



8月6日 | 48届前端早早聊

前端工程化 架构演进 设计落地

研发提效
业务交付
AI 与前端
工程化体系
设计与落地



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月	9月	9月	10月	10月	10月	11月	11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	运维专场	鸿蒙专场	框架专场	开发语言专场	保养健康专场	自由职业专场	创业专场	可视化专场	音视频专场

13:00 交付:《如何借助前端工程化体系实现高效交付》

章鱼

行云集团
体验设计与研发中心
高级前端工程师

14:00 AI:《AI 结合前端工程化实现智能开发提效方案》

小蝌蚪

腾讯
腾讯新闻团队
高级前端工程师

15:00 toB:《前端工程化在金融 B 端的设计与落地》

芮正云

恒生电子
技术平台大前端
web 前端架构师

16:00 演进:《飞书人事前端业务架构演进之路》

刘承川

飞书
飞书前端
前端工程师

17:00 模块:《Modern.js 的模块工程化实践》

高博文

字节跳动
Web Infra - Web Solutions
前端工程师

18:00 SDK:《基于模块联邦 SDK 化的大型前端应用实践》

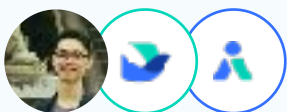
钟正楷

腾讯
腾讯新闻团队
高级前端工程师

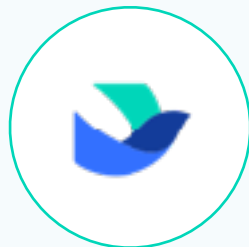


A Brief Introduction of Feishu CoreHR Front-End Architectures

飞书人事前端架构演进之路



刘承川 / 飞书前端团队



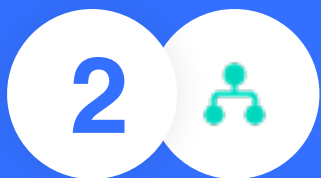
刘承川

飞书人事 · 前端架构方向
核心人事系统前端负责人



产品介绍

Feishu CoreHR



架构演进

1.0 / 2.0 / 3.0





员工人事信息触手可及 随时发现组织人才

会议中、群聊中、文档里，点击头像即可了解员工的全方位信息，让管理者在日常办公场景中可以随时随地挖掘组织人才，告别“脸熟人不熟”才能更好地排兵布阵。



告别割裂系统 人事数据汇集一处

全部员工从面试到离职的全生命周期数据汇集一处，调用和流转便捷，穿透各人力资源模块，让人事数据发挥力量，全方位刻画组织、提供洞察、支持决策。



组织脉络清晰可见 架构调整一步完成

组织架构查看清晰、调整简单，通过极简的“拖拉拽”操作即可一步完成复杂的部门调整和人员变更，自动匹配对应的审批流程，支持组织的每一次进化。



灵活智能的人事流程 轻松为企业提效

无代码配置和调整各类人事流程，无论是审批、工作待办，还是信息收集，皆可流程化设置、自动化推送审批消息。



飞书人事

企业内部系统 – People 系
统

>

商业化产品 – 飞书人事

1.0 后端为主
前端为辅

2.0 前后端分离
建立工程化体系

3.0 基于微前端的
飞书人事新架构



1.0

后端为主， 前端为辅

小型团队

技术 < 10%

业务 > 90%

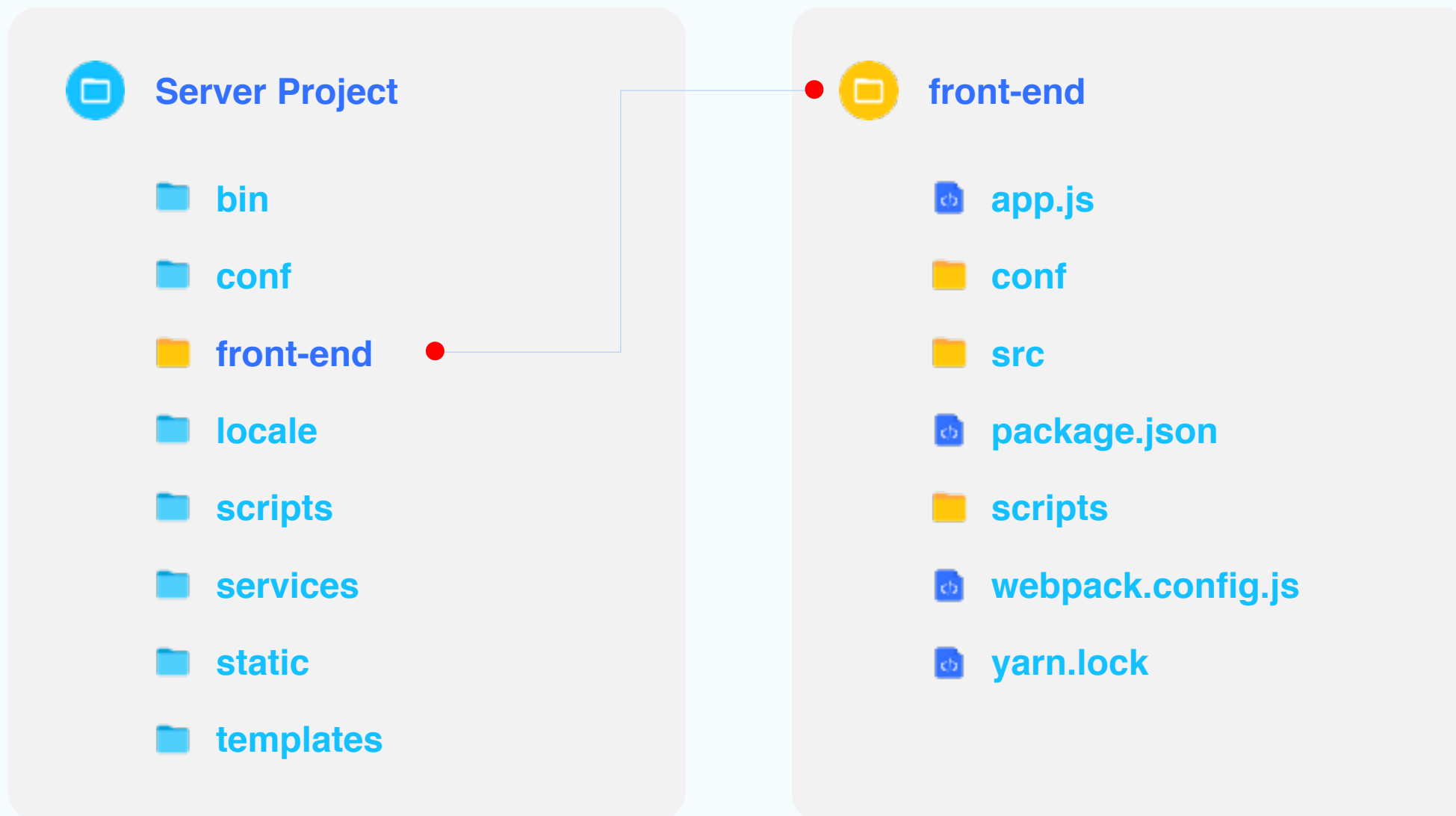
团队
规模

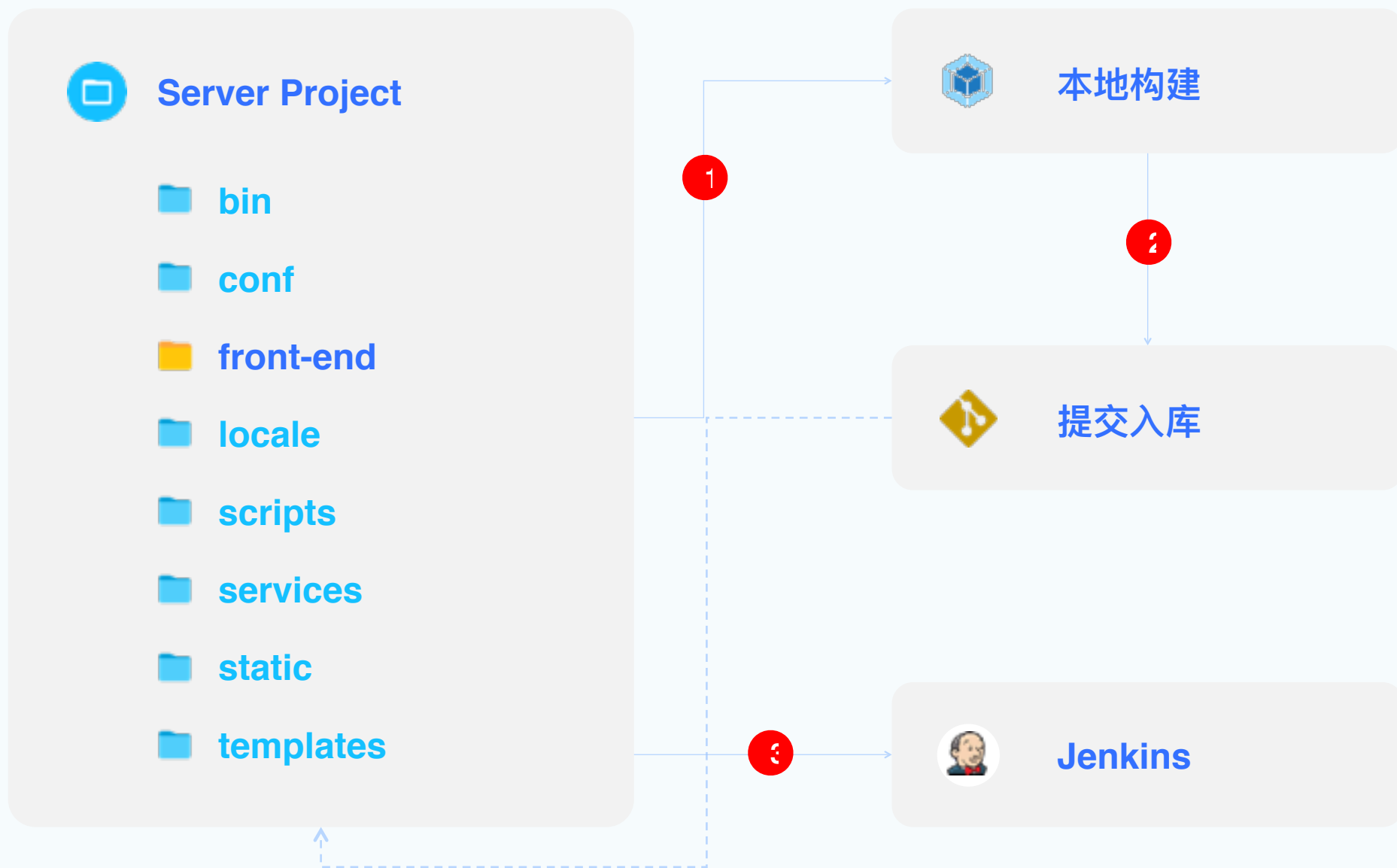
后端技术栈



前端技术栈









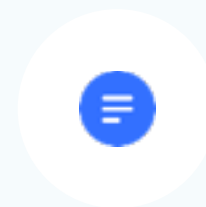
用户



负载均衡



后端服务



HTML 模版



为什么要做架构升级？

流程规范



代码质量



工具组件





2.0

前后端分离 建立工程化体系

开发阶段

测试阶段

上线阶段

线上阶段

项目划分

单元测试

自动化构建

监控平台

脚手架

E2E 测试

产物扫描

日志平台

Git Flow

产物分发

静态资源容灾

Lint / Type Check



Front-end Project



Node Server



UI 组件 & Utils



招聘



人事



OKR



绩效



薪酬



...

