



11月27日 | 55届前端早早聊

AntV

企业方案 远方芬芳

渲染引擎

关系图谱

可视分析组件库

高性能图编辑

移动端可视化



13:00 渲染引擎：《可视化渲染引擎的 API 设计》

沧东

蚂蚁集团
体验技术部
前端工程师

14:00 移动端：《如何玩转移动端数据可视化》

糖莹

蚂蚁集团
数金体验技术部
前端工程师

15:00 图谱：《关系可视化在图谱业务中的挑战与创新》

一四

蚂蚁集团
体验技术部
前端工程师

16:00 组件库：《如何打造开箱即用的位置可视分析组件库》

云极

蚂蚁集团
体验技术部
前端工程师

17:00 图编辑：《如何快速搭建图编辑应用》

文瑀

蚂蚁集团
支付宝体验技术部
前端工程师

2022

全年行程

1/29 2/26 3/19 4/9 4/23 5/14 5/25 5/28 6/18 6/25 7/16 7/17 8/6 8/27 9/3 9/17 10/22 10/29 11/26 11/27 12/3 12/11 12/24

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

裁员仲裁

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

工程化跨端性能

框架专场

可视化专场

AntV 可视化

测试专场

远程工作专场

小程序专场

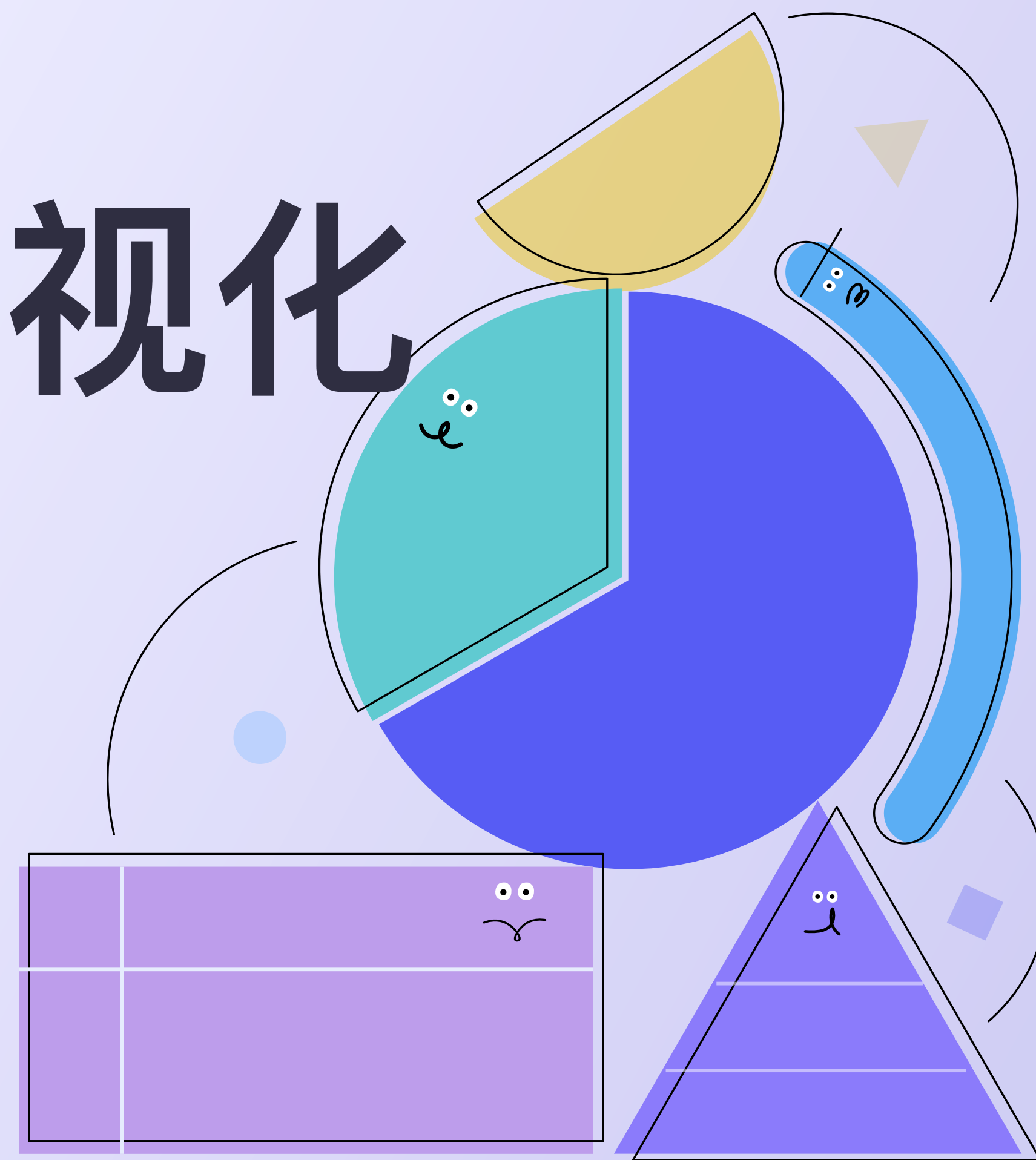


如何玩转移动端可视化

糖莹

蚂蚁集团数字金融线

@antv/f2、@antv/fengine





01 为什么会有移动端可视化

02 移动端可视化怎么这么难

03 移动端可视化解决方案

移动端平台多样各异

设计侧偏好（toC与toB）、交互、代码实现

业务侧移动端可视化诉求

移动端与PC的差异

屏幕大小、代码实现差异

你每天使用手机的时间是多久？

01 为什么会有移动端可视化



02 移动端可视化怎么这么难

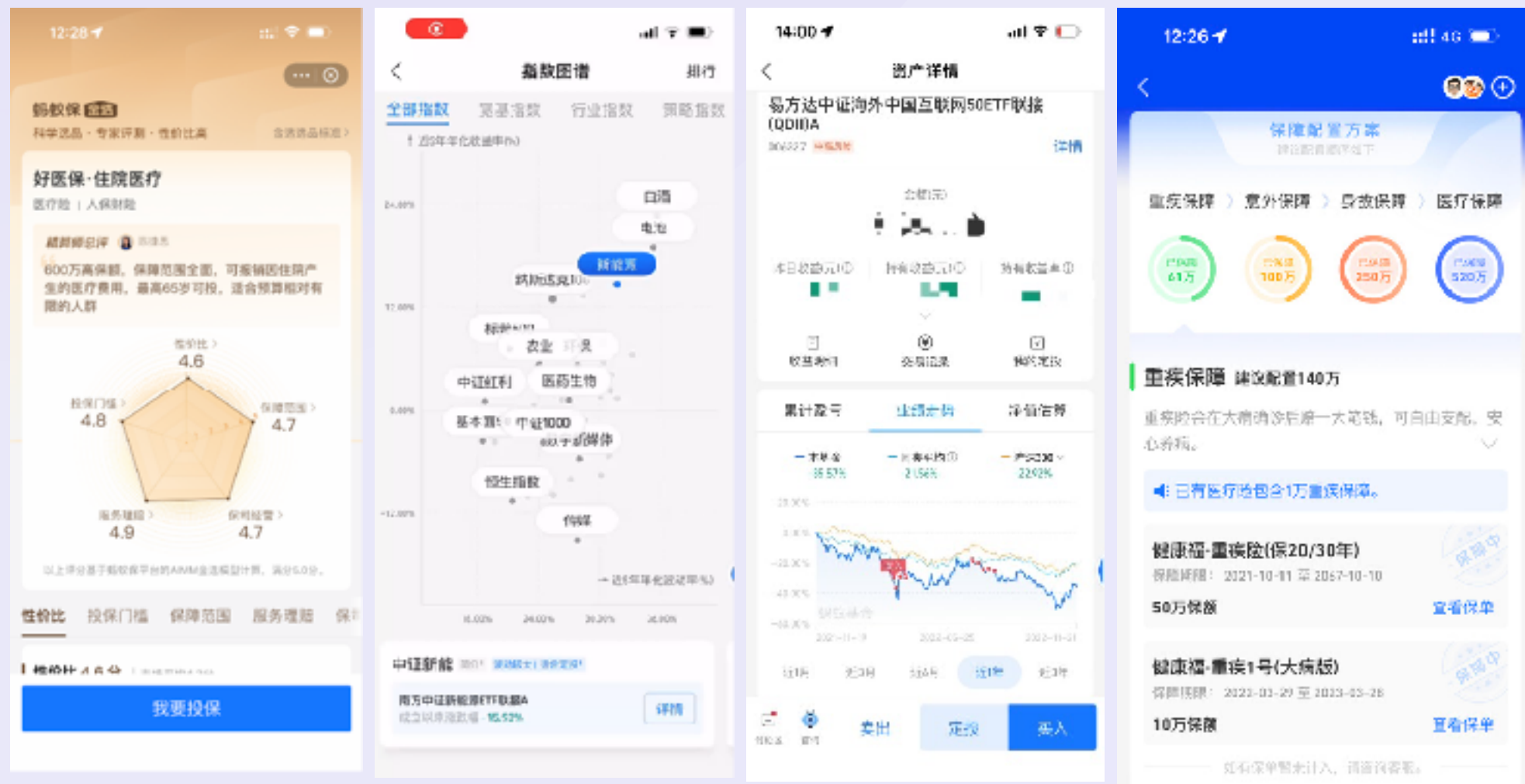
03 移动端可视化解决方案

移动端可视化是什么

白话：移动设备上呈现的可视化作品

业务场景实例

以下展示图表均为上线业务



