与界面显示分离的方法来组织代码，将众多的业务逻辑聚集到一个部件里面，在需要改进和个性化定制界面及用户交互的同时，不需要重新编写业务逻辑，达到减少编码的时间，提高代码复用性。

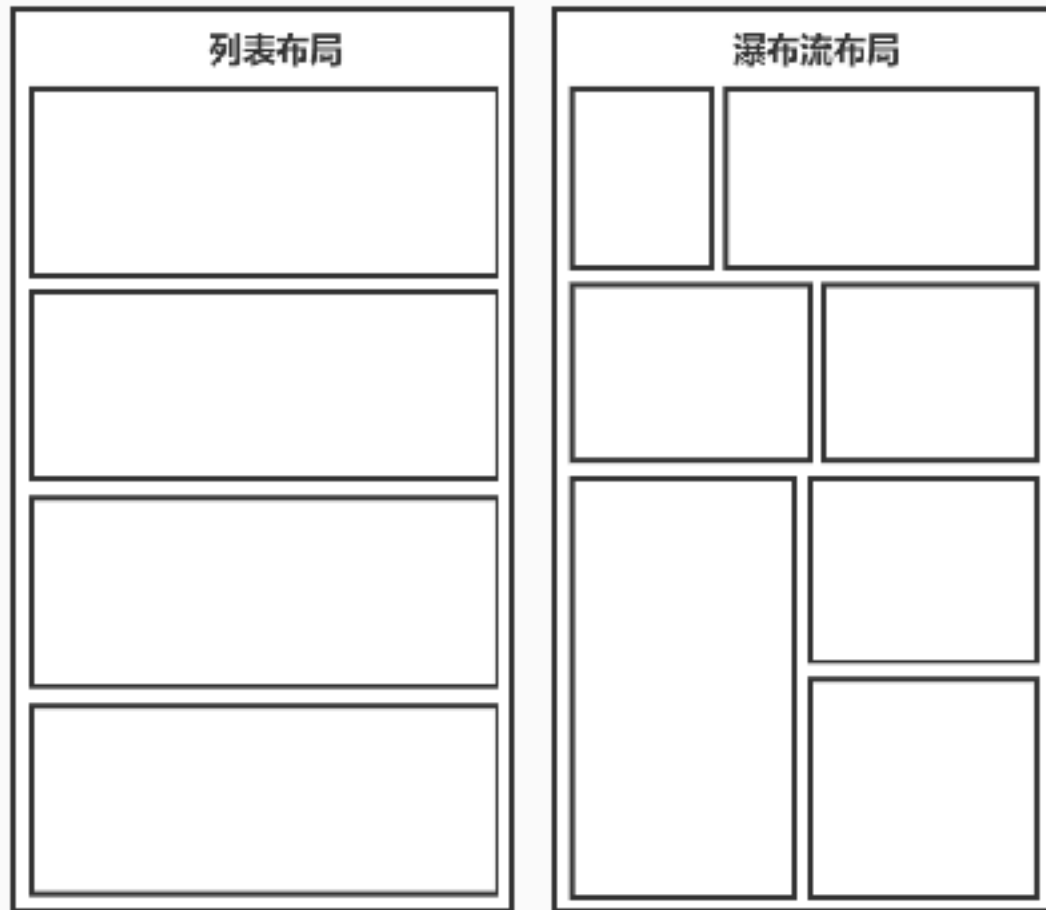
奥创低代码平台设计之初为什么选用MVC?

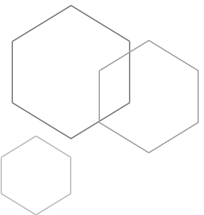
最根本的原因是我们团队对于低代码平台的核心认知就是“**让用户可视化处理业务逻辑、数据交互、界面布局**”



奥创视图层（View）设计

视图层引擎是奥创处理组件布局、屏幕适配、动画的核心，它能保证各组件在不同大小屏幕上的正常显示。我们参考借鉴了iOS Storyboard的布局思路，对于页面的整体布局我们以Cassowary算法为基础自主实现了一套应用于前端低代码场景的Autolayout自动布局逻辑，借助这套布局逻辑我们能实现低代码平台中80%的布局应用场景。再结合大量Web端的各类组件特性，我们辅以实现列表布局、瀑布流布局、Grid布局等这类特殊布局容器来应对剩余的特殊布局场景。





视图布局



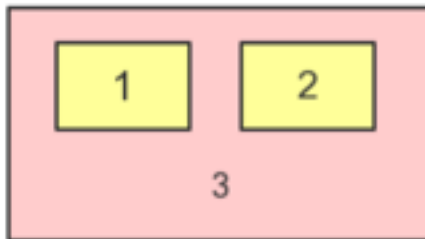
什么是Autolayout?

对于任何一个组件可视单元，确定了它的（相对于父view）位置、大小也就确定了它在整个UI视图中的展示效果。Autolayout就是利用约束和参照物来控制视图的大小和位置，系统会在运行时根据你设置的约束进行计算得到具体的Frame（x，y，width，height）再绘制到屏幕。

计算公式：

$$\text{Component_2_left} = \text{Component_1_right} + \text{constant}$$

(x, y, width, height)



视图布局

为什么选择Autolayout?