**Eden**

自建前端研发体系



```
npx @ies/eden-init@latest
```

```
// or  
eden init
```

**Create React App**

```
npx create-react-app my-app \  
  --template [template-name]
```

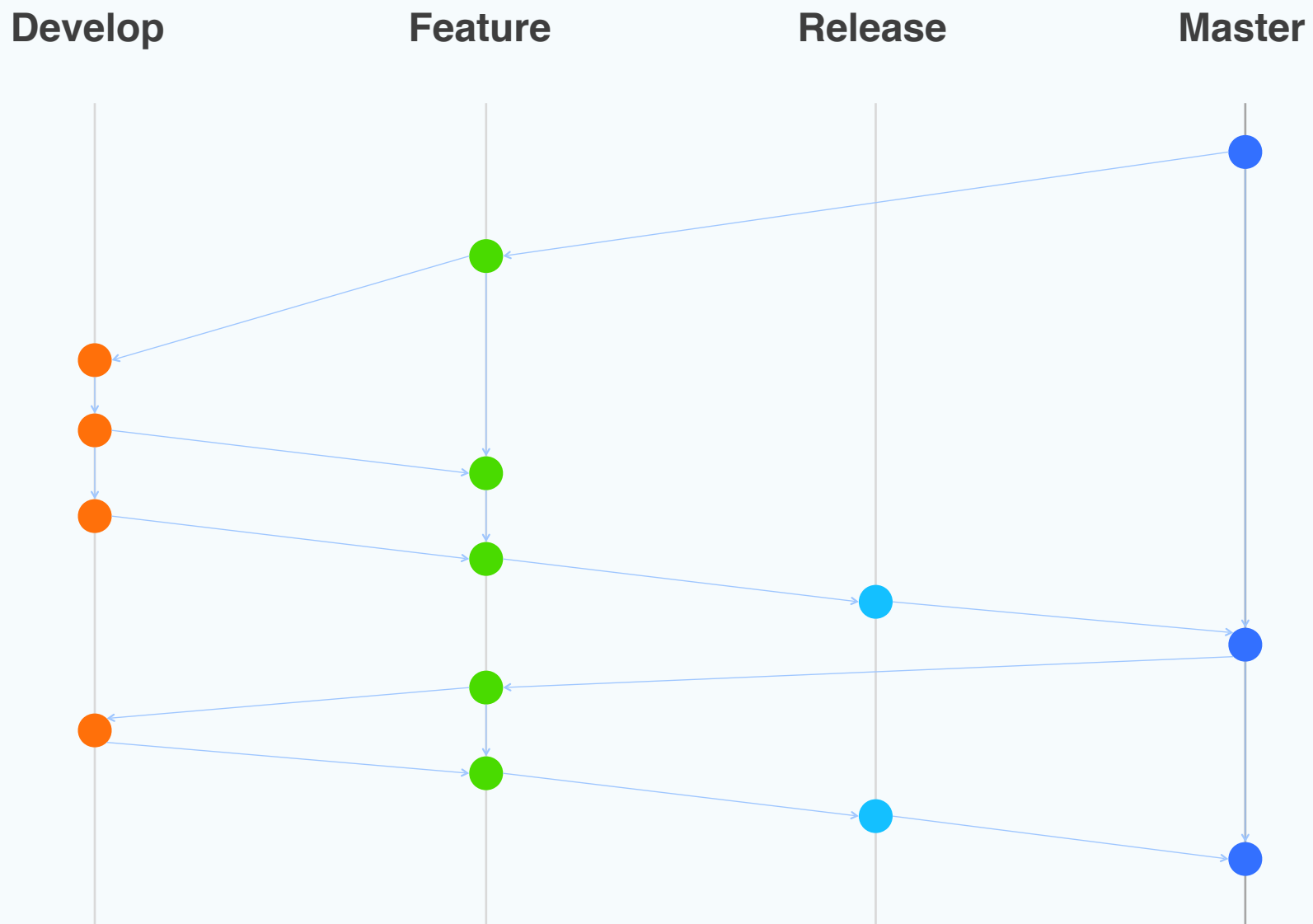
**Vue Cli**

```
yarn global add @vue/cli  
vue create my-project
```



A terminal window titled "eden (npm)" with a standard macOS window header (red, yellow, green buttons and title bar). The terminal shows the command `$ eden init` being executed. Below the command, an error message is displayed in red text: `* error Your npm version is 7.15.1, which may cause some compatibility issues, you can downgrade to npm 6.x or lower to solve.`. A white cursor is positioned on the line following the error message. In the center of the terminal, there is a large, faint, light-blue watermark logo consisting of a stylized hexagon with internal geometric patterns.

```
$ eden init
* error Your npm version is 7.15.1, which may cause some compatibility issues, you can downgrade to npm 6.x or lower to solve.
```



环境切换工具 | BOE | 测试功能 08041536

RD 模式 ☐

切换测试环境泳道

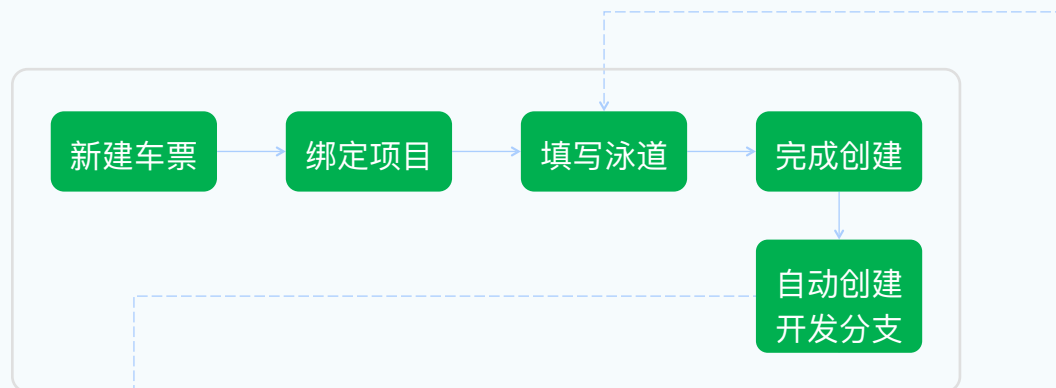
切换环境后，请刷新页面，保证访问正确的环境

飞书人事

切换为全量版本

环境名称	环境标识	创建时间	操作
测试功能 08041536 生效中	lane_001 	2022-08-04 14:29:01	使用该版本
人员管理-功能改进-01	lane_002 	2022-08-04 14:16:07	使用该版本
组织管理-自测技术需求-0728	lane_003 	2022-08-04 11:36:28	使用该版本
组织管理-自测技术需求-0804	lane_004 	2022-08-04 10:47:58	使用该版本
以上需求均为 Mock 数据	lane_005 	2022-08-04 10:47:10	使用该版本