性能差的手机可以兼容么？

可以多种图表组合么？

可以加 lottie 么？

小程序上可以画么？

这种动画么？

画图可太难了啊.gif

这个图表在

可以缩放平移快滑么？

图例可以是线性的么？

设计稿和默认的很像，但柱子可以3D的么？



业务开发同学

性能、包体积

尺寸适配

兼容性

手势交互

定制化需求高

视图聚焦，重表达与解释

目录

01 为什么会有移动端可视化

02 移动端可视化怎么这么难

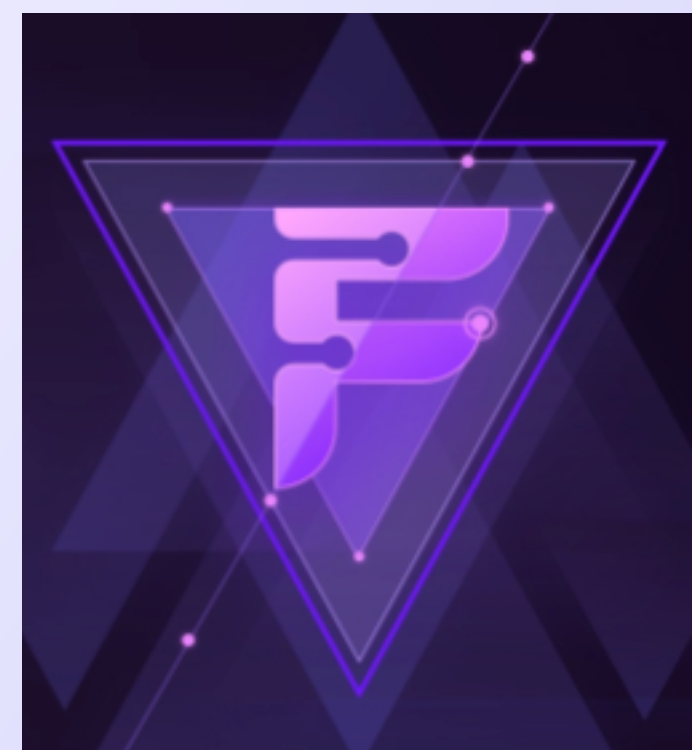
 03 移动端可视化解决方案

移动端解决方案

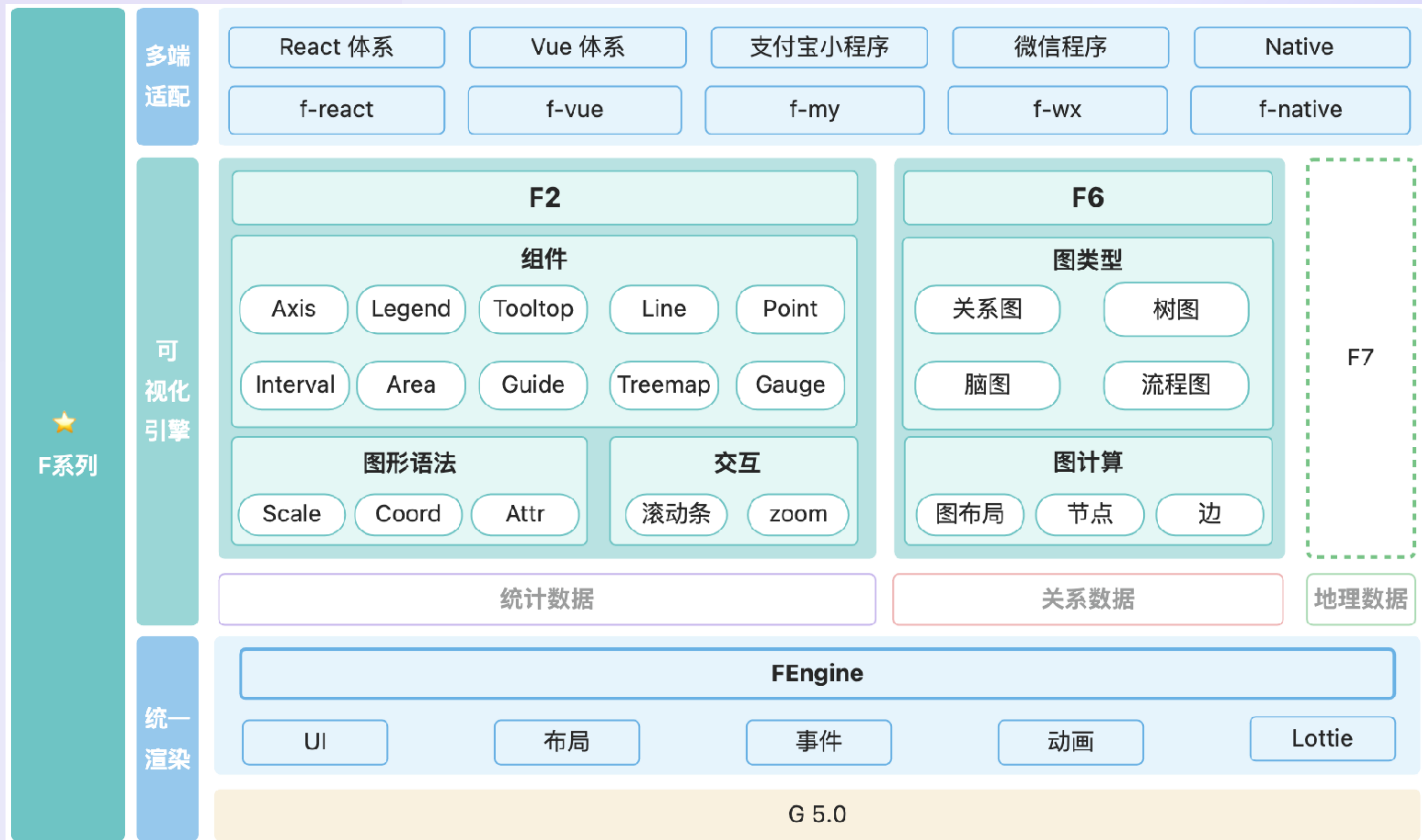
1. 重呈现表达

2. 轻巧

3. 灵活拓展



@antv/f 系列



统一渲染引擎 FEngine:

三种底层渲染引擎

基于 PointerEvent 事件系统

布局系统

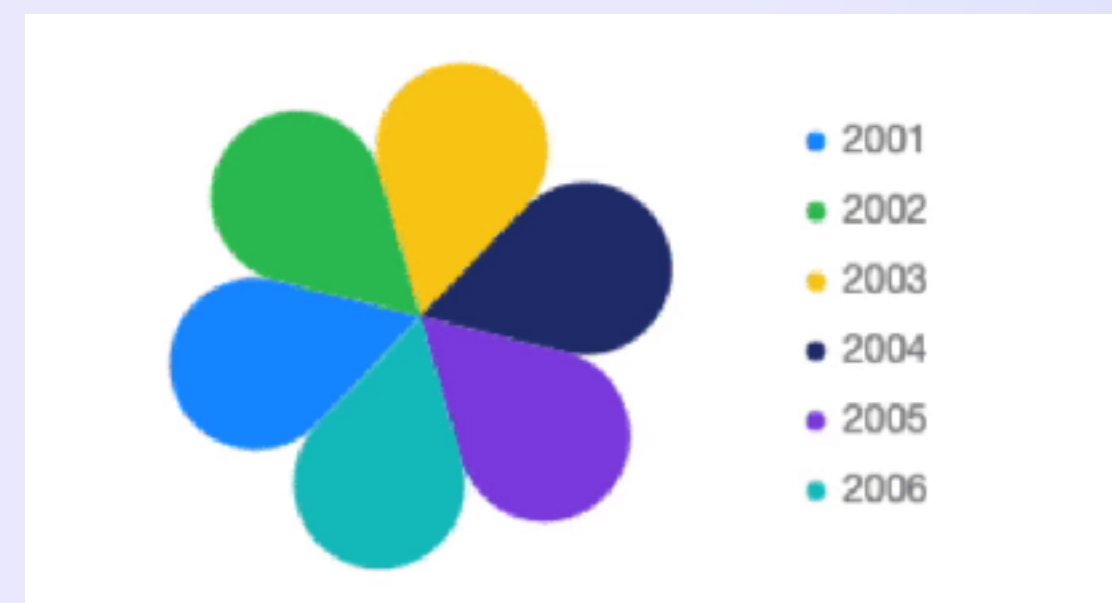
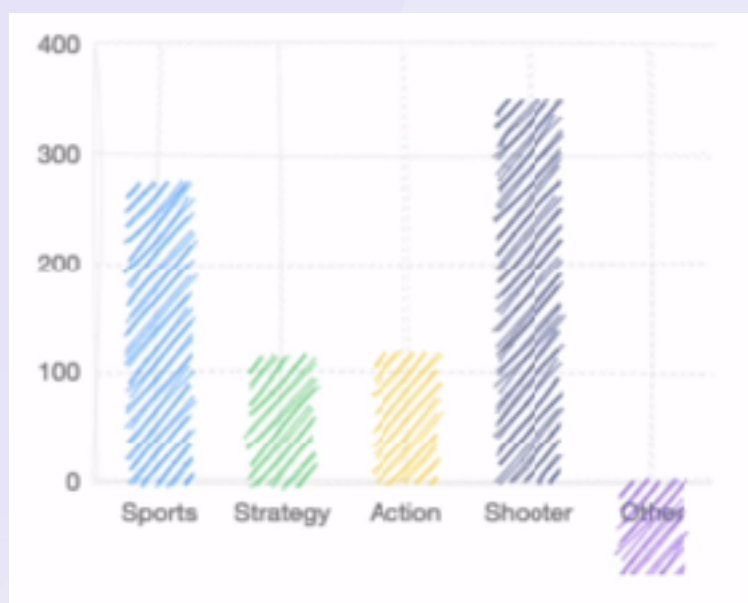
完善的动画机制