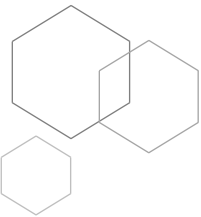
为了解决组件可视单元或者元素是怎样布局、排列的问题。屏幕尺寸会有多种可能的,这就要求我们的组件元素需要具备在不同屏幕尺寸的情况下具有动态可调性,以便于实现较好的UI展示效果。

Autolayout的数据结构

```
▼ autolayout:
  ▼ aligns: Array(1)
    ▼ 0:
      align: "middle"
      componentId: null
      key: "oVnQgxfCZpvcPl442KiIg"
      offset: 0
      offsetPosition: "top"
      ► [[Prototype]]: Object
      length: 1
      ► [[Prototype]]: Array(0)
      height: null
      ► margin: (4) [0, 0, 0, 0]
      width: null
      ► [[Prototype]]: Object
```

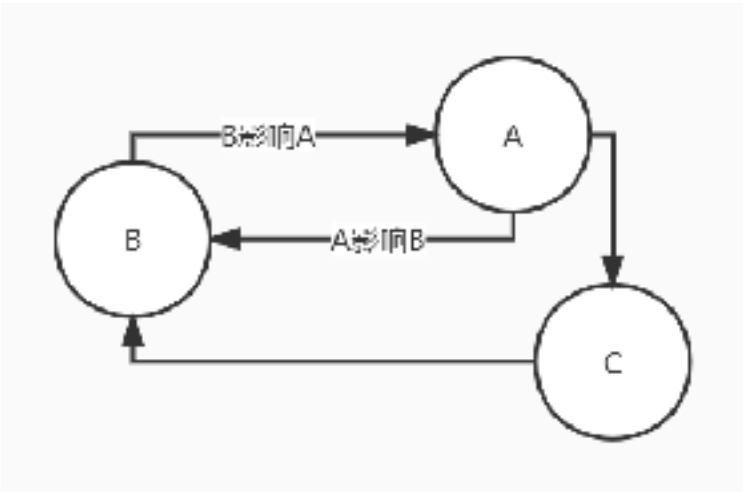
```
▼ [00APTyeKGjUzFV5g7FGo-: Array(4)]
  ▼ 00APTyeKGjUzFV5g7FGo-: Array(4)
    ▼ 0:
      componentId: "5cca2d0d-824b-476a-9b27-db49becaf752"
      neighborEdgeType: "bottom"
      neighborEdgeValue: 0
      value: 0
      ► [[Prototype]]: Object
    ▼ 1:
      componentId: "5cca2d0d-824b-476a-9b27-db49becaf752"
      neighborEdgeType: "left"
      neighborEdgeValue: 500
      value: 500
      ► [[Prototype]]: Object
    ▼ 2:
      componentId: "5cca2d0d-824b-476a-9b27-db49becaf752"
      neighborEdgeType: "top"
      neighborEdgeValue: 500
      value: 500
      ► [[Prototype]]: Object
    ▼ 3:
      componentId: "5cca2d0d-824b-476a-9b27-db49becaf752"
      neighborEdgeType: "right"
      neighborEdgeValue: 0
      value: 0
      ► [[Prototype]]: Object
      length: 4
      ► [[Prototype]]: Array(0)
      ► [[Prototype]]: Object
```

折线图	
基础	Autolayout约束
样式	高度 请输入高度表达式("W"表示页面宽度、"PW"表示父级容器宽度)
布局	高度 请输入高度表达式("H"表示页面高度、"PH"表示父级容器高度)
标题	
网格	边距 上 0 清除 右 0 清除 下 0 清除 左 0 清除
图例	
提示框	
工具栏	
系列	
X轴	对齐 相对组件 请选择组件 对齐方式 纵向居中 对齐偏移 0 上
Y轴	
区域缩放	
主题	删除
数据	新增对齐方式



Autolayout建设中的约束冲突的识别与避免

约束冲突是指当你使用 AutoLayout 布局,你添加的约束不满足或者会造成系统布局无法确定某个组件可视单元的Frame属性值。



借助虚拟Dom的实现思路，生成布局约束的虚拟树形结构，在用户操作某一约束条件时对约束虚拟树形结构进行“增删改查”的操作并测试是否存在“A参照B，B参照A”的情况，若存在则判定约束操作无效，重置该约束条件并作出提示。

设计模式

数据模型层 (Model)

数据模型层是奥创用于控制各组件数据接口标准, 规范各组件的入参

(imports) 和出参(exports)的格式、类型的模块, 它需要保证数据的正常流转及校验。数据模型层包含了对格式化引擎的实现, 格式化引擎负责处理在奥创平台内部数据流转间的各类自定义格式化函数及变量, 在各个组件入参前会通过该格式化引擎处理完后再反馈给组件正确的值。

```
1 数组遍历示例: {{
2    const data = [1,2,3,4];
3    return data.map(num => "Ultron" + num).join('')
4  }}
5  $|
```