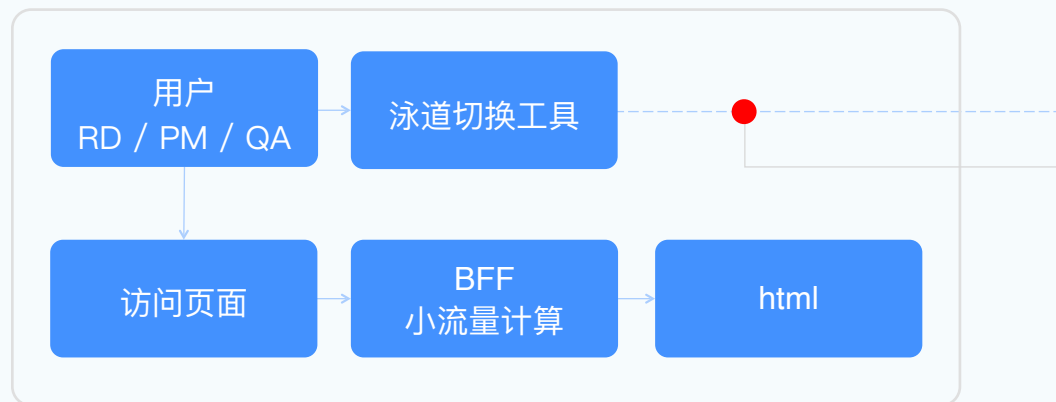
自研发布系统



发布流水线



接入层小流量

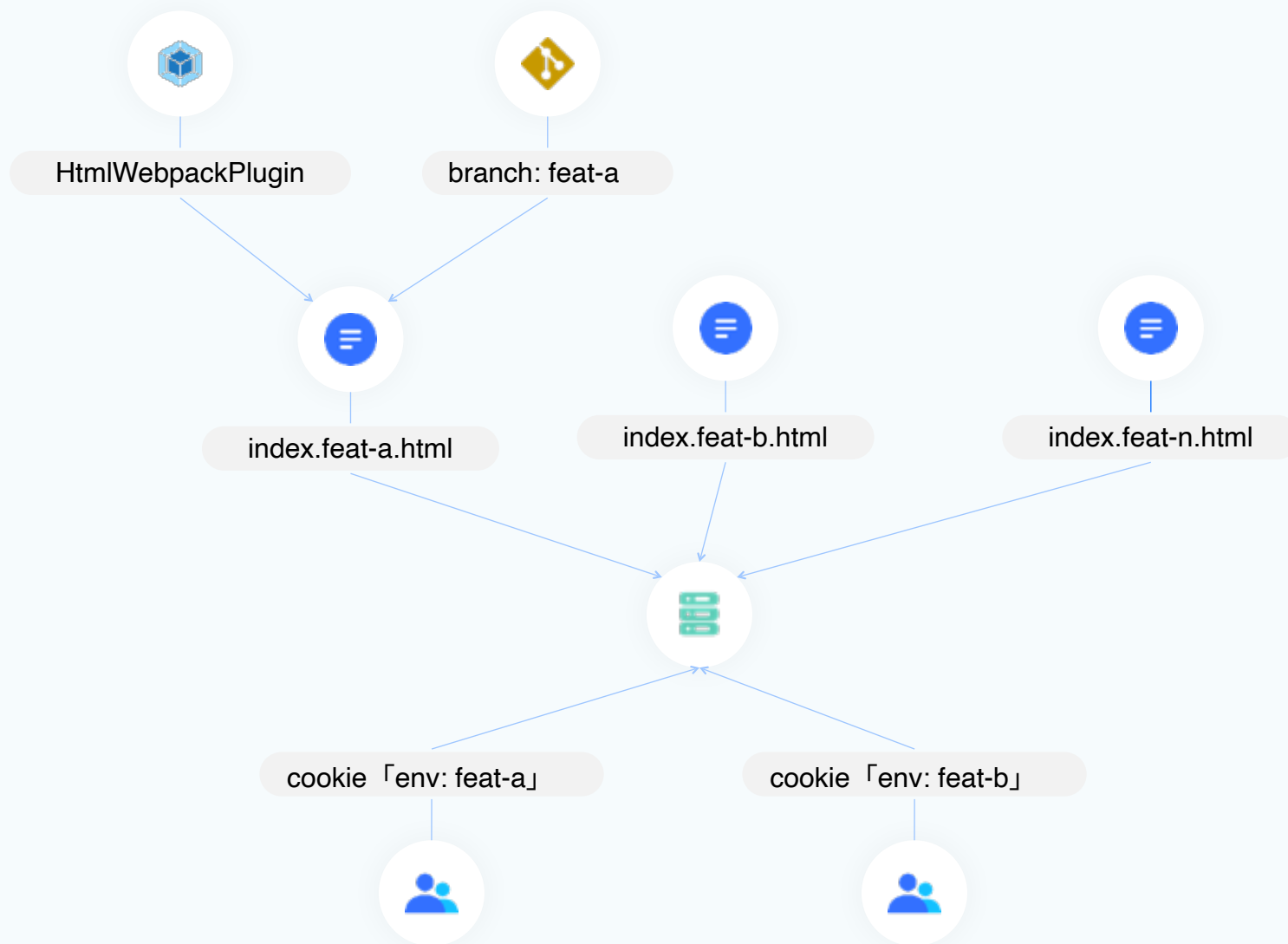


为 html 添加标记

为用户添加 cookie



简易版小流量分发模型



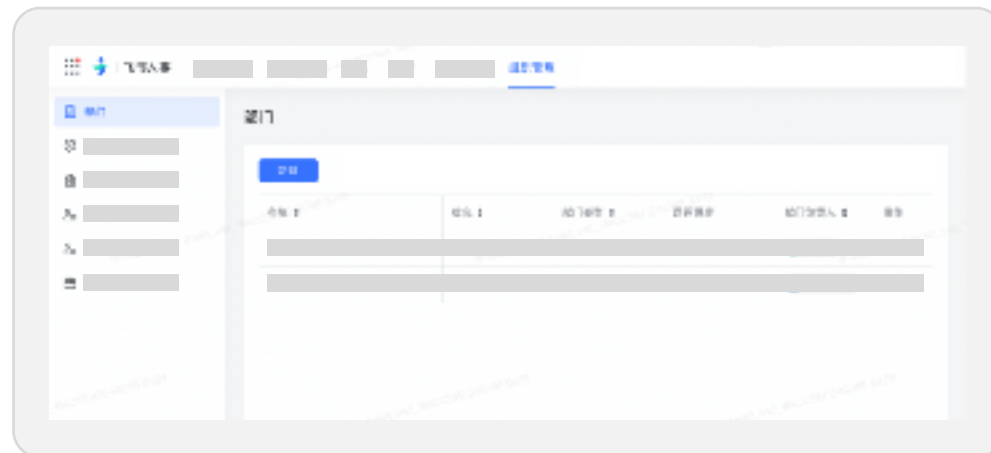
TS fork-ts-checker-webpack-plugin

 eslint-webpack-plugin

```
webpack 5.66.0 compiled with 2 warnings in 1017 ms
ERROR in src/index.tsx:38:5
TS2322: Type 'string' is not assignable to type 'number'.
   36 |
   37 | type test = number;
>  38 | let foo: test = 'hello';
      |      ^^^
   39 |
   40 | async function launch() {
   41 |   try {

ERROR in src/index.tsx:38:5
TS6133: 'foo' is declared but its value is never read.
   36 |
   37 | type test = number;
>  38 | let foo: test = 'hello';
      |      ^^^
   39 |
   40 | async function launch() {
   41 |   try {
```

```
> div > div"
86   );
87   assertion.record(
88     '部门列表: 名称/编辑搜索框存在: ',
89     () => expect(DOM_7cfe54).to.not.be.null
90   );
91   await assertion.screenshot("部门列表页面加载正常!");
92
93   // 新建部门
94   // 点击 BUTTON: "新建"
95   await page.waitForSelector(
96     "div > #org-table-list-box > div > div > button",
97     { timeout: 5000 }
98   );
99   await page.click("div > #org-table-list-box > div > div >
button");
100  // 点击事件后添加的等待时间
101  await page.waitFor(3500);
102
103  // 点击 INPUT: ""
104  await page.waitForSelector("span > div > div > div > input",
{timeout: 5000})
105  await page.click("span > div > div > div > input")
106  // 点击事件后添加的等待时间
107  await page.waitFor(3500);
108
```