JSX声明式图形标签

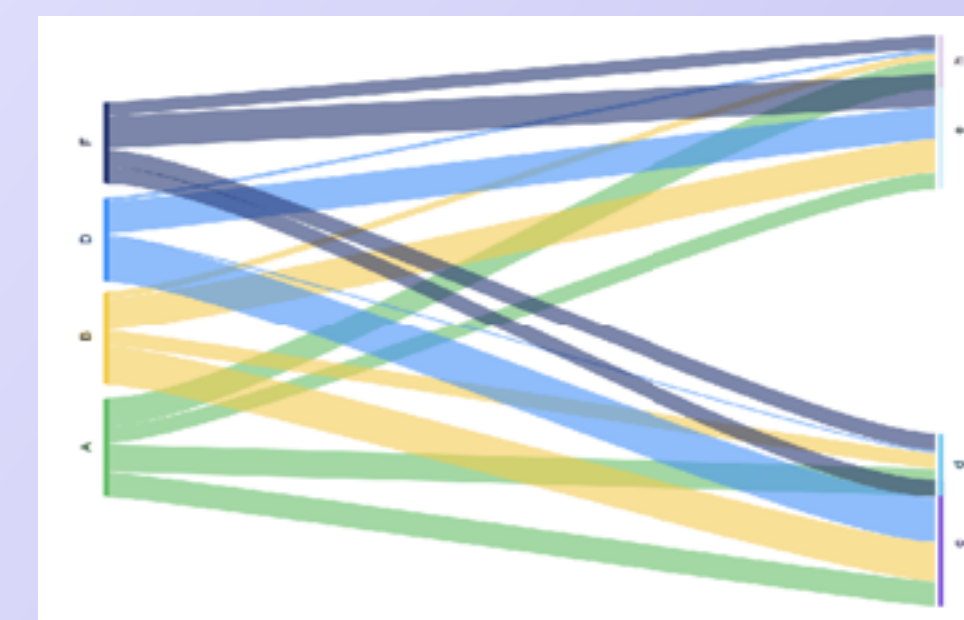
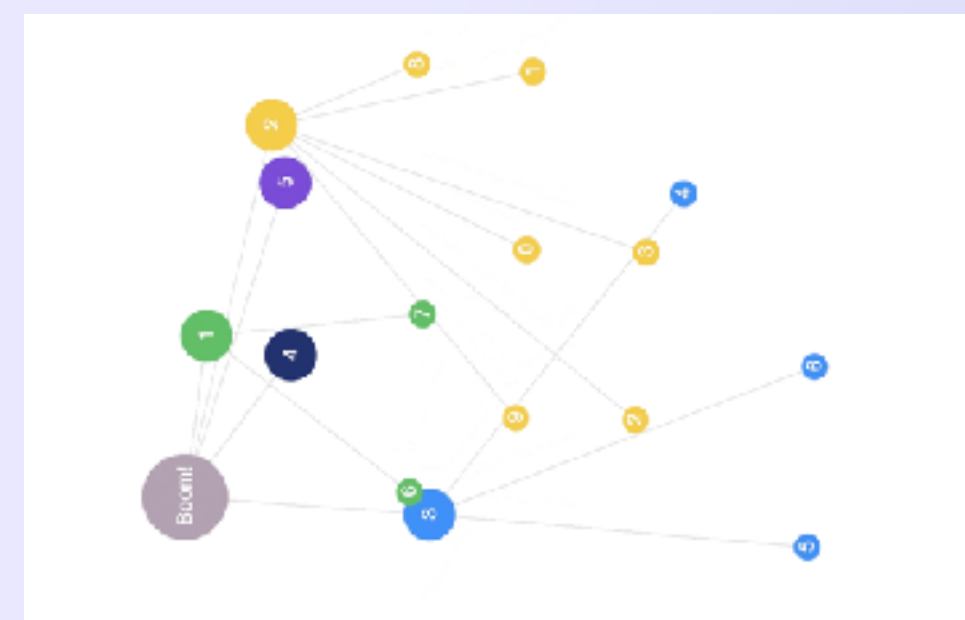
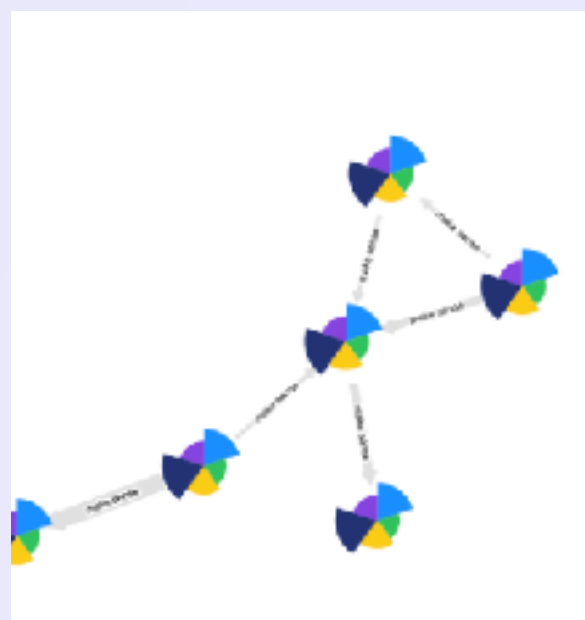
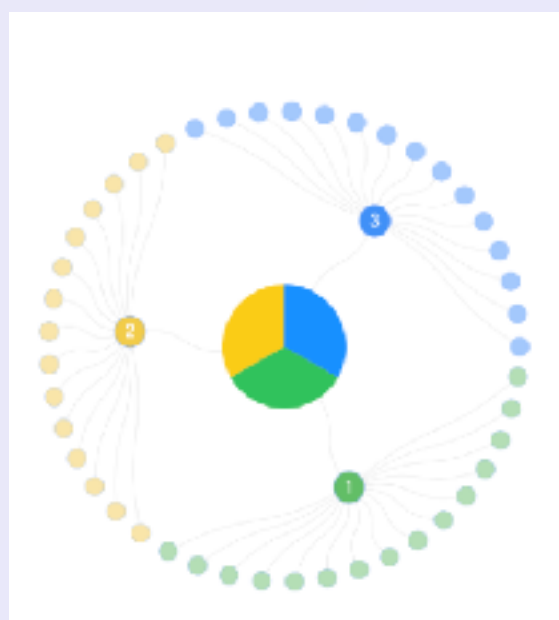
lottie 播放器