- \$系统内置参数_当前时间 标准参数
- \$系统内置参数_当天开始时间 标准参数
- \$用户预置参数_lineData 标准参数
- \$ArrayMapGet 数组遍历取值函数
- \$DateAdd 日期加减函数
- \$NumberToPercent 数字转百分比
- \$ObjectGetVauByPath 对象取值函数
- \$StringSplice 字符串截取函数

```
const config = {
  componentType: FORMITEM_BOX,
  componentName: '单选下拉',
  componentPinyinName: 'danxuanxiala',
  size: [220, 32],
  icon: 'icon-check-circle',
  itemType: 'Select',
  exports: [
    { key: 'option.value', alias: '选中值', type: 'Other' },
    { key: 'option.options', alias: '选项列表', type: 'Array' },
    { key: 'option.searchValue', alias: '搜索值', type: 'Other' }
  ],
  imports: [
    {
      key: 'option.options',
      alias: '选项列表',
      type: 'Array'
    },
    { key: 'option.value', alias: '选中值', type: 'Other' }
  ],
  defaultImports: [{ key: 'option.value', alias: '选中值', type: 'Other' }],
  axiosRequest: false,
  option: {
    options: [
      { label: '张三', value: 'zhangsan' },
      { label: '李四', value: 'lisi' },
      { label: '王五', value: 'wangwu' }
    ],
    mode: null,
    disabled: false,
    bordered: true,
    allowClear: false,
    showSearch: false,
    showArrow: true,
    placeholder: '请选择',
    labelValues: null,
    style: {
      color: 'rgba(0,0,0,0.8)',
      textAlign: 'auto',
      fontStyle: 'normal',
```

数据依赖

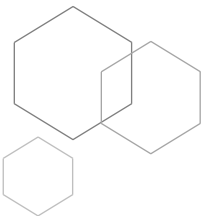


奥创依赖配置图

```
const config = {
  componentType: FORMITEM_BOX,
  componentName: '日期选择',
  componentPinyinName: 'riqixuanze',
  size: [110, 32],
  icon: 'icon-date-single',
  itemType: 'DatePicker',
  exports: [{ key: 'option.value', alias: '选中日期', type: 'Other' }],
  imports: [{ key: 'option.value', alias: '选中日期', type: 'Other' }],
  defaultImports: [{ key: 'option.value', alias: '选中日期', type: 'Other' }],
  option: {
    disabled: false,
    defaultDate: [],
    disabledDate: [],
    bordered: true,
    allowClear: false,
    showTime: false,
    inputReadOnly: false,
    placeholders: '选择日期',
    picker: 'date',
    format: 'YYYY-MM-DD',
    style: {
      color: 'rgba(0,0,0,0.8)',
      textAlign: 'auto',
      fontFamily: 'sans-serif',
      fontStyle: 'normal',
      fontSize: 14,
      fontWeight: 'normal'
    }
  }
}
```

数据依赖是什么？

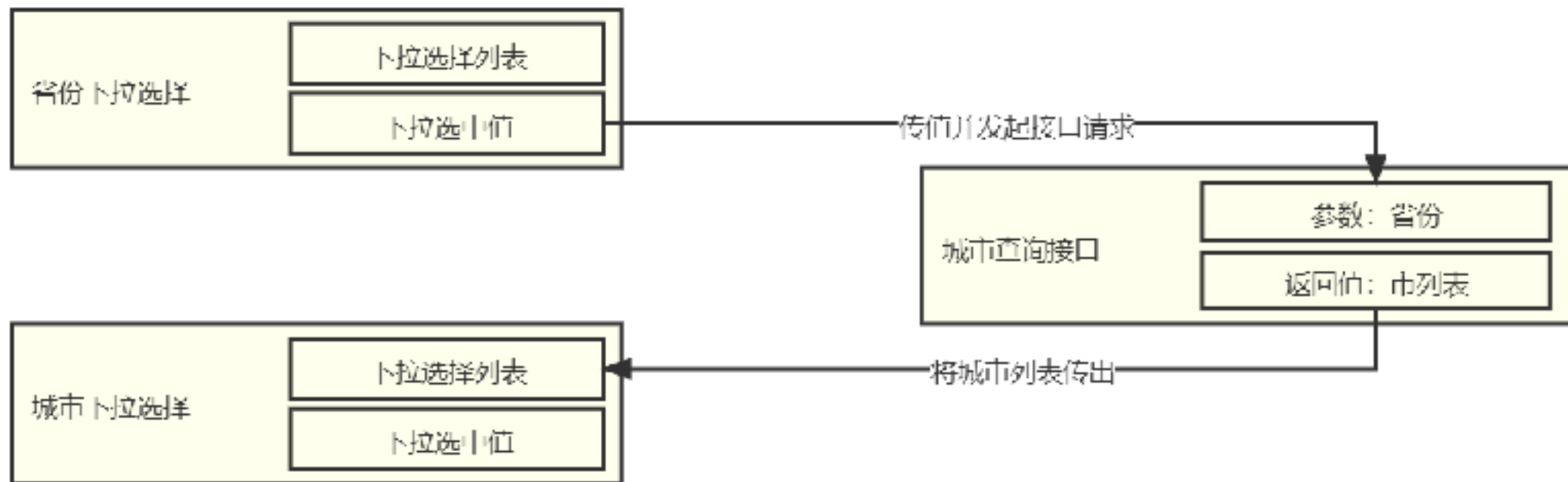
将观察者模式结合回溯算法（深度优先，避免不必要重复搜索的穷举式搜索算法）的面向过程就是奥创的数据依赖。数据依赖是半自动化的逻辑编排，一条完整的数据依赖是包含了数据提供方、数据消费方、计算表达式（非必须）的。