开源解决方案

高效 / 易用 / 可靠的 UI 自动化测试工具

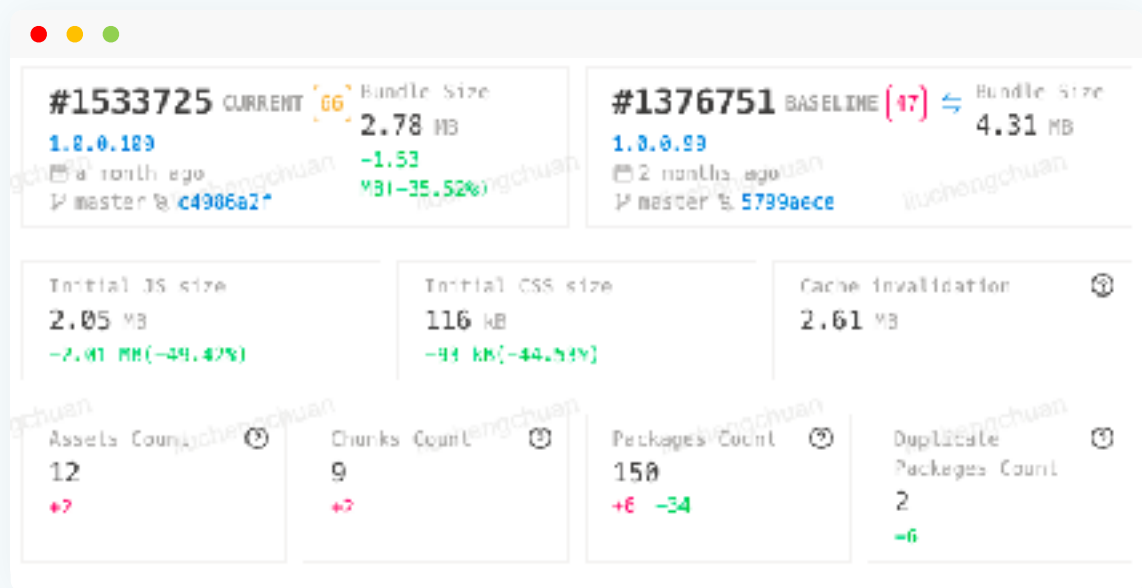


构建产物分析

Select one artifact

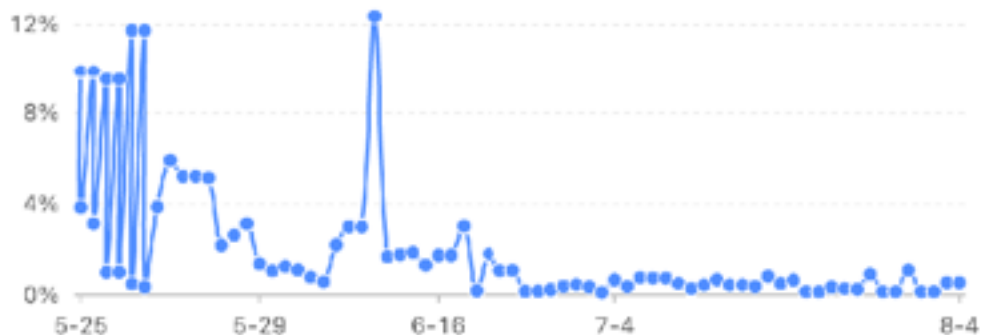
id	version	branch	created at	
#1533725	1.0.0.189	master	2022/07/04 12:04:04	Current
#1470512	1.0.0.187	master	2022/06/22 02:09:37	Select
#1435821	1.0.0.186	master	2022/06/15 01:14:42	Select
#1427919	1.0.0.185	master	2022/06/14 01:33:15	Select
#1422957	1.0.0.184	master	2022/06/13 14:52:07	Select
#1416550	1.0.0.183	master	2022/06/10 15:45:41	Select
#1399195	1.0.0.182	master	2022/06/08 00:57:17	Select
#1398493	1.0.0.181	master	2022/06/07 21:14:32	Select

< 1 2 3 4 5 >





代码质量分析



❗ 代码路径不应过长 P3 共有 1 个文件 [规则详情及建议](#)

检测结果

▼ 在文件 `/src/test.js` 中 共有 1 处

```
import { Token } from '../../../../../../../../constants';
```

代码重复度

语法 Lint

代码逻辑

最佳实践

安全扫描

...

❗ egeqeq 共有 1 个文件 [规则详情及建议](#)

检测结果

▼ 在文件 `/src/test.js` 中 共有 1 处

```
if (objectKey.indexOf('/') == 0) objectKey = objectKey.slice(1);
const res = await axios.post(
```

❗ 禁止在使用渲染框架前提下使用原生 DOM 渲染 P1 共有 2 个文件 [规则详情及建议](#)