常见几类平台兼容

F2: 统计方向



F6: 图方向



F7: 地理方向



一个 🍎



业务开发同学

这个图表 F2 能实现么?



“拆解”



带动画的柱状图

动画配置

lottie

Lottie本身的动画播放
Lottie移动出现的动画

图例

和现有的内置样式不符

1. 选择合适的底层渲染器

三种底层渲染引擎



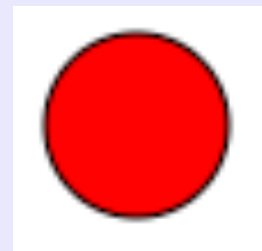
```
● ● ●  
<Canvas context={context} pixelRatio={1}>  
  <Chart data={data}>  
    </Chart>  
</Canvas>
```


插播：JSX声明式标签

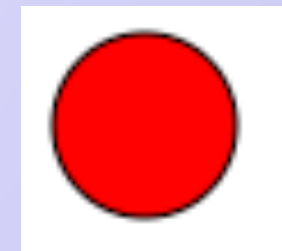
图形 shape

组件 component

```
<circle
  style={{
    x,
    y,
    r: '20px',
    fill: 'red',
  }}
/>
```



```
class Custom extends Component {
  willMount() {}
  didMount() {}
  shouldUpdate(_nextProps): boolean {
    return true;
  }
  willReceiveProps(_props) {}
  willUpdate() {}
  didUpdate() {}
  render() {
    return (
      <circle
        style={{
          x: '10px',
          y: '10px',
          r: '20px',
          fill: 'red',
        }}
      />
    );
  }
}
```



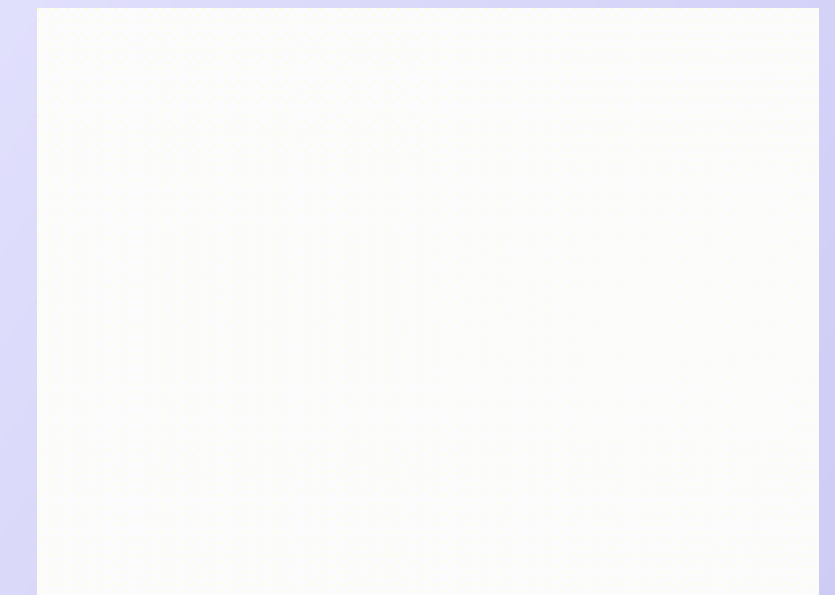
babel编译 jsx 标签 ➡ diff 生成 vnode ➡ 布局 ➡ 动画 ➡ 渲染

2. 画“最像”的柱子 内置的柱状图+动画

完善的动画机制

标签上支持 `appear`、`update`、`leave`

不同图形间自动形变



2. 画“最像”的柱子

```

<Canvas context={context} pixelRatio={1}>
  <Chart data={data}>
    <Interval
      x="genre"
      y="sold"
      color="genre"
      animation={{
        appear: {
          duration: 1000,
          easing: 'linear',
        },
      }}
    />
  </Chart>
</Canvas>