数据依赖



数据依赖在低代码平台中的应用场景解析



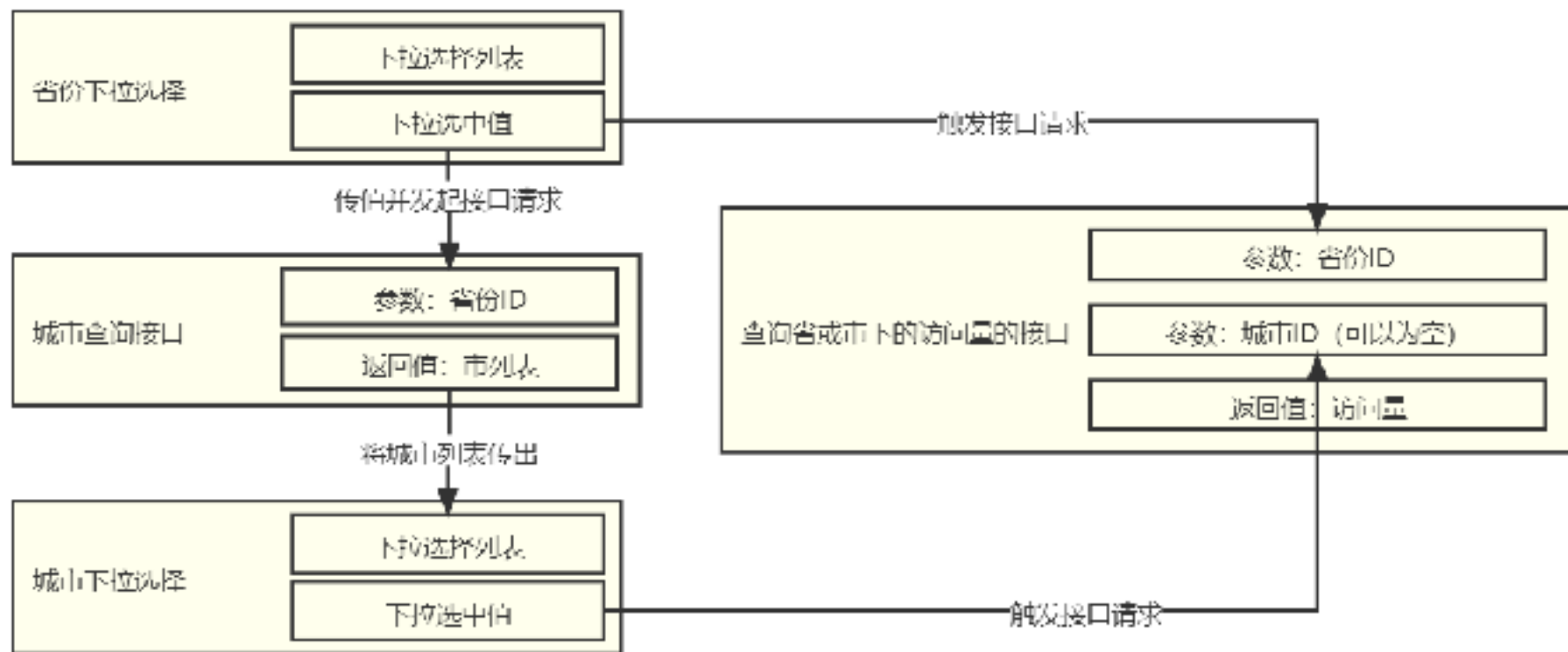
数据依赖



数据依赖流程图

数据依赖

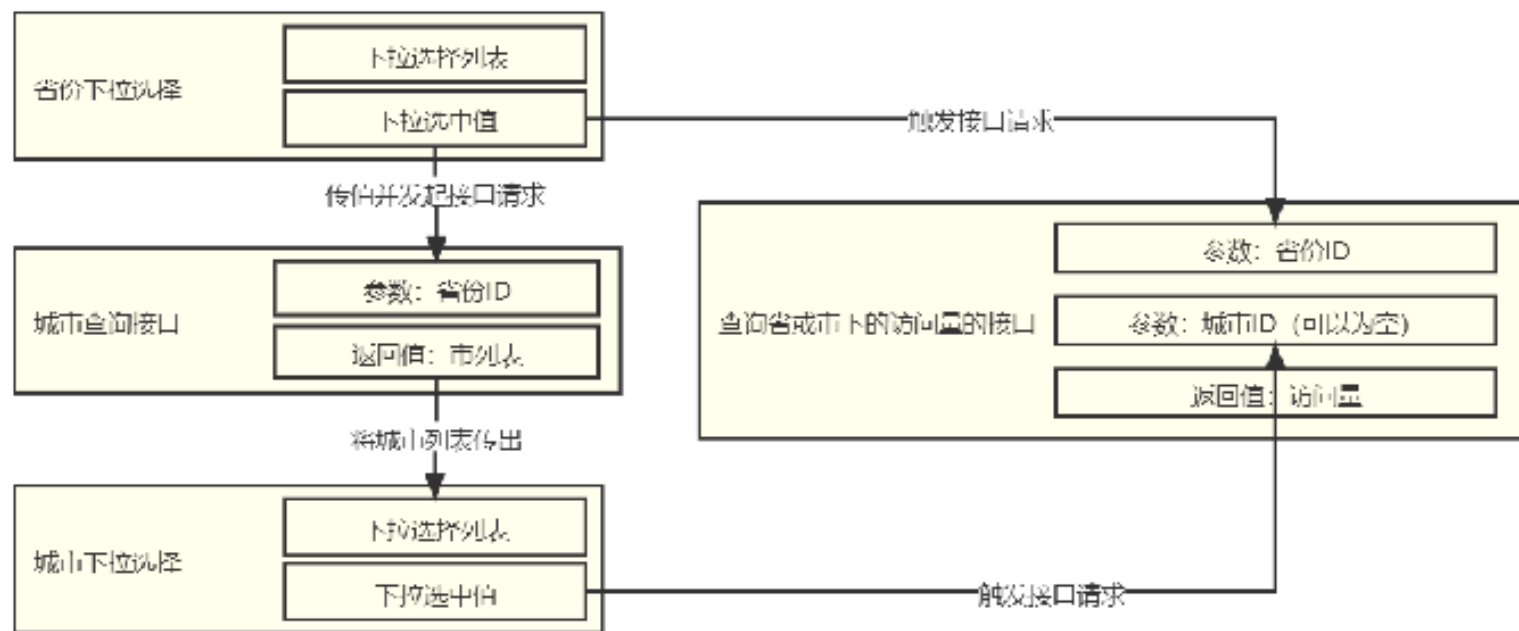
数据依赖重复执行的问题和解决方案



数据依赖

引用计数法解决数据依赖重复执行

引用计数法是通过在对象头中分配一个空间来保存该对象被引用的次数。如果该对象被其它对象引用,则它的引用计数加一,如果删除对该对象的引用,那么它的引用计数就减一,当该对象的引用计数为0时,那么该对象就会被回收。