检测结果

▼ 在文件 `/src/test.js` 中 共有 1 处

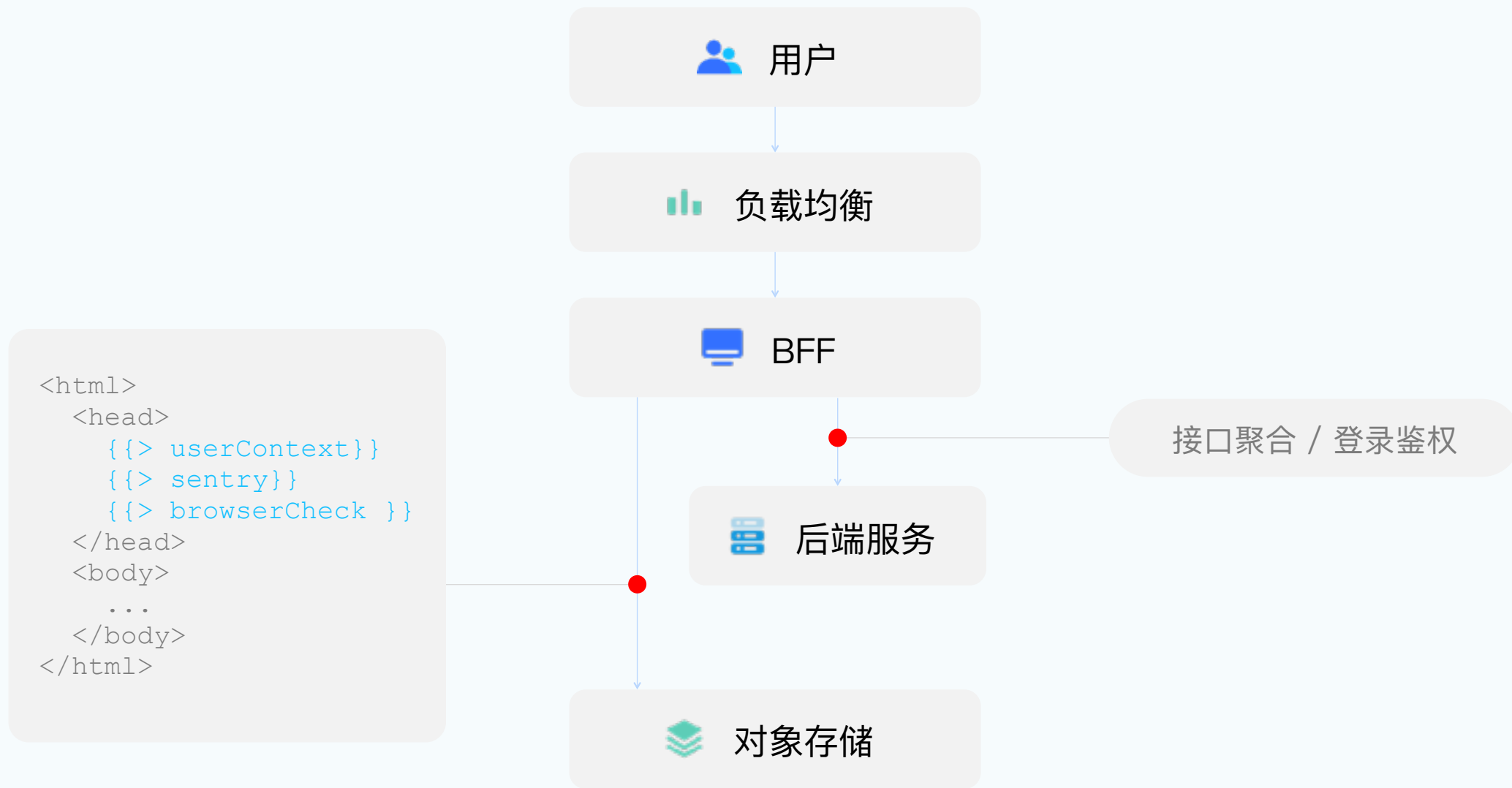
```
document.body.appendChild(dom)
```



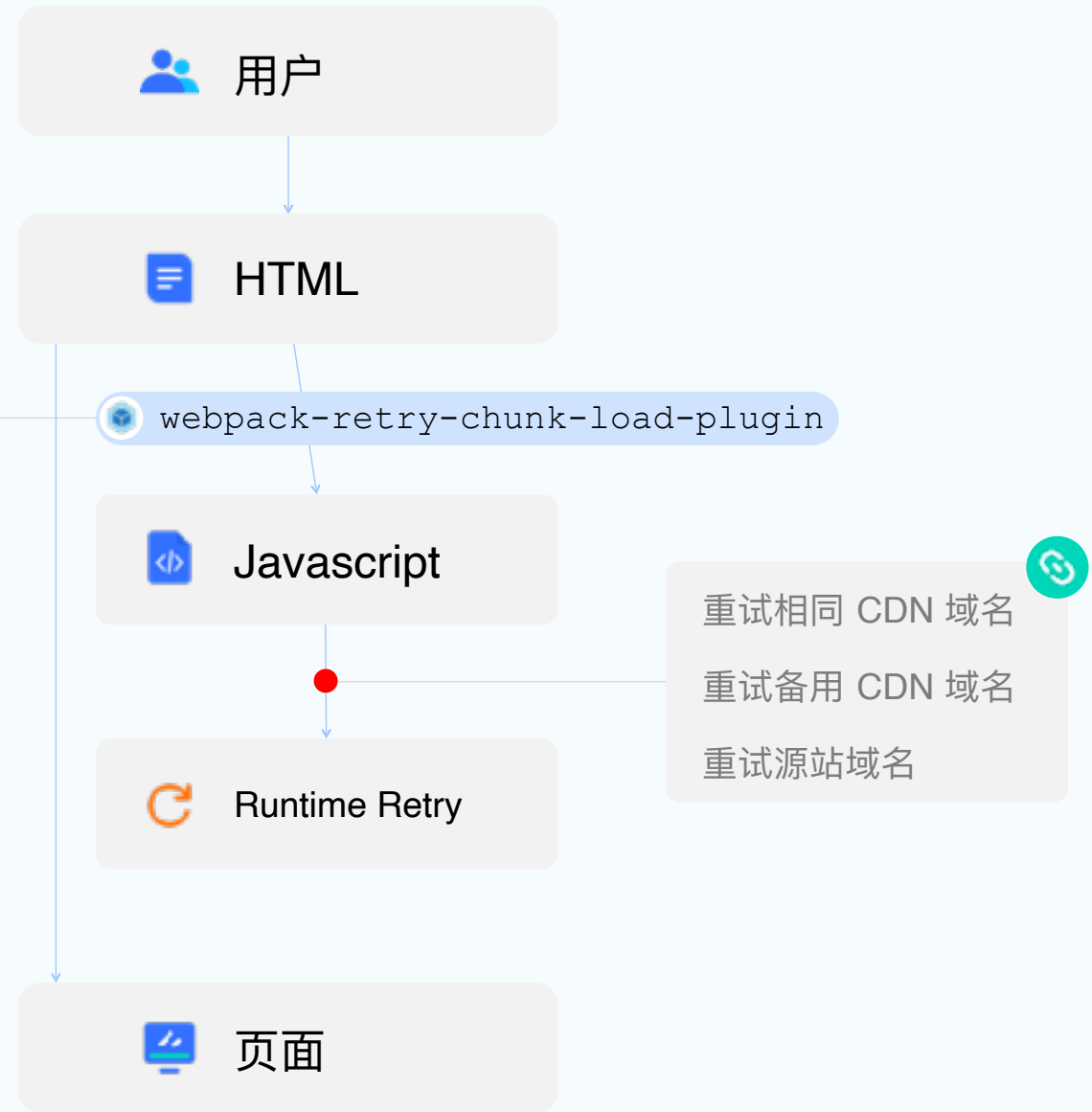
webpack-bundle-analyzer




eslint



```
1 const script = runtimeTemplate.life(
2   '
3   '
4   if(typeof ${RuntimeGlobals.require} !== "undefined") {
5     var oldGetScript = ${RuntimeGlobals.getChunkScriptFilename};
6     var oldLoadScript = ${RuntimeGlobals.ensureChunk};
7     var queryMap = new Map();
8     var countMap = new Map();
9     ${RuntimeGlobals.getChunkScriptFilename} = function(chunkId){
10      var result = oldGetScript(chunkId);
11      return result + (queryMap.has(chunkId) ? '?' + queryMap.get(chunkId) : '');
12    };
13    ${RuntimeGlobals.ensureChunk} = function(chunkId){
14      var result = oldLoadScript(chunkId);
15      return result.catch(function(error){
16        var retries = countMap.has(chunkId) ? countMap.get(chunkId) : ${maxRetries};
17        if (retries < 1) {
18          var realSrc = oldGetScript(chunkId);
19          error.message = 'Loading chunk ' + chunkId + ' failed after ${maxRetries}
20          retries.\\n(' + realSrc + ')';
21          error.request = realSrc;${
22            this.options.lastResortScript ? this.options.lastResortScript : ''
23          }
24          throw error;
25        }