```

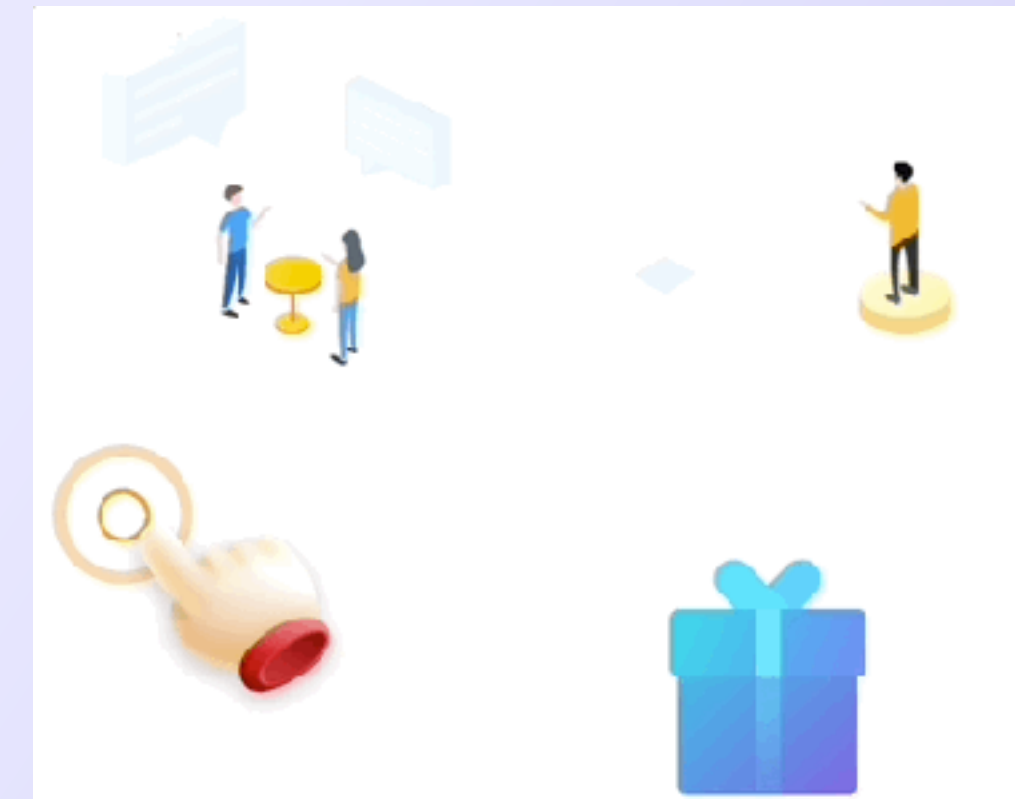


2. 画“有点像”的lottie

需要播控lottie，并在lottie上添加本身整体移动的动画

lottie 播放器

```
<Canvas context={context}>
  { /* 播放 lottie 动画 */ }
  <Lottie data={data} options={{ loop: true, autoplay: true }} />
</Canvas>
```

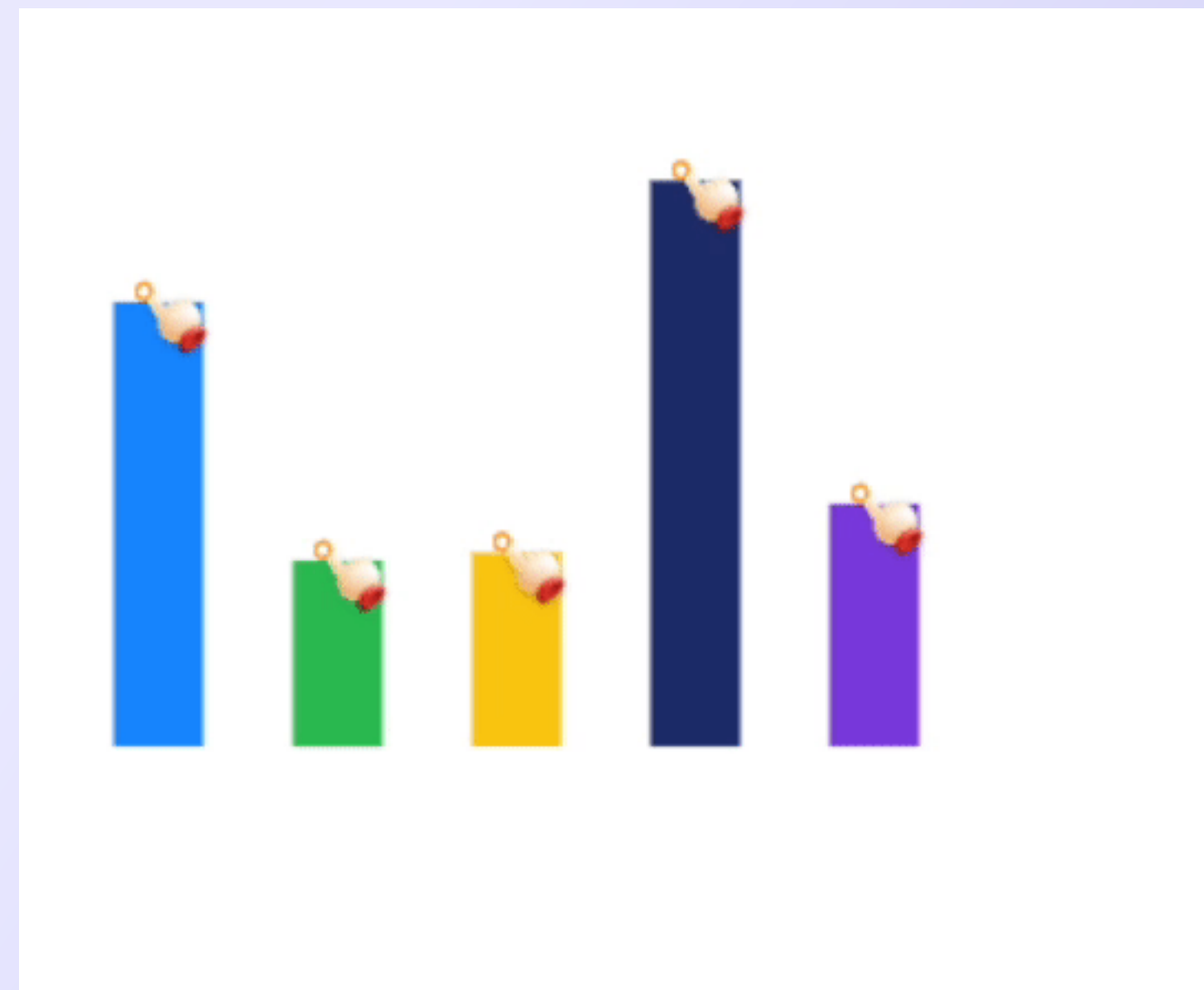


lottie guide

```
<LottieGuide
  offsetX="0px"
  offsetY="0px"
  records=[[item]]
  lottieJson={url}
  style={{
    height: 35,
    width: 35,
  }}/>
```