省份下拉选择: 计数0

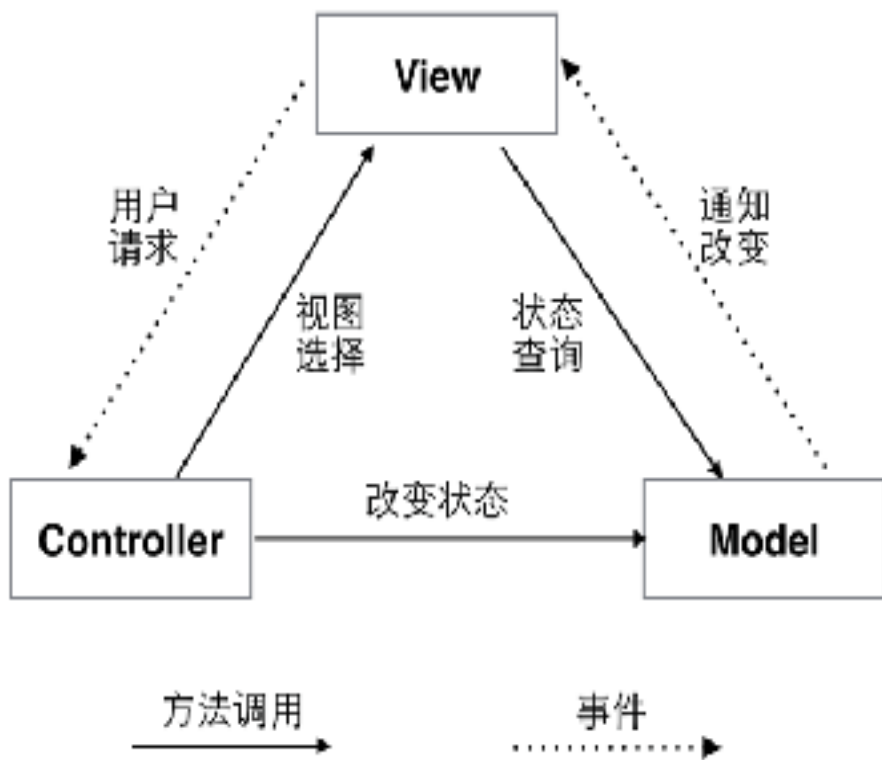
城市查询接口: 计数1

城市下拉选择: 计数1

访问量接口: 计数2

控制器层 (Controller)

控制器层 (Controller)，负责调度数据依赖、把控布局、响应状态变更等。通过引用计数法的方式是提升数据依赖流转效率，减少数据依赖流转次数，通过时间分片操作合理调度长任务函数执行防止界面的卡顿等。