26        return new Promise(function (resolve) {
27          setTimeout(function () {
28            var retryAttempt = ${maxRetries} - retries + 1;
29            var retryAttemptString = '&retry-attempt=' + retryAttempt;
30            var cacheBust = ${getCacheBustString()} + retryAttemptString;
31            queryMap.set(chunkId, cacheBust);
32            countMap.set(chunkId, retries - 1);
33            resolve(${RuntimeGlobals.ensureChunk}(chunkId));
34          }, ${this.options.retryDelay || this.options.timeout || 0})
35        });
36      });
37    }
38  };
39  return source + script;
```



```
<head>
  <meta charset="UTF-8" />
  <title>Feishu CoreHR</title>
  {{#each htmlWebpackPlugin.files.js}}
  <script
    defer="defer"
    src="{{CDN_DOMAIN}}{{ASSETS_PATHNAME}}{{this}}"
  ></script>
  {{/each}}
  {{#each htmlWebpackPlugin.files.css}}
  <link href="{{CDN_DOMAIN}}{{ASSETS_PATHNAME}}{{this}}" rel="stylesheet" />
  {{/each}}
</head>
<body>
  ...
</body>
```

 用户 BFF HTML 页面



webpack-retry-chunk-load-plugin

A X ↴ O S

axios-retry



随着业务的发展
应用层越来越庞大

分散的前端 SPA 项目
无法实现产品形态上的聚合

高频迭代的跨业务公共依赖
维护成本高



3.0

基于微前端的飞书人事新架构







自研开源微前端解决方案 / garfishjs.org



```
Garfish.run({
  basename: '/',
  domGetter: '#render',
  apps: [
    {
      name: 'react',
      activeWhen: '/react',