2. 画“有点像”的lottie

```
<Canvas context={context} pixelRatio={1}>
  <Chart data={data}>
    <Interval
      x="genre"
      y="sold"
      color="genre"
      animation={{
        appear: {
          duration: 1000,
          easing: 'linear',
        },
      }}
    />
    {data.map((item) => {
      return (
        <LottieGuide
          offsetX="0px"
          offsetY="0px"
          records={[item]}
          lottieJson={url}
          style={{
            height: 35,
            width: 35,
          }}
          animation={(points, props) => {
            return {
              appear: {
                easing: 'linear',
                duration: 1000,
                property: ['y'],
                start: {
                  y: props.coord.coord.height,
                  height: 0,
                },
              },
            },
          }}
        />
      );
    })}
  </Chart>
</Canvas>
```



3. 画“最不像”的图例

内置的图例样子我完全不想要，但是我又在相同地方相同数据上画
withLegend钩子，复用数据逻辑，使用图形标签自定义“样子”

```
const Legend = withLegend((props) => {
  const { items, itemWidth } = props;
  return (
    <group>
      {items.map((item) => {
        const { color, name } = item;
        return (
          <group>
            <text
              attrs={{
                fill: color,
                text: name,
              }}
            />
          </group>
        );
      })}
    </group>
  );
});
```



Spider

3. 画“最不像”的图例

如何对齐对齐对齐

完善的布局系统

```
const Legend = withLegend((props) => {
  const { items, itemWidth } = props;

  return (
    <group
      style={{
        display: 'flex',
        flexDirection: 'row',
        alignItems: 'center',
        justifyContent: 'center',
      }}
    >
      {items.map((item) => {
        const { color, name } = item;
        return (
          <group
            className="legend-item"
            style={{
              width: itemWidth,
              display: 'flex',
              flexDirection: 'row',
              alignItems: 'center',
              justifyContent: 'center',
            }}
          >
            <text
              attrs={{
                fill: color,
                text: name,
              }}
            />
          </group>
        );
      })}
    </group>
  );
});
```

Sports Strategy Action Shooter Other

3. 画“最不像”的图例

如何对齐对齐对齐

完善的布局系统

```
const Legend = withLegend((props) => {
  const { items, itemWidth } = props;

  return (
    <group
      style={{
        display: 'flex',
        flexDirection: 'row',
        alignItems: 'center',
        justifyContent: 'center',
      }}
    >
      {items.map((item) => {
        const { color, name } = item;
        return (
          <group
            className="legend-item"
            style={{
              width: itemWidth,
              display: 'flex',
              flexDirection: 'row',
              alignItems: 'center',
              justifyContent: 'center',
            }}
          >
            <text
              attrs={{
                fill: color,
                text: name,
              }}
            />
          </group>
        );
      })}
    </group>
  );
});
```

Sports Strategy Action Shooter Other

整幅图完成! 不, 还没结束....



业务开发同学

产品问图例可以点击进行和图表的联动么?

基于 PointerEvent 事件系统

PointerUp
PointerMove
PointerUp
PointerOutside

“组合”模拟



Pan
Click
Swipe
Press
Pinch

```
<rect  
  style={{  
    width: '200px',  
    height: '200px',  
    fill: 'red',  
  }}  
  onPan={onPan}  
>
```



业务开发同学

反响很好，需要在多平台投放，需要写多份代码么？

常见几类平台兼容

f-react、f-vue、f-my、f-wx

未来&结语

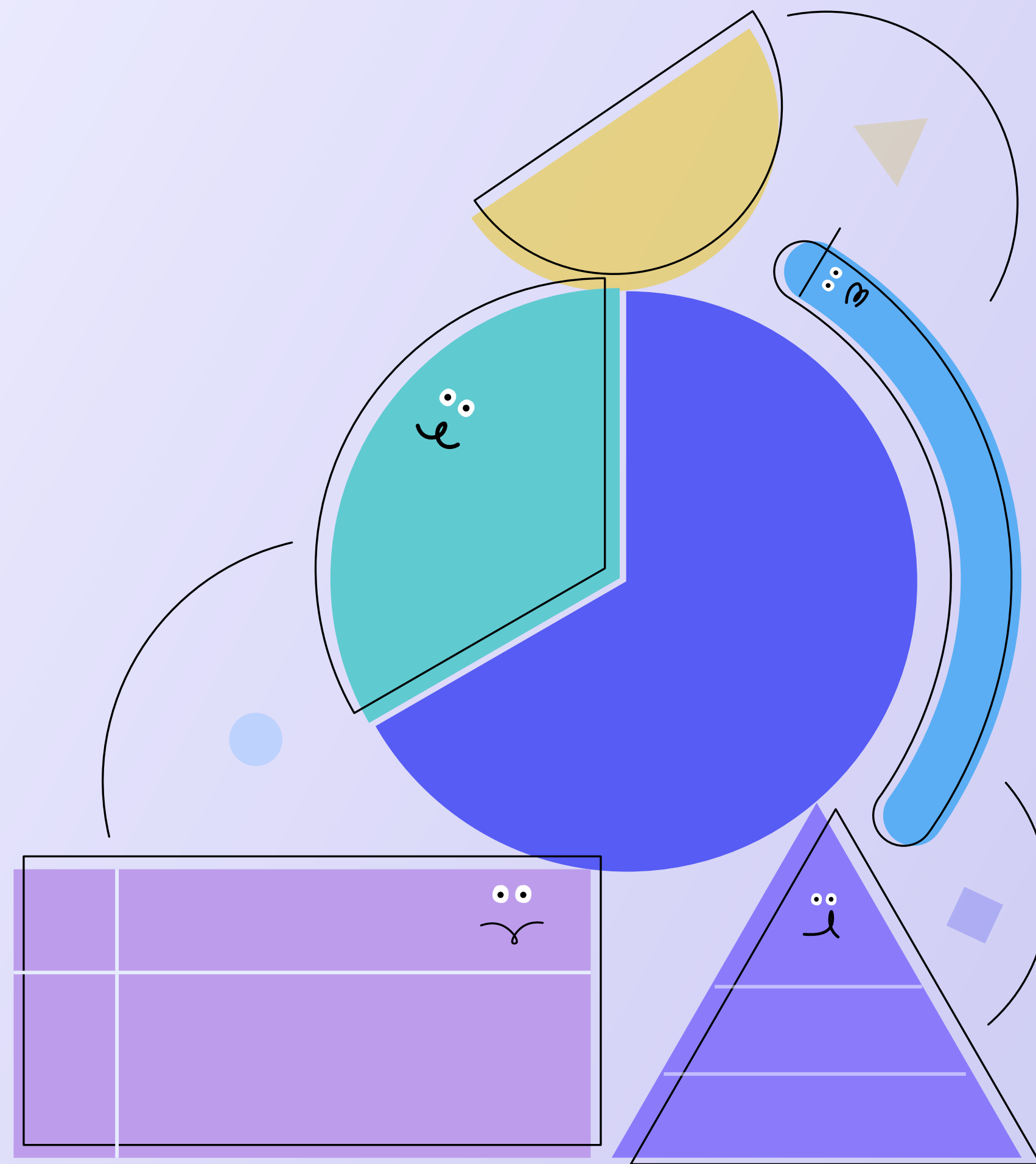
1. 基于 F 系列的底层共通，那么用 F2 与 F6的互通共用将可以实现。
2. 基于 webgl 渲染器以及场景对呈现质感的要求，三维图表是否可以被合适定义和表达。
3. lottie 的支持是否可以与现有图表形式迸发出更多的火花。