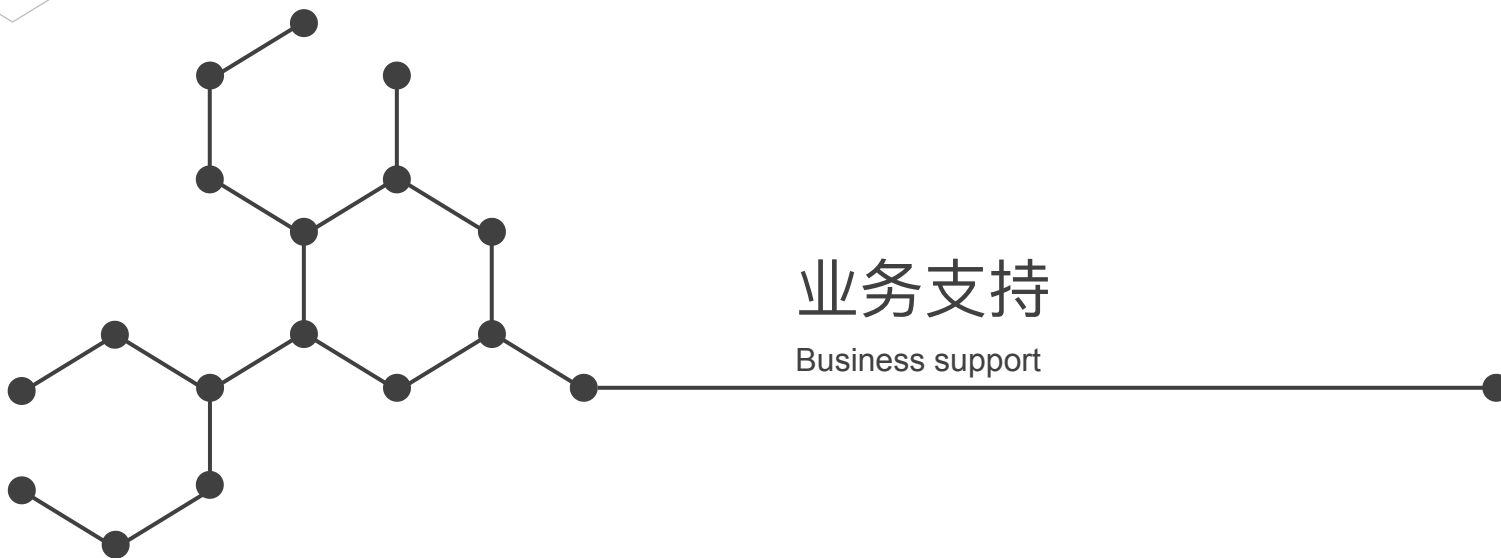
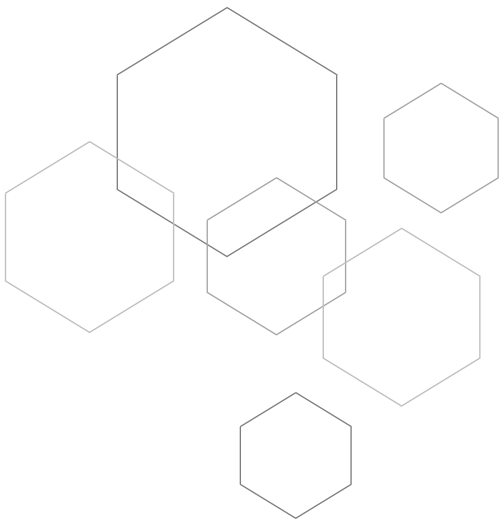


```
// 简易时间分片
function timeSlice (fnc, cb = setTimeout) {
  if(fnc.constructor.name !== 'GeneratorFunction') return fnc()

  return async function (...args) {
    const fnc_ = fnc(...args)
    let data

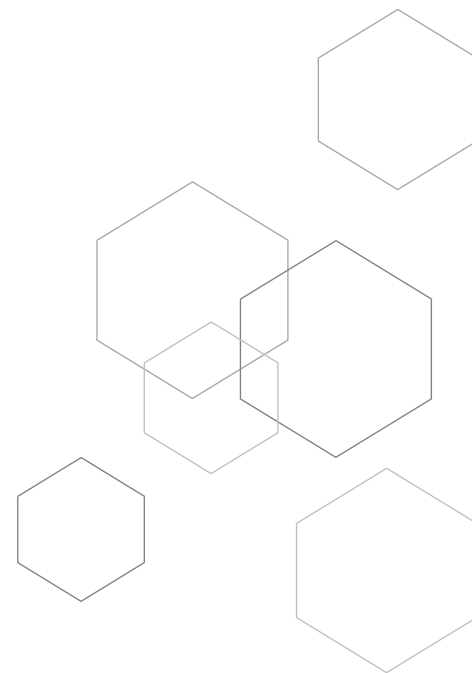
    do {
      data = fnc_.next(await data?.value)
      // 每执行一步就休眠，注册一个宏任务 setTimeout 来叫醒他
      await new Promise( resolve => cb(resolve))
    } while (!data.done)

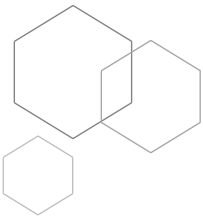
    return data.value
  }
}
```