      entry: 'http://localhost:3000',
    },
    {
      name: 'vue',
      activeWhen: '/vue',
      entry: 'http://localhost:9090',
    }
  ]
});
```

```
export const provider = ({ basename, dom }) => {
  return {
    render () {
      ReactDOM.render(
        <App basename={basename} />
      ),
      dom.querySelector('#root')
    },
    destroy () {
      ReactDOM.unmountComponentAtNode(dom.querySelector('#root'));
    }
  }
}
```



微前端主应用

业务逻辑

登录状态

导航 UI

监控报警

日志打点

通用配置

...

微前端

应用加载

预加载

沙盒

依赖共享

React

UI 组件

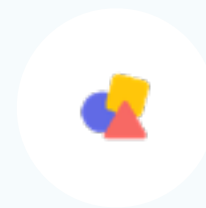
Utils



组件开发脚手架



物料平台



微组件

 接入指南

快速上手

 一键拖拽组件

Transfer

Clamp

Sidebar

MultilingualInput

DepartmentSelect

RuleCondition

RuleDisplay

EditableContent

PrcTable

ConflictModal

 NEXT SDK

AppLauncher

穿梭框 Transfer

穿梭框通常指双栏选择框，需要在多个可选项中进行多选时使用，其用鼠标的方式在两栏间移动元素，完成选择行为。比起 Select，穿梭框占据更大的空间，可以展示可选项的更多信息。

- 组件安装：npm install @next-nemix/transfer 或 yarn add @next-nemix/transfer
- 接入本组件项目 React 版本要求 >= 16.8，且安装 UI 组件库
- 本组件功能样式开发遵循[Transfer 设计规范](#)
- 开发者：@小袁

穿梭框基础用法

基础用法

全选 (100)

☐ 测试 0
 ☐ 测试 1
 ☐ 测试 2
 ☒ 测试 3
 ☐ 测试 4
 ☐ 测试 5

已选 (1)

清除

测试 3

»

穿梭框基础用法

选中内容可拖拽到穿梭框

分组展示的穿梭框

带 Tag 样式选项 Item

带描述形式描述 Item

同时有头像、tag、描述的形式选项 Item

左侧可下拉的穿梭框

左侧为树形的穿梭框

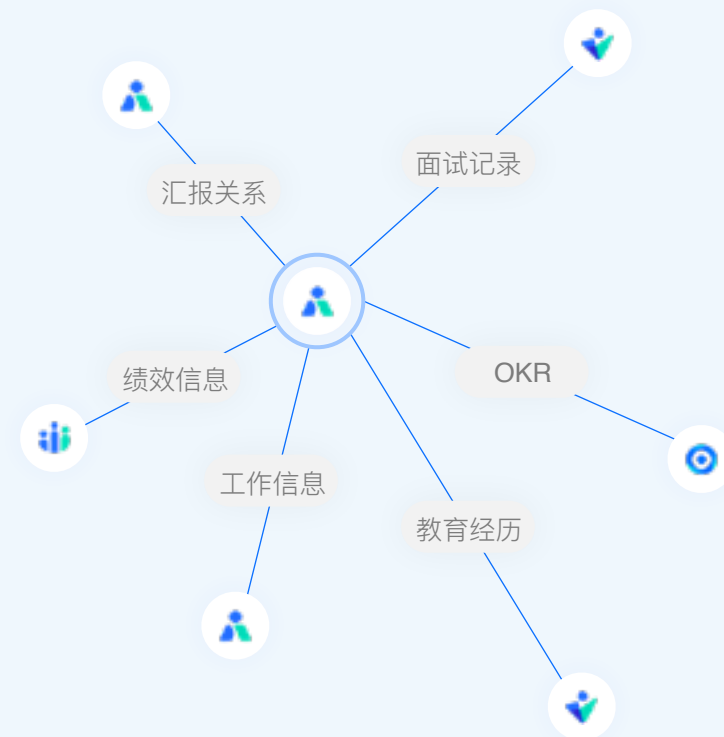
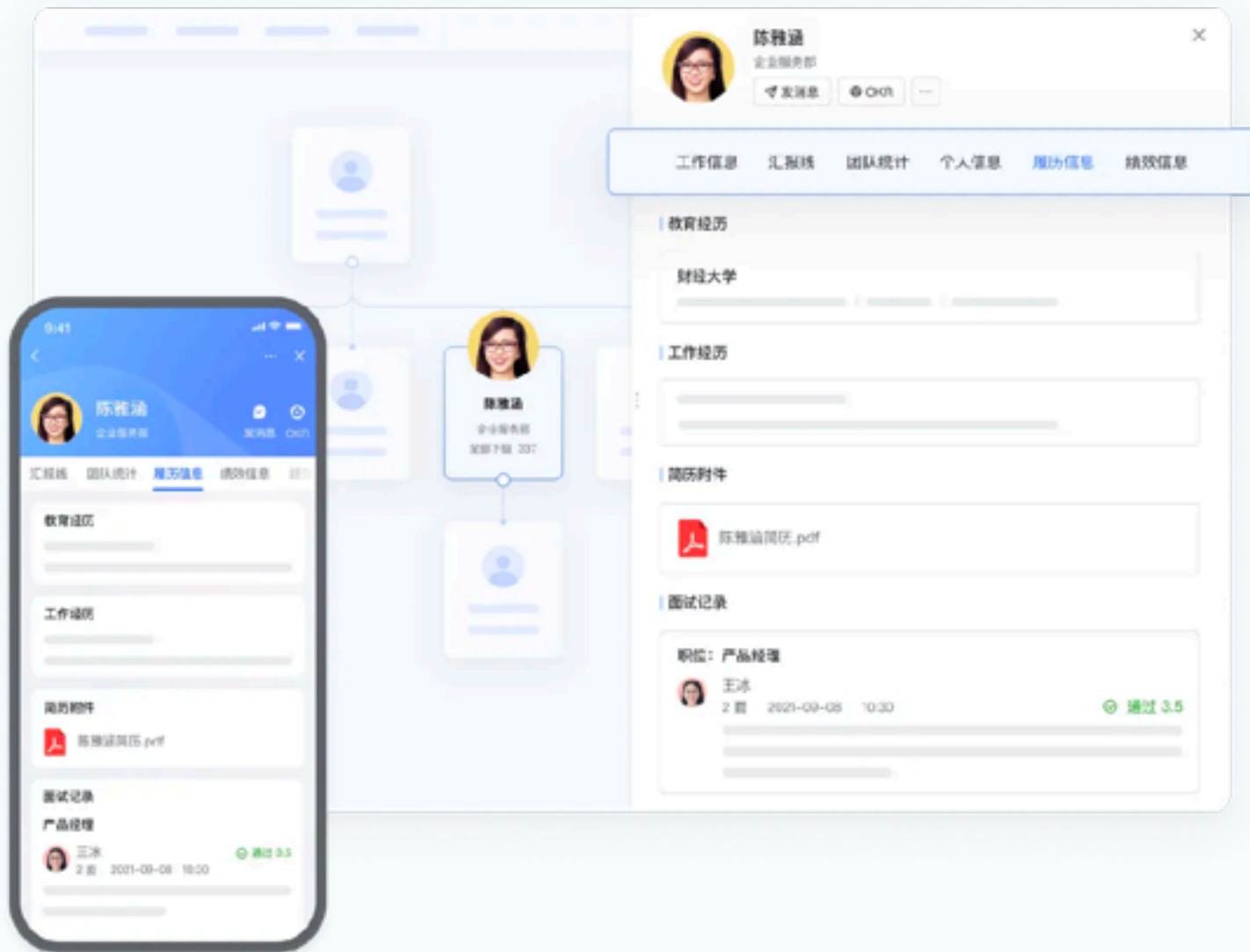
左右均为树形的穿梭框