谢谢大家

欢迎指正与交流~

糖莹



一起走满5年

早早聊天使票预售

三周年限定版天使年票
最后 400 张

这次,你将 **TOGETHER**

解锁 2020 - 2024 年大会年票会员

2020-2024 年 800 场大会直播永久观看权

晋级职场高阶能力拓展营, 40 位顶尖高手带你玩

50 场群内直播分享, 剖析业务设计思路, 现象级产品解读, 复盘经典运营活动, 分享成体系管理实践, 全方位开商业天眼, 加速技能破圈

获得 Scott 职业建议与指导 + 管家式服务

Scott 职业建议与指导 2 次, 早早聊对你的商业作品评估和投资, 跳槽服务折扣及专享招聘服务, 优质人脉共享, 助力技术成果变现...

拥有早早聊 6 年利润分红

成为早早聊天使会员, 享有 2024 ~ 2029 年盈利分红权

扫码咨询
了解更多详情

 早早聊
ZAOZAO.RUN





12月3日 | 56届前端早早聊

玩转测试

工程质量
业务保障

UI 测试

单测

接口测试

自动化测试

智能测试



09:00 自动化: 《语雀的质量体系与自动化技术》

会能

蚂蚁集团
语雀团队
语雀质量负责人

10:00 自动化: 《Cypress E2E 自动化测试在 B 端场景下的实践》

王力国

达观数据
前端产品负责人

11:00 UI 自动化: 《花借 UI 自动化测试实践》

静贤

蚂蚁集团
数金体验技术部
前端技术专家

13:00 DevOps: 《前端工程质量保障体系建设》

张蕾

网易
前端开发工程师

14:00 智能: 《App 智能遍历测试与自动化测试用例生成》

思寒

测吧
霍格沃兹测试开发学社创始人

15:00 单测: 《浅谈 Node.js 单元测试 -- 人丑多读书, 人懒多单测》

天猪

EggJS
核心开发者

16:00 单测: 《单元测试从 0 到 1》

胡继伟

合思·易快报
大前端架构组
大前端架构组组长

17:00 接口: 《如何玩转接口测试》

张丽君

中电海康
数据与软件组
测试工程师

2022

全年行程

1/29 2/26 3/19 4/9 4/23 5/14 5/25 5/28 6/18 6/25 7/16 7/17 8/6 8/27 9/3 9/17 10/22 10/29 11/26 11/27 12/3 12/11 12/24

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 裁员仲裁 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 工程化跨端性能 框架专场 可视化专场 AntV 可视化 测试专场 远程工作专场 小程序专场



T Salon



12月11日 | 57届前端早早聊

远程工作

御寒利器
机遇满满

远程公司组建
远程团队管理
远程工作经验
开源远程协作
圆桌讨论



10:00 远程 10 年: 《远程工作十年的感悟和经验》

大灰
BIGray

电鸭社区
Founder

11:00 公司组建: 《我是如何搭建一支远程创业团队的?》

杨帆

CODE.FUN
CODE.FUN 创始人

12:00 团队管理: 《远程工作的需求管理和高效沟通》

齐伟
Nick

Tubi
资深前端工程师

13:00 开源社区: 《开源工作的远程工作经验》

蒋豪群

Vue Technology LLC
开源开发者

14:00 远程 3 年: 《远程工作的经验之谈》

Corki

海外公司
前端开发工程师

15:00 远程 2 年: 《远程工作感悟: 重要的几件事》

密叔

海外公司
前端 & Golang 开发

16:00 远程实习: 《在校生远程工作经验分享》

关艺俊

字节跳动
前端实习生

17:00 -: 《》

James

-

18:00 圆桌聊聊: 《你感兴趣的远程问题, 有问必答》

徐政

T 技术沙龙
运营负责人

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
5/25 裁员仲裁
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
10/22 工程化跨端性能
10/29 框架专场
11/26 可视化专场
11/27 可视化专场
12/3 测试专场
12/11 远程工作专场
12/24 小程序专场

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 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	5/25	5/28	6/18	6/25	7/16	7/17	8/6	8/27	9/3	9/17	10/22	10/29	11/26	11/27	12/3	12/11	12/24
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	裁员仲裁	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	工程化跨端性能	框架专场	可视化专场	Antv可视化	测试专场	远程工作专场	小程序专场