业务支持

Business support





业务输出的两种模式

全量托管模式是指用户搭建后的网页入口、路由、状态管理等都由奥创引擎处理，完全托管在奥创的服务器中无法指定域名，第三方用户凭借链接访问，整体灵活度偏低。

全量托管
模式

SDK接入模式是指用户搭建的页面通过奥创微前端SDK接入，奥创不再接管路由、状态管理等，奥创仅会将搭建后的成果作为一个远程组件加载到用户指定的Dom容器内。

SDK接入
模式

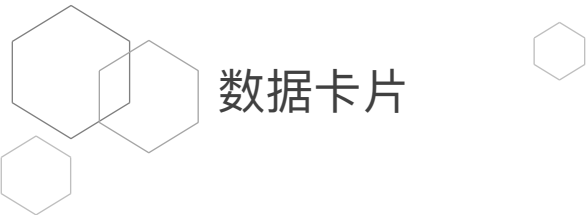
SDK接入模式——微前端数据卡片解决方案

奥创是一个低代码搭建平台工具，虽然致力于降低使用门槛，但它目前还做不到消除使用门槛。因此，我们拥抱开发者便也成了必然，奥创数据卡片微前端SDK便是专为开发者提供的小而精（9.9KB）的微前端接入服务。它可以支持将奥创搭建出来的各类报表、表单、实时数据分析等页面不受限制的精准嵌入到各种前端项目，或者说是前端项目的某个页面中的某一小块区域，开发者所需要做的仅仅是引入奥创数据卡片微前端SDK。

在各位的使用视角中它会是：



接入奥创数据卡片微前端SDK



数据卡片

奥创数据卡片微前端的核心理念

一、将应用作为一个独立的个体来考虑，不像常规微前端一样，不去明确界定父子应用之分。我们可以说奥创应用没有父应用，也可以说任何引用了奥创数据卡片SDK（以下简称SDK）的应用都可称之为奥创的父应用。我们树立这个理念并践行它得到的好处就是极大地便利了开发者对奥创的接入，同时保证了奥创应用内部的高度自治。

二、“高内聚、低耦合、零配置”，目前大家所了解的大部分的微前端框架应用除了去修改子应用的配置，另外还得更改父应用配置，想必用过的同学深有所感。但在奥创数据卡片微前端服务中，不再需要关心父应用。所有的状态刷新、应用重载、消息回调等等一系列的过程已经全部由奥创应用内部实现，与外部应用、SDK都无关。

数据卡片

第三方应用

Vue

React

Angular

.....

奥创SDK

应用管理

注册

挂载

缓存

刷新

XMLHttpRequest

manifest文件请求

静态资源加载

作用域隔离

Css作用域隔离

Css动态作用域

奥创应用

SDK响应

libraryTarget

asset-manifest

消息回调

应用刷新实现

7、开发支持

7.1 SDK引入

添加script标签: `<script src=UltronSDK.min.js?name=Ultron"></script>`

```
1 window.Ultron.createContainer({
2
3   // 挂载id 必传
4   // 奥创页面将加载显示在此Dom节点上
5   mountId: "挂载id",
6
7   // 页面ID 必传
8   // 需要显示的奥创应用ID, 可以在奥创的页面管理中复制得到
9   pageId: "页面ID",
10
11  // 加载的模块 必传
12  // sdkView 表示获取生产页面模块
13  // sdkDesign 表示获取开发设计模块
14  module : 'sdkView',
15
16  // keepAlive
17  // 视情况开启或关闭缓存
18  keepAlive : true,
19
20  // 环境变量 不传默认获取生产环境的图表配置
21  // 开发传 "dev"
22  // 测试传 "test"
23  // 生产传 "prod" 或者不传
24  env : '环境变量',
25
26  // 传递给奥创应用的参数 例如: "token"等 非必传
27  props : {
28
29  },
30  // 钩子函数
31  hooks : {...}
32
33  }).show();
```