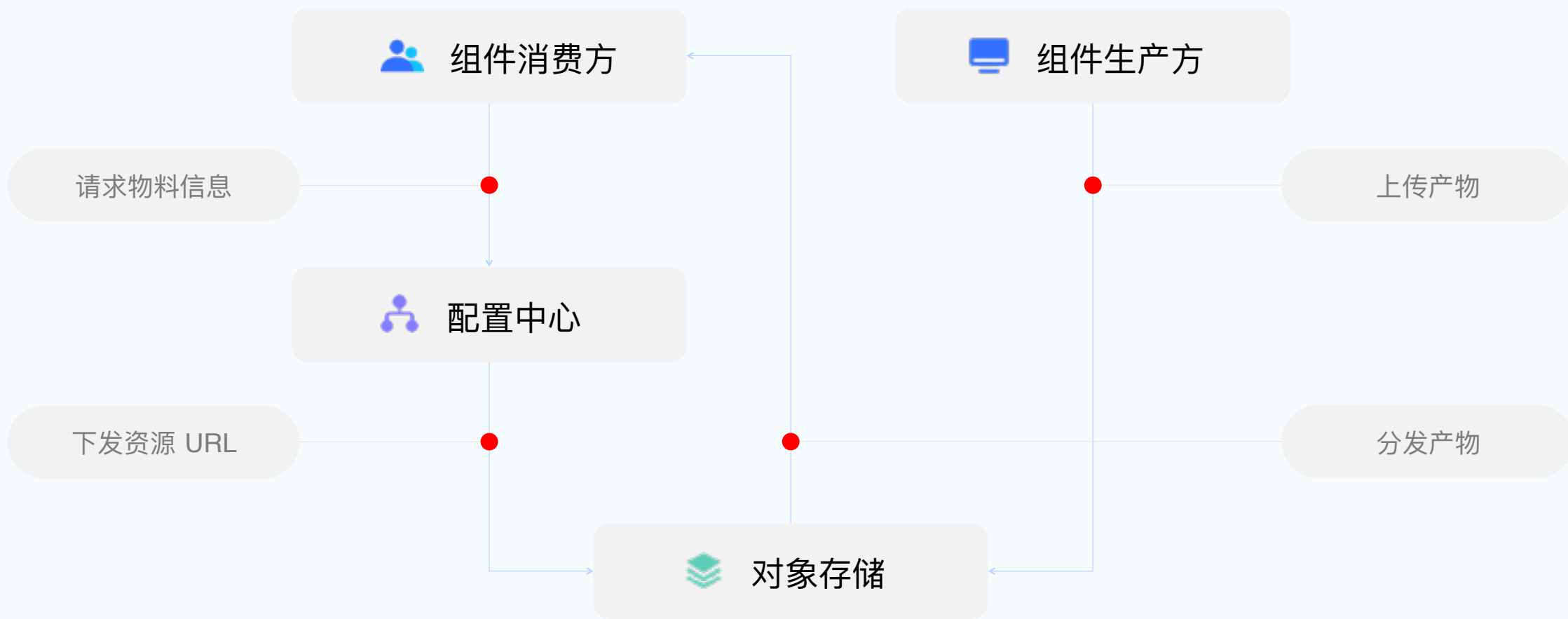
左右侧均为树形且右侧可拖拽排序的穿梭框

可联动展示穿梭框

自定义 Item 渲染

API







Future

找准问题所在
才能更好的进行架构改进



前端开发工程师

前端开发专家

资深前端开发专家

...



北京 / 广州 / 上海 / 杭州 /

深圳 / 成都 / 三亚



大狮兄
飞书个人版



使用飞书扫描二维码
联系我们获取更多信息

Thanks



刘承川 / 飞书前端团队



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 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 9/17 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

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:《云端 IDE 与微组件市场的碰撞》	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:《行云搭建工具落地》	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:《樛卯: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

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

2021 全年行程▶

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

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月	9月	9月	10月	8月	10月	8月	11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	运维专场	鸿蒙专场	框架专场	开发语言专场	保养健康专场	自由职业专场	创业专场	可视化专场	音视频专场