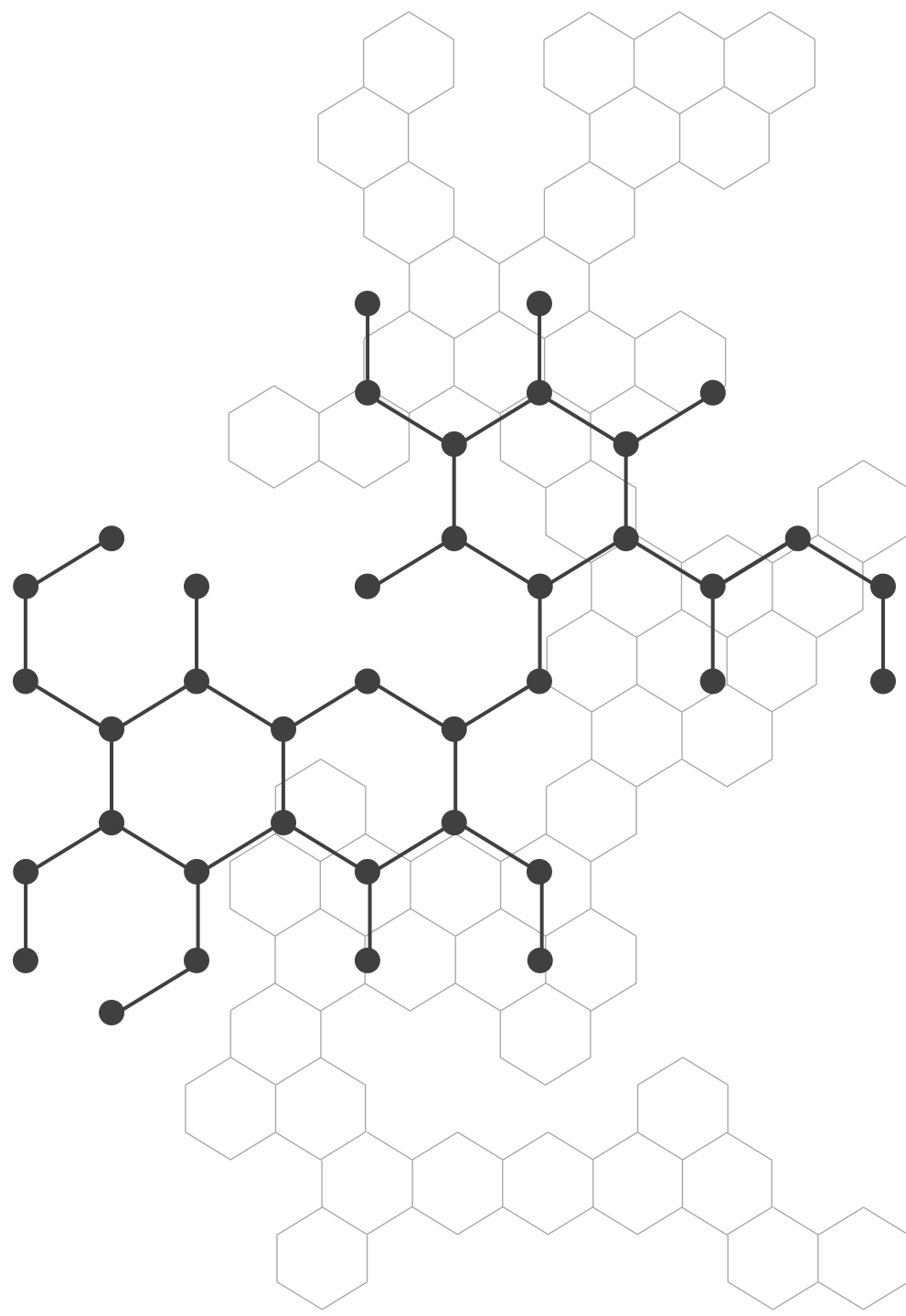
兴盛优选技术社区

谢谢观看

Thanks for watching

兴盛优选研发信息中心/UXT体验技术部

主讲人：钟柳





早早聊
ZAOZAO.RUN

一起走满5年

早早聊天使票预售

扫描右侧二维码

抢前 1000 分红名额：



早

虎年跳个好团队

早早聊 跳槽营



职业路线辅导
跳槽套路指导
模拟面试摸底
题库讲解直播
面经直播分享
优质团队内推
优质简历分享
职业成长视频

八大服务 破解 七大跳槽难题

学历硬伤
跳槽高频
项目简单
底子薄弱
深度不足
业务不精
管理不通



跟我一起学习吧!

每月挑一个周六，一起充电成长!



快扫码上车早早聊
400 个录播等你看

2020

全年行程 ▶

1/11 前端转管理
2/29 前端搞基建
3/28 前端搞搭建
4/11 前端搞规划
4/25 前端搞监控
5/16 Serverless
5/30 玩转微前端
5/31 前端大厂面试
6/13 在线文档专场
6/20 前端跨端跨栈
6/27 女生职业专场
7/18 前端搞可视化
8/15 前端搞构建
8/29 职业成长与晋升
9/26 前端搞报表
10/17 组件化资产
11/21 前端框架专场
12/5 脚手架工具链
12/26 前端性能专场

2021

全年行程 ▶

1/9 前端自由职业
1/23 前端团队管理
2/6 小程序组件化
2/27 页面搭建专场
3/20 大厂招聘面试
4/10 前端搞 CI/CD
4/24 前端玩转算法
5/9 前端迷职晋升
5/15 前端玩转互动
6/5 跨端搞 Flutter
6/12 编译彩蛋专场
6/26 前端 WebGL
7/17 前端搞可视化
8/14 前端玩转 BFF
8/21 GraphQL 专场
8/28 前端质量专场
9/11 前端安全专场
10/23 前端搞 Vue
11/20 前端搞 IDE

2022

全年行程 ▶

1/29 前端职业前路
2/26 TS 技术专场
3/19 管理专场
4/9 跳槽面试专场
4/23 组件专场
5/14 监控专场
5/28 性能专场
6/18 成长与晋升
6/25 跨端专场
7/16 海外工作专场
7/17 一站式基建
8/6 工程化专场
8/27 低代码专场
9/3 低代码微前端
9/17 音视频专场
9/24 创业专场
10月 开发语言专场
10月 保养健康专场
11月 自由职业专场
11月 框架专场
12月 运维专场
12月 鸿蒙专场
12月 可视化专场